API producer zone

API Management

Exported on 06/10/2025

Table of Contents

1 Get started 120
2 Deliver your API 121
3 API register 122
4 Release schedule 123
5 Get help and support 124
6 Get started as an API producer 125
6.1 What are APIs? 125
6.2 Benefits of API platform 125
6.3 Guidance for specific roles 125
6.4 Tools, systems and environments 125
6.5 What are APIs and why do they matter? 126
6.5.1 Overview 126
6.5.2 What is an API? 126
6.5.2.1 What is an application? 126
6.5.2.2 What is a database? 126
6.5.2.3 What is an interface? 126
6.5.2.4 What is a user interface? 127
6.5.2.5 Finally, what is an API? 127
6.5.3 Why are APIs a good idea? 129
6.5.4 NHS England's Open API policy 131
6.5.5 API first 131
6.5.6 Different types of APIs 132
6.5.7 How to build a good API 133
6.5.7.1 New APIs 133
6.5.7.2 Existing APIs 133
6.6 Benefits of using the API platform 134
6.6.1 Slide deck (aka the "pitch deck") 134

6.6.2 What is the API platform? 134
6.6.3 Is the API platform right for me? 134
6.6.4 What do I get as an API producer? 135
6.6.5 What's not included? 137
6.7 Get started as a delivery lead, product owner, user researcher or business analyst 138
6.7.1 Engage 138
6.7.2 Get set up 138
6.7.3 Further reading 138
6.7.4 Help and support 138
6.7.5 User research for APIs 139
6.7.5.1 User research in the different delivery phases 139
6.7.5.2 Identifying users for APIs 139
6.7.5.3 Recruiting users for APIs 139
6.7.5.4 User research methods 140
6.7.5.5 Communicating with Developers 140
6.8 Get started as an architect or technical lead 141
6.8.1 Learn 141
6.8.2 Play 141
6.8.3 Engage 141
6.8.4 Get set up 141
6.8.4.1 Help and support 142
6.9 Get started as a software developer or tester 142
6.9.1 Play 142
6.9.2 Engage 143
6.9.3 Get set up 143
6.9.4 Further reading 143
6.9.5 Help and support 143
6.10 Get started as a technical author 143

6.10.1 Engage 144
6.10.2 Get set up 144
6.10.3 Further reading 144
6.10.4 Help and support 144
6.11 Get access to tools, systems and environments 144
6.11.1 Overview 145
6.11.2 Key 145
6.11.3 Get access to Confluence 150
6.11.4 Get access to Slack and support channels 151
6.11.4.1 Overview 151
6.11.4.2 NHS digital platform Slack workspace 151
6.11.4.3 Manual Slack request 152
6.11.4.4 Getting access to Slack channels once in the the workspace 153
6.11.4.5 Recommended Slack channels 153
6.11.5 Set up your developer environment 153
6.11.5.1 Prerequisites 153
6.11.5.2 Environment set up 154
6.11.5.3 Security requirements 158
6.11.5.4 Troubleshooting 160
6.11.6 Get access to Azure AD groups and Splunk 161
6.11.6.1 Overview 161
6.11.6.2 What type of access do you need? 161
6.11.6.3 Requesting Splunk access 162
6.11.6.4 Requesting Azure AD group access (Change in request process introduced in April 2024) 163
6.11.7 Get smartcard and reader access 165
6.11.7.1 Overview 165
6.11.7.2 Getting set up 165
6.11.7.3 Log in using your Smartcard 166

6.11.7.4 Spine application URLs 167
6.11.7.5 What to do if locked out 167
6.11.7.6 Mac OS 167
6.11.8 Get access to ServiceNow 167
6.11.8.1 Overview 167
6.11.8.2 Register for a ServiceNow account 167
6.11.8.3 Set up a new assignment group 167
6.11.8.4 Add team members to an assigned group 168
6.11.9 AWS HSCN VPN access 168
6.11.9.1 Overview 168
6.11.9.2 If you have access to nhsdigital-platforms.slack.com #platforms channel 169
6.11.9.3 If you don't have access to nhsdigital-platforms.slack.com #platforms channel (or you don't know who your approver is) 169
6.11.10 Set up NHS Smart Cards on OS X 169
6.11.10.1 Overview 169
6.11.10.2 1. Setting up VirtualBox 170
6.11.10.3 2. Installing Windows 10 on VirtualBox 171
6.11.10.4 3. Setting up the smart card on the VM 183
6.11.10.5 4. Setting up the smart card 187
6.11.10.6 5. Tips/Misc Info 190
6.11.11 Apigee prod access - Custom producer roles 192
6.11.11.1 Overview 192
6.11.11.2 Step 1 192
6.11.11.3 Step 2 192
6.11.11.4 Step 3 193
7 Deliver your API 194
7.1 Overview 194
7.2 Different types of API 194
7.3 A bit more detail 194

7.4 Delivery phases 195
7.4.1 Inception 195
7.4.2 Discovery 195
7.4.3 Alpha 196
7.4.4 Beta 196
7.4.5 Live 196
7.4.6 Sunsetting (deprecation and retirement) 196
7.5 Delivery phases versus API statuses 197
7.6 GDS assessment 197
7.7 API delivery options 198
7.8 API process checklist 198
7.9 Get started 198
7.10 API process checklist 198
7.10.1 Overview 199
7.10.1.1 Status codes key 199
7.10.1.2 Overview 199
7.10.1.3 Team members 200
7.10.1.4 Activities 200
7.10.1.5 Overview 202
7.10.1.6 Team members 202
7.10.1.7 Stakeholders 202
7.10.1.8 Activities 203
7.10.1.9 Overview 206
7.10.1.10 Team members 206
7.10.1.11 Activities 207
7.10.1.12 Overview 211
7.10.1.13 Team members 211
7.10.1.14 Enter private beta activities 222
7.10.1.15 Overview 224

7.10.1.16 Team members 224
7.10.1.17 Activities 225
7.11 API process checklist using proxy generator 227
7.11.1 Overview 227
7.11.1.1 Status codes key 228
7.11.1.2 Overview 228
7.11.1.3 Team members 228
7.11.1.4 Activities 228
7.11.1.5 Overview 230
7.11.1.6 Team members 230
7.11.1.7 Stakeholders 231
7.11.1.8 Activities 231
7.11.1.9 Overview 234
7.11.1.10 Team members 234
7.11.1.11 Activities 235
7.11.1.12 Overview 238
7.11.1.13 Team members 238
7.11.1.14 Enter private beta activities 244
7.11.1.15 Overview 246
7.11.1.16 Team members 246
7.11.1.17 Activities 247
7.12 Inception phase 249
7.12.1 Overview 249
7.12.2 Process steps 249
7.12.3 Resources 249
7.12.4 Delivery kick-off meeting 249
7.12.5 Getting your API on the interactive product backlog 250
7.12.5.1 Overview 250
7.12.5.2 How to use it 251

7.12.5.3 Accessing Feature Upvote as a moderator or admin 252
7.13 Discovery phase 252
7.13.1 Overview 252
7.13.2 Process steps 252
7.13.3 Resources 252
7.13.4 Identify and engage with internal stakeholders 253
7.13.4.1 Why it is important to engage with Internal stakeholders 253
7.13.4.2 Stakeholders for API producer teams 253
7.13.5 Understanding the Context and Problem/Opportunity Space 255
7.13.6 Understanding API consumers 256
7.13.6.1 Overview 256
7.13.6.2 Archetypes 256
7.13.6.3 Roles within development organisations 257
7.13.6.4 User journey 257
7.13.7 Mapping Out the User Journey for API Consumers 258
7.13.8 Define your scope and agree this with your sponsoring stakeholder 259
7.13.9 Architecting your API 260
7.13.9.1 Architecture kick-off meeting 262
7.13.9.2 API architecture options template 262
7.13.9.3 API architecture review checklist 266
7.13.9.4 Routing requests to multiple backends 280
7.13.9.5 Is my API a FHIR API? 285
7.13.9.6 Service levels for APIs 288
7.13.9.7 API architecture review 289
7.13.9.8 Wider architecture review 290
7.13.9.9 Large payloads 292
7.13.10 Metrics for success / KPIs for APIs 292
7.13.11 API delivery and support options 293
7.13.11.1 Overview 293

7.13.11.2 API delivery process 294
7.13.11.3 API delivery options 295
7.13.11.4 Long term support options 296
7.13.11.5 API delivery costs and timescales 296
7.13.12 Different types of API 297
7.13.12.1 Overview 297
7.13.12.2 Central API services 297
7.13.12.3 Peer to peer APIs (aka API standards) 298
7.13.12.4 Intermediary API services 298
7.13.12.5 Interoperability standards 299
7.13.13 Access modes - compliance - security and simplifying running your API 299
7.13.13.1 Overview 299
7.13.13.2 Approach 301
7.13.13.3 Implications 302
7.13.14 Cost / Benefit Analysis 304
7.13.15 API platform costs and raising a New Work Request (NWR) 305
7.13.15.1 Overview 305
7.13.15.2 API delivery costs and timescales 306
7.13.15.3 How to raise a New Work Request (NWR) 306
7.13.15.4 How we prioritise work 307
7.13.16 Discovery exit review 307
7.13.17 Products and Product IDs 308
7.13.17.1 Overview 308
7.13.17.2 Global Product ID 308
7.13.17.3 Organisations, product teams and products 309
7.13.17.4 Products, assurance and connection agreements 310
7.13.17.5 Versions of products 310
7.13.17.6 API Producer pages 311

7.13.17.7 Bigger picture 312
7.13.18 Analysis of API Migration from Non-OAS to OAS Format 314
7.13.18.1 A comprehensive list of all identified non-OAS APIs. 314
7.13.18.2 A detailed outline of the steps involved in migrating a Non OAS document into OAS spec JSON. 316
7.13.18.3 An estimated development time for migrating a single API to OAS. 317
7.13.18.4 An estimated time for migrating all APIs to OAS. 317
7.13.18.5 A summary of any potential challenges or dependencies identified during the analysis. 317
7.13.18.6 API Migration from Non-OAS to OAS Format 318
7.14 Alpha phase 321
7.14.1 Overview 321
7.14.2 Process steps 321
7.14.3 Resources 321
7.14.4 Designing your API 322
7.14.4.1 Overview 322
7.14.4.2 Guiding design principles 322
7.14.4.3 Moving into the detailed design 323
7.14.4.4 FHIR 324
7.14.4.5 API naming and grouping endpoints 359
7.14.4.6 Consistency between APIs 362
7.14.4.7 Proxy base path vs API base path 364
7.14.4.8 RESTful HTTP design points 365
7.14.4.9 RESTful APIs good practice 368
7.14.4.10 Transforming data on the APIM platform and containers 371
7.14.4.11 API design review 375
7.14.4.12 Version control 380
7.14.4.13 HTTP headers for your API 395
7.14.4.14 Proxies in Apigee and what is an API Producer responsible for 402
7.14.4.15 Key elements of any API published on the platform 404

7.14.4.16 Privacy considerations in your design 406
7.14.4.17 Spine based APIs 408
7.14.4.18 Continuous deployment target model 422
7.14.4.19 Proxy complexity and sharing policies 433
7.14.4.20 Guidance for making API security decisions 437
7.14.4.21 Spine based APIs and single ASID model 444
7.14.5 Building your API alpha 578
7.14.5.1 Overview 579
7.14.5.2 Prerequisites 579
7.14.5.3 Process steps 579
7.14.5.4 Creating a GitHub repository and deployment pipelines for your API 580
7.14.5.5 Setting up your API sandbox 592
7.14.5.6 Path to live environments 596
7.14.5.7 Setting up a target server 607
7.14.5.8 Passing information to your API backend 611
7.14.5.9 Handling errors 624
7.14.5.10 APIM AWS Hosted containers 637
7.14.5.11 Async to sync conversion (sync wrap) 647
7.14.5.12 Using multi-line secrets in the pipeline 661
7.14.5.13 ExtendedAttributes shared flow 662
7.14.5.14 Storing secrets securely 667
7.14.5.15 EnhancedVerifyApiKey shared flow 668
7.14.5.16 Handling Messaging Response Error code - For Review 669
7.14.5.17 SingleAsidApply shared flow 676
7.14.6 Building and deploying your API alpha with Proxy Generator 680
7.14.6.1 Overview 680
7.14.6.2 Prerequisites 680
7.14.6.3 Process steps 680
7.14.6.4 Getting set up with proxy generator 681

7.14.6.5 Authenticating with Proxy Generator 682
7.14.6.6 Setting up your API sandbox with Proxy Generator 684
7.14.6.7 Configuring your connection to your API backend with Proxy Generator 688
7.14.6.8 Creating your spec-only API specification with Proxy Generator 696
7.14.6.9 Deploying your API with Proxy Generator 698
7.14.6.10 Configuring target attributes 708
7.14.6.11 Configuring Target Identity Headers 712
7.14.6.12 Deploy an instance into DEP using Proxygen 713
7.14.7 Testing your API 715
7.14.7.1 Starting point 716
7.14.7.2 Test Utils 2.0: pytest plugin for APIM 718
7.14.7.3 Testing with mock auth 737
7.14.7.4 Automated quality testing of your API 753
7.14.7.5 INT environment and multiple identity services 755
7.14.7.6 Performance testing 756
7.14.7.7 Environments 758
7.14.8 Securing your API 774
7.14.8.1 Overview 774
7.14.8.2 Wider security approach 777
7.14.8.3 Security patterns and implementations 777
7.14.8.4 Security for API Backend 778
7.14.8.5 Security for Third-Party Applications 789
7.14.8.6 Security for Apigee 795
7.14.9 Securing your API with proxy generator 856
7.14.9.1 Security segments 856
7.14.9.2 Topics 857
7.14.9.3 API backend security 857
7.14.9.4 API consumer access security patterns 871

7.14.10 Deploying your API ... 882
7.14.10.1 Overview ... 882
7.14.10.2 Process steps ... 883
7.14.10.3 Configuring the API deployment ... 883
7.14.10.4 Managing your API deployments ... 884
7.14.10.5 Manifest.yml reference ... 896
7.14.10.6 Pipeline caching using the s3 cache task ... 908
7.14.10.7 Using manual approval on pipelines ... 913
7.14.10.8 Accessing your deployed API ... 925
7.14.10.9 Source control ... 928
7.14.11 Alpha exit review ... 933
7.14.12 Proxygen CLI user guide ... 933
7.14.12.1 Overview ... 933
7.14.12.2 Getting started ... 934
7.14.12.3 Managing your credentials ... 937
7.14.12.4 Managing your settings ... 938
7.14.12.5 Deploying an API instance ... 938
7.14.12.6 Managing your API specifications ... 940
7.14.12.7 Managing secrets used by your API ... 941
7.14.12.8 Interacting with your API's docker repository ... 943
7.14.12.9 Retrieve a token for use with the pytest-nhsd-apim python testing package ... 944
7.15 Beta phase ... 944
7.15.1 Overview ... 944
7.15.2 Process steps ... 945
7.15.3 Resources ... 945
7.15.4 Preparing your API beta for production ... 945
7.15.4.1 Overview ... 945
7.15.4.2 Prerequisites ... 946

7.15.4.3 Process steps 946
7.15.4.4 Get approvals to deploy to production 947
7.15.4.5 Engineering tasks..... 1036
7.15.4.6 Engineering tasks for proxy generator 1115
7.15.5 First time deploying into production review 1120
7.15.6 Deploying to production 1120
7.15.6.1 Overview 1121
7.15.6.2 Self-serve releases..... 1121
7.15.6.3 Deploying to production for the first time 1122
7.15.6.4 Determining what needs an RFC and timing of the release 1122
7.15.6.5 Creating an RFC 1122
7.15.6.6 Scheduling a release for an API Platform team deployment 1130
7.15.6.7 Changing the date of release 1130
7.15.6.8 Deploying to production for the first time 1131
7.15.7 Deploying to production with Proxy Generator 1135
7.15.7.1 Overview 1135
7.15.7.2 Deploying to production for the first time 1135
7.15.7.3 Managing production deployments with RFCs 1136
7.15.7.4 Creating an RFC 1136
7.15.7.5 Scheduling a release for an API Platform team deployment 1143
7.15.7.6 Changing the date of release 1144
7.15.8 Testing in production..... 1144
7.15.8.1 Overview 1144
7.15.8.2 Smoke testing 1144
7.15.8.3 Smoke testing 1145
7.15.9 API Consumer connection topologies..... 1149
7.15.9.1 Securing connections to NHS England..... 1149
7.15.9.2 Compliance and security controls to protect patient data..... 1150
7.15.10 Beta exit review 1154

7.15.11 Developer community guidance 1154
7.15.11.1 Overview 1154
7.15.11.2 Owning and managing sub-communities 1155
7.15.11.3 Operations 1155
7.15.12 Identifying Who Are Your API Consumers? 1172
7.15.12.1 Internal Consumers 1172
7.15.12.2 External Consumers 1174
7.15.13 Qualtrics Survey to Power BI 1176
7.15.13.1 Qualtrics Popup Survey Implementation 1176
7.15.13.2 Automated Feedback Data Processing 1176
7.15.13.3 Scheduled Execution via GitHub Actions 1177
7.15.13.4 Data Integration Requirements from Qualtrics to Power BI & SharePoint 1177
7.15.13.5 Steps to Configure Microsoft Entra App & Power BI Integration 1178
7.16 Live phase 1179
7.16.1 Overview 1179
7.16.2 Process steps 1180
7.16.3 Resources 1180
7.17 Sunsetting (deprecation and retirement) phases 1180
7.17.1 Overview 1180
7.17.2 Deprecating an API 1181
7.17.3 Retiring an API 1182
7.17.4 Deprecating / retiring API endpoints 1182
7.17.5 Resources 1183
7.17.6 Sunsetting phases for an API standard 1183
7.17.6.1 Overview 1183
7.17.6.2 Deprecation 1184
7.17.6.3 Retirement 1185
7.17.6.4 Sunsetting process checklist for API standards 1186

7.17.7 Sunsetting checklist template ... 1186
7.17.7.1 Overview ... 1186
7.17.7.2 Checklists ... 1187
7.17.7.3 API details ... 1192
7.17.7.4 Reason for deprecation ... 1192
7.17.7.5 Usage analysis ... 1192
7.17.7.6 Stakeholders ... 1193
7.17.7.7 Deprecation comms plan ... 1193
7.17.8 Sunsetting checklist template (API standards) ... 1194
7.17.8.1 Overview ... 1194
7.17.8.2 Checklists ... 1194
7.17.8.3 API standard details ... 1198
7.17.8.4 Reason for deprecation ... 1199
7.17.8.5 Stakeholders ... 1199
7.17.8.6 Deprecation comms messaging ... 1199
7.17.8.7 Retirement comms messaging ... 1200
7.17.9 WGLL - sample comms (both internal & external) ... 1200
7.17.9.1 What good looks like ... 1200
8 API register ... 1204
8.1 API register - new APIs ... 1204
8.2 API sunsetting register ... 1307
8.3 Producer team member audit ... 1313
8.3.1 Audit tasks ... 1313
8.3.1.1 Link to AMR board ... 1314
8.3.2 Audit information table ... 1314
8.3.3 Comms Template ... 1334
8.3.4 Audit ticket template ... 1335
8.3.4.1 Title ... 1336
8.3.4.2 Description ... 1336

8.4 What Good Looks Like checklist 1336
8.4.1 Overview 1336
8.4.2 Why does the checklist exist? 1336
8.4.3 Who should use the checklist 1337
8.4.4 How this relates to our API policies and "quality measures" 1337
8.4.5 How to use the checklist 1337
8.4.6 Checklist summaries 1337
8.4.7 Cyber Alerts API - What Good Looks Like (WGLL) checklist 1339
8.4.7.1 Overview 1339
8.4.7.2 Why does this checklist exist? 1339
8.4.7.3 Who should use this checklist 1339
8.4.7.4 How this relates to our "quality measures" 1340
8.4.7.5 How to use this checklist 1340
8.4.7.6 Participants 1340
8.4.7.7 Status codes 1341
8.4.7.8 Checklist 1342
8.4.7.9 Summary of recommendations 1347
8.4.8 MESH API - What Good Looks Like (WGLL) checklist 1347
8.4.8.1 Overview 1347
8.4.8.2 Why does this checklist exist? 1348
8.4.8.3 Who should use this checklist 1348
8.4.8.4 How to use this checklist 1348
8.4.8.5 Participants 1349
8.4.8.6 Status codes 1349
8.4.8.7 Checklist 1349
8.4.9 NHS CIS2 CIA - What Good Looks Like (WGLL) checklist 1355
8.4.9.1 Overview 1355
8.4.9.2 Why does this checklist exist? 1355
8.4.9.3 Who should use this checklist 1355

8.4.9.4 How this relates to our "quality measures" 1355
8.4.9.5 How to use this checklist 1356
8.4.9.6 Participants 1356
8.4.9.7 Status codes 1357
8.4.9.8 Checklist 1357
8.4.10 NHS login - What Good Looks Like (WGLL) checklist 1363
8.4.10.1 Overview 1363
8.4.10.2 Why does this checklist exist? 1363
8.4.10.3 Who should use this checklist 1364
8.4.10.4 How this relates to our "quality measures" 1364
8.4.10.5 How to use this checklist 1364
8.4.10.6 Participants 1364
8.4.10.7 Status codes 1365
8.4.10.8 Checklist 1366
8.4.11 Terminology Server - What Good Looks Like (WGLL) checklist 1372
8.4.11.1 Overview 1372
8.4.11.2 Why does this checklist exist? 1372
8.4.11.3 Who should use this checklist 1373
8.4.11.4 How to use this checklist 1373
8.4.11.5 Participants 1373
8.4.11.6 Status codes 1374
8.4.11.7 Checklist 1375
8.4.11.8 Summary of recommendations 1382
8.4.12 What Good Looks Like (WGLL) checklist - template 1383
8.4.12.1 Overview 1383
8.4.12.2 How to use this checklist 1384
8.4.12.3 Participants 1384
8.4.12.4 Status codes 1385
8.4.12.5 Checklist 1385

9 Release Schedule 1390
9.1 Planned Releases (with RFC) 1391
9.2 CHG NumberHere - Template 1485
9.2.1 Tickets 1485
9.2.2 Prepared Release 1485
9.2.3 Release Process 1486
9.2.3.1 Post Release Checks 1486
9.2.4 Roll-Back 1486
9.3 Archive 1486
9.3.1 2021-02-16 identity-service-api deployment plan 1489
9.3.1.1 Plan Summary 1489
9.3.1.2 Detail 1490
9.3.2 Release Plan APIM-2021-03-02 1492
9.3.3 RFC 2021-12-14 TBD: Upgrade netcompany app ratelimit in production 1492
9.3.3.1 Release plan 1492
9.3.3.2 Rollback plan 1494
9.3.4 RFC 30025111 - Monitoring Service Discovery in Prod 1495
9.3.4.1 Revised Plan - Tuesday 9th March 1495
9.3.4.2 Revised Plan - Thursday 4th March 1496
9.3.4.3 Revised Plan - Tuesday 2nd March 1497
9.3.4.4 Original Plan Summary 1498
9.3.5 RFC 30025379: APIM Identity Service Updates - v1.0.859-alpha 1499
9.3.5.1 Release Plan - Tuesday 16th March 1500
9.3.5.2 Release Process 1500
9.3.6 RFC 30025426 - Immunisation History API Updates 1500
9.3.6.1 Tickets in this release 1500
9.3.6.2 Release Procedure 1501
9.3.6.3 Rollback Procedure 1501

9.3.7 RFC 30025432: APIM Ambulance Data Submissions API Updates - v1.0.330-alpha 1501
9.3.7.1 Status of tickets to be deployed 1502
9.3.7.2 RFC Change Request Details 1502
9.3.7.3 Manual Changes to be applied 1502
9.3.7.4 Dependencies 1502
9.3.7.5 Release Plan Summary 1502
9.3.8 RFC 30025529: APIM EPS Updates 1503
9.3.8.1 Tickets 1503
9.3.8.2 Release Plan - Wednesday 7th April 1503
9.3.8.3 Release Process 1503
9.3.9 RFC 30025588: APIM Apigee Config - Update Production Certificates 1503
9.3.9.1 Tickets 1503
9.3.9.2 Release Plan - Tuesday 23rd March 1504
9.3.9.3 Release Process 1504
9.3.10 RFC 30025611: APIM - Remove API Platform caveat for limiting the number of connecting parties 1504
9.3.10.1 Tickets 1504
9.3.10.2 Release Plan - Tuesday 23rd March 1505
9.3.10.3 Release Process 1505
9.3.11 RFC 30025620: APIM - Remove API Platform caveat for TPS rate limit of 5 TPS 1505
9.3.11.1 Tickets 1505
9.3.11.2 Release Plan - Tuesday 23rd March 1505
9.3.11.3 Release Process 1506
9.3.12 RFC 30025723: APIM Identity Service Updates 1506
9.3.12.1 Tickets 1506
9.3.12.2 Release Plan - Tuesday 30th March 1507
9.3.13 RFC 30025798: Splunk logging improvements and Changes to support Identity Service 1507

9.3.13.1 Tickets 1507
9.3.13.2 Prepared Release 1507
9.3.13.3 Instructions 1507
9.3.14 RFC 30025803: APIM PDS Base URL Deprecation Updates 1508
9.3.14.1 Tickets 1508
9.3.14.2 Prepared Release 1508
9.3.14.3 Release Process 1508
9.3.15 RFC 30025874: APIM Identity Service Updates 1508
9.3.15.1 Tickets 1508
9.3.15.2 Release Plan - Thursday 8th April 1509
9.3.15.3 Release Process 1509
9.3.16 RFC 30026054: Fix Spike Arrest Error Message 1510
9.3.16.1 Tickets 1510
9.3.16.2 Prepared Release 1510
9.3.16.3 Release Process 1510
9.3.17 RFC 30026104: APIM Identity Service Updates 1510
9.3.17.1 Tickets 1510
9.3.17.2 Release Plan - Tuesday 19th April 1511
9.3.17.3 Release Process 1511
9.3.18 RFC 30026140 - API Management Infrastructure Updates 1512
9.3.18.1 Tickets in this RFC 1512
9.3.18.2 Release Procedure 1512
9.3.18.3 Rollback Procedure 1513
9.3.19 RFC 30026495: APIM - Identity Service Updates 1513
9.3.19.1 Tickets 1513
9.3.20 RFC 30026553- API Management Infrastructure Updates 1514
9.3.20.1 Tickets in this RFC 1514
9.3.20.2 Release Procedure 1514
9.3.20.3 Rollback Procedure 1515

9.3.21 RFC 30026822 - Infrastructure Service Update 1515
9.3.21.1 Tickets in this RFC 1515
9.3.22 RFC 30027067, 30027100 & 30027153: APIM - Infrastructure Release 1515
9.3.22.1 Tickets in this RFC 1515
9.3.22.2 Purpose of release 1516
9.3.22.3 Release Number: 1516
9.3.22.4 Release Procedure 1516
9.3.22.5 Rollback Procedure 1517
9.3.23 RFC 30027083: APIM - PDS API - Sync in Production 1518
9.3.23.1 Tickets in this RFC 1518
9.3.23.2 Purpose of release 1518
9.3.23.3 Release Number: 1518
9.3.23.4 Release Procedure 1519
9.3.23.5 Rollback Procedure 1519
9.3.23.6 Release prep checklist 1519
9.3.24 RFC 30027232: APIM Run script in prod to update JWKS resource URLs for applications 1520
9.3.24.1 Tickets in this RFC 1520
9.3.24.2 Purpose of RFC 1520
9.3.24.3 Release Number: 1520
9.3.24.4 Release Procedure 1520
9.3.24.5 Rollback Procedure 1521
9.3.25 RFC 30027332 - Digital Onboarding Service Release 1521
9.3.25.1 Tickets in this RFC 1521
9.3.25.2 Purpose of release 1521
9.3.25.3 Release Number: 1522
9.3.25.4 Release Procedure 1522
9.3.25.5 Rollback Procedure 1522
9.3.26 RFC 30027481 PDS API - Sync Wrap PROD 1522

9.3.26.1 Cell G Checklist 1523
9.3.26.2 Release Number 1523
9.3.26.3 Prepared Release 1523
9.3.27 RFC 30027510 - API Management Infrastructure - Identity Service KVM 1524
9.3.27.1 Tickets in this RFC 1524
9.3.27.2 Release Procedure 1524
9.3.27.3 Rollback Procedure 1524
9.3.28 RFC 30027613: APIM PDS Production Quota Update 1525
9.3.28.1 Tickets 1525
9.3.28.2 Prepared Release 1525
9.3.28.3 Release Process 1525
9.3.28.4 Roll-Back 1525
9.3.29 RFC 30027655: APIM PDS Disable Async Update Mode 1525
9.3.29.1 Tickets 1525
9.3.29.2 Prepared Release 1526
9.3.29.3 Release Process 1526
9.3.29.4 Roll-Back 1526
9.3.30 RFC 30027682: APIM Infrastructure Updates to enable prod Spine connectivity 1526
9.3.30.1 Tickets 1526
9.3.30.2 Prepared Release 1526
9.3.30.3 Release Process 1527
9.3.30.4 Roll-Back 1527
9.3.31 RFC 30027868 - Verifying URID for user updates 1527
9.3.31.1 High level checklist 1527
9.3.31.2 Release Process 1528
9.3.31.3 Roll-Back 1529
9.3.32 RFC 30027924 - Digital Onboarding Service - High Volume SMTP fix . 1530
9.3.32.1 Tickets 1530

9.3.32.2 Prepared Release 1530
9.3.32.3 Release Process 1530
9.3.32.4 Roll-Back 1530
9.3.33 RFC 30028033: APIM Identity Service Updates 1531
9.3.33.1 Tickets 1531
9.3.33.2 Release Plan - Tuesday 30th March 1531
9.3.34 RFC 30028203 - Digital Onboarding Service - GA/Hotjar fixes 1531
9.3.34.1 Tickets 1531
9.3.34.2 Prepared Release 1532
9.3.34.3 Release Process 1532
9.3.34.4 Roll-Back 1532
9.3.35 RFC 30028231 - PDS FHIR API Conformance Resource URL Change . 1532
9.3.35.1 Tickets 1532
9.3.35.2 Prepared Release 1533
9.3.35.3 Release Process 1533
9.3.35.4 Roll-Back 1533
9.3.36 RFC 30028313 - Digital Onboarding Service - GA/Hotjar fixes 1533
9.3.36.1 Tickets 1533
9.3.36.2 Prepared Release 1533
9.3.36.3 Release Process 1534
9.3.36.4 Roll-Back 1534
9.3.37 RFC 30028333 - API Management Infrastructure Updates 1534
9.3.37.1 Release Version: v2021.07.29.10.39 1534
9.3.37.2 Tickets in this RFC 1534
9.3.37.3 Release Procedure 1535
9.3.37.4 Rollback Procedure 1535
9.3.38 RFC 30028351 - APIM Covid-19 Test Results API - Token Exchange .. 1535
9.3.38.1 Tickets 1535
9.3.38.2 Prepared Release 1536

9.3.38.3 Release Process 1536
9.3.38.4 Roll-Back 1536
9.3.38.5 Results Service 8.9.3 - Release Notes 1536
9.3.39 RFC 30028396 Apigee application restricted for immunisation history api 1537
9.3.39.1 Tickets 1537
9.3.39.2 Prepared Release 1537
9.3.39.3 Release Process 1537
9.3.39.4 Roll-Back 1538
9.3.40 RFC 30028431 - TPS raised to 270 - PDS API 1538
9.3.40.1 Tickets 1538
9.3.40.2 Prepared Release 1538
9.3.40.3 Release Process 1538
9.3.40.4 Roll-Back 1539
9.3.40.5 High level checklist 1539
9.3.41 RFC 30028449: APIM PDS Patient (Citizen) Access Integration 1540
9.3.41.1 Tickets 1540
9.3.41.2 Prepared Release 1540
9.3.41.3 Release Process 1540
9.3.41.4 Roll-Back 1540
9.3.42 RFC 30028450 - APIM Infrastructure Changes 1540
9.3.42.1 Tickets 1540
9.3.42.2 Prepared Release 1541
9.3.42.3 Release Process 1541
9.3.42.4 Roll-Back 1541
9.3.43 RFC 30028523 - Management UI SAML Testing - In hours 1541
9.3.43.1 Tickets 1541
9.3.43.2 Prepared Release 1542
9.3.43.3 Release Process 1542

9.3.43.4 Roll-Back ... 1542
9.3.44 RFC 30028609 - Digital Onboarding Service - disable user, resend invite, DAQ changelog, misc fixes ... 1542
9.3.44.1 Tickets ... 1542
9.3.44.2 Prepared Release ... 1543
9.3.44.3 Release Process ... 1543
9.3.44.4 Roll-Back ... 1543
9.3.45 RFC 30028763 - Digital Onboarding Service Fixes ... 1543
9.3.45.1 Tickets ... 1543
9.3.45.2 Prepared Release ... 1544
9.3.45.3 Release Process ... 1544
9.3.45.4 Roll-Back ... 1544
9.3.46 RFC 30028795 - APIM Infrastructure Changes ... 1544
9.3.46.1 Tickets ... 1544
9.3.46.2 Prepared Release ... 1545
9.3.46.3 Release Process ... 1545
9.3.46.4 Roll-Back ... 1545
9.3.47 RFC 30028836 - Management UI Enable Azure AD SAML - In hours ... 1546
9.3.47.1 Tickets ... 1546
9.3.47.2 Prepared Release ... 1546
9.3.47.3 Release Process ... 1546
9.3.47.4 Roll-Back ... 1546
9.3.48 RFC 30028855 - APIM Covid-19 Test Results API - Enabling Token Exchange in the Covid Pass APIGEE app configuration ... 1547
9.3.48.1 Tickets ... 1547
9.3.48.2 Prepared Release ... 1547
9.3.48.3 Release Process ... 1547
9.3.48.4 Roll-Back ... 1548
9.3.49 RFC 30028932 - Digital Onboarding Service fixes ... 1548
9.3.49.1 Tickets ... 1548

9.3.49.2 Prepared Release 1548
9.3.49.3 Release Process 1548
9.3.49.4 Roll-Back 1549
9.3.50 RFC 30028975 - APIM Covid-19 Test Results API - Remove legacy product from Covid Pass APIGEE app configuration 1549
9.3.50.1 Description 1549
9.3.50.2 Prepared Release 1549
9.3.50.3 Release Process 1549
9.3.50.4 Roll-Back 1550
9.3.51 RFC 30029047 - APIM NHS App API - Standard Release (first since 22 January 2021) 1550
9.3.51.1 Description 1550
9.3.51.2 Prepared Release 1552
9.3.51.3 Release Process 1552
9.3.51.4 Roll-Back 1552
9.3.52 RFC 30029377 - APIM Infrastructure Changes 1553
9.3.52.1 Tickets 1553
9.3.52.2 Prepared Release 1553
9.3.52.3 Release Process 1553
9.3.52.4 Roll-Back 1554
9.3.53 RFC 30029381 - SDS API forward reliable and forward express address fix and new actor mappings 1554
9.3.53.1 Tickets 1554
9.3.53.2 Prepared Release 1554
9.3.53.3 Release Process 1554
9.3.53.4 Roll-Back 1555
9.3.54 RFC 30029451 - SDS API new query parameters 1555
9.3.54.1 Tickets 1555
9.3.54.2 Prepared Release 1555
9.3.54.3 Release Process 1555

9.3.54.4 Roll-Back 1556
9.3.55 RFC 30029482 - Set ApplyRateLimiting Quota policies to rollingwindow 1556
9.3.55.1 Tickets 1556
9.3.56 RFC 30029491 - APIM Infrastructure Changes 1556
9.3.56.1 Tickets 1556
9.3.56.2 Prepared Release 1556
9.3.56.3 Release Process 1557
9.3.56.4 Roll-Back 1557
9.3.57 RFC 30029506: APIM Ambulance Data Submissions API Updates - v1.0.376-alpha 1557
9.3.57.1 Status of tickets to be deployed 1557
9.3.57.2 RFC Change Request Details 1558
9.3.57.3 Manual Changes to be applied 1558
9.3.57.4 Dependencies 1558
9.3.57.5 Release Plan Summary 1558
9.3.58 RFC 30029519 - EPS Primary care prescribing technical go live 1558
9.3.58.1 Tickets 1558
9.3.58.2 Prepared Release 1564
9.3.58.3 Release Process 1564
9.3.58.4 Roll-Back 1564
9.3.59 RFC 30029596: APIM Infrastructure Changes 1564
9.3.59.1 Tickets 1564
9.3.59.2 Release Process 1565
9.3.59.3 Post Release Checks 1565
9.3.59.4 Roll-Back 1565
9.3.60 RFC 30029702 - Digital Onboarding Service 1565
9.3.60.1 Tickets 1565
9.3.60.2 Prepared Release 1566
9.3.60.3 Release Process 1566

9.3.60.4 Post Release Checks 1566
9.3.60.5 Roll-Back 1566
9.3.61 RFC 30029855: Auth Services Changes 1566
9.3.61.1 Tickets 1566
9.3.61.2 Release Process 1567
9.3.61.3 Post Release Checks 1567
9.3.61.4 Roll-Back 1567
9.3.62 RFC 30029871: APIM Ambulance Data Submissions API Updates - v1.0.387-alpha 1567
9.3.62.1 Status of tickets to be deployed 1568
9.3.62.2 RFC Change Request Details 1568
9.3.62.3 Manual Changes to be applied 1568
9.3.62.4 Dependencies 1568
9.3.62.5 Release Plan Summary 1568
9.3.62.6 Rollback 1569
9.3.63 RFC 30029904 - Digital Onboarding Service 1569
9.3.63.1 Tickets 1569
9.3.63.2 Prepared Release 1569
9.3.63.3 Release Process 1569
9.3.63.4 Post Release Checks 1569
9.3.63.5 Roll-Back 1570
9.3.64 RFC 30029940: APIM Infrastructure Release - Fix ApplyRateLimiting shared flow priority order of arguments 1570
9.3.64.1 Tickets 1570
9.3.64.2 Release Process 1570
9.3.64.3 Post Release Checks 1571
9.3.64.4 Roll-Back 1571
9.3.65 RFC 30029982- APIM NHS App API - Apply per-application rate limits 1571
9.3.65.1 Description 1571

9.3.65.2 Prepared Release 1571
9.3.65.3 Release Process 1571
9.3.65.4 Roll-Back 1572
9.3.66 RFC 30030097 - Digital Onboarding Service 1572
9.3.66.1 Tickets 1572
9.3.66.2 Prepared Release 1572
9.3.66.3 Release Process 1572
9.3.66.4 Post Release Checks 1573
9.3.66.5 Roll-Back 1573
9.3.67 RFC 30030118 - NHS e-RS APIM Release R1.3 Production Deployment 1573
9.3.67.1 Tickets 1573
9.3.67.2 Prepared Release 1573
9.3.67.3 Release Process 1573
9.3.67.4 Roll-Back 1574
9.3.68 RFC 30030152 - EPS bug fix 1574
9.3.68.1 Tickets 1574
9.3.68.2 Prepared Release 1574
9.3.68.3 Release Process 1574
9.3.68.4 Roll-Back 1575
9.3.69 RFC 30030217: Auth Services Changes - backchannel logout 1575
9.3.69.1 Tickets 1575
9.3.69.2 Release Process 1575
9.3.69.3 Post Release Checks 1576
9.3.69.4 Roll-Back 1576
9.3.70 RFC 30030299 - Digital Onboarding Service 1576
9.3.70.1 Tickets 1576
9.3.70.2 Prepared Release 1576
9.3.70.3 Release Process 1576

9.3.70.4 Roll-Back ... 1577
9.3.71 RFC 30030374 - Identity Service ... 1577
9.3.71.1 Tickets ... 1577
9.3.71.2 Prepared Release ... 1577
9.3.71.3 Release Process ... 1577
9.3.71.4 Roll-Back ... 1578
9.3.72 RFC 30030383 - Digital Onboarding Service ... 1578
9.3.72.1 Tickets ... 1578
9.3.72.2 Prepared Release ... 1578
9.3.72.3 Release Process ... 1578
9.3.72.4 Roll-Back ... 1579
9.3.73 RFC 30030422 - SDS API new query params + mapping change ... 1579
9.3.73.1 Tickets ... 1579
9.3.73.2 Prepared Release ... 1580
9.3.73.3 Release Process ... 1580
9.3.73.4 Roll-Back ... 1580
9.3.74 RFC 30030423 - NHS e-RS APIM Release R1.3 Production Deployment ... 1580
9.3.74.1 Tickets ... 1580
9.3.74.2 Prepared Release ... 1581
9.3.74.3 Release Process ... 1581
9.3.74.4 Roll-Back ... 1581
9.3.75 RFC 30030433 - Digital Onboarding Service ... 1582
9.3.75.1 Tickets ... 1582
9.3.75.2 Prepared Release ... 1582
9.3.75.3 Release Process ... 1582
9.3.75.4 Roll-Back ... 1582
9.3.76 RFC 30030479 - NHS e-RS APIM Release R1.4 Production Deployment ... 1583
9.3.76.1 Tickets ... 1583

9.3.76.2 Prepared Release 1583
9.3.76.3 Release Process 1583
9.3.76.4 Roll-Back 1584
9.3.77 RFC 30030574 - Alignment of infrastructure version in Production 1584
9.3.77.1 Tickets 1584
9.3.77.2 Prepared Release 1584
9.3.77.3 Release Process 1585
9.3.77.4 Roll-Back 1585
9.3.78 RFC 30030787: APIM Infrastructure Changes 1585
9.3.78.1 Tickets 1585
9.3.78.2 Release Process 1585
9.3.78.3 Post Release Checks 1586
9.3.78.4 Roll-Back 1586
9.3.79 RFC 30030831: APIM Infrastructure Changes 1586
9.3.79.1 Tickets 1586
9.3.79.2 Release Process 1587
9.3.79.3 Post Release Checks 1587
9.3.79.4 Roll-Back 1587
9.3.80 RFC 30030852 - Identity Service 1588
9.3.80.1 NOTE - Must be done after successful release of Infrastructure (planned for 11th Jan) - RFC 30030831: APIM Infrastructure Changes 1588
9.3.80.2 Tickets 1588
9.3.80.3 Prepared Release 1588
9.3.80.4 Release Process 1588
9.3.80.5 Roll-Back 1589
9.3.81 RFC 30030933 - PDS Add ASID and Reinstate ASID-Checking in Prod 1589
9.3.81.1 Tickets 1589
9.3.81.2 Prepared Release 1589
9.3.81.3 Release Process 1589

9.3.81.4 Roll-Back ... 1590
9.3.82 RFC 30030990: APIM Infrastructure Changes ... 1590
9.3.82.1 Tickets ... 1590
9.3.82.2 Release Process ... 1590
9.3.82.3 Post Release Checks ... 1591
9.3.82.4 Roll-Back ... 1591
9.3.83 RFC 30031030: APIM Infrastructure Changes ... 1591
9.3.83.1 Tickets ... 1591
9.3.83.2 Release Process ... 1591
9.3.83.3 Post Release Checks ... 1592
9.3.83.4 Roll-Back ... 1592
9.3.84 RFC 30031068: APIM Infrastructure Updates ... 1592
9.3.84.1 Tickets ... 1592
9.3.84.2 Prepared Release ... 1592
9.3.84.3 Release Process ... 1592
9.3.85 RFC 30031092 - NHS e-RS APIM Release R1.5 Production Deployment ... 1593
9.3.85.1 Tickets ... 1593
9.3.85.2 Prepared Release ... 1593
9.3.85.3 Release Process ... 1593
9.3.85.4 Roll-Back ... 1594
9.3.86 RFC 30031095 - Digital Onboarding Service/Developer Identity SSO ... 1594
9.3.86.1 Tickets ... 1594
9.3.86.2 Prepared Release ... 1594
9.3.86.3 Release Process ... 1594
9.3.86.4 Roll-Back ... 1595
9.3.87 RFC 30031265 Digital Onboarding Service ... 1595
9.3.87.1 Tickets ... 1595
9.3.87.2 Prepared Release ... 1595

9.3.87.3 Release Process 1595
9.3.87.4 Roll-Back 1596
9.3.88 RFC 30031345 - Add Developer Identity SSO to Production Apigee Portal 1596
9.3.88.1 Tickets 1596
9.3.88.2 Prepared Release 1596
9.3.88.3 Roll-Back 1596
9.3.89 RFC 30031392 - Digital Onboarding Service 1597
9.3.89.1 Tickets 1597
9.3.89.2 Prepared Release 1597
9.3.89.3 Release Process 1597
9.3.89.4 Roll-Back 1597
9.3.90 RFC 30031501 - API Management Infrastructure: Switch to High Bandwith HSCN connection 1598
9.3.90.1 Tickets 1598
9.3.90.2 Release Process 1598
9.3.90.3 Post Release Checks 1598
9.3.90.4 Roll-Back 1598
9.3.91 RFC 30031595 - APIM Infrastructure Changes 1599
9.3.91.1 Tickets 1599
9.3.91.2 Prepared Release 1599
9.3.91.3 Release Process 1599
9.3.91.4 Roll-Back 1599
9.3.92 RFC 30031685: APIM Infrastructure Changes 1600
9.3.92.1 Tickets 1600
9.3.92.2 Release Process 1600
9.3.92.3 Post Release Checks 1600
9.3.92.4 Roll-Back 1601
9.3.93 RFC 30030425: APIM Infrastructure - Add ERS Production Target Server 1601

9.3.93.1 Tickets 1601
9.3.93.2 Prepared Release 1601
9.3.93.3 Release Process 1601
9.3.93.4 Roll-Back 1601
9.3.94 RFC-30028297 Migrate Immunisation History Client Apps -> User Restricted 1602
9.3.94.1 Advise RFC starting in slack 1602
9.3.94.2 Check traffic on Immunisation History, make everything looks ok. 1602
9.3.94.3 Make changes in APIgee 1602
9.3.94.4 Observe API traffic 1603
9.3.94.5 Update Live Access Logbook 1603
9.3.94.6 Notify RFC Complete In Slack 1603
9.3.95 RFC - APIM Infrastructure Updates - Feature Flag 1603
9.3.95.1 Status of tickets to be deployed 1604
9.3.95.2 RFC Change Request Details 1604
9.3.95.3 Manual Changes to be applied (Pre-Deploy) 1604
9.3.95.4 Deployment Process 1604
9.3.95.5 Dependencies 1604
9.3.95.6 Manual Changes to be applied (Post-Deploy) 1605
9.3.95.7 Post-Deployment Checks 1605
9.3.96 RFC TBD - Digital Onboarding Service 1605
9.3.96.1 Tickets 1605
9.3.96.2 Prepared Release 1605
9.3.96.3 Release Process 1605
9.3.96.4 Roll-Back 1605
9.4 RFC 30031886: APIM Infrastructure Changes 1606
9.4.1 Tickets 1606
9.4.2 Release Process 1606
9.4.3 Post Release Checks 1606

9.4.4 Roll-Back ... 1607
9.5 RFC 30031983: APIM Infrastructure Changes ... 1607
9.5.1 Tickets ... 1607
9.5.2 Release Process ... 1608
9.5.3 Post Release Checks ... 1608
9.5.4 Roll-Back ... 1608
9.6 RFC 30032025: APIM Infrastructure Changes ... 1608
9.6.1 Tickets ... 1608
9.6.2 Prepared Release ... 1609
9.6.3 Release Process ... 1609
9.6.3.1 Post Release Checks ... 1609
9.6.4 Roll-Back ... 1610
9.7 RFC 30031802 - NHS e-RS APIM Release R1.6 Production Deployment ... 1610
9.7.1 Tickets ... 1610
9.7.2 Prepared Release ... 1611
9.7.3 Release Process ... 1611
9.7.3.1 Post Release Checks ... 1611
9.7.4 Roll-Back ... 1611
9.8 RFC 30032113 - APIM Infrastructure Changes ... 1611
9.8.1 Tickets ... 1611
9.8.2 Release Process ... 1612
9.8.2.1 Post Release Checks ... 1612
9.8.3 Roll-Back ... 1612
9.9 RFC 30032151 - NHS e-RS APIM Release R1.6.4 hotfix Production Deployment ... 1612
9.9.1 Tickets ... 1612
9.9.2 Prepared Release ... 1613
9.9.3 Release Process ... 1613
9.9.3.1 Post Release Checks ... 1613

9.9.4 Roll-Back ... 1613
9.10 RFC 30032161: APIM Infrastructure Changes ... 1613
9.10.1 Tickets ... 1613
9.10.2 Release Process ... 1614
9.10.3 Post Release Checks ... 1614
9.10.4 Roll-Back ... 1614
9.11 RFC 30032243: Digital Onboarding Service - My applications page ... 1614
9.11.1 Tickets ... 1614
9.11.2 Prepared Release ... 1615
9.11.3 Release Process ... 1615
9.11.3.1 Post Release Checks ... 1616
9.11.4 Roll-Back ... 1616
9.12 RFC 30032103 - NHS e-RS APIM Release R1.7 Production Deployment ... 1616
9.12.1 Tickets ... 1616
9.12.2 Prepared Release ... 1616
9.12.3 Release Process ... 1617
9.12.3.1 Post Release Checks ... 1617
9.12.4 Roll-Back ... 1617
9.13 RFC 30032221 - APIM Infrastructure Changes (DOS SSM Parameter) ... 1617
9.13.1 Tickets ... 1617
9.13.2 Prepared Release ... 1617
9.13.3 Release Process ... 1618
9.13.3.1 Post Release Checks ... 1618
9.13.4 Roll-Back ... 1618
9.14 RFC 30032366 - APIM Infrastructure Change ... 1618
9.14.1 Tickets ... 1618
9.14.2 Prepared Release ... 1619
9.14.3 Release Process ... 1619

9.14.3.1 Post Release Checks 1619
9.14.4 Roll-Back 1619
9.15 RFC 30032406 - APIM Infrastructure Change 1619
9.15.1 Tickets 1619
9.15.2 Prepared Release 1620
9.15.3 Release Process 1620
9.15.3.1 Post Release Checks 1620
9.15.4 Roll-Back 1620
9.16 RFC 30032474 - APIM Infrastructure Update 1620
9.16.1 Tickets 1620
9.16.2 Prepared Release 1620
9.16.3 Release Process 1621
9.16.3.1 Post Release Checks 1621
9.16.4 Roll-Back 1621
9.17 RFC 30032569 - Digital Onboarding Service 1621
9.17.1 Tickets 1621
9.17.2 Prepared Release 1622
9.17.3 Release Process 1622
9.17.3.1 Post Release Checks 1622
9.17.4 Roll-Back 1622
9.18 RFC 30032605 - NHS e-RS APIM Referral Release R1.8 Production Deployment 1622
9.18.1 Tickets 1622
9.18.2 Prepared Release 1623
9.18.3 Release Process 1623
9.18.3.1 Post Release Checks 1623
9.18.4 Roll-Back 1623
9.19 RFC 30032787 - NHS e-RS APIM Patient Care Release R1.0.5 Production Deployment 1624
9.19.1 Tickets 1624

9.19.2 Prepared Release 1624
9.19.3 Release Process 1624
9.19.3.1 Post Release Checks 1624
9.19.4 Roll-Back 1624
9.20 RFC 30032815 - Digital Onboarding Service 1625
9.20.1 Tickets 1625
9.20.2 Prepared Release 1625
9.20.3 Release Process 1625
9.20.3.1 Post Release Checks 1625
9.20.4 Roll-Back 1626
9.21 RFC 30032821: APIM Infrastructure Changes 1626
9.21.1 Tickets 1626
9.21.2 Release Process 1627
9.21.3 Post Release Checks 1627
9.21.4 Roll-Back 1627
9.22 RFC 30032765 Covid Medical Exemptions 1627
9.22.1 Tickets 1627
9.22.2 Prepared Release 1628
9.22.3 Release Procedure 1628
9.22.4 Rollback Procedure 1628
9.23 RFC 30032835- Identity Service Changes 1628
9.23.1 Tickets 1628
9.23.2 Prepared Release 1629
9.23.3 Release Process 1629
9.23.3.1 Post Release Checks 1629
9.23.4 Roll-Back 1629
9.24 RFC 30032959 - Digital Onboarding Service 1629
9.24.1 Tickets 1629
9.24.2 Prepared Release 1630

9.24.3 Release Process 1630
9.24.3.1 Post Release Checks 1630
9.24.4 Roll-Back 1630
9.25 RFC 30033031 - Digital Onboarding Service 1631
9.25.1 Tickets 1631
9.25.2 Prepared Release 1631
9.25.3 Release Process 1631
9.25.3.1 Post Release Checks 1631
9.25.4 Roll-Back 1631
9.26 RFC CHG0031039 - Digital Onboarding Service 1632
9.26.1 Tickets 1632
9.26.2 Prepared Release 1632
9.26.3 Release Process 1632
9.26.3.1 Post Release Checks 1632
9.26.4 Roll-Back 1632
9.27 RFC CHG0031189: APIM Infrastructure Changes 1633
9.27.1 Tickets 1633
9.27.2 Release Process 1633
9.27.3 Post Release Checks 1633
9.27.4 Roll-Back 1633
9.28 RFC CHG0031403: APIM Infrastructure Changes 1634
9.28.1 Tickets 1634
9.28.2 Release Process 1634
9.28.3 Post Release Checks 1634
9.28.4 Roll-Back 1635
9.29 RFC CHG0031664 - Digital Onboarding Service 1635
9.29.1 Tickets 1635
9.29.2 Prepared Release 1635
9.29.3 Release Process 1636

9.29.3.1 Post Release Checks 1636
9.29.4 Roll-Back 1636
9.30 RFC CHG0031705: APIM Infrastructure Changes 1636
9.30.1 Tickets 1636
9.30.2 Release Process 1637
9.30.3 Post Release Checks 1637
9.30.4 Roll-Back 1637
9.31 RFC CHG0031779 - Digital Onboarding Service 1637
9.31.1 Tickets 1637
9.31.2 Prepared Release 1638
9.31.3 Release Process 1638
9.31.3.1 Post Release Checks 1638
9.31.4 Roll-Back 1638
9.32 RFC 0031930 - NHS e-RS APIM Patient Care Release R1.1.6 Production Deployment 1639
9.32.1 Tickets 1639
9.32.2 Prepared Release 1639
9.32.3 Release Process 1639
9.32.3.1 Post Release Checks 1639
9.32.4 Roll-Back 1640
9.33 RFC 0031929 - NHS e-RS APIM Referral Release R1.9 Production Deployment 1640
9.33.1 Tickets 1640
9.33.2 Prepared Release 1640
9.33.3 Release Process 1640
9.33.3.1 Post Release Checks 1641
9.33.4 Roll-Back 1641
9.34 CHG0031941 - Digital Onboarding Service 1641
9.34.1 Tickets 1641
9.34.2 Prepared Release 1641

9.34.3 Release Process 1641
9.34.3.1 Post Release Checks 1642
9.34.4 Roll-Back 1642
9.35 RFC CHG0032105 - NHS App API change 1642
9.35.1 Tickets 1642
9.35.2 Prepared Release 1642
9.35.3 Release Process 1642
9.35.3.1 Post Release Checks 1643
9.35.4 Roll-Back 1643
9.36 CHG0032137 - Digital Onboarding Service 1643
9.36.1 Tickets 1643
9.36.2 Prepared Release 1643
9.36.3 Release Process 1643
9.36.3.1 Post Release Checks 1643
9.36.4 Roll-Back 1644
9.37 CHG0032180: APIM Infrastructure Changes 1644
9.37.1 Tickets 1644
9.37.2 Release Process 1645
9.37.3 Post Release Checks 1645
9.37.4 Roll-Back 1645
9.38 CHG0032379: APIM Infrastructure Changes 1645
9.38.1 Tickets 1645
9.38.2 Release Process 1646
9.38.3 Post Release Checks 1646
9.38.4 Roll-Back 1646
9.39 CHG0032484: Digital Onboarding Service 1646
9.39.1 Tickets 1646
9.39.2 Release Process 1647
9.39.3 Post Release Checks 1647

9.39.4 Roll-Back 1647
9.40 CHG0032533: APIM Infrastructure Changes 1647
9.40.1 Tickets 1647
9.40.2 Release Process 1647
9.40.2.1 Post Release Checks 1648
9.40.3 Roll-Back 1648
9.41 CHG0032690 Immunisation History API 1648
9.41.1 Tickets 1648
9.41.2 Prepared Release 1648
9.41.3 Release Process 1649
9.41.4 Roll-Back 1649
9.42 CHG0032731: Digital Onboarding Service 1649
9.42.1 Tickets 1649
9.42.2 Release Process 1650
9.42.3 Post Release Checks 1650
9.42.4 Roll-Back 1650
9.43 CHG0032859 - APIM Infrastructure Updates 1650
9.43.1 Tickets 1650
9.43.2 Prepared Release 1651
9.43.3 Release Process 1651
9.43.3.1 Post Release Checks 1651
9.43.4 Roll-Back 1651
9.44 RFC 0032744 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment 1651
9.44.1 Tickets 1651
9.44.2 Prepared Release 1653
9.44.3 Release Process 1653
9.44.3.1 Post Release Checks 1653
9.44.4 Roll-Back 1653

9.45 RFC 0032733 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment 1654
9.45.1 Tickets 1654
9.45.2 Prepared Release 1654
9.45.3 Release Process 1654
9.45.3.1 Post Release Checks 1655
9.45.4 Roll-Back 1655
9.46 CHG0033065 - APIM Infrastructure Updates 1655
9.46.1 Tickets 1655
9.46.2 Prepared Release 1655
9.46.3 Release Process 1655
9.46.3.1 Post Release Checks 1656
9.46.4 Roll-Back 1656
9.47 CHG0033227 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment 1656
9.47.1 Tickets 1656
9.47.2 Prepared Release 1657
9.47.3 Release Process 1658
9.47.3.1 Post Release Checks 1658
9.47.4 Roll-Back 1658
9.48 CHG0033228 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment 1658
9.48.1 Tickets 1658
9.48.2 Prepared Release 1659
9.48.3 Release Process 1659
9.48.3.1 Post Release Checks 1659
9.48.4 Roll-Back 1659
9.49 CHG0033306 - Digital Onboarding Service 1660
9.49.1 Tickets 1660
9.49.2 Prepared Release 1660

9.49.3 Release Process 1660
9.49.3.1 Post Release Checks 1660
9.49.4 Roll-Back 1661
9.50 CHG0033524 - Digital Onboarding Service 1661
9.50.1 Tickets 1661
9.50.2 Prepared Release 1661
9.50.3 Release Process 1661
9.50.3.1 Post Release Checks 1662
9.50.4 Roll-Back 1662
9.51 CHG0033763 - Digital Onboarding Service 1662
9.51.1 Tickets 1662
9.51.2 Prepared Release 1662
9.51.3 Release Process 1662
9.51.3.1 Post Release Checks 1663
9.51.4 Roll-Back 1663
9.52 CHG0033871 Sync-wrap API changes 1663
9.52.1 Tickets 1663
9.52.2 Prepared Release 1663
9.52.3 Release Process 1663
9.52.3.1 Post Release Checks 1664
9.52.4 Roll-Back 1664
9.53 CHG0033320 - NHS e-RS APIM Referral Release R1.11.0 Production Deployment 1664
9.53.1 Content 1664
9.53.2 Prepared Release 1665
9.53.3 Release Process 1665
9.53.3.1 Post Release Checks 1665
9.53.4 Roll-Back 1666
9.54 CHG0033321 - NHS e-RS APIM Patient Care Release R1.3.0 Production Deployment 1666

9.54.1 Content 1666
9.54.2 Prepared Release 1666
9.54.3 Release Process 1667
9.54.3.1 Post Release Checks 1667
9.54.4 Roll-Back 1667
9.55 CHG0033762 - Apigee environment attributes 1667
9.55.1 Tickets 1667
9.55.2 Prepared Release 1667
9.55.3 Release Process 1668
9.55.3.1 Post Release Checks 1668
9.55.4 Roll-Back 1668
9.56 CHG0034015 - APIM Infrastructure Updates - Add patient care aggregator prod target server 1668
9.56.1 Tickets 1668
9.56.2 Prepared Release 1669
9.56.3 Release Process 1669
9.56.3.1 Post Release Checks 1669
9.56.4 Roll-Back 1669
9.57 CHG0034157 - Digital Onboarding Service 1669
9.57.1 Tickets 1669
9.57.2 Prepared Release 1670
9.57.3 Release Process 1670
9.57.3.1 Post Release Checks 1670
9.57.4 Roll-Back 1670
9.58 CHG 0034302 - Immunisation History API 1670
9.58.1 Tickets 1670
9.58.2 Prepared Release 1671
9.58.3 Release Process 1671
9.58.3.1 Post Release Checks 1671

9.58.4 Roll-Back ... 1671
9.59 CHG0034551 - Digital Onboarding Service ... 1671
9.59.1 Tickets ... 1671
9.59.2 Prepared Release ... 1672
9.59.3 Release Process ... 1672
9.59.3.1 Post Release Checks ... 1672
9.59.4 Roll-Back ... 1672
9.60 CHG0040981 - Initial deployment of Electronic Signing Service to production ... 1673
9.60.1 Tickets ... 1673
9.60.2 Prepared Release ... 1673
9.60.3 Release Process ... 1673
9.60.3.1 Technical Steps to Deploy to Live ... 1673
9.60.3.2 Post Release Checks ... 1675
9.60.4 Roll-Back ... 1676
9.61 CHG0034532 - Initial deployment of Electronic Signing Service to production ... 1677
9.61.1 Tickets ... 1677
9.61.2 Prepared Release ... 1678
9.61.2.1 Technical Steps to Deploy to Live ... 1678
9.61.2.2 Post Release Checks ... 1680
9.61.3 Roll-Back ... 1681
9.62 CHG0034608- Identity Service Changes ... 1682
9.62.1 Tickets ... 1682
9.62.2 Prepared Release ... 1683
9.62.3 Release Process ... 1683
9.62.3.1 Post Release Checks ... 1683
9.62.4 Roll-Back ... 1683
9.63 CHG0034039 - NHS e-RS APIM Referral Release R1.12.0 Production Deployment ... 1683

9.63.1 Content 1683
9.63.2 Prepared Release 1684
9.63.3 Release Process 1684
9.63.3.1 Post Release Checks 1684
9.63.4 Roll-Back 1684
9.64 CHG0034041 - NHS e-RS APIM Patient Care Release R1.4.0 Production Deployment 1685
9.64.1 Content 1685
9.64.2 Prepared Release 1685
9.64.3 Release Process 1685
9.64.3.1 Post Release Checks 1685
9.64.4 Roll-Back 1686
9.65 CHG0034907: APIM Infrastructure Changes 1686
9.65.1 Tickets 1686
9.65.2 Release Process 1686
9.65.3 Post Release Checks 1686
9.65.4 Roll-Back 1687
9.66 CHG0033743 - GP Registration Management Information 1687
9.66.1 Tickets 1687
9.66.2 Prepared Release 1687
9.66.3 Release Process 1687
9.66.3.1 Post Release Checks 1687
9.66.4 Roll-Back 1687
9.67 CHG0035002 - Digital Onboarding Service 1688
9.67.1 Tickets 1688
9.67.2 Prepared Release 1688
9.67.3 Release Process 1688
9.67.3.1 Post Release Checks 1688
9.67.4 Roll-Back 1689

9.68 CHG0035191 - Electronic Signing Service - First of Type Readiness ... 1689
9.68.1 This will be self-released by the Signing Service team ... 1689
9.68.2 Tickets ... 1689
9.68.3 Prepared Release ... 1689
9.68.3.1 Technical Steps to Deploy to Live ... 1690
9.68.3.2 Post Release Checks ... 1691
9.68.4 Roll-Back ... 1693
9.69 CHG0035187 - Digital Onboarding Service ... 1694
9.69.1 Tickets ... 1694
9.69.2 Prepared Release ... 1695
9.69.3 Release Process ... 1695
9.69.3.1 Post Release Checks ... 1695
9.69.4 Roll-Back ... 1695
9.70 CHG0035334 - Digital Onboarding Service ... 1695
9.70.1 Tickets ... 1695
9.70.2 Prepared Release ... 1696
9.70.3 Release Process ... 1696
9.70.3.1 Post Release Checks ... 1696
9.70.4 Roll-Back ... 1696
9.71 CHG0035412: APIM Infrastructure Changes ... 1696
9.71.1 Tickets ... 1696
9.71.2 Release Process ... 1697
9.71.3 Post Release Checks ... 1697
9.71.4 Roll-Back ... 1697
9.72 CHG0035211 - NHS e-RS APIM Referral Release R1.13 Production Deployment ... 1697
9.72.1 Content ... 1697
9.72.2 Prepared Release ... 1698
9.72.3 Release Process ... 1698

9.72.3.1 Post Release Checks 1698
9.72.4 Roll-Back 1698
9.73 CHG0035210 - NHS e-RS APIM Patient Care Release R1.5 Production Deployment 1698
9.73.1 Content 1698
9.73.2 Prepared Release 1699
9.73.3 Release Process 1699
9.73.3.1 Post Release Checks 1699
9.73.4 Roll-Back 1699
9.74 CHG0035615 - Infrastructure Standard Change 1699
9.74.1 Tickets 1700
9.74.2 Prepared Release 1700
9.74.3 Release Process 1700
9.74.3.1 Post Release Checks 1700
9.74.4 Roll-Back 1700
9.75 CHG0035683: APIM Infrastructure Changes 1700
9.75.1 Tickets 1701
9.75.2 Prepared Release 1701
9.75.3 Release Process 1701
9.75.3.1 Post Release Checks 1701
9.75.4 Roll-Back 1701
9.76 CHG0035816 - Digital Onboarding Service 1701
9.76.1 Tickets 1701
9.76.2 Prepared Release 1702
9.76.3 Release Process 1702
9.76.3.1 Post Release Checks 1702
9.76.4 Roll-Back 1702
9.77 CHG0036081 - Digital Onboarding Service 1703
9.77.1 Tickets 1703

9.77.2 Prepared Release 1703
9.77.3 Release Process 1703
9.77.3.1 Post Release Checks 1703
9.77.4 Roll-Back 1704
9.78 CHG0036192- Digital Onboarding Service 1704
9.78.1 Tickets 1704
9.78.2 Prepared Release 1704
9.78.3 Release Process 1704
9.78.3.1 Post Release Checks 1704
9.78.4 Roll-Back 1705
9.79 CHG0036171 - NHS e-RS APIM Referral Release R1.14 Production Deployment 1705
9.79.1 Content 1705
9.79.2 Prepared Release 1705
9.79.3 Release Process 1705
9.79.3.1 Post Release Checks 1706
9.79.4 Roll-Back 1706
9.80 CHG0036172 - NHS e-RS APIM Patient Care Release R1.6 Production Deployment 1706
9.80.1 Content 1706
9.80.2 Prepared Release 1706
9.80.3 Release Process 1707
9.80.3.1 Post Release Checks 1707
9.80.4 Roll-Back 1707
9.81 CHG0036522 - Digital Onboarding 1707
9.81.1 Tickets 1707
9.81.2 Prepared Release 1708
9.81.3 Release Process 1708
9.81.3.1 Post Release Checks 1708
9.81.4 Roll-Back 1708

9.82 CHG0036796 - APIM Identity Service Updates 1708
9.82.1 Tickets 1708
9.82.2 Prepared Release 1709
9.82.3 Release Process 1709
9.82.3.1 Post Release Checks 1709
9.82.4 Roll-Back 1709
9.83 CHG0036917 - Digital Onboarding Service 1710
9.83.1 Tickets 1710
9.83.2 Prepared Release 1710
9.83.3 Release Process 1710
9.83.3.1 Post Release Checks 1710
9.83.4 Roll-Back 1711
9.84 CHG0036991 - NHS e-RS APIM Patient Care Release R1.7 Production Deployment 1711
9.84.1 Content 1711
9.84.2 Prepared Release 1711
9.84.3 Release Process 1711
9.84.3.1 Post Release Checks 1712
9.84.4 Roll-Back 1712
9.85 CHG0036990 - NHS e-RS APIM Referral Release R1.15 Production Deployment 1712
9.85.1 Content 1712
9.85.2 Prepared Release 1712
9.85.3 Release Process 1713
9.85.3.1 Post Release Checks 1713
9.85.4 Roll-Back 1713
9.86 CHG0037210 - APIM Identity Service Updates 1713
9.86.1 Tickets 1713
9.86.2 Prepared Release 1714
9.86.3 Release Process 1714

9.86.3.1 Post Release Checks 1714
9.86.4 Roll-Back 1714
9.87 CHG0037214: APIM Infrastructure Changes 1715
9.87.1 Tickets 1715
9.87.2 Release Process 1716
9.87.3 Post Release Checks 1716
9.87.4 Roll-Back 1716
9.88 CHG0037292 - Digital Onboarding Service 1716
9.88.1 Tickets 1716
9.88.2 Prepared Release 1717
9.88.3 Release Process 1717
9.88.3.1 Post Release Checks 1717
9.88.4 Roll-Back 1717
9.89 CHG0037404- APIM Identity Service Update 1717
9.89.1 Tickets 1717
9.89.2 Prepared Release 1718
9.89.3 Release Process 1718
9.89.3.1 Post Release Checks 1718
9.89.4 Roll-Back 1718
9.90 CHG0037410 - APIM EPS Updates 1718
9.90.1 Tickets 1719
9.90.2 Prepared Release 1721
9.90.3 Release Process 1721
9.90.3.1 Post Release Checks 1721
9.90.4 Roll-Back 1721
9.91 CHG0037450: APIM Infrastructure Changes 1721
9.91.1 Tickets 1722
9.91.2 Prepared Release 1722
9.91.3 Release Process 1722

9.91.3.1 Post Release Checks 1722
9.91.4 Roll-Back 1722
9.92 CHG0037724 - Digital Onboarding Service 1723
9.92.1 Tickets 1723
9.92.2 Prepared Release 1723
9.92.3 Release Process 1723
9.92.3.1 Post Release Checks 1724
9.92.4 Roll-Back 1724
9.93 CHG0037795: APIM Infrastructure Changes 1724
9.93.1 Tickets 1724
9.93.2 Release Process 1725
9.93.3 Post Release Checks 1725
9.93.4 Roll-Back 1725
9.94 CHG0037877 - APIM Identity Service Updates 1725
9.94.1 Tickets 1725
9.94.2 Prepared Release 1726
9.94.3 Release Process 1726
9.94.3.1 Post Release Checks 1726
9.94.4 Roll-Back 1726
9.95 CHG0037892 - Digital Onboarding Service 1726
9.95.1 Tickets 1726
9.95.2 Prepared Release 1727
9.95.3 Release Process 1727
9.95.3.1 Post Release Checks 1727
9.95.4 Roll-Back 1727
9.96 CHG0037865 - APIM EPS Updates 1727
9.96.1 Tickets 1727
9.96.2 Prepared Release 1728
9.96.3 Release Process 1728

9.96.3.1 Post Release Checks 1728
9.96.4 Roll-Back 1728
9.97 CHG0038027 - Digital Onboarding Service 1728
9.97.1 Tickets 1728
9.97.2 Prepared Release 1729
9.97.3 Release Process 1729
9.97.3.1 Post Release Checks 1729
9.97.4 Roll-Back 1729
9.98 CHG0038099 - NHS e-RS APIM Patient Care Release R1.8 Production Deployment 1729
9.98.1 Content 1729
9.98.2 Prepared Release 1730
9.98.3 Release Process 1730
9.98.3.1 Post Release Checks 1730
9.98.4 Roll-Back 1730
9.99 CHG0038100 - NHS e-RS APIM Referral Release R1.16 Production Deployment 1731
9.99.1 Content 1731
9.99.2 Prepared Release 1731
9.99.3 Release Process 1731
9.99.3.1 Post Release Checks 1731
9.99.4 Roll-Back 1732
9.100 CHG0038330: APIM Infrastructure Changes 1732
9.100.1 Tickets 1732
9.100.2 Release Process 1732
9.100.3 Post Release Checks 1732
9.100.4 Roll-Back 1733
9.101 CHG0038358: Personal Demographics Service 1733
9.101.1 Service Now CHG0038358 1733
9.101.2 Tickets 1733

9.101.3 Release Process 1733
9.101.4 Post Release Checks 1733
9.101.5 Roll-Back 1734
9.102 CHG0038470: APIM Infrastructure Changes 1734
9.102.1 Tickets 1734
9.102.2 Release Process 1734
9.102.3 Post Release Checks 1734
9.102.4 Roll-Back 1734
9.103 CHG0038571: APIM Infrastructure Changes 1735
9.103.1 Tickets 1735
9.103.2 Release Process 1735
9.103.3 Post Release Checks 1735
9.103.4 Roll-Back 1735
9.104 CHG0038936 - Digital Onboarding Service 1735
9.104.1 Tickets 1736
9.104.2 Prepared Release 1736
9.104.3 Release Process 1736
9.104.3.1 Post Release Checks 1737
9.104.4 Roll-Back 1737
9.105 CHG0038946 - Identity Service Updates 1737
9.105.1 Tickets 1737
9.105.2 Prepared Release 1737
9.105.3 Release Process 1737
9.105.3.1 Post Release Checks 1738
9.105.4 Roll-Back 1738
9.106 CHG0039474 - APIM Infrastructure Changes 1738
9.106.1 Tickets 1738
9.106.2 Prepared Release 1738
9.106.3 Release Process 1738

9.106.3.1 Post Release Checks 1739
9.106.4 Roll-Back 1739
9.107 CHG0039266 - Digital Onboarding Service 1739
9.107.1 Tickets 1739
9.107.2 Prepared Release 1739
9.107.3 Release Process 1739
9.107.3.1 Post Release Checks 1740
9.107.4 Roll-Back 1740
9.108 CHG0039377 - NHS e-RS APIM Referral Release R1.17 Production Deployment 1740
9.108.1 Content 1740
9.108.2 Prepared Release 1740
9.108.3 Release Process 1741
9.108.3.1 Post Release Checks 1741
9.108.4 Roll-Back 1741
9.109 CHG0039378 - NHS e-RS APIM Patient Care Release R1.9 Production Deployment 1741
9.109.1 Content 1741
9.109.2 Prepared Release 1742
9.109.3 Release Process 1742
9.109.3.1 Post Release Checks 1742
9.109.4 Roll-Back 1742
9.110 CHG0039499: APIM Infrastructure Changes 1742
9.110.1 Tickets 1742
9.110.2 Release Process 1743
9.110.3 Post Release Checks 1743
9.110.4 Roll-Back 1743
9.111 CHG0039788 - APIM Infrastructure Changes 1743
9.111.1 Tickets 1743
9.111.2 Prepared Release 1744

9.111.3 Release Process 1744
9.111.3.1 Post Release Checks 1744
9.111.4 Roll-Back 1744
9.112 CHG0039901 - APIM Infrastructure changes 1744
9.112.1 Tickets 1744
9.112.2 Prepared Release 1745
9.112.3 Release Process 1745
9.112.3.1 Post Release Checks 1745
9.112.4 Roll-Back 1745
9.113 CHG0040020 - Digital Onboarding Service 1745
9.113.1 Tickets 1746
9.113.2 Prepared Release 1746
9.113.3 Release Process 1746
9.113.3.1 Post Release Checks 1746
9.113.4 Roll-Back 1746
9.114 CHG0040127 - APIM Infrastructure Updates 1747
9.114.1 Tickets 1747
9.114.2 Prepared Release 1747
9.114.3 Release Process 1747
9.114.3.1 Post Release Checks 1747
9.114.4 Roll-Back 1747
9.115 CHG0040159 - APIM EPS Updates 1748
9.115.1 Tickets 1748
9.115.2 Prepared Release 1749
9.115.3 Release Process 1749
9.115.3.1 Post Release Checks 1749
9.115.4 Roll-Back 1749
9.116 CHG0040263 - APIM Infrastructure Updates 1750
9.116.1 Tickets 1750

9.116.2 Prepared Release 1750
9.116.3 Release Process 1750
9.116.3.1 Post Release Checks 1750
9.116.4 Roll-Back 1750
9.117 CHG0040295 - Digital Onboarding Service 1751
9.117.1 Tickets 1751
9.117.2 Prepared Release 1751
9.117.3 Release Process 1751
9.117.3.1 Post Release Checks 1751
9.117.4 Roll-Back 1751
9.118 CHG0040638 - Digital Onboarding Service 1752
9.118.1 Tickets 1752
9.118.2 Prepared Release 1752
9.118.3 Release Process 1752
9.118.3.1 Post Release Checks 1753
9.118.4 Roll-Back 1753
9.119 CHG0040876 - APIM Infrastructure Updates 1753
9.119.1 Tickets 1753
9.119.2 Prepared Release 1753
9.119.3 Release Process 1753
9.119.3.1 Post Release Checks 1754
9.119.4 Roll-Back 1754
9.120 CHG0040978 - APIM EPS Updates 1754
9.120.1 Tickets 1754
9.120.2 Prepared Release 1754
9.120.3 Release Process 1754
9.120.3.1 Post Release Checks 1755
9.120.4 Roll-Back 1755
9.121 CHG0041013 - APIM Infrastructure Updates 1755

9.121.1 Tickets 1755
9.121.2 Prepared Release 1755
9.121.3 Release Process 1755
9.121.3.1 Post Release Checks 1756
9.121.4 Roll-Back 1756
9.122 CHG0041198 - Identity Service Updates 1756
9.122.1 Tickets 1756
9.122.2 Prepared Release 1756
9.122.3 Release Process 1757
9.122.3.1 Post Release Checks 1757
9.122.4 Roll-Back 1757
9.123 CHG0041210 - APIM Infrastructure Updates 1757
9.123.1 Tickets 1757
9.123.2 Prepared Release 1757
9.123.3 Release Process 1758
9.123.3.1 Post Release Checks 1758
9.123.4 Roll-Back 1758
9.124 CHG0041275 - APIM Infrastructure updates 1758
9.124.1 Tickets 1758
9.124.2 Prepared Release 1758
9.124.3 Release Process 1759
9.124.3.1 Post Release Checks 1759
9.124.4 Roll-Back 1759
9.125 CHG0041319 - APIM Infrastructure updates 1759
9.125.1 Tickets 1759
9.125.2 Prepared Release 1759
9.125.3 Release Process 1760
9.125.3.1 Post Release Checks 1760
9.125.4 Roll-Back 1760

9.126 CHG0041490 - APIM EPS Updates 1760
9.126.1 Tickets 1760
9.126.2 Prepared Release 1761
9.126.3 Release Process 1761
9.126.3.1 Post Release Checks 1761
9.126.4 Roll-Back 1761
9.127 CHG0041498 - APIM Infrastructure Updates 1761
9.127.1 Tickets 1761
9.127.2 Prepared Release 1762
9.127.3 Release Process 1762
9.127.3.1 Post Release Checks 1762
9.127.4 Roll-Back 1762
9.128 CHG0041507 - NHS e-RS APIM Patient Care Release R1.10 Production Deployment 1762
9.128.1 Content 1762
9.128.2 Prepared Release 1763
9.128.3 Release Process 1763
9.128.3.1 Post Release Checks 1763
9.128.4 Roll-Back 1763
9.129 CHG0041499 - NHS e-RS APIM Referral Release R1.18 Production Deployment 1763
9.129.1 Content 1763
9.129.2 Prepared Release 1764
9.129.3 Release Process 1764
9.129.3.1 Post Release Checks 1764
9.129.4 Roll-Back 1764
9.130 CHG0041524 - Digital Onboarding Service 1764
9.130.1 Tickets 1765
9.130.2 Prepared Release 1765
9.130.3 Release Process 1765

9.130.3.1 Post Release Checks 1765
9.130.4 Roll-Back 1765
9.131 CHG0041597 - APIM Infrastructure Updates 1766
9.131.1 Tickets 1766
9.131.2 Prepared Release 1766
9.131.3 Release Process 1766
9.131.3.1 Post Release Checks 1766
9.131.4 Roll-Back 1767
9.132 CHG0041622 - APIM Infrastructure Releases 1767
9.132.1 Tickets 1767
9.132.2 Prepared Release 1767
9.132.3 Release Process 1767
9.132.3.1 Post Release Checks 1768
9.132.4 Roll-Back 1768
9.133 CHG0042084 - APIM Infrastructure Updates 1768
9.133.1 Tickets 1768
9.133.2 Prepared Release 1768
9.133.3 Release Process 1768
9.133.3.1 Post Release Checks 1769
9.133.4 Roll-Back 1769
9.134 CHG0042126 - APIM EPS Updates 1769
9.134.1 Tickets 1769
9.134.2 Prepared Release 1769
9.134.3 Release Process 1769
9.134.3.1 Post Release Checks 1770
9.134.4 Roll-Back 1770
9.135 CHG0042198 - Digital Onboarding Content release for DAQ 11.2 Production 1770
9.135.1 Tickets 1770

9.135.2 Prepared Release 1770
9.135.3 Release Process 1770
9.135.3.1 Post Release Checks 1771
9.135.4 Roll-Back 1771
9.136 CHG0042217 - APIM Infrastructure updates 1771
9.136.1 Tickets 1771
9.136.2 Prepared Release 1771
9.136.3 Release Process 1771
9.136.3.1 Post Release Checks 1772
9.136.4 Roll-Back 1772
9.137 CHG0042220 - Digital Onboarding Service 1772
9.137.1 Tickets 1772
9.137.2 Prepared Release 1773
9.137.3 Release Process 1773
9.137.3.1 Post Release Checks 1773
9.137.4 Roll-Back 1773
9.138 CHG0042299 - APIM Infrastructure updates 1773
9.138.1 Tickets 1773
9.138.2 Prepared Release 1774
9.138.3 Release Process 1774
9.138.3.1 Post Release Checks 1774
9.138.4 Roll-Back 1774
9.139 CHG0042547 - NHS e-RS APIM Referral Release R1.19 Production Deployment 1775
9.139.1 Content 1775
9.139.2 Prepared Release 1775
9.139.3 Release Process 1775
9.139.3.1 Post Release Checks 1775
9.139.4 Roll-Back 1776

9.140 CHG0042546 - NHS e-RS APIM Patient Care Release R1.11 Production Deployment 1776
9.140.1 Content 1776
9.140.2 Prepared Release 1776
9.140.3 Release Process 1776
9.140.3.1 Post Release Checks 1777
9.140.4 Roll-Back 1777
9.141 CHG0042663 - APIM Infrastructure Updates 1777
9.141.1 Tickets 1777
9.141.2 Prepared Release 1777
9.141.3 Release Process 1777
9.141.3.1 Post Release Checks 1778
9.141.4 Roll-Back 1778
9.142 CHG0042712- Digital Onboarding Service 1778
9.142.1 Tickets 1778
9.142.2 Prepared Release 1778
9.142.3 Release Process 1778
9.142.3.1 Post Release Checks 1779
9.142.4 Roll-Back 1779
9.143 CHG0042721 - Digital Onboarding Content release for DAQ 11.3 Production 1779
9.143.1 Tickets 1779
9.143.2 Prepared Release 1779
9.143.3 Release Process 1780
9.143.3.1 Post Release Checks 1780
9.143.4 Roll-Back 1780
9.144 CHG0042720 - APIM Infrastructure Updates 1780
9.144.1 Tickets 1780
9.144.2 Prepared Release 1780
9.144.3 Release Process 1780

9.144.3.1 Post Release Checks 1781
9.144.4 Roll-Back 1781
9.145 CHG0042725 - APIM Infrastructure Updates 1781
9.145.1 Tickets 1781
9.145.2 Prepared Release 1781
9.145.3 Release Process 1782
9.145.3.1 Post Release Checks 1782
9.145.4 Roll-Back 1782
9.146 CHG0042923 - Digital Onboarding Service 1782
9.146.1 Tickets 1782
9.146.2 Prepared Release 1782
9.146.3 Release Process 1783
9.146.3.1 Post Release Checks 1783
9.146.4 Roll-Back 1783
9.147 CHG0043067 - Digital Onboarding Content release for DAQ 11.4 Production 1783
9.147.1 Tickets 1783
9.147.2 Prepared Release 1784
9.147.3 Release Process 1784
9.147.3.1 Post Release Checks 1784
9.147.4 Roll-Back 1784
9.148 CHG0043151 - APIM EPS Updates 1784
9.148.1 Tickets 1784
9.148.2 Prepared Release 1784
9.148.3 Release Process 1785
9.148.3.1 Post Release Checks 1785
9.148.4 Roll-Back 1785
9.149 CHG0043556 - APIM Infrastructure Updates 1785
9.149.1 Tickets 1785

9.149.2 Prepared Release 1785
9.149.3 Release Process 1786
9.149.3.1 Post Release Checks 1786
9.149.4 Roll-Back 1786
9.150 CHG0043595 - Digital Onboarding Service 1786
9.150.1 Tickets 1786
9.150.2 Prepared Release 1787
9.150.3 Release Process 1787
9.150.3.1 Post Release Checks 1787
9.150.4 Roll-Back 1787
9.151 CHG0043908 - APIM Infrastructure Updates 1787
9.151.1 Prepared Release 1788
9.151.2 Release Process 1788
9.151.2.1 Post Release Checks 1788
9.151.3 Roll-Back 1788
9.152 CHG0043949- APIM EPS Updates 1788
9.152.1 Tickets 1788
9.152.2 Prepared Release 1789
9.152.3 Release Process 1789
9.152.3.1 Post Release Checks 1789
9.152.4 Roll-Back 1789
9.153 CHG0043967 - Digital Onboarding Service 1789
9.153.1 Tickets 1789
9.153.2 Prepared Release 1790
9.153.3 Release Process 1790
9.153.3.1 Post Release Checks 1790
9.153.4 Roll-Back 1790
9.154 CHG0044062 - NHS e-RS APIM Patient Care Release R1.12 Production Deployment 1790

9.154.1 Content 1790
9.154.2 Prepared Release 1791
9.154.3 Release Process 1791
9.154.3.1 Post Release Checks 1791
9.154.4 Roll-Back 1791
9.155 CHG0044059 - NHS e-RS APIM Referral Release R1.20 Production Deployment 1791
9.155.1 Content 1791
9.155.2 Prepared Release 1792
9.155.3 Release Process 1792
9.155.3.1 Post Release Checks 1792
9.155.4 Roll-Back 1792
9.156 CHG0044171 - APIM Infrastructure Updates 1793
9.156.1 Prepared Release 1793
9.156.2 Release Process 1793
9.156.2.1 Post Release Checks 1794
9.156.3 Roll-Back 1794
9.157 CHG0044202 - Digital Onboarding Content release for DAQ 11.5 Production 1794
9.157.1 Tickets 1794
9.157.2 Prepared Release 1794
9.157.3 Release Process 1794
9.157.3.1 Post Release Checks 1795
9.157.4 Roll-Back 1795
9.158 CHG0044409 - APIM EPS Updates 1795
9.158.1 Tickets 1795
9.158.2 Prepared Release 1795
9.158.3 Release Process 1795
9.158.3.1 Post Release Checks 1795
9.158.4 Roll-Back 1796

9.159 CHG0044433 - Digital Onboarding Service 1796
9.159.1 Tickets 1796
9.159.2 Prepared Release 1796
9.159.3 Release Process 1796
9.159.3.1 Post Release Checks 1797
9.159.4 Roll-Back 1797
9.160 CHG0044790 - APIM Infrastructure Updates 1797
9.160.1 Tickets 1797
9.160.2 Prepared Release 1797
9.160.3 Release Process 1797
9.160.3.1 Post Release Checks 1798
9.160.4 Roll-Back 1798
9.161 CHG0044977 - APIM EPS Updates 1798
9.161.1 Tickets 1798
9.161.2 Prepared Release 1798
9.161.3 Release Process 1798
9.161.3.1 Post Release Checks 1799
9.161.4 Roll-Back 1799
9.162 CHG0045027 - NHS e-RS APIM Patient Care Release R1.13 Production Deployment 1799
9.162.1 Content 1799
9.162.2 Prepared Release 1799
9.162.3 Release Process 1800
9.162.3.1 Post Release Checks 1800
9.162.4 Roll-Back 1800
9.163 CHG0045025 - NHS e-RS APIM Referral Release R1.21 Production Deployment 1800
9.163.1 Content 1800
9.163.2 Prepared Release 1801
9.163.3 Release Process 1801

9.163.3.1 Post Release Checks 1801
9.163.4 Roll-Back 1801
9.164 CHG0045116 - APIM Infrastructure Updates 1801
9.164.1 Tickets 1802
9.164.2 Prepared Release 1802
9.164.3 Release Process 1802
9.164.3.1 Post Release Checks 1802
9.164.4 Roll-Back 1802
9.165 CHG0045315 - APIM EPS Updates 1802
9.165.1 Tickets 1803
9.165.2 Prepared Release 1803
9.165.3 Release Process 1803
9.165.3.1 Post Release Checks 1803
9.165.4 Roll-Back 1803
9.166 CHG0045108 - Digital Onboarding Service 1803
9.166.1 Tickets 1803
9.166.2 Prepared Release 1804
9.166.3 Release Process 1804
9.166.3.1 Post Release Checks 1804
9.166.4 Roll-Back 1804
9.167 CHG0039918 - APIM - NHS App Updates for Comms Hub: 1804
9.167.1 Tickets 1804
9.167.2 Prepared Release 1805
9.167.3 Release Process 1805
9.167.3.1 Post Release Checks 1806
9.167.4 Roll-Back 1806
9.168 CHG0045519 - APIM EPS Updates 1806
9.168.1 Tickets 1806
9.168.2 Prepared Release 1806

9.168.3 Release Process 1806
9.168.3.1 Post Release Checks 1807
9.168.4 Roll-Back 1807
9.169 CHG0045620- APIM EPS Updates 1807
9.169.1 Tickets 1807
9.169.2 Prepared Release 1807
9.169.3 Release Process 1807
9.169.3.1 Post Release Checks 1808
9.169.4 Roll-Back 1808
9.170 CHG0045708 - APIM Infrastructure Updates 1808
9.170.1 Tickets 1808
9.170.2 Prepared Release 1808
9.170.3 Release Process 1808
9.170.3.1 Post Release Checks 1809
9.170.4 Roll-Back 1809
9.171 CHG0045726 - Digital Onboarding Service 1809
9.171.1 Tickets 1809
9.171.2 Prepared Release 1809
9.171.3 Release Process 1810
9.171.3.1 Post Release Checks 1810
9.171.4 Roll-Back 1810
9.172 CHG0045741 - APIM Infrastructure Updates 1810
9.172.1 Tickets 1810
9.172.2 Prepared Release 1810
9.172.3 Release Process 1811
9.172.3.1 Post Release Checks 1811
9.172.4 Roll-Back 1811
9.173 CHG0045793 - APIM EPS Updates 1811
9.173.1 Tickets 1811

9.173.2 Prepared Release 1811
9.173.3 Release Process 1812
9.173.3.1 Post Release Checks 1812
9.173.4 Roll-Back 1812
9.174 CHG0046077 - APIM EPS Updates 1812
9.174.1 Tickets 1812
9.174.2 Prepared Release 1812
9.174.3 Release Process 1813
9.174.3.1 Post Release Checks 1813
9.174.4 Roll-Back 1813
9.175 CHG0046184 - APIM Infrastructure Updates 1813
9.175.1 Tickets 1813
9.175.2 Prepared Release 1813
9.175.3 Release Process 1814
9.175.3.1 Post Release Checks 1814
9.175.4 Roll-Back 1814
9.176 CHG0046325 - APIM Infrastructure Updates 1814
9.176.1 Tickets 1814
9.176.2 Prepared Release 1814
9.176.3 Release Process 1815
9.176.3.1 Post Release Checks 1815
9.176.4 Roll-Back 1815
9.177 CHG0046555 - APIM Infrastructure Updates 1815
9.177.1 Tickets 1815
9.177.2 Prepared Release 1815
9.177.3 Release Process 1816
9.177.3.1 Post Release Checks 1816
9.177.4 Roll-Back 1816
9.178 CHG0046713 - Digital Onboarding Service 1816

9.178.1 Tickets 1816
9.178.2 Prepared Release 1816
9.178.3 Release Process 1817
9.178.3.1 Post Release Checks 1817
9.178.4 Roll-Back 1817
9.179 CHG0046042 - NHS e-RS APIM Patient Care Release R1.14 Production Deployment 1817
9.179.1 Content 1817
9.179.2 Prepared Release 1818
9.179.3 Release Process 1818
9.179.3.1 Post Release Checks 1818
9.179.4 Roll-Back 1818
9.180 CHG0046041 NHS e-RS APIM Referral Release R1.22 Deployment (Prod) 1819
9.180.1 Content 1819
9.180.2 Prepared Release 1819
9.180.3 Release Process 1819
9.180.3.1 Post Release Checks 1819
9.180.4 Roll-Back 1820
9.181 CHG0046633 - APIM EPS Updates 1820
9.181.1 Tickets 1820
9.181.2 Prepared Release 1820
9.181.3 Release Process 1820
9.181.3.1 Post Release Checks 1820
9.181.4 Roll-Back 1821
9.182 CHG0046863 - APIM EPS Updates 1821
9.182.1 Tickets 1821
9.182.2 Prepared Release 1821
9.182.3 Release Process 1821
9.182.3.1 Post Release Checks 1821

9.182.4 Roll-Back ... 1822
9.183 CHG0047223 - Digital Onboarding Service ... 1822
9.183.1 Tickets ... 1822
9.183.2 Prepared Release ... 1822
9.183.3 Release Process ... 1822
9.183.3.1 Post Release Checks ... 1822
9.183.4 Roll-Back ... 1822
9.184 CHG0047419 - APIM Infrastructure Updates ... 1823
9.184.1 Tickets ... 1823
9.184.2 Prepared Release ... 1823
9.184.3 Release Process ... 1823
9.184.3.1 Post Release Checks ... 1823
9.184.4 Roll-Back ... 1823
9.185 CHG0047578 - Digital Onboarding Content release for DAQ 11.6 Production ... 1824
9.185.1 Tickets ... 1824
9.185.2 Prepared Release ... 1824
9.185.3 Release Process ... 1824
9.185.3.1 Post Release Checks ... 1824
9.185.4 Roll-Back ... 1824
9.186 CHG0047662 - Digital Onboarding Service ... 1825
9.186.1 Tickets ... 1825
9.186.2 Prepared Release ... 1825
9.186.3 Release Process ... 1825
9.186.3.1 Post Release Checks ... 1825
9.186.4 Roll-Back ... 1825
9.187 CHG0047461 - NHS e-RS APIM Patient Care Release R1.15 Production Deployment ... 1826
9.187.1 Content ... 1826
9.187.2 Prepared Release ... 1826

9.187.3 Release Process 1826
9.187.3.1 Post Release Checks 1826
9.187.4 Roll-Back 1827
9.188 CHG0047463 - NHS e-RS APIM Referral Release R1.23 Deployment (Prod) 1827
9.188.1 Content 1827
9.188.2 Prepared Release 1827
9.188.3 Release Process 1827
9.188.3.1 Post Release Checks 1828
9.188.4 Roll-Back 1828
9.189 CHG0047777 - Digital Onboarding Service 1828
9.189.1 Tickets 1828
9.189.2 Prepared Release 1828
9.189.3 Release Process 1828
9.189.3.1 Post Release Checks 1829
9.189.4 Roll-Back 1829
9.190 CHG0048068 - Digital Onboarding Service 1829
9.190.1 Tickets 1829
9.190.2 Prepared Release 1829
9.190.3 Release Process 1829
9.190.3.1 Post Release Checks 1830
9.190.4 Roll-Back 1830
9.191 CHG0048062- Digital Onboarding Content release 1830
9.191.1 Tickets 1830
9.191.2 Prepared Release 1830
9.191.3 Release Process 1830
9.191.3.1 Post Release Checks 1831
9.191.4 Roll-Back 1831
9.192 CHG0048116 - APIM Infrastructure Updates 1831

9.192.1 Tickets 1831
9.192.2 Prepared Release 1831
9.192.3 Release Process 1831
9.192.3.1 Post Release Checks 1832
9.192.4 Roll-Back 1832
9.193 CHG0048266- Digital Onboarding Content release 1832
9.193.1 Tickets 1832
9.193.2 Prepared Release 1832
9.193.3 Release Process 1832
9.193.3.1 Post Release Checks 1833
9.193.4 Roll-Back 1833
9.194 CHG0048359 - APIM Infrastructure Updates 1833
9.194.1 Tickets 1833
9.194.2 Prepared Release 1833
9.194.3 Release Process 1833
9.194.3.1 Post Release Checks 1834
9.194.4 Roll-Back 1834
9.195 CHG0048398 - APIM Infrastructure Updates 1834
9.195.1 Tickets 1834
9.195.2 Prepared Release 1834
9.195.3 Release Process 1835
9.195.3.1 Post Release Checks 1835
9.195.4 Roll-Back 1835
9.196 CHG0048360- Digital Onboarding Content release 1835
9.196.1 Tickets 1835
9.196.2 Prepared Release 1836
9.196.3 Release Process 1836
9.196.3.1 Post Release Checks 1836
9.196.4 Roll-Back 1836

9.197 CHG0048455 - Digital Onboarding Content release 1836
9.197.1 Tickets 1836
9.197.2 Prepared Release 1837
9.197.3 Release Process 1837
9.197.3.1 Post Release Checks 1837
9.197.4 Roll-Back 1837
9.198 CHG0048511 - Digital Onboarding Content release 1837
9.198.1 Tickets 1837
9.198.2 Prepared Release 1838
9.198.3 Release Process 1838
9.198.3.1 Post Release Checks 1838
9.198.4 Roll-Back 1838
9.199 CHG0048714 - APIM Infrastructure Updates 1838
9.199.1 Tickets 1838
9.199.2 Prepared Release 1839
9.199.3 Release Process 1839
9.199.3.1 Post Release Checks 1839
9.199.4 Roll-Back 1839
9.200 CHG0048869 - Digital Onboarding Service 1839
9.200.1 Tickets 1839
9.200.2 Prepared Release 1840
9.200.3 Release Process 1840
9.200.3.1 Post Release Checks 1840
9.200.4 Roll-Back 1840
9.201 CHG0049012 - APIM Infrastructure Updates 1840
9.201.1 Tickets 1840
9.201.2 Prepared Release 1841
9.201.3 Release Process 1841
9.201.3.1 Post Release Checks 1841

9.201.4 Roll-Back 1841
9.202 CHG0049112 - Digital Onboarding Service 1841
9.202.1 Tickets 1841
9.202.2 Prepared Release 1842
9.202.3 Release Process 1842
9.202.3.1 Post Release Checks 1842
9.202.4 Roll-Back 1842
9.203 CHG0049183 - APIM EPS Updates 1842
9.203.1 Tickets 1842
9.203.2 Prepared Release 1843
9.203.3 Release Process 1843
9.203.3.1 Post Release Checks 1843
9.203.4 Roll-Back 1843
9.204 CHG0049251 - Digital Onboarding Content release 1843
9.204.1 Tickets 1843
9.204.2 Prepared Release 1844
9.204.3 Release Process 1844
9.204.3.1 Post Release Checks 1844
9.204.4 Roll-Back 1844
9.205 CHG0049159 - Booking and Referrals 1844
9.205.1 Tickets 1844
9.205.2 Prepared Release 1845
9.205.3 Release Process 1845
9.205.3.1 Post Release Checks 1845
9.205.4 Roll-Back 1845
9.205.5 CHG0051804 - Booking and Referrals 1845
9.205.5.1 Tickets 1845
9.205.5.2 Prepared Release 1845
9.205.5.3 Release Process 1846

9.205.5.4 Roll-Back 1846
9.206 CHG0049315 - APIM Infrastructure Updates 1846
9.206.1 Tickets 1846
9.206.2 Prepared Release 1846
9.206.3 Release Process 1847
9.206.3.1 Post Release Checks 1847
9.206.4 Roll-Back 1847
9.207 CHG0048743 - NHS e-RS APIM Patient Care Release R1.16 Production Deployment 1847
9.207.1 Content 1847
9.207.2 Prepared Release 1847
9.207.3 Release Process 1848
9.207.3.1 Post Release Checks 1848
9.207.4 Roll-Back 1848
9.208 CHG0048741 - NHS e-RS APIM Referral Release R1.24 Deployment (Prod) 1848
9.208.1 Content 1848
9.208.2 Prepared Release 1849
9.208.3 Release Process 1849
9.208.3.1 Post Release Checks 1849
9.208.4 Roll-Back 1849
9.209 CHG0049615- Identity Service Changes 1849
9.209.1 Tickets 1849
9.209.2 Prepared Release 1850
9.209.3 Release Process 1850
9.209.3.1 Post Release Checks 1850
9.209.4 Roll-Back 1850
9.210 CHG0049711 - Digital Onboarding Service 1850
9.210.1 Tickets 1850
9.210.2 Prepared Release 1851

9.210.3 Release Process 1851
9.210.3.1 Post Release Checks 1851
9.210.4 Roll-Back 1851
9.211 RFC CHG0049916 - APIM Infrastructure Updates 1851
9.211.1 Tickets 1851
9.211.2 Prepared Release 1852
9.211.3 Release Process 1852
9.211.3.1 Post Release Checks 1852
9.211.4 Roll-Back 1852
9.212 CHG0050015 - Digital Onboarding Content release 1852
9.212.1 Tickets 1852
9.212.2 Prepared Release 1853
9.212.3 Release Process 1853
9.212.3.1 Post Release Checks 1853
9.212.4 Roll-Back 1853
9.213 CHG0050060 - Digital Onboarding Service 1853
9.213.1 Tickets 1853
9.213.2 Prepared Release 1854
9.213.3 Release Process 1854
9.213.3.1 Post Release Checks 1854
9.213.4 Roll-Back 1854
9.214 CHG0050127 - APIM - Add rate limiting custom attribute to app 1854
9.214.1 Tickets 1854
9.214.2 Prepared Release 1855
9.214.3 Release Process 1855
9.214.3.1 Post Release Checks 1855
9.214.4 Roll-Back 1856
9.215 CHG0050342 - APIM Infrastructure Updates 1856
9.215.1 Tickets 1856

9.215.2 Prepared Release 1856
9.215.3 Release Process 1856
9.215.3.1 Post Release Checks 1856
9.215.4 Roll-Back 1857
9.216 CHG0050379 - Digital Onboarding Service 1857
9.216.1 Tickets 1857
9.216.2 Prepared Release 1857
9.216.3 Release Process 1857
9.216.3.1 Post Release Checks 1857
9.216.4 Roll-Back 1858
9.217 CHG0050564 - Digital Onboarding Content release 1858
9.217.1 Tickets 1858
9.217.2 Prepared Release 1858
9.217.3 Release Process 1858
9.217.3.1 Post Release Checks 1858
9.217.4 Roll-Back 1859
9.218 CHG0050574 - APIM Infrastructure Updates 1859
9.218.1 Tickets 1859
9.218.2 Prepared Release 1859
9.218.3 Release Process 1859
9.218.3.1 Post Release Checks 1859
9.218.4 Roll-Back 1860
9.219 CHG0050653 - APIM Infrastructure Updates 1860
9.219.1 Tickets 1860
9.219.2 Prepared Release 1860
9.219.3 Release Process 1860
9.219.3.1 Post Release Checks 1860
9.219.4 Roll-Back 1861
9.220 CHG0050994 - APIM Infrastructure Updates 1861

9.220.1 Tickets 1861
9.220.2 Prepared Release 1861
9.220.3 Release Process 1861
9.220.3.1 Post Release Checks 1862
9.220.4 Roll-Back 1862
9.221 CHG0051070 - Digital Onboarding Service 1862
9.221.1 Tickets 1862
9.221.2 Prepared Release 1862
9.221.3 Release Process 1863
9.221.3.1 Post Release Checks 1863
9.221.4 Roll-Back 1863
9.222 CHG0051201 - APIM Infrastructure Updates 1863
9.222.1 Tickets 1863
9.222.2 Prepared Release 1864
9.222.3 Release Process 1864
9.222.3.1 Post Release Checks 1864
9.222.4 Roll-Back 1864
9.223 CHG0051205 - APIM EPS FHIR Façade Updates 1864
9.223.1 Tickets 1865
9.223.2 Prepared Release 1865
9.223.3 Release Process 1865
9.223.3.1 Post Release Checks 1865
9.223.4 Roll-Back 1865
9.224 CHG0051432 - APIM Infrastructure Updates 1865
9.224.1 Tickets 1866
9.224.2 Prepared Release 1866
9.224.3 Release Process 1866
9.224.3.1 Post Release Checks 1866
9.224.4 Roll-Back 1866

9.225 CHG0051704 - Digital Onboarding Service 1867
9.225.1 Tickets 1867
9.225.2 Prepared Release 1867
9.225.3 Release Process 1867
9.225.3.1 Post Release Checks 1867
9.225.4 Roll-Back 1868
9.226 CHG0051849 - APIM Infrastructure Updates 1868
9.226.1 Tickets 1868
9.226.2 Prepared Release 1868
9.226.3 Release Process 1868
9.226.3.1 Post Release Checks 1868
9.226.4 Roll-Back 1869
9.227 CHG0051855 - APIM EPS FHIR Façade Updates 1869
9.227.1 Tickets 1869
9.227.2 Prepared Release 1869
9.227.3 Release Process 1869
9.227.3.1 Post Release Checks 1869
9.227.4 Roll-Back 1870
9.228 CHG0051994 - Digital Onboarding Service 1870
9.228.1 Tickets 1870
9.228.2 Prepared Release 1870
9.228.3 Release Process 1870
9.228.3.1 Post Release Checks 1871
9.228.4 Roll-Back 1871
9.229 CHG0052282 - Digital Onboarding Content release for DAQ 13.4 Production 1871
9.229.1 Tickets 1871
9.229.2 Prepared Release 1872
9.229.3 Release Process 1872

9.229.3.1 Post Release Checks 1872
9.229.4 Roll-Back 1872
9.230 CHG0052295 - Digital Onboarding Service 1872
9.230.1 Tickets 1872
9.230.2 Prepared Release 1873
9.230.3 Release Process 1873
9.230.3.1 Post Release Checks 1873
9.230.4 Roll-Back 1873
9.231 CHG0052407 - APIM EPS FHIR Façade Updates 1873
9.231.1 Tickets 1873
9.231.2 Prepared Release 1873
9.231.3 Release Process 1874
9.231.3.1 Post Release Checks 1874
9.231.4 Roll-Back 1874
9.232 CHG0052605 - APIM PfP FHIR API Apigee Updates v1.0.148-beta 1874
9.232.1 Tickets 1874
9.232.2 Prepared Release 1874
9.232.3 Release Process 1875
9.232.3.1 Post Release Checks 1875
9.232.4 Roll-Back 1875
9.233 CHG0052689 - APIM Infrastructure Updates 1875
9.233.1 Tickets 1875
9.233.2 Prepared Release 1876
9.233.3 Release Process 1876
9.233.3.1 Post Release Checks 1876
9.233.4 Roll-Back 1876
9.234 CHG0052687 - Digital Onboarding Content release for DAQ 13.5 Production 1877
9.234.1 Tickets 1877

9.234.2 Prepared Release 1877
9.234.3 Release Process 1877
9.234.3.1 Post Release Checks 1877
9.234.4 Roll-Back 1877
9.235 CHG0052866 - APIM Infrastructure Updates 1878
9.235.1 Tickets 1878
9.235.2 Prepared Release 1879
9.235.3 Release Process 1879
9.235.3.1 Post Release Checks 1879
9.235.4 Roll-Back 1879
9.236 CHG0052918 - APIM EPS FHIR Façade Updates 1879
9.236.1 Tickets 1880
9.236.2 Prepared Release 1880
9.236.3 Release Process 1880
9.236.3.1 Post Release Checks 1880
9.236.4 Roll-Back 1880
9.237 CHG0052921 - APIM PfP FHIR API Apigee Updates v1.0.150-beta 1880
9.237.1 Tickets 1881
9.237.2 Prepared Release 1881
9.237.3 Release Process 1881
9.237.3.1 Post Release Checks 1881
9.237.4 Roll-Back 1881
9.238 CHG0053201 - Digital Onboarding Service 1881
9.238.1 Tickets 1881
9.238.2 Prepared Release 1882
9.238.3 Release Process 1882
9.238.3.1 Post Release Checks 1882
9.238.4 Roll-Back 1882
9.239 CHG0053252 - APIM Infrastructure Updates 1883

9.239.1 Tickets 1883
9.239.2 Prepared Release 1883
9.239.3 Release Process 1883
9.239.3.1 Post Release Checks 1883
9.239.4 Roll-Back 1884
9.240 CHG0053287 - APIM EPS FHIR Façade Updates 1884
9.240.1 Tickets 1884
9.240.2 Prepared Release 1884
9.240.3 Release Process 1884
9.240.3.1 Post Release Checks 1884
9.240.4 Roll-Back 1885
9.241 CHG0053288 - APIM PfP FHIR API Apigee Updates v1.0.152-beta 1885
9.241.1 Tickets 1885
9.241.2 Prepared Release 1885
9.241.3 Release Process 1885
9.241.3.1 Post Release Checks 1885
9.241.4 Roll-Back 1886
9.242 CHG0053370 - APIM Infrastructure Updates 1886
9.242.1 Tickets 1886
9.242.2 Prepared Release 1886
9.242.3 Release Process 1886
9.242.3.1 Post Release Checks 1886
9.242.4 Roll-Back 1886
9.243 CHG0053374 - APIM Infrastructure Updates 1887
9.243.1 Tickets 1887
9.243.2 Prepared Release 1887
9.243.3 Release Process 1887
9.243.3.1 Post Release Checks 1887
9.243.4 Roll-Back 1887

9.244 CHG0053407 - Digital Onboarding Service 1888
9.244.1 Tickets 1888
9.244.2 Prepared Release 1888
9.244.3 Release Process 1888
9.244.3.1 Post Release Checks 1888
9.244.4 Roll-Back 1888
9.245 CHG0053638 - Digital Onboarding Service 1889
9.245.1 Tickets 1889
9.245.2 Prepared Release 1889
9.245.3 Release Process 1889
9.245.3.1 Post Release Checks 1889
9.245.4 Roll-Back 1889
9.246 CHG0053705 - Identity Service Updates 1890
9.246.1 Tickets 1890
9.246.2 Prepared Release 1890
9.246.3 Release Process 1890
9.246.3.1 Post Release Checks 1890
9.246.4 Roll-Back 1890
9.247 CHG0053719 - APIM Infrastructure Updates 1891
9.247.1 Tickets 1891
9.247.2 Prepared Release 1891
9.247.3 Release Process 1891
9.247.3.1 Post Release Checks 1891
9.247.4 Roll-Back 1892
9.248 CHG0054065 - APIM Infrastructure Updates 1892
9.248.1 Tickets 1892
9.248.2 Prepared Release 1892
9.248.3 Release Process 1892
9.248.3.1 Post Release Checks 1892

9.248.4 Roll-Back 1893
9.249 CHG0054160 - Digital Onboarding Service 1893
9.249.1 Tickets 1893
9.249.2 Prepared Release 1893
9.249.3 Release Process 1893
9.249.3.1 Post Release Checks 1893
9.249.4 Roll-Back 1894
9.250 CHG0054271 - APIM Infrastructure Updates 1894
9.250.1 Tickets 1894
9.250.2 Prepared Release 1894
9.250.3 Release Process 1894
9.250.3.1 Post Release Checks 1895
9.250.4 Roll-Back 1895
9.251 CHG0054438 - APIM Infrastructure Updates 1895
9.251.1 Tickets 1895
9.251.2 Prepared Release 1895
9.251.3 Release Process 1895
9.251.3.1 Post Release Checks 1896
9.251.4 Roll-Back 1896
9.252 CHG0054514 - APIM Infrastructure Updates 1896
9.252.1 Tickets 1896
9.252.2 Prepared Release 1896
9.252.3 Release Process 1896
9.252.3.1 Post Release Checks 1897
9.252.4 Roll-Back 1897
9.253 CHG0054686 - APIM Infrastructure Updates 1897
9.253.1 Tickets 1897
9.253.2 Prepared Release 1897
9.253.3 Release Process 1897

9.253.3.1 Post Release Checks 1898
9.253.4 Roll-Back 1898
9.254 CHG0054703 - APIM Infrastructure Updates 1898
9.254.1 Tickets 1898
9.254.2 Prepared Release 1899
9.254.3 Release Process 1899
9.254.3.1 Post Release Checks 1899
9.254.4 Roll-Back 1899
9.255 CHG0054951 - APIM Infrastructure Updates 1899
9.255.1 Tickets 1899
9.255.2 Prepared Release 1900
9.255.3 Release Process 1900
9.255.3.1 Post Release Checks 1900
9.255.4 Roll-Back 1900
9.256 CHG0055041 - Digital Onboarding Service 1900
9.256.1 Tickets 1900
9.256.2 Prepared Release 1901
9.256.3 Release Process 1901
9.256.3.1 Post Release Checks 1901
9.256.4 Roll-Back 1901
9.257 CHG0055343 - APIM Infrastructure Updates 1901
9.257.1 Tickets 1902
9.257.2 Prepared Release 1902
9.257.3 Release Process 1902
9.257.3.1 Post Release Checks 1902
9.257.4 Roll-Back 1902
9.258 CHG0055353 - Digital Onboarding Content release for DAQ 14.0 Production 1903
9.258.1 Tickets 1903

9.258.2 Prepared Release 1903
9.258.3 Release Process 1903
9.258.3.1 Post Release Checks 1903
9.258.4 Roll-Back 1903
9.259 CHG0055559 - Digital Onboarding Service 1904
9.259.1 Tickets 1904
9.259.2 Prepared Release 1904
9.259.3 Release Process 1904
9.259.3.1 Post Release Checks 1904
9.259.4 Roll-Back 1904
9.260 CHG0055831- APIM Infrastructure Updates 1905
9.260.1 Tickets 1905
9.260.2 Prepared Release 1905
9.260.3 Release Process 1905
9.260.3.1 Post Release Checks 1905
9.260.4 Roll-Back 1905
9.261 CHG0056020 - Digital Onboarding Service 1906
9.261.1 Tickets 1906
9.261.2 Prepared Release 1906
9.261.3 Release Process 1906
9.261.3.1 Post Release Checks 1906
9.261.4 Roll-Back 1906
9.262 CHG0056746 - Digital Onboarding Service 1907
9.262.1 Tickets 1907
9.262.2 Prepared Release 1907
9.262.3 Release Process 1907
9.262.3.1 Post Release Checks 1907
9.262.4 Roll-Back 1908
9.262.5 Release Process 1908

9.262.5.1 Post Release Checks 1908
9.262.6 Roll-Back 1908
9.263 CHG0056988- APIM Infrastructure Updates 1908
9.263.1 Tickets 1908
9.263.2 Prepared Release 1909
9.263.3 Release Process 1909
9.263.3.1 Post Release Checks 1909
9.263.4 Roll-Back 1909
9.264 CHG0057120 - Digital Onboarding Content release for DAQ 14.1 Production 1909
9.264.1 Tickets 1909
9.264.2 Prepared Release 1910
9.264.3 Release Process 1910
9.264.3.1 Post Release Checks 1910
9.264.4 Roll-Back 1910
9.265 CHG0057534 - APIM Infrastructure Updates 1910
9.265.1 Tickets 1910
9.265.2 Prepared Release 1911
9.265.3 Release Process 1911
9.265.3.1 Post Release Checks 1911
9.265.4 Roll-Back 1911
9.266 CHG0057559 - APIM Infrastructure Updates 1911
9.266.1 Tickets 1911
9.266.2 Prepared Release 1912
9.266.3 Release Process 1912
9.266.3.1 Post Release Checks 1912
9.266.4 Roll-Back 1912
9.267 CHG0057636 - APIM Infrastructure Updates 1912
9.267.1 Tickets 1912

9.267.2 Prepared Release 1913
9.267.3 Release Process 1913
9.267.3.1 Post Release Checks 1913
9.267.4 Roll-Back 1913
9.268 CHG0057776 - APIM Infrastructure Updates 1913
9.268.1 Tickets 1914
9.268.2 Prepared Release 1914
9.268.3 Release Process 1914
9.268.3.1 Post Release Checks 1914
9.268.4 Roll-Back 1914
9.269 CHG0057788 - Digital Onboarding Service 1915
9.269.1 Tickets 1915
9.269.2 Prepared Release 1915
9.269.3 Release Process 1915
9.269.3.1 Post Release Checks 1915
9.269.4 Roll-Back 1915
9.270 CHG0057834 - APIM Infrastructure Updates 1916
9.270.1 Tickets 1916
9.270.2 Prepared Release 1916
9.270.3 Release Process 1916
9.270.3.1 Post Release Checks 1916
9.270.4 Roll-Back 1916
9.271 CHG0057875 - APIM Infrastructure Updates 1917
9.271.1 Tickets 1917
9.271.2 Prepared Release 1917
9.271.3 Release Process 1917
9.271.3.1 Post Release Checks 1917
9.271.4 Roll-Back 1918
9.272 CHG0057999 - APIM Infrastructure Updates 1918

9.272.1 Tickets 1918
9.272.2 Prepared Release 1918
9.272.3 Release Process 1918
9.272.3.1 Post Release Checks 1918
9.272.4 Roll-Back 1919
9.273 CHG0058074 - APIM Infrastructure Updates 1919
9.273.1 Tickets 1919
9.273.2 Prepared Release 1919
9.273.3 Release Process 1919
9.273.3.1 Post Release Checks 1919
9.273.4 Roll-Back 1920
9.274 CHG0058130 - APIM Infrastructure Updates 1920
9.274.1 Tickets 1920
9.274.2 Prepared Release 1920
9.274.3 Release Process 1920
9.274.3.1 Post Release Checks 1920
9.274.4 Roll-Back 1921
9.275 CHG0058183 - APIM Infrastructure Updates 1921
9.275.1 Tickets 1921
9.275.2 Prepared Release 1921
9.275.3 Release Process 1921
9.275.3.1 Post Release Checks 1922
9.275.4 Roll-Back 1922
9.276 CHG0058247 - APIM Infrastructure Updates 1922
9.276.1 Tickets 1922
9.276.2 Prepared Release 1922
9.276.3 Release Process 1922
9.276.3.1 Post Release Checks 1923
9.276.4 Roll-Back 1923

9.277 CHG0058407 - APIM Infrastructure Updates 1923
9.277.1 Tickets 1923
9.277.2 Prepared Release 1924
9.277.3 Release Process 1924
9.277.3.1 Post Release Checks 1924
9.277.4 Roll-Back 1924
9.278 CHG0058536 - Digital Onboarding Service 1924
9.278.1 Tickets 1924
9.278.2 Prepared Release 1925
9.278.3 Release Process 1925
9.278.3.1 Post Release Checks 1925
9.278.4 Roll-Back 1925
9.279 CHG0058624 - APIM Infrastructure Updates 1925
9.279.1 Tickets 1925
9.279.2 Prepared Release 1926
9.279.3 Release Process 1926
9.279.3.1 Post Release Checks 1926
9.279.4 Roll-Back 1926
9.280 CHG0058712 - APIM Infrastructure Updates 1926
9.280.1 Tickets 1926
9.280.2 Prepared Release 1927
9.280.3 Release Process 1927
9.280.3.1 Post Release Checks 1927
9.280.4 Roll-Back 1927
9.281 CHG0058728 - APIM Infrastructure Updates 1927
9.281.1 Tickets 1928
9.281.2 Prepared Release 1928
9.281.3 Release Process 1928
9.281.3.1 Post Release Checks 1928

9.281.4 Roll-Back 1928
9.282 CHG0058779 - APIM Infrastructure Updates 1929
9.282.1 Tickets 1929
9.282.2 Prepared Release 1929
9.282.3 Release Process 1929
9.282.3.1 Post Release Checks 1929
9.282.4 Roll-Back 1929
9.283 CHG0058794 - APIM Infrastructure Updates 1930
9.283.1 Tickets 1930
9.283.2 Prepared Release 1930
9.283.3 Release Process 1931
9.283.3.1 Post Release Checks 1931
9.283.4 Roll-Back 1931
9.284 CHG0058838 - APIM Infrastructure Updates 1931
9.284.1 Tickets 1931
9.284.2 Prepared Release 1931
9.284.3 Release Process 1932
9.284.3.1 Post Release Checks 1932
9.284.4 Roll-Back 1932
9.285 CHG0058988- Digital Onboarding Service 1932
9.285.1 Tickets 1932
9.285.2 Prepared Release 1933
9.285.3 Release Process 1933
9.285.3.1 Post Release Checks 1933
9.285.4 Roll-Back 1933
9.286 CHG0059059 - APIM Infrastructure Updates 1933
9.286.1 Tickets 1933
9.286.2 Prepared Release 1934
9.286.3 Release Process 1934

9.286.3.1 Post Release Checks 1934
9.286.4 Roll-Back 1934
9.287 CHG0059171 - APIM Infrastructure Updates 1934
9.287.1 Tickets 1935
9.287.2 Prepared Release 1935
9.287.3 Release Process 1935
9.287.3.1 Post Release Checks 1935
9.287.4 Roll-Back 1935
9.288 CHG0059319 - Digital Onboarding Content release for DAQ 14.4 Production 1936
9.288.1 Tickets 1936
9.288.2 Prepared Release 1936
9.288.3 Release Process 1936
9.288.3.1 Post Release Checks 1936
9.288.4 Roll-Back 1936
9.289 CHG0059463 - APIM Infrastructure Updates 1937
9.289.1 Tickets 1937
9.289.2 Prepared Release 1937
9.289.3 Release Process 1937
9.289.3.1 Post Release Checks 1937
9.289.4 Roll-Back 1938
9.290 CHG0059608 - APIM Infrastructure Updates 1938
9.290.1 Tickets 1938
9.290.2 Prepared Release 1938
9.290.3 Release Process 1938
9.290.3.1 Post Release Checks 1939
9.290.4 Roll-Back 1939
9.291 CHG0059853- Digital Onboarding Service 1939
9.291.1 Tickets 1939

9.291.2 Prepared Release 1939
9.291.3 Release Process 1940
9.291.3.1 Post Release Checks 1940
9.291.4 Roll-Back 1940
9.292 CHG0059908 - APIM Infrastructure Updates 1940
9.292.1 Tickets 1940
9.292.2 Prepared Release 1940
9.292.3 Release Process 1941
9.292.3.1 Post Release Checks 1941
9.292.4 Roll-Back 1941
9.293 CHG0060017 - APIM Infrastructure Updates 1941
9.293.1 Tickets 1941
9.293.2 Prepared Release 1941
9.293.3 Release Process 1942
9.293.3.1 Post Release Checks 1942
9.293.4 Roll-Back 1942
9.294 CHG0060198 - APIM Infrastructure Updates 1942
9.294.1 Tickets 1942
9.294.2 Prepared Release 1942
9.294.3 Release Process 1943
9.294.3.1 Post Release Checks 1943
9.294.4 Roll-Back 1943
9.295 CHG0060298 - APIM Infrastructure Updates 1943
9.295.1 Tickets 1943
9.295.2 Prepared Release 1944
9.295.3 Release Process 1944
9.295.3.1 Post Release Checks 1944
9.295.4 Roll-Back 1944
9.296 CHG0060407 - APIM Infrastructure Updates 1944

9.296.1 Tickets 1944
9.296.2 Prepared Release 1945
9.296.3 Release Process 1945
9.296.3.1 Post Release Checks 1945
9.296.4 Roll-Back 1945
9.297 CHG0060650 - APIM Infrastructure Updates 1945
9.297.1 Tickets 1945
9.297.2 Prepared Release 1946
9.297.3 Release Process 1946
9.297.3.1 Post Release Checks 1946
9.297.4 Roll-Back 1946
9.298 CHG0060675 - APIM Infrastructure Updates 1946
9.298.1 Tickets 1947
9.298.2 Prepared Release 1947
9.298.3 Release Process 1947
9.298.3.1 Post Release Checks 1947
9.298.4 Roll-Back 1947
9.299 CHG0060761 - APIM Infrastructure Updates 1948
9.299.1 Tickets 1948
9.299.2 Prepared Release 1948
9.299.3 Release Process 1948
9.299.3.1 Post Release Checks 1948
9.299.4 Roll-Back 1948
9.300 CHG0060857 - APIM Infrastructure Updates 1949
9.300.1 Tickets 1949
9.300.2 Prepared Release 1949
9.300.3 Release Process 1949
9.300.3.1 Post Release Checks 1949
9.300.4 Roll-Back 1950

9.301 CHG0060881 - APIM Infrastructure Updates 1950
9.301.1 Tickets 1950
9.301.2 Prepared Release 1950
9.301.3 Release Process 1950
9.301.3.1 Post Release Checks 1951
9.301.4 Roll-Back 1951
9.302 CHG0061730 - APIM Infrastructure Updates 1951
9.302.1 Tickets 1951
9.302.2 Prepared Release 1952
9.302.3 Release Process 1952
9.302.3.1 Post Release Checks 1952
9.302.4 Roll-Back 1952
9.303 CHG0061790 - APIM Infrastructure Updates 1952
9.303.1 Tickets 1952
9.303.2 Prepared Release 1953
9.303.3 Release Process 1953
9.303.3.1 Post Release Checks 1953
9.303.4 Roll-Back 1953
9.304 CHG0062110 - APIM Infrastructure Updates 1953
9.304.1 Tickets 1954
9.304.2 Prepared Release 1954
9.304.3 Release Process 1954
9.304.3.1 Post Release Checks 1954
9.304.4 Roll-Back 1954
9.305 CHG0061960 - Identity Service Updates 1955
9.305.1 Tickets 1955
9.305.2 Prepared Release 1955
9.305.3 Release Process 1955
9.305.3.1 Post Release Checks 1955

9.305.4 Roll-Back 1955
9.306 CHG0062021 - Identity Service Updates 1956
9.306.1 Tickets 1956
9.306.2 Prepared Release 1956
9.306.3 Release Process 1956
9.306.3.1 Post Release Checks 1956
9.306.4 Roll-Back 1956
9.307 CHG0062247- Digital Onboarding Service 1957
9.307.1 Tickets 1957
9.307.2 Prepared Release 1957
9.307.3 Release Process 1957
9.307.3.1 Post Release Checks 1957
9.307.4 Roll-Back 1957
9.308 CHG0062306 - Identity Service Updates 1958
9.308.1 Tickets 1958
9.308.2 Prepared Release 1958
9.308.3 Release Process 1958
9.308.3.1 Post Release Checks 1958
9.308.4 Roll-Back 1958
9.309 CHG0062331 - APIM Infrastructure Updates 1959
9.309.1 Tickets 1959
9.309.2 Prepared Release 1959
9.309.3 Release Process 1959
9.309.3.1 Post Release Checks 1959
9.309.4 Roll-Back 1960
9.310 CHG0062472 - APIM Infrastructure Updates 1960
9.310.1 Tickets 1960
9.310.2 Prepared Release 1960
9.310.3 Release Process 1960

9.310.3.1 Post Release Checks 1961
9.310.4 Roll-Back 1961
9.311 CHG0063443 - APIM Infrastructure Updates 1961
9.311.1 Tickets 1961
9.311.2 Prepared Release 1962
9.311.3 Release Process 1962
9.311.3.1 Post Release Checks 1962
9.311.4 Roll-Back 1962
9.312 CHG0064048 - Digital Onboarding Content release for DAQ 15.1 Production 1962
9.312.1 Tickets 1963
9.312.2 Prepared Release 1963
9.312.3 Release Process 1963
9.312.3.1 Post Release Checks 1963
9.312.4 Roll-Back 1963
9.313 CHG0064121 - APIM Infrastructure Updates 1963
9.313.1 Tickets 1964
9.313.2 Prepared Release 1964
9.313.3 Release Process 1964
9.313.3.1 Post Release Checks 1964
9.313.4 Roll-Back 1964
9.314 CHG0064142 - APIM Infrastructure Updates 1965
9.314.1 Tickets 1965
9.314.2 Prepared Release 1965
9.314.3 Release Process 1965
9.314.3.1 Post Release Checks 1965
9.314.4 Roll-Back 1965
9.315 CHG0064418 - APIM Infrastructure Updates 1966
9.315.1 Tickets 1966

9.315.2 Prepared Release 1966
9.315.3 Release Process 1966
9.315.3.1 Post Release Checks 1966
9.315.4 Roll-Back 1967
9.316 CHG0064624 - APIM Infrastructure Updates 1967
9.316.1 Tickets 1967
9.316.2 Prepared Release 1967
9.316.3 Release Process 1967
9.316.3.1 Post Release Checks 1968
9.316.4 Roll-Back 1968
9.317 CHG0065516 - APIM Infrastructure Updates 1968
9.317.1 Tickets 1968
9.317.2 Prepared Release 1968
9.317.3 Release Process 1969
9.317.3.1 Post Release Checks 1969
9.317.4 Roll-Back 1969
9.318 CHG0065725 - APIM Infrastructure Updates 1969
9.318.1 Tickets 1969
9.318.2 Prepared Release 1970
9.318.3 Release Process 1970
9.318.3.1 Post Release Checks 1970
9.318.4 Roll-Back 1970
9.319 CHG0065727 - APIM - Add rate limiting custom attribute to app 1970
9.319.1 Tickets 1970
9.319.2 Prepared Release 1971
9.319.3 Release Process 1971
9.319.3.1 Post Release Checks 1971
9.319.4 Roll-Back 1971
9.320 CHG0065769 - Identity Service Updates 1972

9.320.1 Tickets 1972
9.320.2 Prepared Release 1972
9.320.3 Release Process 1972
9.320.3.1 Post Release Checks 1972
9.320.4 Roll-Back 1972
9.321 CHG0065961 - NHS App Custom Rate Limit Update 1973
9.321.1 Tickets 1973
9.321.2 Prepared Release 1973
9.321.3 Release Process 1974
9.321.3.1 Post Release Checks 1974
9.321.4 Roll-Back 1974
9.322 CHG0066136 - APIM Infrastructure Updates 1974
9.322.1 Tickets 1974
9.322.2 Prepared Release 1974
9.322.3 Release Process 1975
9.322.3.1 Post Release Checks 1975
9.322.4 Roll-Back 1975
9.323 CHG0066257 - APIM Infrastructure Updates 1975
9.323.1 Tickets 1975
9.323.2 Prepared Release 1976
9.323.3 Release Process 1976
9.323.3.1 Post Release Checks 1976
9.323.4 Roll-Back 1976
9.324 CHG0066319 - APIM Infrastructure Updates 1976
9.324.1 Tickets 1976
9.324.2 Prepared Release 1977
9.324.3 Release Process 1977
9.324.3.1 Post Release Checks 1977
9.324.4 Roll-Back 1977

9.325 CHG0066347 - Digital Onboarding Service 1977
9.325.1 Tickets 1978
9.325.2 Prepared Release 1978
9.325.3 Release Process 1978
9.325.3.1 Post Release Checks 1978
9.325.4 Roll-Back 1978
9.326 CHG0066582 - Digital Onboarding Service 1978
9.326.1 Tickets 1978
9.326.2 Prepared Release 1979
9.326.3 Release Process 1979
9.326.3.1 Post Release Checks 1979
9.326.4 Roll-Back 1979
9.327 CHG0066743 - Digital Onboarding Service 1979
9.327.1 Tickets 1979
9.327.2 Prepared Release 1979
9.327.3 Release Process 1980
9.327.3.1 Post Release Checks 1980
9.327.4 Roll-Back 1980
9.328 CHG0066814 - Immunisation FHIR API deployment to PROD 1980
9.328.1 Tickets 1980
9.328.2 Prepared Release 1980
9.328.3 Release Process 1981
9.328.3.1 Post Release Checks 1981
9.328.4 Roll-Back 1981
9.329 CHG0066905 - APIM Infrastructure Updates 1981
9.329.1 Tickets 1982
9.329.2 Prepared Release 1982
9.329.3 Release Process 1982
9.329.3.1 Post Release Checks 1982

9.329.4 Roll-Back ... 1982
9.330 CHG0067046 - APIM Infrastructure Updates ... 1983
9.330.1 Tickets ... 1983
9.330.2 Prepared Release ... 1983
9.330.3 Release Process ... 1983
9.330.3.1 Post Release Checks ... 1983
9.330.4 Roll-Back ... 1984
9.331 CHG0067443 - Digital Onboarding Service ... 1984
9.331.1 Tickets ... 1984
9.331.2 Prepared Release ... 1985
9.331.3 Release Process ... 1985
9.331.3.1 Post Release Checks ... 1985
9.331.4 Roll-Back ... 1985
9.332 CHG0067490- Digital Onboarding Content release for DAQ 16.0 Production ... 1986
9.332.1 Tickets ... 1986
9.332.2 Prepared Release ... 1986
9.332.3 Release Process ... 1986
9.332.3.1 Post Release Checks ... 1986
9.332.4 Roll-Back ... 1986
9.333 CHG0067527 - APIM Infrastructure Updates ... 1987
9.333.1 Tickets ... 1987
9.333.2 Prepared Release ... 1987
9.333.3 Release Process ... 1987
9.333.3.1 Post Release Checks ... 1987
9.333.4 Roll-Back ... 1988
9.334 CHG0067588 - APIM Infrastructure Updates ... 1988
9.334.1 Tickets ... 1988
9.334.2 Prepared Release ... 1988

9.334.3 Release Process 1988
9.334.3.1 Post Release Checks 1989
9.334.4 Roll-Back 1989
9.335 CHG0067640- APIM Infrastructure Updates 1989
9.335.1 Tickets 1989
9.335.2 Prepared Release 1989
9.335.3 Release Process 1990
9.335.3.1 Post Release Checks 1990
9.335.4 Roll-Back 1990
9.336 CHG0067720- APIM Infrastructure Updates 1990
9.336.1 Tickets 1990
9.336.2 Prepared Release 1991
9.336.3 Release Process 1991
9.336.3.1 Post Release Checks 1991
9.336.4 Roll-Back 1991
9.337 CHG0067945 - Digital Onboarding Content release for DAQ 16.1 Production 1991
9.337.1 Tickets 1992
9.337.2 Prepared Release 1992
9.337.3 Release Process 1992
9.337.3.1 Post Release Checks 1992
9.337.4 Roll-Back 1992
9.338 CHG0068017 - Digital Onboarding Service 1992
9.338.1 Tickets 1992
9.338.2 Prepared Release 1993
9.338.3 Release Process 1993
9.338.3.1 Post Release Checks 1993
9.338.4 Roll-Back 1993
9.339 CHG0068037 - Digital Onboarding Content release for DAQ 17.0 Production 1993

9.339.1 Tickets 1994
9.339.2 Prepared Release 1994
9.339.3 Release Process 1994
9.339.3.1 Post Release Checks 1994
9.339.4 Roll-Back 1994
9.340 CHG0068157- APIM Infrastructure Updates 1994
9.340.1 Tickets 1995
9.340.2 Prepared Release 1995
9.340.3 Release Process 1995
9.340.3.1 Post Release Checks 1995
9.340.4 Roll-Back 1995
9.341 CHG0068402 - Digital Onboarding Content release for DAQ 18.0 Production 1996
9.341.1 Tickets 1996
9.341.2 Prepared Release 1996
9.341.3 Release Process 1996
9.341.3.1 Post Release Checks 1996
9.341.4 Roll-Back 1996
9.342 CHG0068438- APIM Infrastructure Updates 1997
9.342.1 Tickets 1997
9.342.2 Prepared Release 1997
9.342.3 Release Process 1997
9.342.3.1 Post Release Checks 1998
9.342.4 Roll-Back 1998
9.343 CHG0068545 - APIM Infrastructure Updates 1998
9.343.1 Tickets 1998
9.343.2 Prepared Release 1998
9.343.3 Release Process 1998
9.343.3.1 Post Release Checks 1999

9.343.4 Roll-Back 1999
9.344 CHG0068584 - Digital Onboarding Content release for DAQ 18.1 Production 1999
9.344.1 Tickets 1999
9.344.2 Prepared Release 1999
9.344.3 Release Process 2000
9.344.3.1 Post Release Checks 2000
9.344.4 Roll-Back 2000
9.345 CHG0068626 - Multicast Notifications Service 2000
9.345.1 Tickets 2000
9.345.2 Prepared Release 2000
9.345.3 Release Process 2000
9.345.3.1 Post Release Checks 2001
9.345.4 Roll-Back 2001
9.346 CHG0068898 - Digital Onboarding Service 2001
9.346.1 Tickets 2001
9.346.2 Prepared Release 2001
9.346.3 Release Process 2001
9.346.3.1 Post Release Checks 2002
9.346.4 Roll-Back 2002
9.347 CHG0069094 - APIM Infrastructure Updates 2002
9.347.1 Tickets 2002
9.347.2 Prepared Release 2002
9.347.3 Release Process 2002
9.347.3.1 Post Release Checks 2003
9.347.4 Roll-Back 2003
9.348 CHG0069202 - APIM Infrastructure Updates 2003
9.348.1 Tickets 2003
9.348.2 Prepared Release 2003

9.348.3 Release Process 2004
9.348.3.1 Post Release Checks 2004
9.348.4 Roll-Back 2004
9.349 CHG0069390 - APIM Infrastructure Updates 2004
9.349.1 Tickets 2004
9.349.2 Prepared Release 2005
9.349.3 Release Process 2005
9.349.3.1 Post Release Checks 2005
9.349.4 Roll-Back 2005
9.350 CHG0069773 - Digital Onboarding Content release for DAQ 18.2 Production 2005
9.350.1 Tickets 2006
9.350.2 Prepared Release 2006
9.350.3 Release Process 2006
9.350.3.1 Post Release Checks 2006
9.350.4 Roll-Back 2006
9.351 CHG0070092 - Digital Onboarding Service 2006
9.351.1 Tickets 2007
9.351.2 Prepared Release 2007
9.351.3 Release Process 2007
9.351.3.1 Post Release Checks 2007
9.351.4 Roll-Back 2007
9.352 CHG0070107 - APIM Infrastructure Updates 2008
9.352.1 Tickets 2008
9.352.2 Prepared Release 2008
9.352.3 Release Process 2008
9.352.3.1 Post Release Checks 2008
9.352.4 Roll-Back 2008
9.353 CHG0070342 - NHS App Custom Rate Limit Update 2009

9.353.1 Tickets 2009
9.353.2 Prepared Release 2009
9.353.3 Release Process 2010
9.353.3.1 Post Release Checks 2010
9.353.4 Roll-Back 2010
9.354 CHG0070584 - Multicast Notifications Service 2010
9.354.1 Tickets 2010
9.354.2 Prepared Release 2010
9.354.3 Release Process 2010
9.354.3.1 Post Release Checks 2011
9.354.4 Roll-Back 2011
9.355 CHG0072360 - PDS Custom Rate Limit Update for Anima 2011
9.355.1 Tickets 2011
9.355.2 Prepared Release 2011
9.355.3 Release Process 2012
9.355.3.1 Post Release Checks 2012
9.355.4 Roll-Back 2013
9.356 CHG0070601 - Digital Onboarding Service 2013
9.356.1 Tickets 2013
9.356.2 Prepared Release 2013
9.356.3 Release Process 2013
9.356.3.1 Post Release Checks 2013
9.356.4 Roll-Back 2014
9.357 CHG0070517- Immunisation FHIR API deployment to PROD 2014
9.357.1 Tickets 2014
9.357.2 Prepared Release 2014
9.357.3 Release Process 2014
9.357.3.1 Post Release Checks 2015
9.357.4 Roll-Back 2015

9.358 CHG0070675 - Digital Onboarding Content release for DAQ 19.0 Production 2015
9.358.1 Tickets 2015
9.358.2 Prepared Release 2016
9.358.3 Release Process 2016
9.358.3.1 Post Release Checks 2016
9.358.4 Roll-Back 2016
9.359 CHG0070705 - APIM Infrastructure Updates 2016
9.359.1 Tickets 2016
9.359.2 Prepared Release 2017
9.359.3 Release Process 2017
9.359.3.1 Post Release Checks 2017
9.359.4 Roll-Back 2017
9.360 CHG0071016 - NHS App Custom Rate Limit Update 2017
9.360.1 Tickets 2017
9.360.2 Prepared Release 2018
9.360.3 Release Process 2018
9.360.3.1 Post Release Checks 2018
9.360.4 Roll-Back 2019
9.361 CHG0071101 - Digital Onboarding Content release for DAQ 19.1 Production 2019
9.361.1 Tickets 2019
9.361.2 Prepared Release 2019
9.361.3 Release Process 2019
9.361.3.1 Post Release Checks 2019
9.361.4 Roll-Back 2020
9.362 CHG0071606 - APIM Infrastructure Updates 2020
9.362.1 Tickets 2020
9.362.2 Prepared Release 2020
9.362.3 Release Process 2020

9.362.3.1 Post Release Checks 2021
9.362.4 Roll-Back 2021
9.363 CHG0071920 - APIM Infrastructure Updates 2021
9.363.1 Tickets 2021
9.363.2 Prepared Release 2021
9.363.3 Release Process 2022
9.363.3.1 Post Release Checks 2022
9.363.4 Roll-Back 2022
9.364 CHG0071923 - Digital Onboarding Content release for DAQ 20.0 Production 2022
9.364.1 Tickets 2022
9.364.2 Prepared Release 2023
9.364.3 Release Process 2023
9.364.3.1 Post Release Checks 2023
9.364.4 Roll-Back 2023
9.365 CHG0071979 - Digital Onboarding Content release for DAQ 21.0 Production 2023
9.365.1 Tickets 2023
9.365.2 Prepared Release 2024
9.365.3 Release Process 2024
9.365.3.1 Post Release Checks 2024
9.365.4 Roll-Back 2024
9.366 CHG0072339 - Digital Onboarding Service 2024
9.366.1 Tickets 2024
9.366.2 Prepared Release 2025
9.366.3 Release Process 2025
9.366.3.1 Post Release Checks 2025
9.366.4 Roll-Back 2025
9.367 CHG0072472 - APIM Infrastructure Updates 2025
9.367.1 Tickets 2025

9.367.2 Prepared Release 2026
9.367.3 Release Process 2026
9.367.3.1 Post Release Checks 2026
9.367.4 Roll-Back 2026
9.368 CHG0072682 - APIM Infrastructure Updates 2026
9.368.1 Tickets 2027
9.368.2 Prepared Release 2027
9.368.3 Release Process 2027
9.368.3.1 Post Release Checks 2027
9.368.4 Roll-Back 2027
9.369 CHG0073010 - Digital Onboarding Content release for DAQ 22.0 Production 2028
9.369.1 Tickets 2028
9.369.2 Prepared Release 2028
9.369.3 Release Process 2028
9.369.3.1 Post Release Checks 2028
9.369.4 Roll-Back 2028
9.370 CHG0073067 - Digital Onboarding Service 2029
9.370.1 Tickets 2029
9.370.2 Prepared Release 2029
9.370.3 Release Process 2029
9.370.3.1 Post Release Checks 2030
9.370.4 Roll-Back 2030
9.371 CHG0073262 - Digital Onboarding Service 2030
9.371.1 Tickets 2030
9.371.2 Prepared Release 2030
9.371.3 Release Process 2030
9.371.3.1 Post Release Checks 2031
9.371.4 Roll-Back 2031

9.372 CHG0073383 - APIM Infrastructure Updates 2031
9.372.1 Tickets 2031
9.372.2 Prepared Release 2031
9.372.3 Release Process 2032
9.372.3.1 Post Release Checks 2032
9.372.4 Roll-Back 2032
9.373 CHG0073656 - Digital Onboarding Service 2032
9.373.1 Tickets 2032
9.373.2 Prepared Release 2033
9.373.3 Release Process 2033
9.373.3.1 Post Release Checks 2033
9.373.4 Roll-Back 2033
9.374 CHG0073902 - APIM monitoring-service-discovery Updates 2033
9.374.1 Tickets 2033
9.374.2 Prepared Release 2034
9.374.3 Release Process 2034
9.374.3.1 Post Release Checks 2034
9.374.4 Roll-Back 2034
9.374.4.1 Version v1.0.31-alpha 2034
9.375 CHG0073957 - APIM Splunk Monitor Updates 2034
9.375.1 Tickets 2034
9.375.2 Prepared Release 2035
9.375.3 Release Process 2035
9.375.3.1 Post Release Checks 2035
9.375.4 Roll-Back 2035
9.376 CHG0073626 - Immunisation FHIR API deployment to PROD 2036
9.376.1 Tickets 2036
9.376.2 Prepared Release 2036
9.376.3 Release Process 2036

9.376.3.1 Post Release Checks 2037
9.376.4 Roll-Back 2037
9.377 CHG0073996 - Identity Service Updates 2037
9.377.1 Tickets 2037
9.377.2 Prepared Release 2037
9.377.3 Release Process 2038
9.377.3.1 Post Release Checks 2038
9.377.4 Roll-Back 2038
9.378 CHG0074029 - Digital Onboarding Service 2038
9.378.1 Tickets 2038
9.378.2 Prepared Release 2038
9.378.3 Release Process 2039
9.378.3.1 Post Release Checks 2039
9.378.4 Roll-Back 2039
9.379 CHG0074642 - APIM Sync Wrap Updates 2039
9.379.1 Tickets 2039
9.379.2 Prepared Release 2039
9.379.3 Release Process 2040
9.379.3.1 Post Release Checks 2040
9.379.4 Roll-Back 2040
9.380 CHG0074860 - APIM Infrastructure Updates 2040
9.380.1 Tickets 2040
9.380.2 Prepared Release 2041
9.380.3 Release Process 2041
9.380.3.1 Post Release Checks 2041
9.380.4 Roll-Back 2041
9.381 CHG0075022 - PDS Custom Rate Limit Update for IPlato 2041
9.381.1 Tickets 2042
9.381.2 Prepared Release 2042

9.381.3 Release Process 2043
9.381.3.1 Post Release Checks 2043
9.381.4 Roll-Back 2043
9.382 CHG0075379 - Digital Onboarding Service 2043
9.382.1 Tickets 2043
9.382.2 Prepared Release 2044
9.382.3 Release Process 2044
9.382.3.1 Post Release Checks 2044
9.382.4 Roll-Back 2044
9.383 CHG0075648 - Mock JWKS Updates 2045
9.383.1 Tickets 2045
9.383.2 Prepared Release 2045
9.383.3 Release Process 2045
9.383.3.1 Post Release Checks 2045
9.383.4 Roll-Back 2045
9.384 CHG0075754 (N) & CHG0075915 (Er) - APIM Infrastructure Updates. 2046
9.384.1 Tickets 2046
9.384.2 Prepared Release 2046
9.384.3 Release Process 2047
9.384.3.1 Post Release Checks 2047
9.384.4 Roll-Back 2047
9.385 CHG0075918 (E) - APIM Infrastructure Updates 2047
9.385.1 Tickets 2048
9.385.2 Prepared Release 2048
9.385.3 Release Process 2048
9.385.3.1 Post Release Checks 2048
9.385.4 Roll-Back 2048
9.386 CHG0076180 - Identity Custom Rate Limit Update for IPlato 2049
9.386.1 Tickets 2049

9.386.2 Prepared Release 2049
9.386.3 Release Process 2050
9.386.3.1 Post Release Checks 2050
9.386.4 Roll-Back 2050
9.387 CHG0076190 - Multicast Notifications Service 2050
9.387.1 Tickets 2051
9.387.2 Prepared Release 2051
9.387.3 Release Process 2051
9.387.3.1 Post Release Checks 2051
9.387.4 Roll-Back 2051
9.388 CHG0076569 - Digital Onboarding Service 2051
9.388.1 Tickets 2052
9.388.2 Prepared Release 2052
9.388.3 Release Process 2052
9.388.3.1 Post Release Checks 2052
9.388.4 Roll-Back 2052
9.389 CHG0076652 - APIM Infrastructure Updates 2053
9.389.1 Tickets 2053
9.389.2 Prepared Release 2053
9.389.3 Release Process 2053
9.389.3.1 Post Release Checks 2053
9.389.4 Roll-Back 2054
9.390 CHG0076937 - APIM Infrastructure Updates 2054
9.390.1 Tickets 2054
9.390.2 Prepared Release 2054
9.390.3 Release Process 2054
9.390.3.1 Post Release Checks 2054
9.390.4 Roll-Back 2055
10 Help and support 2056

10.1 Supporting your API delivery 2056
10.1.1 Self-service support 2056
10.1.2 Support via Slack 2056
10.1.3 Service notifications via Slack 2056
10.1.4 Feature requests 2056
10.1.5 Architecture support 2056
10.1.6 FHIR API payload design support from IOPS 2057
10.2 Live service support for your API 2057
10.2.1 Service level agreements (SLAs) 2057
10.3 Support for API consumers 2057
11 API Producer Workflow Requests 2058
11.1 Redaction Request New Process using Service Now 2059
12 API Producer Team - Key contacts: 2065
13 Release Management 2077
13.1 Components 2077
13.2 Approach and when is an RFC needed 2077
13.2.1 Making changes to Apigee Product and API Consumer Apps 2079
13.3 How to raise an RFC 2079
13.4 How to raise a release RFC change request in ServiceNow 2080
13.4.1 The RFC Ticket Process 2080
13.4.2 How to raise a release RFC in ServiceNow 2081
13.4.2.1 Login to ServiceNow 2081
13.4.2.2 Creating a new Change Request 2081
13.4.2.3 Selecting the type of Change Request 2082
13.4.2.4 Filling in the Request 2082
13.4.2.5 Select a Category 2083
13.4.2.6 Select the Service 2083
13.4.2.7 Select the Service offering 2083
13.4.2.8 Select an Assignment group 2083

13.4.2.9 Configuration Item 2083
13.4.2.10 Select Risk level 2083
13.4.2.11 Select Impact level 2084
13.4.2.12 Retrospective Change 2084
13.4.2.13 Provide a short description (title) for the Change Request 2084
13.4.2.14 Enter a Description 2084
13.4.2.15 Remedial Change 2085
13.4.2.16 Planning Tab 2085
13.4.2.17 Schedule Tab 2087
13.4.2.18 Approvals Tab 2087
13.4.2.19 Change readiness group 2087
13.4.2.20 Assessment approval group 2087
13.4.2.21 Azure DevOps pipelines metadata tab 2087
13.4.2.22 Next steps 2088
13.4.2.23 Approvals 2088
13.4.3 Changing the date of release 2088
13.5 In-hours Releases 2088
13.6 Raising RFCs - Cheatsheet 2091
13.6.1 When is an RFC needed 2091
13.6.2 High level checklist 2092
13.6.3 Good advice 2092
13.6.4 How to fill in an RFC in Service 2092
13.7 Standard Changes 2093
13.7.1 Raising a Standard Change 2093
13.7.2 How to create Standard Change Templates 2093
13.7.2.1 APIM Standard Changes 2094
13.7.3 Triage for Standard Changes 2096

Learn how to deliver APIs on the NHS Digital API platform.

Search

1 Get started

Learn how our API platform works, how to deliver an API on the platform, and how to get started.

Get started as an API producer (see page 125)

2 Deliver your API

Get detailed guidance on all aspects of delivering an API - both technical and non-technical.

Deliver your API (see page 194)

3 API register

See a list of built or in-progress APIs.

API register (see page 1204)

4 Release schedule

Get guidance on how to create a schedule a release and a RFC (Release For Change)

Release Schedule (see page 1390)

5 Get help and support

Find out how to get additional help.

Help and support (see page 2056)

6 Get started as an API producer

6.1 What are APIs?

If you're new to APIs, see What are APIs and why do they matter? (see page 126).

6.2 Benefits of API platform

For the benefits of using the API platform (and whether it's right for you) - see Benefits of using the API platform (see page 134).

6.3 Guidance for specific roles

For role-specific guidance, choose one of our "get started" guides, based on your role:

Delivery lead (see page 138)

Product owner (see page 138)

Business analyst (see page 138)

Architect (see page 141)

Technical lead (see page 141)

Developer (see page 142)

Tester (see page 142)

Technical author (see page 143)

User researcher (see page 138)

6.4 Tools, systems and environments

See our guide: Get access to tools, systems and environments (see page 144).

6.5 What are APIs and why do they matter?

6.5.1 Overview

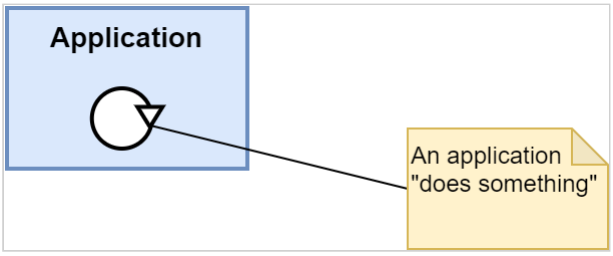
If you aren't sure what an API is or why you should care about them, read this page.

6.5.2 What is an API?

Good question. But before we answer it, let's answer some other questions first.

6.5.2.1 What is an application?

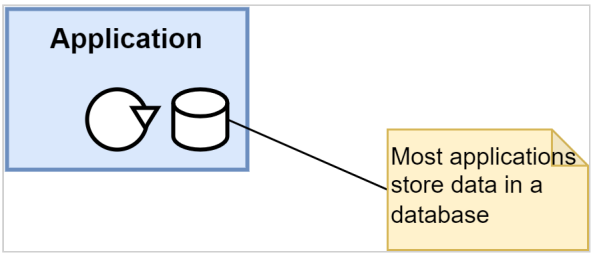
An "**application**" is a piece of computer software that "does something":



6.5.2.2 What is a database?

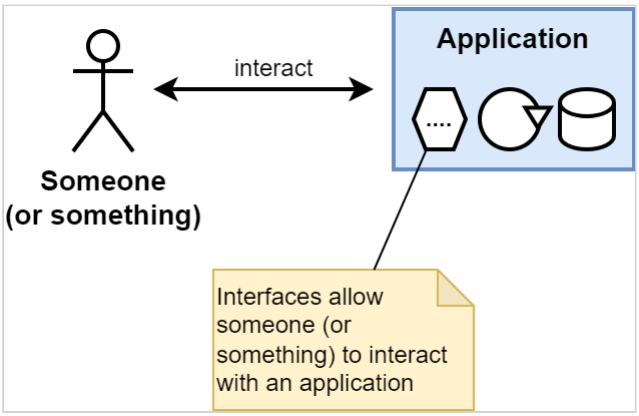
Most applications involve the processing and storage of information, also known as "data".

A "**database**" is where an application stores data:



6.5.2.3 What is an interface?

An "**interface**" allows someone (or something) else to interact with an application:



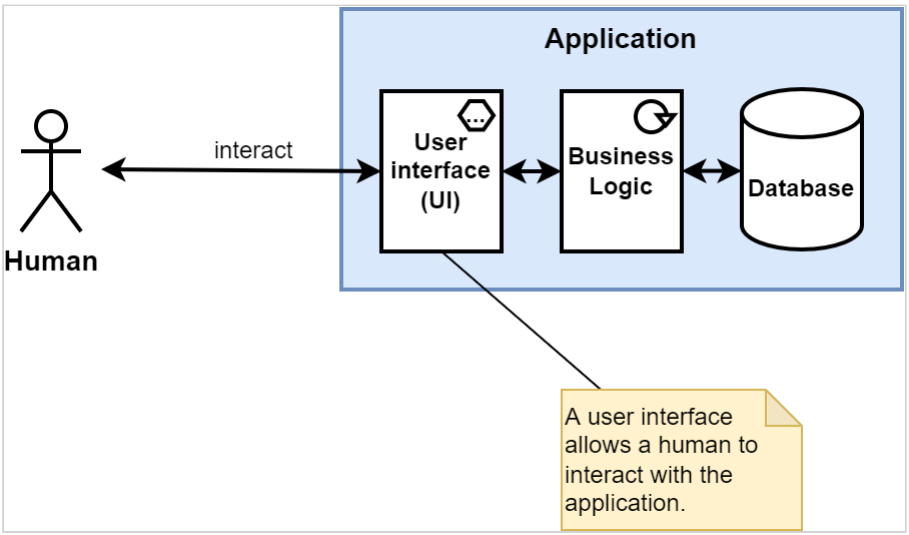
The interaction might be:

- input some data
- retrieve some data
- request some processing or calculation

An application with no interfaces is a useless black box - there might be something going on in there, but you have no idea what.

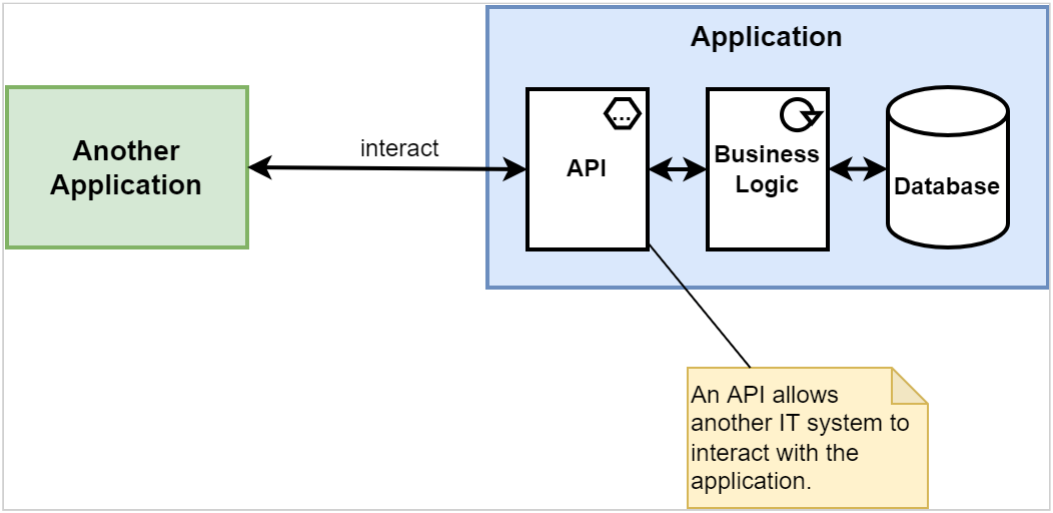
6.5.2.4 What is a user interface?

A "**user interface**" allows a **human** to interact with an application:

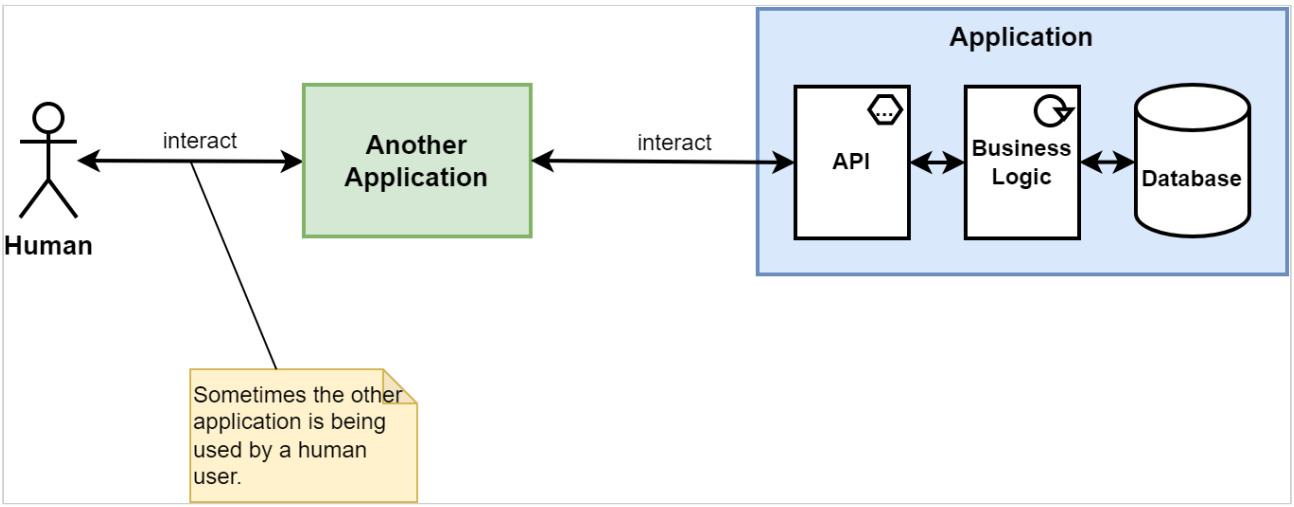


6.5.2.5 Finally, what is an API?

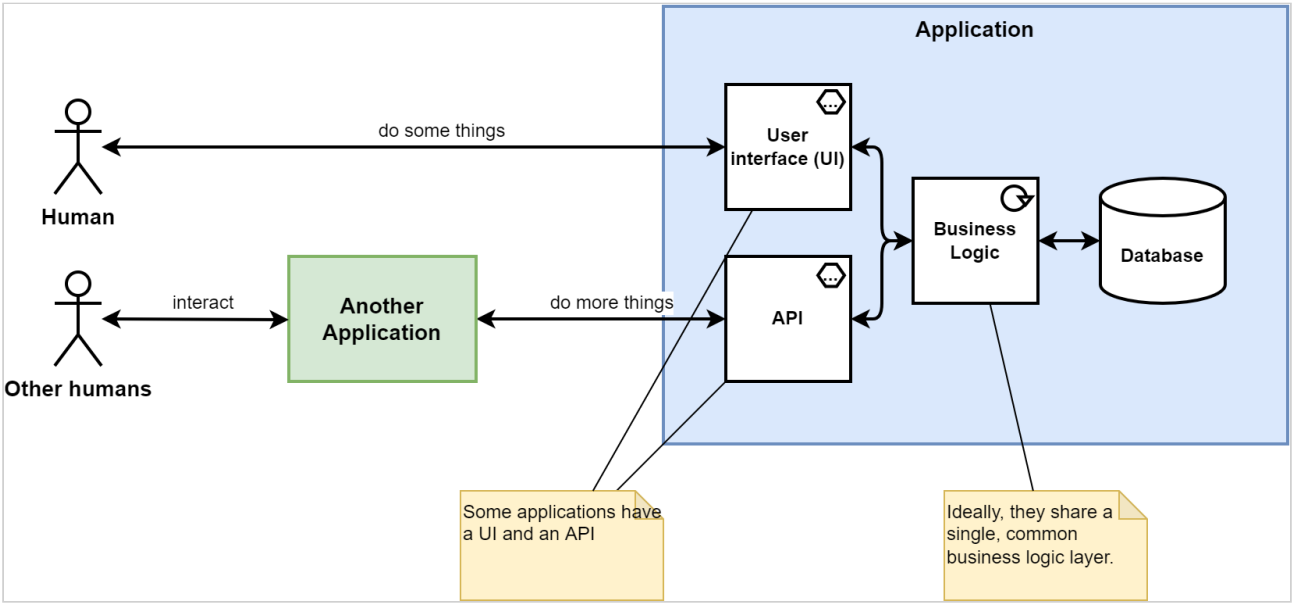
An **application programming interface** (or "API" for short) allows another application to interact with an application:



Sometimes, the other application is itself being used by a human:



And sometimes, an application has both a user interface and an API:

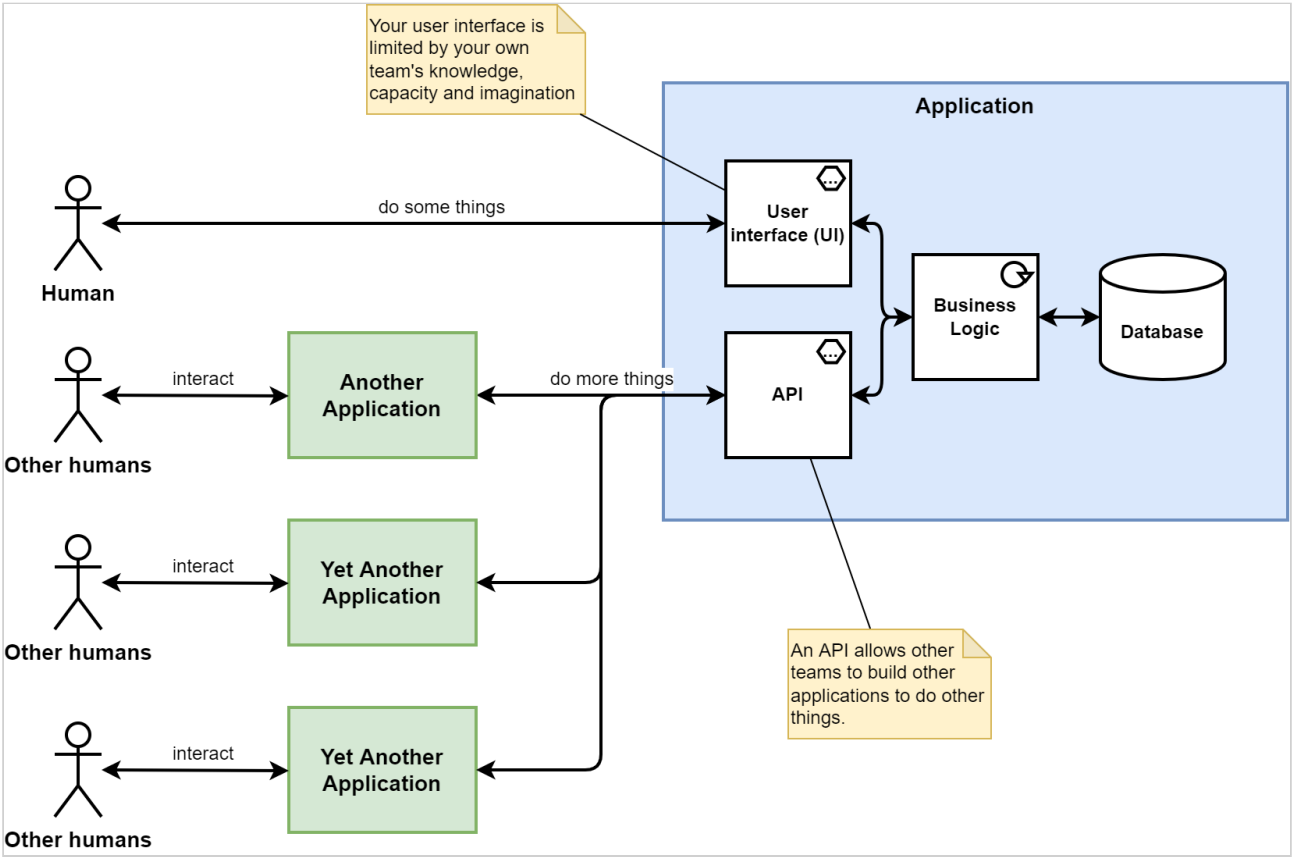


6.5.3 Why are APIs a good idea?

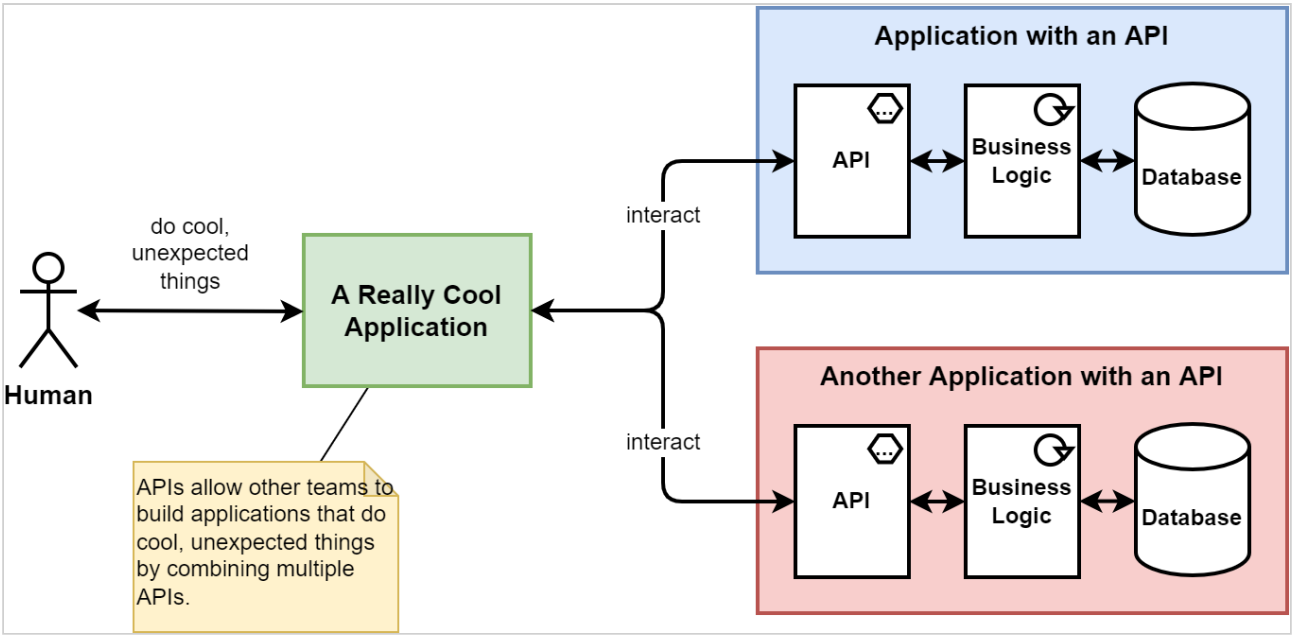
If your application only has a user interface, not an API, it will be limited by:

- **your** knowledge of your users' needs
- **your** imagination as to what cool new features to add to your UI
- **your** capacity to add new features to your UI

If you build an API, this allows multiple other teams to build other applications that provide more functions using their own knowledge, imagination and capacity:



It also allows other teams to build other applications that combine functions from multiple API-exposing applications in cool, unexpected ways:



6.5.4 NHS England's Open API policy

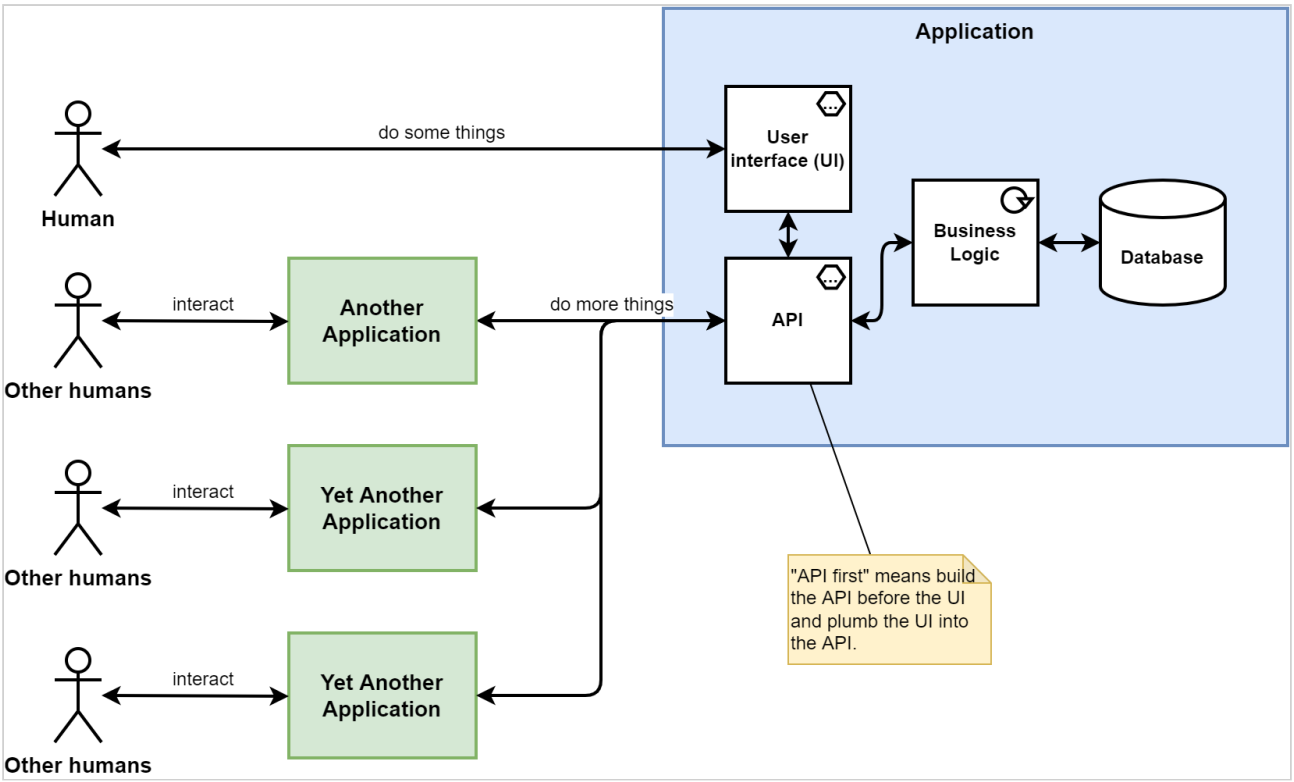
Not only are APIs a good idea, they are mandated by NHS England's Open API policy[1], which says, amongst other things:

"All significant business functionality should be exposed via an Open API"

So, if you own, or are building, a digital service, you need to be thinking about APIs.

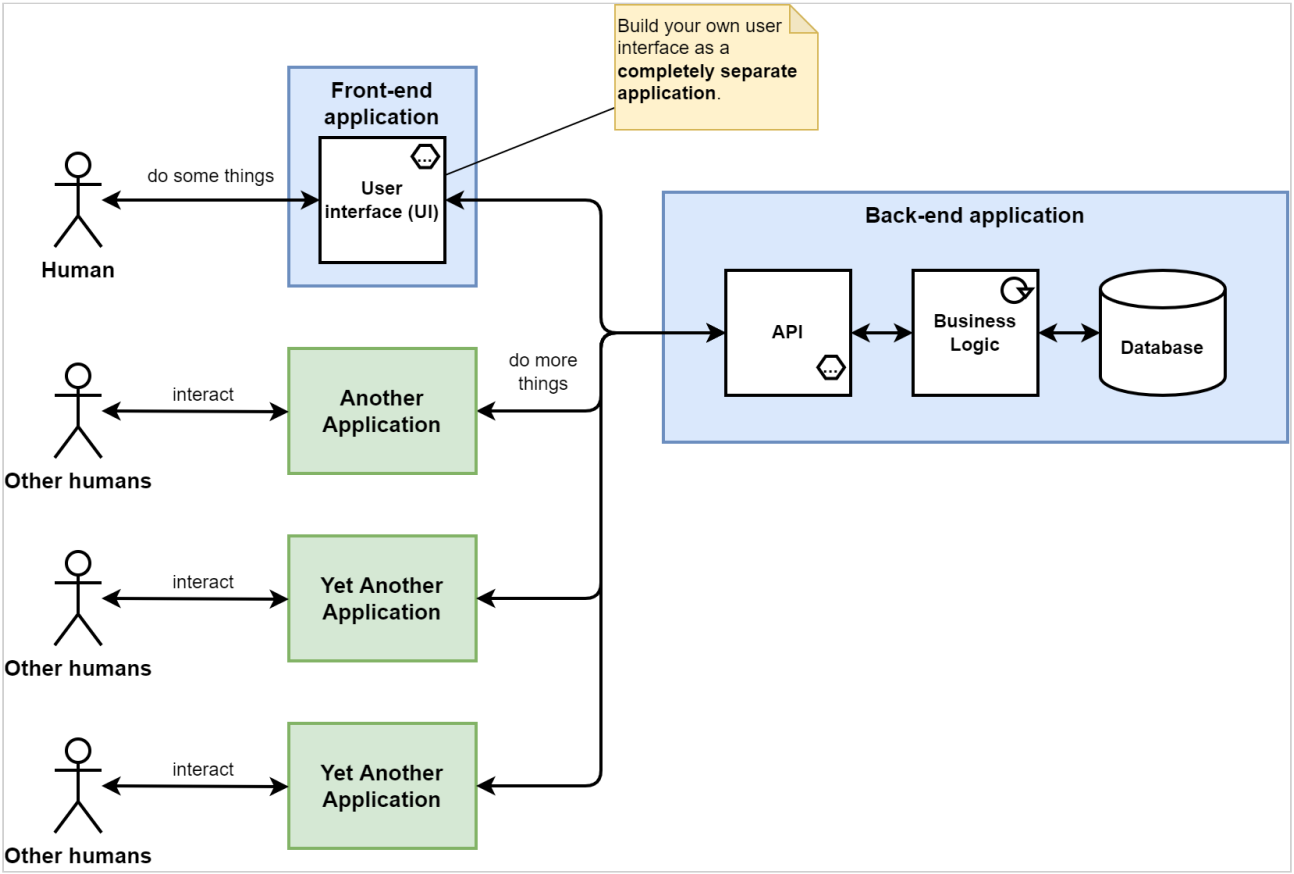
6.5.5 API first

"API first" is a particular way of building APIs in applications. As the name implies, you build the API first. You then build the user interface and, importantly, the user interface talks to the database and executes business functionality **through the API**:



A variant on this idea is that you build your user interface as a **completely separate application**:

1 https://www.england.nhs.uk/publication/open-api-architecture-policy/



API first was popularised by Jeff Bezos at Amazon with his API mandate[2].

It also aligns with the NHS architecture principle to build a data layer with registers and APIs[3].

Benefits of API first:

- using your own API allows you to check it's fit for purpose (sometimes referred to as "eating your own dog food")
- gets more APIs built sooner

6.5.6 Different types of APIs

Not all APIs are alike. In fact not all APIs are APIs.

Here's a summary:

2 https://nordicapis.com/the-bezos-api-mandate-amazons-manifesto-for-externalization/
3 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/principles/build-a-data-layer-with-registers-and-apis

Integration type	Description
Request-response - aka "synchronous"	One application requests - for example get something or update something. The other application responds - for example with the requested thing, or to confirm the update was done.
Message integration - aka "asynchronous"	One application sends a message - for example details of a transfer of care. The other application confirms ("acknowledges") it has received the message. It processes the message in its own good time.
Publish-subscribe event	One application "publishes" some information, also known as an "event" - for example details of a patient's change of address. Many applications have previously subscribed to receive this type of event. The event publisher forwards the event on to them all.

To read more about the different types of APIs, see https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/integration-and-apis.

6.5.7 How to build a good API

6.5.7.1 New APIs

If you are building a new API, you should:

- build it on our API platform[4]
- follow our API delivery process (see page 194)

To get started, see Get started as an API producer (see page 125).

6.5.7.2 Existing APIs

If you have an existing API, contact us about using our What Good Looks Like (WGLL) checklist - template (see page 1383) to assess your API and identify gaps and opportunities to improve it.

4 https://digital.nhs.uk/services/api-platform

6.6 Benefits of using the API platform

6.6.1 Slide deck (aka the "pitch deck")

Here's a slide deck version of this page - useful for "pitching" the API platform to colleagues:

https://nhs.sharepoint.com/:p:/r/sites/msteams_1186b5/_layouts/15/Doc.aspx?sourcedoc=%7B85EF2968-BB4D-46D1-93D7-A8244B4A7C31%7D&file=NHS%20Digital%20API%20Platform%20-%20API%20Producer%20Overview.pptx&action=edit&mobileredirect=true

6.6.2 What is the API platform?

See Product overview[5] or Architecture[6]

The NHS England API Platform is a centralised hub for health and care APIs within the National Health Service (NHS). It serves as a gateway for developers and organisations to access and utilise various NHS data and services through APIs.

Key Features and Benefits:

- Centralised Access: Provides a single point of access to a wide range of NHS APIs, making it easier for developers to find and integrate with relevant services.
- Standardised Interfaces: Ensures consistent API design and documentation, simplifying integration efforts.
- Security and Compliance: Adheres to strict security and data privacy standards to protect sensitive patient information.
- Developer Support: Offers comprehensive developer documentation, tutorials, and support resources.
- Innovation Catalyst: Enables the development of innovative digital health solutions by providing access to valuable healthcare data and services.

6.6.3 Is the API platform right for me?

Use the API platform if:

- you want to expose your API externally or want to do an NHS CIS2/NHS login integration
- An API is the right solution choice for your business use case - see Integration Patterns Book[7] for guidance

If your API is for internal use only, then you don't need to use APIM... However:

- The API First concept works because you never know who will be able to use your API

5 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114892404/Product+overview
6 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97747483/Architecture
7 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

- We have some very draft guidelines on how to work this out: D068 - API First policy and the use of API Management[8]

If unsure, please speak to us - see Help and support (see page 2056)

6.6.4 What do I get as an API producer?

Here's a summary:

Feature	Description
Internet end point	Your API is made available on our internet-facing domain api.service.nhs.uk[9]. We do this by deploying an API proxy into our API platform tool (Apigee Edge) which forwards traffic on to your backend API.
Security	You can choose from a variety of plug-and-play security mechanisms, including: • open access • application-restricted with simple API key • application-restricted with strong authentication • user-restricted for healthcare workers using OAuth 2.0 and NHS Identity • user-restricted for citizens using OAuth 2.0 and NHS Login Security plugs into your API proxy, so your back end API doesn't need to worry about it - see Security patterns for API consumer access (see page 823) OAuth 2.0 is both more secure and more self-service than using TLS-MA. (**future**) National RBAC / ABAC coarse grained authorisation - see Authorisation Design[10].
Rate limiting	With a simple configuration, you can limit the number of transactions to your API, either overall or per calling application - see Rate limiting (see page 1037)
Documentation and the developer hub	Based on an Open API Specification (OAS) file you manage, the API platform automatically publishes your API specification on the NHS England website - see: https://digital.nhs.uk/developer/api-catalogue. The developer hub also has many general articles on topics like security, testing and onboarding, so you don't have to write them yourself.
Sandbox for learning	The API platform automatically generates a public-facing interactive sandbox for developers to play with and learn how your API works. This is generated from your OAS specification and allows an in-browser "Try this API" functionality directly from the API specification page.

8 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/898793499/D068+-+API+First+policy+and+the+use+of+API+Management
9 http://api.service.nhs.uk
10 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/108292225/Authorisation+Design

Feature	Description
Logging and analytics	Simple configuration in your API proxy allows all requests to be logged into NHS England's Splunk indexes for API Management. You get access to the NHS England Splunk indexes for API Management, allowing dashboards and reporting.
Alerting	Simple configuration allows your API proxy and API backend to be monitored in all environments and for you to be alerted via Slack of any issues. We're also working on uptime dashboards.
Path-to-live environments	The API platform has path-to-live environments for your team's own testing and also for external developer testing, including the sandbox environment and a formal integration test environment. See https://nhsd-confluence.digital.nhs.uk/display/APM/Architecture#Architecture-environments.
GitHub repo and CI/CD Pipeline	The API platform will generate you a GitHub repo from a template which gives you a "starter" API proxy, including a CI/CD pipeline which deploys your proxy and spec to all path-to-live environments including production.
Automated testing	The CI/ CD pipeline has a number of slots to allow you to add automated testing to your pipelines
Non-functional testing	Where you need pen testing and performance testing, we have environments available to do that with.
Self-service developer onboarding	Developers can register an account, their developer teams, and their applications, all self-service. They can get access to path-to-live environments self-service. You can make access to production self-service or manually controlled by you. We've done the hard work to align the API platform with the NHS England onboarding process (based on the SCAL) and we are working to improve that process over time.
Compliance	The Platform itself has gone through the full NHS England IG and Cyber Security approvals and is designed to manage Class V data. Solutions Assurance is embedded in the Tribe and help guide Proxy design and build out. For on-boarding the Tribe is working on the new (faster) Supplier Conformance Assessment List (SCAL) based on-boarding process, but we are looking to speed that up even further by: • Implementing security in a way to reduce assurance burden on third party suppliers • Including ways to help to automate the on-boarding

Feature	Description
API delivery process	Our API delivery process is based on public sector best practice from Government Digital Service (GDS) and guides you though how to build an API that meets user needs.
User research and feedback	The developer hub uses HotJar and Google Analytics for user research, including page hits, context-sensitive feedback, pop-up polls, heatmaps and surveys. The interactive product backlog allows developers to suggest, comment and vote on API and platform features. See https://nhs-digital-api-management.featureupvote.com/
Developer comms and API marketing	Because developers register an account, we have a ready-made method for communicating service and API updates to developers. We also work closely with the comms and industry engagement teams to market the API platform and the APIs on it.
Service Levels	The Apigee Platform has an NHS England Platinum SLA. If an API Producer is using the AWS Hosted Containers option to run Docker light weight processing, then the current SLA is Bronze (in practice it is Platinum, but final service readiness work still to complete) As an API Producer you are responsible for supporting your API to the service level you require
Help and support	The API Management team are available to support you, with expertise in API design, documentation, development, testing, user research and more.

6.6.5 What's not included?

Feature	Description
Hosting your back end API	The API platform does not host your back end API, nor any of your API business logic or data persistence - you have the flexibility to decide where and how you host that. We can give advice if needed. The API platform can in theory do orchestration, but we haven't tried it yet. We can discuss.
Writing your documentation	Your team needs to actually write the documentation, using the OAS specification style. We then publish that automatically for you.

Feature	Description
Ownership	The APIM Management Tribe will not own the proxy configuration, nor the backend API - the responsibility will remain with you as an API Producer.

6.7 Get started as a delivery lead, product owner, user researcher or business analyst

Learn

Before you start, you should learn about:

- the benefits of using the API platform (and whether it's right for you) - see Benefits of using the API platform (see page 134)
- the API delivery process - see Deliver your API (see page 194)
- who will build and support your API when it becomes 'live' - see API delivery and support options (see page 293)

6.7.1 Engage

If your team is not already formally engaged with us, kick off our API delivery process, starting with the inception phase (see page 249).

6.7.2 Get set up

See Get access to tools, systems and environments (see page 144) on how to get access to various tools and environments for your role.

6.7.3 Further reading

- API platform[11], and what API Management are building
- User research for APIs (see page 139) for help on user research

---

6.7.4 Help and support

You can get help and support from us for the API Platform and for general build/planning advice. See Help and support (see page 2056) page for more information.

11 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/95229863/API+platform

We can only offer minimal help and support until you formally engage with us.

6.7.5 User research for APIs

Overview

User research helps teams learn about users and create digital services that meet their needs.

There's a lot of great information on user research in the GOV.UK Service Manual[12]. The trouble is, it's generally aimed at teams building front end digital services, not APIs.

This page gives details on how to do user research for APIs at NHS England.

6.7.5.1 User research in the different delivery phases

User research is needed in all phases of delivering your API service.

For more details see Deliver your API (see page 194).

6.7.5.2 Identifying users for APIs

Doing user research for APIs is different (and probably harder) because there are two sets of users to consider:

- Understanding API consumers (see page 256)
- End users, who use third party software to do their job

You need to have a good understanding of both classes of user - software developers and end users.

6.7.5.3 Recruiting users for APIs

6.7.5.3.1 Recruiting software developers

At NHS England, there are a number of channels you could use to recruit software developers.

For details of the channels we've used to recruit software developers for user research on the API Platform, see User recruitment[13].

12 https://www.gov.uk/service-manual/user-research
13 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125409533/User+recruitment

6.7.5.3.2 Recruiting end users

In most cases you won't need to recruit end users of third party software directly, you'll rely on feedback coming from software developers. However, it sometimes useful to get feedback directly from end users, either via contextual testing sessions or questionnaires.

For these activities, recruit end users via your software developer partners.

6.7.5.4 User research methods

See https://www.gov.uk/service-manual/user-research for a general list of user research methods.

The methods we've found work well for APIs are:

- Contextual research and observation - watching developers trying to use your documentation
- Surveys and questionnaires - getting large volumes of feedback from either developers or end users - use:
    - Google Forms
    - HotJar surveys
- HotJar feedback on the developer hub, including:
    - Feedback widgets - there is one on every page under https://digital.nhs.uk/developer (unless you have added a pop-up poll to the page)
    - Pop-up polls - asking for specific feedback like "did you find what you were looking for"
    - Heatmaps - so you can see where on your spec page users are looking/clicking
    - Surveys - ask pointed questions
- Google analytics - you can see page hits and user journeys - see https://datastudio.google.com/reporting/ea753208-2814-41d1-b79a-58dc583fc3a1
- Hackathons - getting a group of developers in a room for a day to "have a play" with your API

6.7.5.5 Communicating with Developers

You will want to communicate with the developer community for a variety of reasons:

- To recruit developers for user testing
- To let developers know your service is available (in Alpha, in Beta etc)

The channels for communicating, in general, are the same channels you might use for user recruitment - see User recruitment[14].

You might want to develop a formal comms plan for your API - in which case you should engage with the Comms team. At the time of writing, our contact is rebecca.wilson79@nhs.net[15] or catherine.farrell1@nhs.net[16]. Another useful ally is the Industry Engagement team - our contact is Helen Fisher.

14 https://nhsd-confluence.digital.nhs.uk/display/APM/User+recruitment
15 mailto:rebecca.wilson79@nhs.net
16 mailto:catherine.farrell1@nhs.net.

6.8 Get started as an architect or technical lead

6.8.1 Learn

Before you start, the following pages provide useful information to help you get into the tech:

- Architecting your API (see page 260)
- Wider architecture review (see page 290) - how does APIM, TRG (and other Architectural Governance), and your API Producer responsibilities interact
- Designing your API (see page 322)

Also, you do need to understand the delivery process:

- Deliver your API (see page 194)

Finally, below are some quick reference links to building your API:

- Designing your API (see page 322)
- Building your API alpha (see page 578)
- Securing your API (see page 774)
- Testing your API (see page 715)
- Monitoring your API (see page 1064)
- Documenting your API (see page 445)

6.8.2 Play

It you want to be hands on and make you API available in a testing environment, use our one-click tooling to get your repository, build and release pipeline set-up - see Set up and deploy your API (see page 580)

6.8.3 Engage

If your team is not already formally engaged with us, kick off our API delivery process, starting with the inception phase (see page 249).

6.8.4 Get set up

See Get access to tools, systems and environments (see page 144) on how to get access to various tools and environments for your role.

Further reading

- learn about the Benefits of using the API platform - What do I get as an API producer? (see page 135)
- learn about how the API platform[17] works

17 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/95229863/API+platform

- learn about the API Management Roadmap[18]
- learn about our Architecture[19]
- learn about the API delivery and support options (see page 293)
- learn about the entry/exit criteria for each delivery phase - see Alpha phase (see page 321), Beta phase (see page 944)

6.8.4.1 Help and support

You can get help and support from us for the API Platform and for general build/planning advice. See Help and support (see page 2056) page for more information.

> We can only offer minimal help and support until you formally engage with us.

6.9 Get started as a software developer or tester

Learn

Before you start delivering your API, you should learn about the API delivery process by reading Deliver your API (see page 194).

6.9.1 Play

Before you engage with us formally, if you want you can have a play with the tech. You can do this easily by trying the "try this API" feature in the "endpoints" section against any of the APIs in our API catalogue[20]. This will give you pre-canned responses, giving you a feel for what the requests and responses look like. A good place to start is with our exemplar API, the Personal Demographics (PDS) FHIR API[21].

Alternatively, if you want to get hands-on with the API platform, including authorisation and container backends, we can help set you up with a trial API in our non-production environment (follow the links below to find out how). As part of your trial, you'll get access to:

- an API proxy on our internal-dev environment
- a skeleton OAS file for your API specification
- a GitHub repository and CI/CD pipelines (see page 580) for you to use to update your API proxy config and OAS file, including publishing your API spec in our internal developer portal
- (limited) support from us via our API producer support Slack channel

---

18 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/95230918/Roadmap
19 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97747483/Architecture
20 https://digital.nhs.uk/developer/api-catalogue
21 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-endpoints

6.9.2 Engage

If your team is not already formally engaged with us, kick off our API delivery process, starting with the inception phase (see page 249).

6.9.3 Get set up

Before you can start building APIs on the API Platform you'll need to:

- get access to various tools and environments for your role - see Get access to environments, systems and tools (see page 144)
- in particular, Set up your developer environment (see page 153)
- create a new GitHub repository and CI/CD pipelines (see page 580)to deploy your changes to our `internal-dev` environment

6.9.4 Further reading

- learn about the API Platform[22], and what API Management are building
- learn about the Benefits of using the API platform (see page 134)

6.9.5 Help and support

You can get help and support from us for the API Platform and for general build/planning advice. See Help and support (see page 2056) page for more information.

We can only offer minimal help and support until you formally engage with us.

.

6.10 Get started as a technical author

Learn

Before you start, you should learn about:

- how to Deliver your API (see page 194)
- how to Document your API (see page 445)

22 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Platform

6.10.1 Engage

If your team is not already formally engaged with us, kick off our API delivery process, starting with the inception phase (see page 249).

6.10.2 Get set up

See Get access to tools, systems and environments (see page 144) on how to get access to various tools and environments for your role.

6.10.3 Further reading

- API platform[23], and what API Management are building.
- Benefits of using the API platform (see page 134) (and whether it's appropriate for you)
- NHS Digital A to Z of house style[24]
- GDS Style guide A to Z[25]

See Get access to tools, systems and environments (see page 144) on how to get access to various tools and environments for your role

6.10.4 Help and support

You can get help and support from us for the API Platform and for general build/planning advice. See Help and support (see page 2056) page for more information.

We can only offer minimal help and support until you formally engage with us.

6.11 Get access to tools, systems and environments

eSub pages:

- Get access to Confluence (see page 150)
- Get access to Slack and support channels (see page 151)
- Set up your developer environment (see page 153)
- Get access to Azure AD groups and Splunk (see page 161)
- Get smartcard and reader access (see page 165)

23 https://nhsd-confluence.digital.nhs.uk/display/APM/API+platform

24 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style

25 https://www.gov.uk/guidance/style-guide/a-to-z-of-gov-uk-style

- Get access to ServiceNow (see page 167)
- AWS HSCN VPN access (see page 168)
- Set up NHS Smart Cards on OS X (see page 169)
- Apigee prod access - Custom producer roles (see page 192)

6.11.1 Overview

This page tells you what you'll need access to as an API producer and how to get it.

It is the responsibility of producer teams to notify us and Platforms Resourcing of any leavers. To do so please use the 'Report a Leaver' workflow in the #platforms-apim-producer-support (for guidance on how to use workflows please see the pinned message in the channel) **and** email platformsresourcing@nhs.net[26].
Untracked leavers are a security concern so please make sure to action this promptly when necessary. Quartley audits are conducted to ensure this is being managed effectivley.

6.11.2 Key

Code	Description	Code	Description
PO	Product owner	DV	Developer
DL	Delivery lead	TS	Tester
BA	Business analyst	AR	Architect
UR	User researcher	TA	Technical author

Summary

26 mailto:platformsresourcing@nhs.net

Tool, system or environment	Usage	Who needs it?								How to get access
		PO	DL	BA	UR	DV	TS	TA	AR	
API producer zone Confluence space (see page 119)	API producer technical documentation and guidance on the API delivery process Provides copies of various checklists and templates	PO	DL	BA	UR	DV	TS	TA	AR	If you already have access to Confluence then you may be able to view this space, otherwise see Get access to Confluence[27] to get access for other team members.
Your own Confluence space	Writing your API's internal documentation.	PO	DL	BA	UR	DV	TS	TA	AR	All your documentation should be located under **your own project space in Confluence.** Depending on the permission on your own Confluence space you may need to contact your Administrator to get access your project space.
NHS Digital Slack[28]	API producer support from us and outbound notifications from us.	PO	DL	BA	UR	DV	TS	TA	AR	See Get access to Slack and support channels (see page 151).
Interactive product backlog (see page 250)	Gives API consumers and other API producers early visibility of your API and also enables them to vote on any features.	PO	DL	BA						Contact the API producer liaison to get access (currently Jonathan Eagle).

27 https://digital.nhs.uk/services/api-platform/accessing-confluence#top
28 http://nhsdigital-platforms.slack.com

Tool, system or environment	Usage	Who needs it?								How to get access
		PO	DL	BA	UR	DV	TS	TA	AR	
API register (see page 1204)	Shows all APIs that have been built (or are in progress) on the API platform and contains links to documentation in your Confluence space such as the process checklists, API architecture review etc...	PO	DL	BA						Contact the API producer liaison to get access (currently Jonathan Eagle).
API tracker[29]	A JIRA board (AMR) to track all APIs on the API platform	PO	DL	BA						Contact the API producer liaison to get access (currently Unknown User (dasi8))
Splunk (see page 1082)	Monitoring API traffic, alerting, analytics and diagnostics. There 2 redacted (personal data) dashboards which are have open access : • API management health dashboard which displays details of requests on the Platform across environments, proxies and applications. • APIM KPI reporting dashboard which displays KPIs for APIs in production and the applications that use them.	PO	DL	BA		DV	TS		AR	See Get access to Azure AD groups and Splunk (see page 161) to get access

29 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110871586/API+tracker

Tool, system or environment	Usage	Who needs it?								How to get access
		PO	DL	BA	UR	DV	TS	TA	AR	
Grafana (see page 1065)	Monitoring the health of all our APIs in all environments through two API endpoints: a "shallow" ping confirming the API proxy is up and running and a "deep" ping confirming that the back end service is up.	PO	DL	BA		DV	TS		AR	See Get access to Azure AD groups and Splunk (see page 161) to get access
Azure DevOps (see page 884)	Managing your API deployments.					DV	TS		AR	See Get access to Azure AD groups and Splunk (see page 161) to get access
Developer machine environment	Setting up your developer environment, security requirements for your machine					DV	TS		AR	See Set up your developer environment (see page 153)

Tool, system or environment	Usage	Who needs it?								How to get access
		PO	DL	BA	UR	DV	TS	TA	AR	
Apigee	Debugging your API proxies, updating API specifications for Previewing API specifications on the Bloomreach training environment (see page 515).					DV	TS		AR	**Non-Prod Org** Request an Apigee non-prod account by using our Slack workflow named 'Apigee Non-prod Account'. For guidance on how to use the workflow please see the pinned message in #platforms-apim-producer-support (for slack access see Help and support (see page 2056)). **Prod Org (which includes INT)** Permissions on prod are more tightly controlled so ask Producer Support for a custom role for your producer team if one does not already exist (they will follow Apigee prod access - Custom producer roles (see page 192)). **Note**: You must deploy to environments held in Apigee prod before requesting the role (i.e. int, sandbox)
GitHub account and access	To develop, store any code or documentation artefacts					DV	TS	TA	AR	See Set up your Github account (see page 153)

Tool, system or environment	Usage	Who needs it?								How to get access
		PO	DL	BA	UR	DV	TS	TA	AR	
Smartcard and reader (where appropriate)	Developing and testing of APIs requiring Health worker restricted Smartcard access					DV	TS			See Get smartcard and reader access (see page 165)
Bloomreach CMS - User Acceptance Test instance	To publish your API specification for preview in Bloomreach UAT.							TA		See Using Bloomreach CMS - Accessing Bloomreach (see page 509) See also AWS HSCN VPN below.
Bloomreach CMS - Production instance	Only needed if you also have some supplementary API documents to publish							TA		See Using Bloomreach CMS - Accessing Bloomreach (see page 509) See also AWS HSCN VPN below.
AWS HSCN VPN	If you are doing a lot of content authoring in Bloomreach, AWS HSCN VPN is the best way to do it (as opposed to over VDI).							TA		See AWS HSCN VPN access (see page 168)
ServiceNow	For raising RFC and dealing with incidents		DL			DV	TS			See Get access to ServiceNow (see page 167)

6.11.3 Get access to Confluence

To maintain a single source of truth, the instructions for accessing CDT Confluence have been moved to the external facing Accessing Confluence[30] page now visible on internet.

30 https://digital.nhs.uk/services/api-platform/accessing-confluence#top

6.11.4 Get access to Slack and support channels

6.11.4.1 Overview

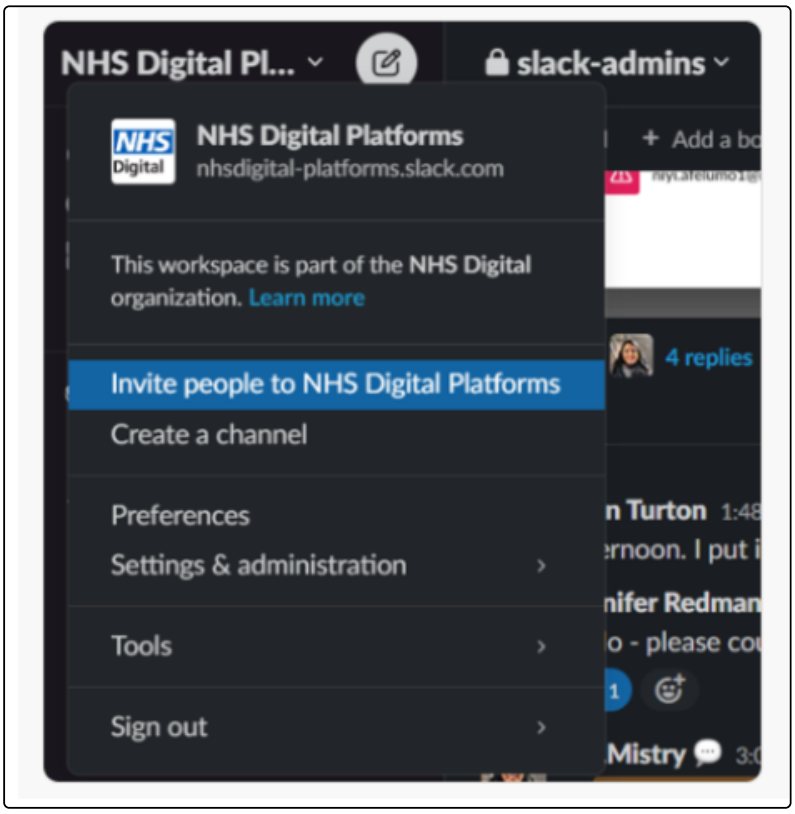
You will need access to the NHS Digital Slack workspace to get help and support from us when in the pre-inception, inception, discovery phases and also when delivering your API.

Note: If you can access or know a colleague that can access Slack, you can request access for yourself/ other team members by inviting them via NHS digital platform Slack workspace option (see page 151) ; otherwise you will need to follow the instructions described in Manual Slack request (see page 152) section.

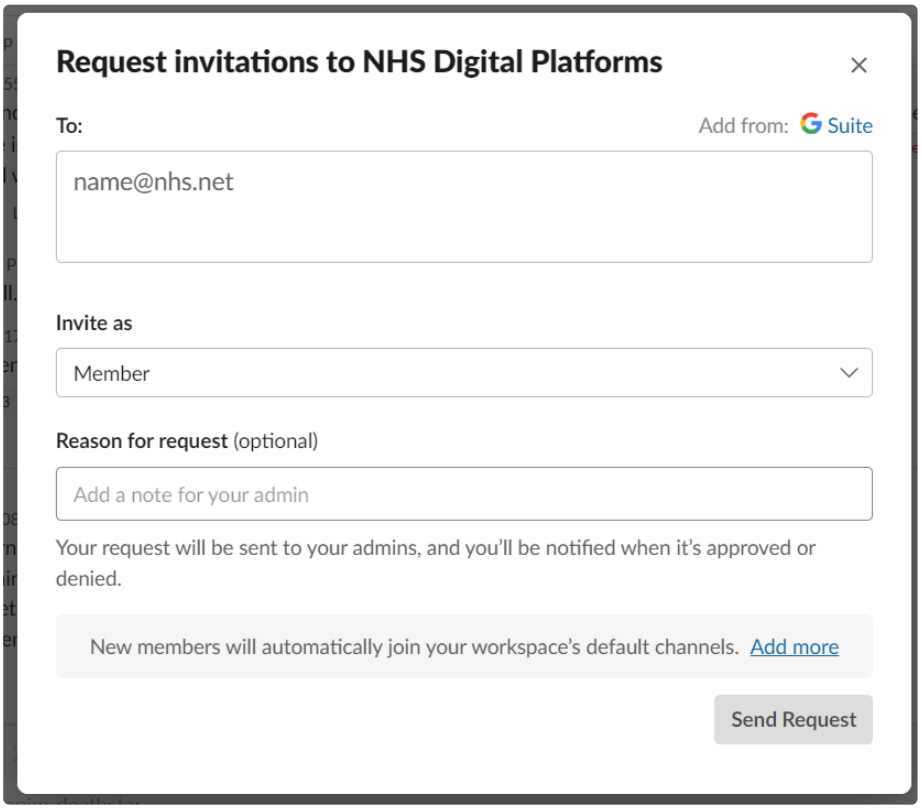
6.11.4.2 NHS digital platform Slack workspace

1.In Slack, select the 'NHS Digital Platform' Slack Workspace in the top-left-hand corner of Slack window.

2. Select the 'Invite people to the NHS Digital Platforms" option



3. Complete the 'request invitations' form



Request invitations to NHS digital platforms form

- **To** - The email of the person who needs access. It is preferable that the nhs.net email address is used (if not using a nhs.net email address see note below)
- **Invite as** - if using a nhs.net email select 'member'. If not using a nhs.net email address, select 'Guest'. Guests can only be added to one channel with a time limit (the 'set an expiration date' option becomes visible when the 'Guest' option is selected).
- **Reason for the request** - Explain why the person need access, what area of platforms they work on or if they're not platforms then the reason they need access. This will speed up your request being actioned.
- **Add more** option (hyperlink) - Select this option to specify which channels need to be added to. See Recommended Slack channels (see page 153) for guidance on which channels to include.

Once the request is sent it will then be reviewed by a Slack administrator.

6.11.4.3 Manual Slack request

If you do not have access to Slack or to a colleague with Slack access, contact: api.management@nhs.net[31]. Ensure that you include all details (where applicable) of the Slack invite request form in your email.

31 mailto:api.management@nhs.net

6.11.4.4 Getting access to Slack channels once in the the workspace

If you don't have access or cannot see certain channels in Slack follow NHS digital platform Slack workspace option (see page 151) to request access to any specific channel..

6.11.4.5 Recommended Slack channels

Channel Name	Purpose
#platforms-apim-producer-announcements ***(Archived)***	Announcements about changes, outages, etc
#platforms-apim-producer-support	Requesting support from the API Management platform team

6.11.5 Set up your developer environment

6.11.5.1 Prerequisites

6.11.5.1.1 Dev Machine

Your dev machine must meet the security requirements. (see page 158)

6.11.5.1.2 Github Account

1. Create an account at GitHub.com[32]:
    a. Use your @nhs.net[33] email address
    b. Username can be whatever you like, but using a format such as "`[firstname][lastname]-nhs`" helps colleagues to identify you
    c. *Note that this is classed as a personal/public account*
2. Email the NHSE GitHub Admins at nhsdigital.githubadmins@nhs.net[34] requesting to be added as a member of the NHSDigital GitHub Organisation[35]:

32 https://github.com/join
33 http://nhs.net
34 mailto:nhsdigital.githubadmins@nhs.net
35 https://github.com/NHSDigital

a. *NB: licenses are cross-charged and limited in availability so only request a license if you need to actively contribute to a git repository*
b. You will need to provide the following information:
    - Your full name,
    - Your GitHub username,
    - Your NHS.net[36] mail address,
    - Your HSCIC shortcode,
    - The name of the team you are working with,
    - The name and email address of a sponsor from your team.
c. Full details around making the request can be found here: https://nhs.sharepoint.com/sites/X26_EngineeringCOE/SitePages/GitHub-User---how-to-request-access.aspx

3. Once approved, you will receive an invitation via email to join the Organisation - you must configure MFA when you accept the invitation (authenticator app or security key only, SMS is prohibited[37])
4. To contribute to an APIM repository, you will need to be added to the api-gateway[38] team - click on the "Request to join" button. Then go to the Slack Platforms APIM Producer Support[39] channel asking for the request to be actioned (an APIM Github admin can pick up this request, or if none are available email the Github Admins at nhsdigital.githubadmins@nhs.net[40])

For permissions for the relevant API repo see the Tech Lead of your team.

For Tribe members there are two other teams that might be relevant:

- `api-gateway-admin`: full access to repos, for staff who need to manage platform related GitHub activities
- `apim-security[41]`: read-only access to all APIM repos and security alerts, for staff who don't need full admin but need to look at security related activity

Source control (see page 928) contains details of the default branching strategies and naming conventions.

6.11.5.2 Environment set up

6.11.5.2.1 Windows set up

Windows users should install Windows Subsystem for Linux (WSL)[42]. Any distro is fine, though ubuntu/debian are recommended.

36 http://NHS.net
37 https://www.icsynergy.com/2020/01/reconsider-using-sms-mfa-needs/
38 https://github.com/orgs/NHSDigital/teams/api-gateway
39 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY
40 mailto:nhsdigital.githubadmins@nhs.net
41 https://github.com/orgs/NHSDigital/teams/apim-security
42 https://docs.microsoft.com/en-us/windows/wsl/install-win10

6.11.5.2.1.1 Install Windows Terminal (Optional)

Then install Windows Terminal from the app store on windows:

https://www.microsoft.com/en-gb/p/windows-terminal-preview/9n0dx20hk701?activetab=pivot:overviewtab

6.11.5.2.1.2 Install Remote - WSL for VSCode

Open visual studio code extensions, then install the package called:

"Remote - WSL"

6.11.5.2.1.3 Open in VSCode

Once WSL and VSCode is setup we can now use it.

1. Open Terminal and type to switch to linux

   ```
   $ wsl
   ```

2. Then change to the relevant directory and then enter:

   ```
   $ code .
   ```

   Which will open a new window of VSCode and are now able to run linux

6.11.5.2.2 MacOS set up

6.11.5.2.2.1 Install GNU sed

GNU sed is used in a couple of places in API factory builds, so make sure you're using GNU sed and not the sed that is supplied with MacOS.

First, install sed (and if you don't have homebrew, install it):

```
$ brew install gnu-sed
```

Then, in your .profile, .bashrc, .zshrc or whatever floats your boat:

```
alias sed=gsed
```

6.11.5.2.3 Ubuntu/Debian/Spine Developer set up

For Spine developers use the Ubuntu 18 VM.

6.11.5.2.3.1 Installing requirements

Installing pyenv & python

Install build requirements. This will make sure you don't hit any weird python issues later.

```
sudo apt update
sudo apt install make build-essential libssl-dev zlib1g-dev libbz2-dev libreadline-
dev libsqlite3-dev wget curl llvm libncurses5-dev libncursesw5-dev xz-utils tk-dev
libffi-dev liblzma-dev python-openssl git
```

Install pyenv[43]. This is most easily accomplished using pyenv-installer[44]. (If you are not comfortable running straight into bash, download and read the script first.)

```
curl https://pyenv.run | bash
exec $SHELL
```

Install the latest stable python 3.

```
pyenv install 3.8.2
```

if the above command throws error **Command 'pyenv' not found** then try setting up pyenv path by executing this command and then try installing python 3.8.2 again.

```
export PYENV_ROOT="$HOME/.pyenv"
export PATH="$PYENV_ROOT/bin:$PATH"
eval "$(pyenv init --path)"
```

Either set this as your global python (if this is not incompatible with your other projects),

```
pyenv global 3.8.2
```

or local to repository, if there is not a python-version file installed (you might have to raise a PR to add the file that's created).

43 https://github.com/pyenv/pyenv
44 https://github.com/pyenv/pyenv-installer

```
pyenv local 3.8.2
python --version
```

Installing Poetry

Next, install poetry.

```
pip install --user poetry
```

Installing nvm & npm

Next up, install nvm[45], and use that to install node.

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.35.2/install.sh | bash
# Close and reopen your terminal window (or run perform the steps mentioned in the
run script)
nvm install stable
nvm use stable
npm --version
```

Install Java

```
sudo apt install default-jre default-jdk
java -version
```

6.11.5.2.3.2 Repository Setup

Talk to the tech staff in your team to find out which repositories you need adding to. Once you are added, follow the repository instructions to get started. Source control (see page 928) contains details of the recommend branching and naming conventions.

6.11.5.2.4 Troubleshooting

If you are having problems setting up your machine see Troubleshooting (see page 160)

45 https://github.com/nvm-sh/nvm

6.11.5.3 Security requirements

6.11.5.3.1 Machine Requirements

6.11.5.3.1.1 Overview

Here is a summary of requirements for machines:

- Encrypted storage devices
- Password Manager
- Multi-factor auth app/and or device
- Antivirus
- Up-to-date OS and software

6.11.5.3.1.2 Encrypted Storage Devices

All permanent or regularly used storage devices **must** be encrypted using BitLocker, FileVault, LUKS, eCryptFS, fscrypt, or similar.

6.11.5.3.1.3 Password Manager

You **must** use a password manager to store your work passwords (and you really should be using one in your personal life).

You **must** disable the facility in your browser of choice for storing passwords.

You **must not** use the same master password for your password manager that you use for anything else.

Popular cloud password managers include:

- Lastpass
- OnePassword
- Dashlane

Popular offline password managers include (you must ensure regular backups are made):

- Keepass
- pass

6.11.5.3.1.4 MFA

You **must not** use SMS as an MFA method as SIM cards are trivial to hijack. Not even as a backup. It is more secure to print backup codes and keep them somewhere safe.

You must use an MFA device with all accounts that support it, such as:

- Authy
- Google Authenticator

- Microsoft Authenticator
- Yubikeys
- FIDO U2F Devices
- Integrated U2F devices

6.11.5.3.1.5 Antivirus

You should have a modern antivirus, **and its definitions should be kept up to date**.

Windows

Windows defender is perfectly sufficient for most uses.

MacOS

Avira or similar are recommended.

Linux

ClamAV is well regarded on linux. Make sure you have set it up to automatically update at least once a day.

6.11.5.3.2 Everyday Security

6.11.5.3.2.1 Lock your laptop

Lock your laptop if you walk away from it.

6.11.5.3.2.2 Don't allow tailgaters orshoulder surfers

If someone tries to get in a door behind you, or hovers over your shoulder (who you don't know or recognise), ensure they are allowed in the building/are part of your team.

6.11.5.3.2.3 Consider using privacy filters

This will prevent shoulder surfing.

6.11.5.3.2.4 Password Strength

The most important contributor to password entropy is length. Make a long password that is easy for you to remember, or generate long passwords in a password manager.

20 and above is recommended.

Diceware with at least 6 dice is a good way to generate passwords: http://world.std.com/~reinhold/diceware.html

https://diceware.dmuth.org/

6.11.5.3.2.5 SSH Keys

Generate SSH Keys with at least 4096+ bits on RSA or 256+ bits on ECDSA or Ed25519. It always pays to overdo it by a factor of two.

6.11.5.4 Troubleshooting

Please add to this list if you discover any other errors when running local tests.

6.11.5.4.1 Guides

6.11.5.4.1.1 Unhelpful errors from the makefile

When trying to run tests locally you may run into unhelpful errors caused by the makefile. First try updating the poetry dependencies by running

```
poetry update
```

6.11.5.4.1.2 Poetry using the wrong Python version

If poetry seems to be using an outdated version of Python but your system version and pyenv version seem to be correct, run

```
poetry env use $(which python)
```

6.11.5.4.1.3 update_version throwing gitdb error

If the update_version.py script gives you errors regarding gitdb but you can see it installed after running

```
poetry run pip list
```

it may be caused by smmap. Try running

```
poetry run pip3 install --upgrade --force-reinstall gitdb; poetry run pip3 install --
upgrade --force-reinstall smmap
```

6.11.6 Get access to Azure AD groups and Splunk

6.11.6.1 Overview

Currently, access to the following resources is controlled by Azure AD groups:

- Prod Apigee: https://nhs-digital-prod.login.apigee.com/login
- Azure DevOps: https://dev.azure.com/NHSD-APIM/
- Monitoring: https://grafana.ptl.api.platform.nhs.uk/
- JWKS Helper Service: https://jwks-helper.ptl.api.platform.nhs.uk/
- One click API Setup: https://setup-api.ptl.api.platform.nhs.uk[46]

Also access to the following resource (and various indexes within that resource) is controlled by Splunk groups

- Splunk Cloud: https://nhsdigital.splunkcloud.com/

6.11.6.2 What type of access do you need?

Developer Access

Typically Developers will require access to the following groups

- GRP-AzureNHSD-API-Management-PTL-Producers-DL
- GRP-AzureNHSD-API-Management-PROD-Producers-DL
- GRP-SPLUNK-CLOUD-APM-PROD-PRODUCER

6.11.6.2.1 Splunk Groups

We send logs from the APIM Platform ecosystem to Splunk Cloud at https://nhsdigital.splunkcloud.com/. Access to the logs from those environments is controlled by allowing users access to the Splunk indexes. All devs, testers, etc. on the team should request access to the PTL indexes.

Our indexes are split between PTL and Prod, but they are all contained in the Splunk Prod instance. It is only Deathstar / Platform teams that would look at Splunk PTL.

APIM has two Splunk Apps:

- APIM Open Dashboards (nhsd_apm_public_all_sh_all_viz)

46 https://setup-api.ptl.api.platform.nhs.uk/

- APIM DevOps (nhsd_apm_all_sh_all_viz)

Both these apps allow access to the APIM indexes, but segregate the reports, dashboards, alerts between more platform operations and developer focused data vs a simpler / redacted view of the data.

> ⚠ You will not be given access to any PID data in the APIM logs - if you require PID access you can do that in your backend. APIM implemented the redacted indexes as a simple and easy way to grant access to all other critical log data. Read more about log indexes here: Logging to Splunk (see page 1082).

Currently there are only three groups for normal Splunk users:

Group	Env	Who
GRP-SPLUNK-CLOUD-APM-PTL-USER	PTL	For non technical staff who also need to look at PTL open dashboard redacted view. (This gives access to fewer indexes)
GRP-SPLUNK-CLOUD-APM-PROD-USER	Prod	For non technical staff who don't have an existing Splunk login, but need to see Open Dashboards redacted view in production
GRP-SPLUNK-CLOUD-APM-PROD-PRODUCER	Prod & PTL	API Producer devs - the Prod index has access to Client IP addresses so this should **only** be given to staff who have a real use case. This gives more access to more indexes

6.11.6.3 Requesting Splunk access

For access to the above Splunk groups, please notify the Splunk group data custodians by following the steps :

Send an email addressed to vikas.nautiyal1@nhs.net[47] and aubyn.crawford1@nhs.net[48] with the following information:

- The text "Data custodian notification"
- Your name
- Your shortcode
- Your team name
- Your api product name

47 mailto:vikas.nautiyal1@nhs.net
48 mailto:aubyn.crawford1@nhs.net

- Your role in the team
- The Splunk groups you are requesting to join

Don't wait for response, you have completed this step.

At this point **requesting Splunk access is complete (although it takes a short while for it to be applied to your account).** If you also need **azure group access**. then see Requesting Azure AD group access

6.11.6.3.1 Azure AD Groups

The following are the APIM AD Groups. They **do not** directly manage RBAC for many apps, but provide entry to an application - the users then need to be configured with the correct permissions:

Group	Env	Who
GRP-AzureNHSD-API-Management-PROD-Producers-DL	Prod	API Producer Teams. Useful for developers.
GRP-AzureNHSD-API-Management-PROD-DL	Prod	View access in production (primarily Grafana). Useful for product owners, business analysts, and non-technical users
GRP-AzureNHSD-API-Management-PTL-DL	PTL	View access in PTL for less technical tribe members, e.g. product owners, business analysts, who still want to see some PTL resources
GRP-AzureNHSD-API-Management-PTL-Producers-DL	PTL	For developers in API producer teams, including Azure DevOps

For adding users to the APIM Azure AD groups above, please request APIM authorisation by following the steps below:

6.11.6.4 Requesting Azure AD group access (*Change in request process introduced in April 2024*)

NOTE: For all groups (except *GRP-AzureNHSD-API-Management-PTL-DL and GRP-AzureNHSD-API-Management-PROD-DL*) further approval is required.

1. Please send an email to Navin Bose and Fayez Ahmed to request permission to obtain access to the groups beforehand.

Example email:

~~~
~~~

NOTE: DO NOT include access request for **Splunk AD Groups** (any starting with GRP-SPLUNK-*) with your **Azure AD group** request. For Splunk, please follow the guidance on the top of the page for "Requesting access to APIM Splunk Groups"

6.11.7 Get smartcard and reader access

> Smartcards are only supported on Windows devices at the moment. If you are not using a Windows device, contact `#platforms-apim-producer-support` on Slack for guidance.

6.11.7.1 Overview

NHS smartcards are similar to chip and PIN bank cards, and enable healthcare professionals to access clinical and personal information appropriate to their role, through a smartcard reader. If this smartcard is removed whilst logged in, the access you had will be revoked. To regain access all that is required is to re-enter the smartcard and restart the log in process.

You can see a full guide about Smartcards here: https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/nhs-smartcards-for-developers

6.11.7.2 Getting set up

The following steps are required to get set to use a smartcard and reader:

1. Order a smartcard reader.
2. Order a smartcard for the Path To Live (PTL) environments.
3. Set up your development machine by downloading all the necessary software. **The software download links can only be accessed via a VDI session or if you are connected to the VPN.**

Once all the above steps are completed you will be able to Log in using your Smartcard (see page 166)

6.11.7.2.1 Ordering a Smartcard reader

Send email with addresses to: techservices.mgmtoffice@nhs.net[51].

(NB If you are WFH give your address details and the reader will be posted out to you at home)

6.11.7.2.2 Ordering a Smartcard for PTL environments

To request a Smartcard for testing purposes you will need to identify which roles and business functions you would like setting up on the card and all roles and business functions can be found at:

51 mailto:techservices.mgmtoffice@nhs.net

https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/national-rbac-for-developers

> ⚠ Very important to note that, at this stage, for APIs that go to Spine endpoints only one generic role is need on that card: **R8015 (Systems Support Access Role)**
> This is because there is no National RBAC Authorisation happening on API Management - this is planned for later phases.

Once you have identified your roles and business functions you will need to complete the online form https://digital.nhs.uk/forms/smartcard-request-for-the-path-to-live-environments

- The role should be as production R8015 (see above)
- The ODS code for INT is T141D
- No new positions are needed

6.11.7.2.3 Setting up your machine

Your developer machine will need to be configured to use the card and reader(Smartcards are only supported on Windows devices). **NB This is done on your machine and not on your VDI.**

Note: To be able to access the software download link below, you will need to use your VDI connection or be connected to the VPN

Use this link http://nww.hscic.gov.uk/dir/downloads to install the following software:

1. Identity agent - ensure you select NHS Test Environment Certificates
2. NHS Credential Management
3. Smartcard reader drivers. Note: you only need one of these drivers so read carefully before selecting one
4. The correct sub and Root CAs for Integration depending on which environment you have requested on your smartcard - see https://digital.nhs.uk/services/path-to-live-environments/integration-environment#rootca-and-subca-certificates
5. IA registry tool editor

6.11.7.3 Log in using your Smartcard

1. Connect to VPN
2. Connect Smartcard reader
3. Put in the Smartcard
4. Login with pin 1234 when prompted

6.11.7.4 Spine application URLs

See spine application urls[52] for a full list of URLs for the applications in the integration environments.

Note - some of these applications only run in internet explorer.

6.11.7.5 What to do if locked out

If you are locked out of your Smartcard device contact itoc.supportdesk@nhs.net[53]

6.11.7.6 Mac OS

If using Mac OS please refer to this[54] document instead.

6.11.8 Get access to ServiceNow

6.11.8.1 Overview

To access ServiceNow you will need :

- ServiceNow account
- an "assigned group" set up for your team
- be allocated to your team's "assignment group"

6.11.8.2 Register for a ServiceNow account

ServiceNow requires it's own account and password details, and is not part of single sign-on.

1. Go to https://nhsdigitallive.service-now.com/csm
2. Click 'REGISTER' in the top right corner to create a ServiceNow account.

6.11.8.3 Set up a new assignment group

If this is the first time your team has used ServiceNow, it is likely that you won't have an assignment group set for your team and will have to request one to be set up.

To request a new assignment group email ssd.nationalservicedesk@nhs.net[55], with the following information:

- desired team name

52 https://digital.nhs.uk/services/path-to-live-environments/integration-environment#spine-application-urls
53 mailto:itoc.supportdesk@nhs.net
54 https://nhsd-confluence.digital.nhs.uk/display/APM/Setting+up+NHS+Smart+Cards+on+OS+X?moved=true
55 mailto:ssd.nationalservicedesk@nhs.net

- team manager name and shortcode- they will be responsible for approving any changes to the group members
- team member names and shortcodes
- group email address

6.11.8.4 Add team members to an assigned group

If you have Set up a new assignment group (see page 167) then you may have already added team members to your group. An easy way to check if someone is already assigned to a group, is by navigating to https://nhsdigitallive.service-now.com/ . If they are able to get access and can see something, then this means that they are already member of group. To check which group they are in, they can search for their name using the magnifying glass (top right) and then clicking on their name to view the groups they assigned to.

To add additional members or to reassign team members, anyone who is a member of the assignment group can raise a request by emailing ssd.nationalservicedesk@nhs.net[56] with details of members, their shortcodes and the name of the group they need to be assigned to.

6.11.9 AWS HSCN VPN access

6.11.9.1 Overview

If you only need Virtual Desktop Infrastructure (VDI) to access Bloomreach or other network restricted applications; then using Amazon Web Services (AWS) Heath and Social Care Network (HSCN) Virtual Private Network (VPN)[57] is a better option as it allows you to use your browser natively, giving better performance than through your VDI.

Do NOT use the Slack workflow, it no longer works... KOP-004 - Adding a user to the AWS SSO HSCN VPN[58] is the new starting point.

The process for requesting access is as follows:

56 mailto:ssd.nationalservicedesk@nhs.net
57 https://nhsd-confluence.digital.nhs.uk/spaces/AWS/pages/108276927/AWS+HSCN+VPN+Documentation
58 https://nhsd-confluence.digital.nhs.uk/spaces/AWS/pages/108277056/KOP-004+-+Adding+a+user+to+the+AWS+SSO+HSCN+VPN

6.11.9.2 If you have access to nhsdigital-platforms.slack.com[59] #platforms channel

1. Click the lightning bolt
2. Select "Request AWS VPN Account" and fill in the form
3. FInd out who the AWS VPN approver is for the area of NHS Digital you work in (ask your Delivery Manager if you don't know)
4. Ask your approver to approve your shortcut request
5. Receive email with your VPN credentials
6. Install OpenVPN[60], or another VPN tool of your choosing
7. Install the config file from the email into OpenVPN :
    - Open OpenVPN tool from your machine
    - When promoted to create a profile (will only happen 1st time in). Here you will have the option to import a file which should be the config file from your email
8. Connect to the VPN
9. Enter the user name and password details from the email to connect
10. Navigate to https://cms.digital.nhs.uk[61] to test it's working

6.11.9.3 If you don't have access to nhsdigital-platforms.slack.com[62] #platforms channel (or you don't know who your approver is)

Contact awsadmins@nhs.net[63] to ask for support

6.11.10 Set up NHS Smart Cards on OS X

6.11.10.1 Overview

This page provides guidance on how to get NHS smart cards working on Mac devices.

If you do not already have a smart card and a smart card reader please refer to this (see page 165) guide and complete all stages before **Setting up your machine**.

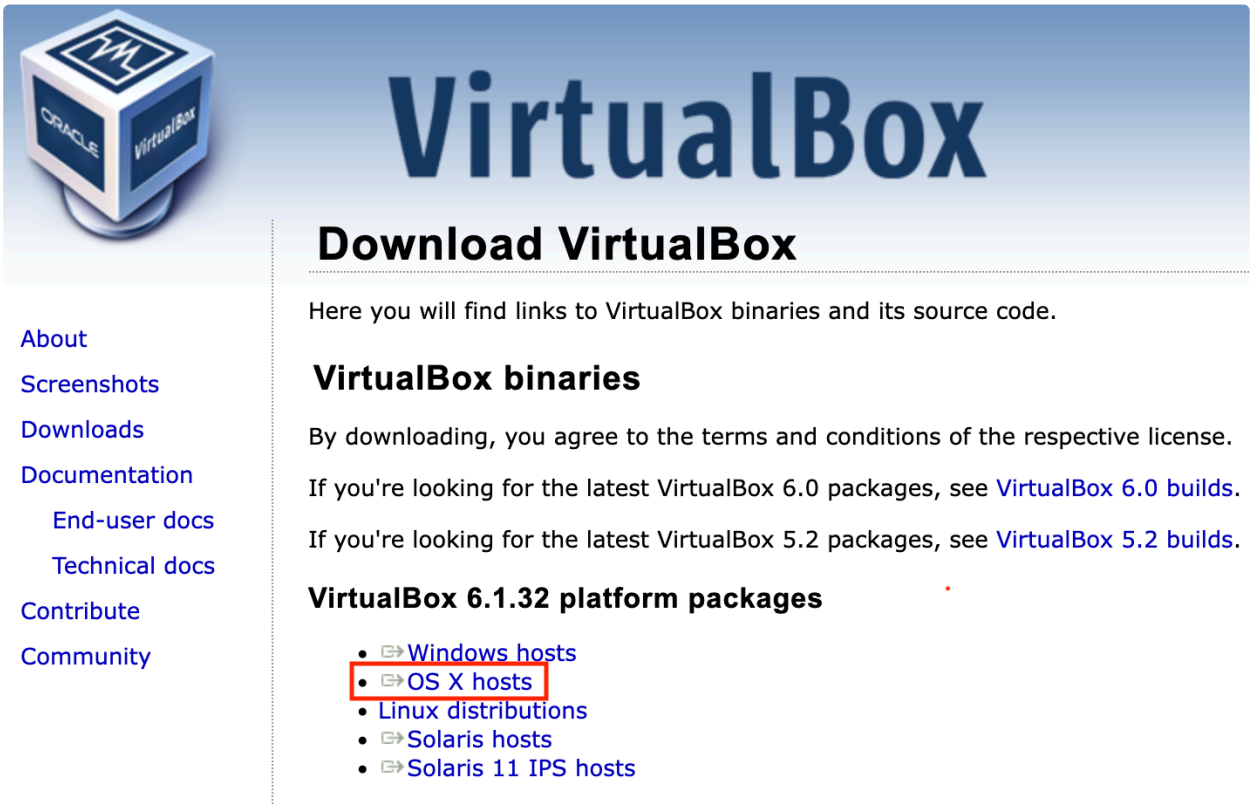
The software used to support smart card integrations is only compatible with **Windows** operating systems so in order to use one on Mac we have to use a piece of VM software called **VirtualBox.**

59 http://nhsdigital-platforms.slack.com
60 https://aws.amazon.com/vpn/client-vpn-download/
61 https://cms.digital.nhs.uk/
62 http://nhsdigital-platforms.slack.com
63 mailto:awsadmins@nhs.net

6.11.10.2 1. Setting up VirtualBox

6.11.10.2.1 1.1 Installation

To get started download the VirtualBox installer by following this link[64] and clicking **OS X Hosts** as shown below:

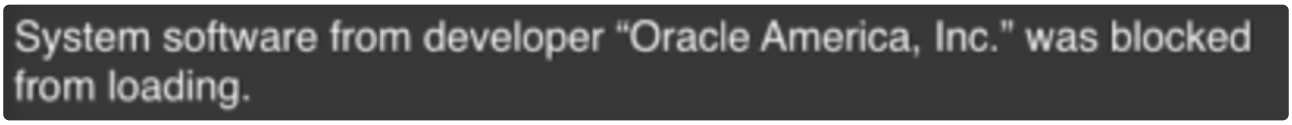


Once it has downloaded run the installer, I recommend using the default installation settings.

6.11.10.2.2 1.2 Verifying the installation

Click the magnifying glass in the top left corner and search for **Security and Privacy.**

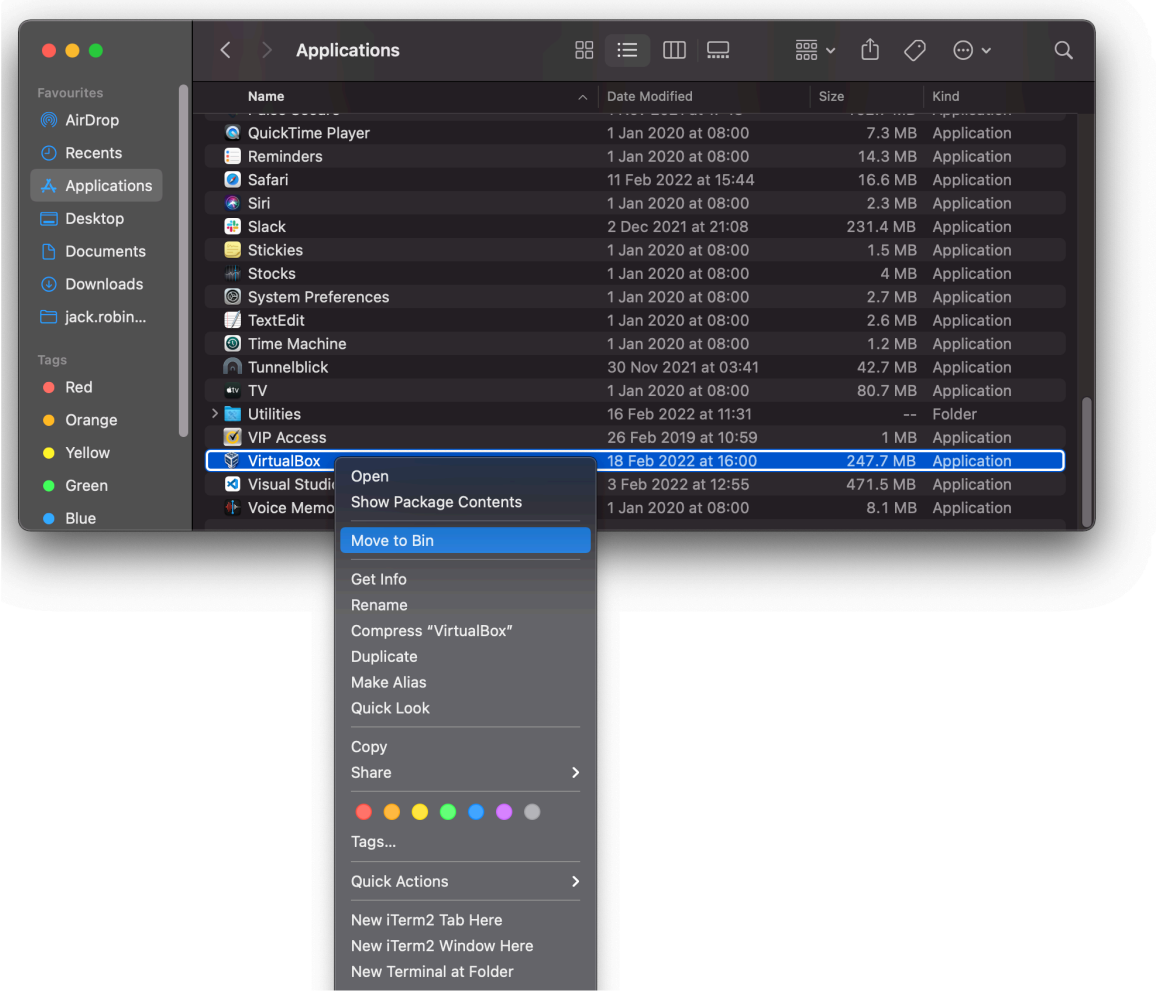
Navigate onto the **General** tab, at the bottom you will likely see the text shown below:



If you don't, move onto step **1.3**. Otherwise, click the padlock in the bottom left corner, click **Allow** and then click the padlock again. We now need to reinstall VirtualBox.

Open **Applications** in the finder, right click **VirtualBox** and click 'Move to Bin' to uninstall.

64 https://www.virtualbox.org/wiki/Downloads

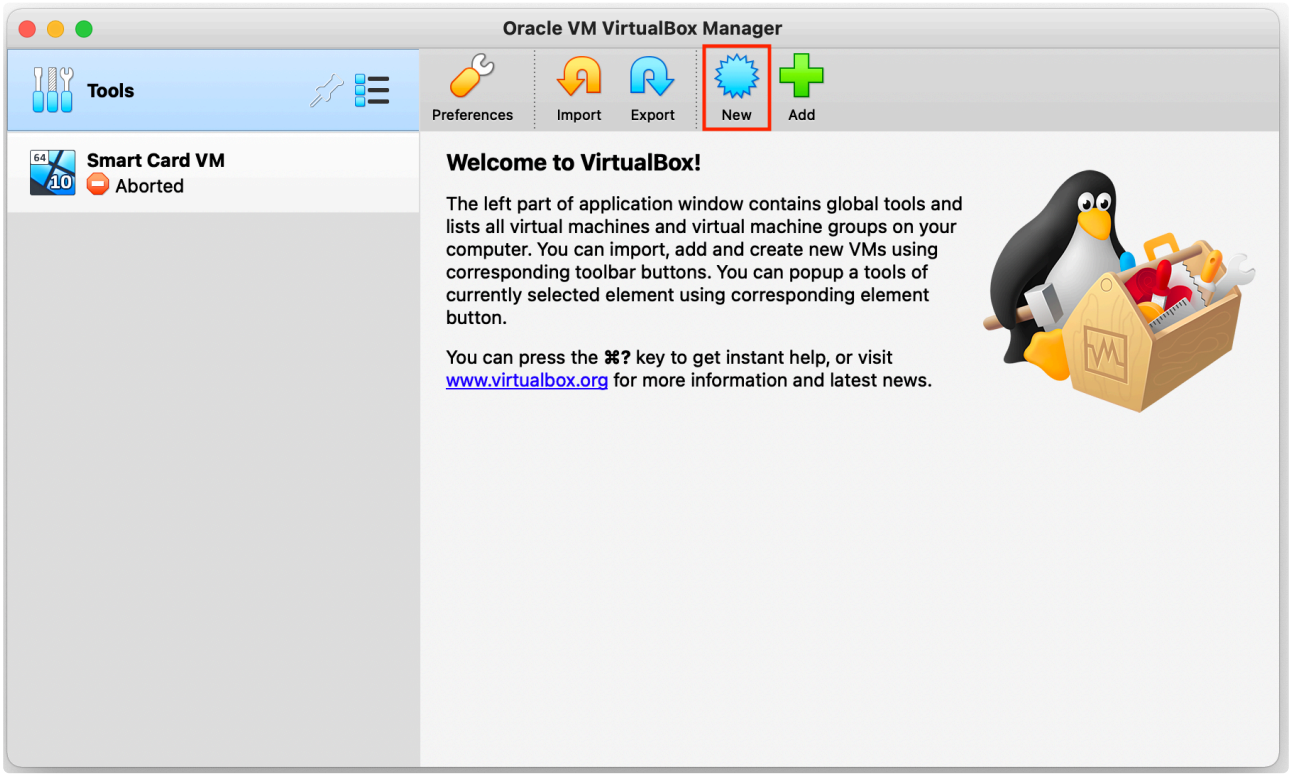


Now run the installer again just as before in step **1.1**.

6.11.10.3 2. Installing Windows 10 on VirtualBox

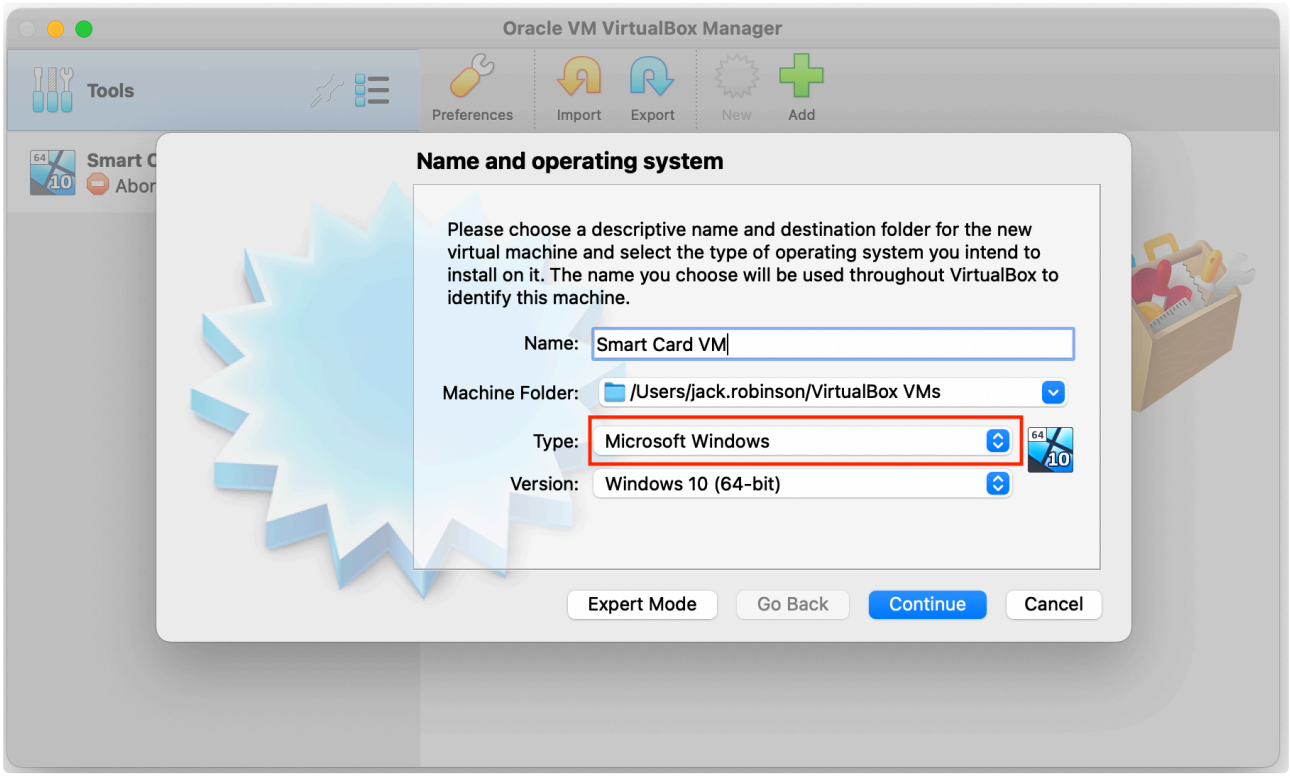
6.11.10.3.1 2.1 Create the virtual machine

Find and run VirtualBox, when it opens click on the **New** button at the top.

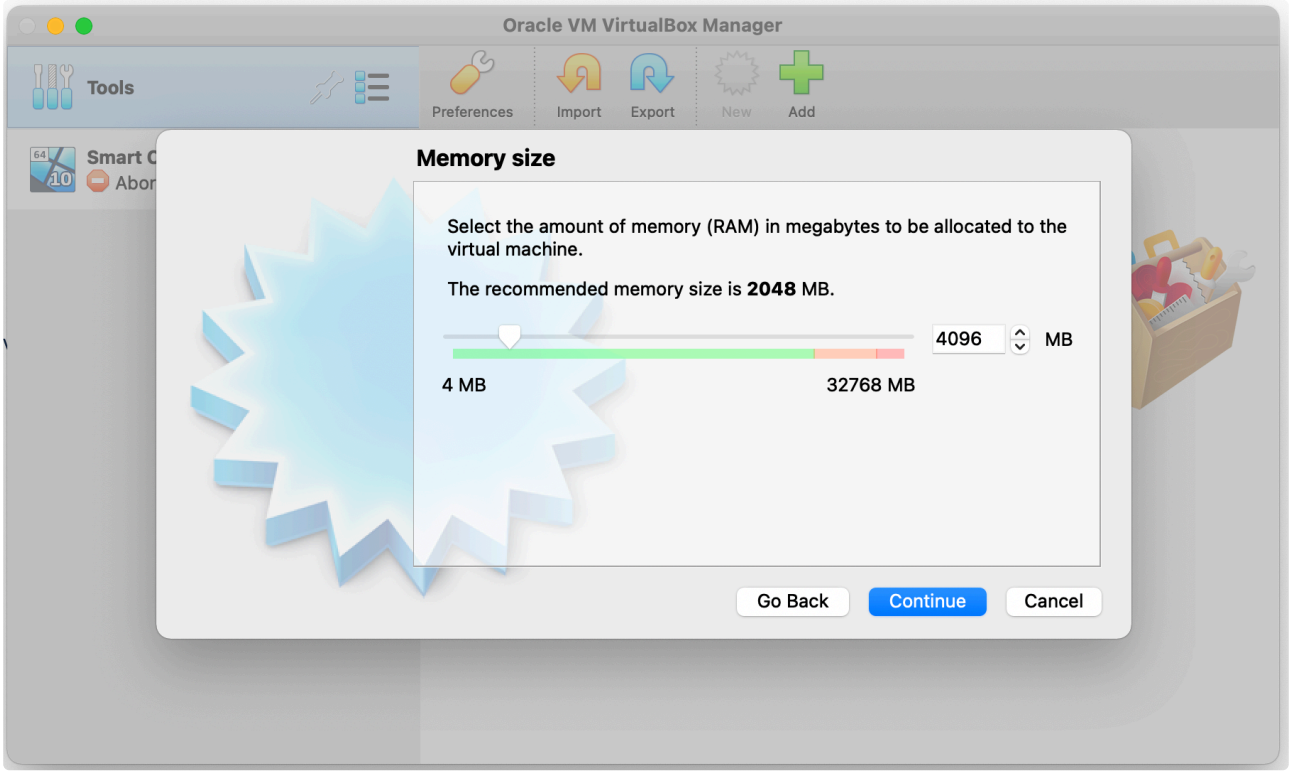


On the following screen chose a name for your VM (I used **Smart Card VM**) and change the **Version** to **Windows 10 (64-bit)**.

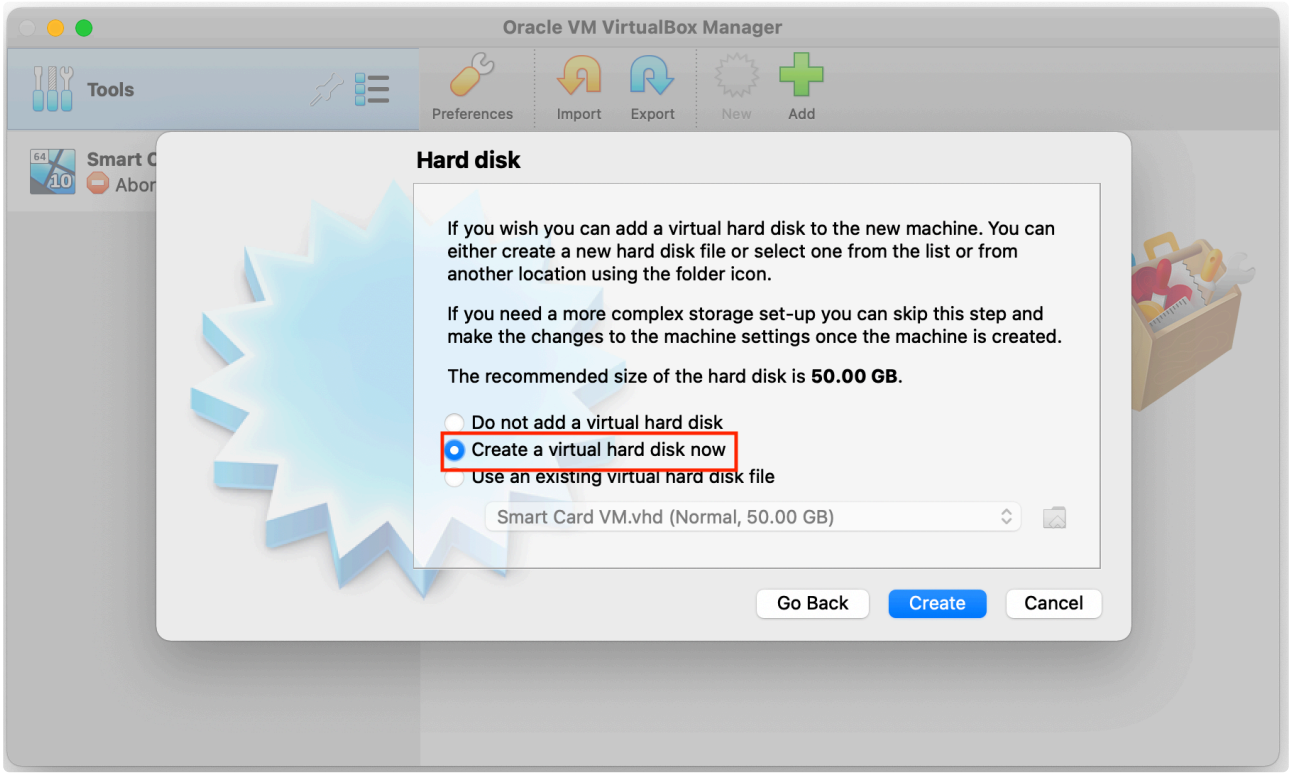
Note - the NHS smart card software is supported for many versions of Windows, if you wanted to you could try to use an older version that may result in a less resource intensive virtual machine (e.g. Windows 7) but for simplicity I have chosen to stick with Windows 10.

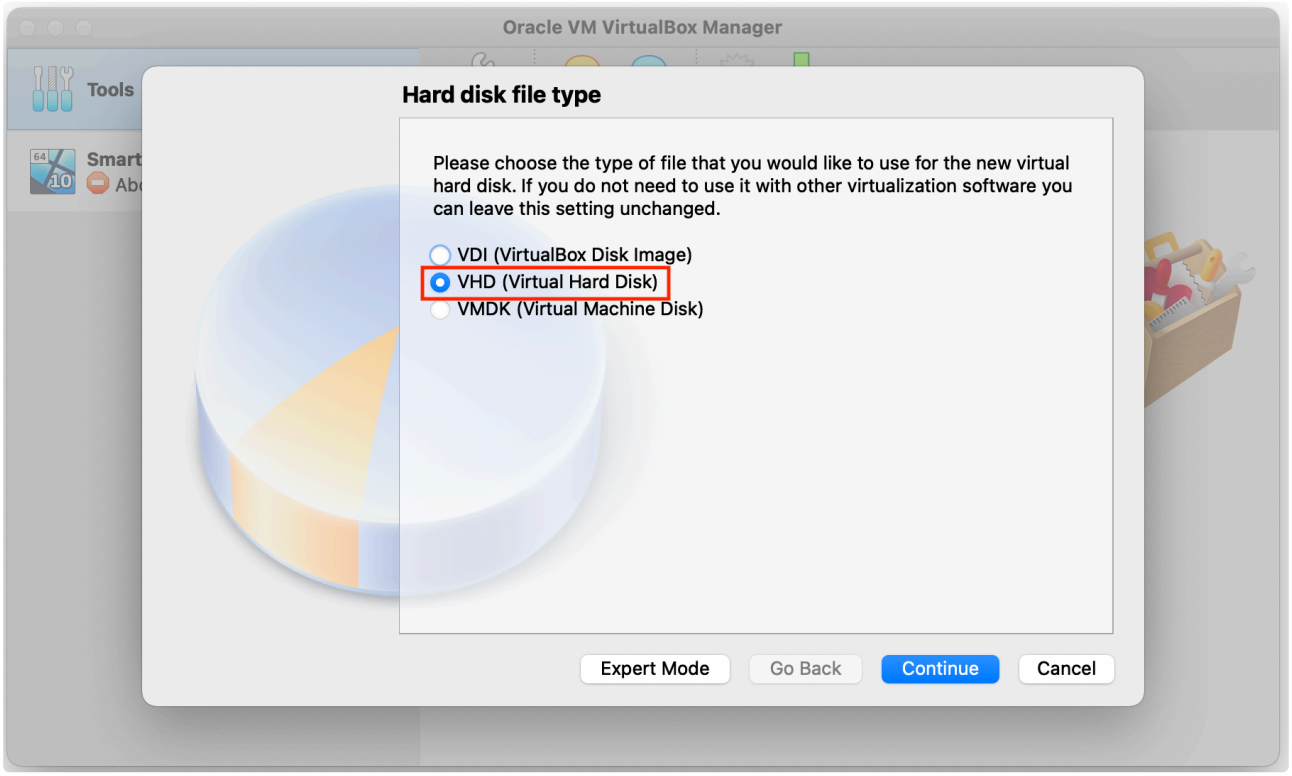


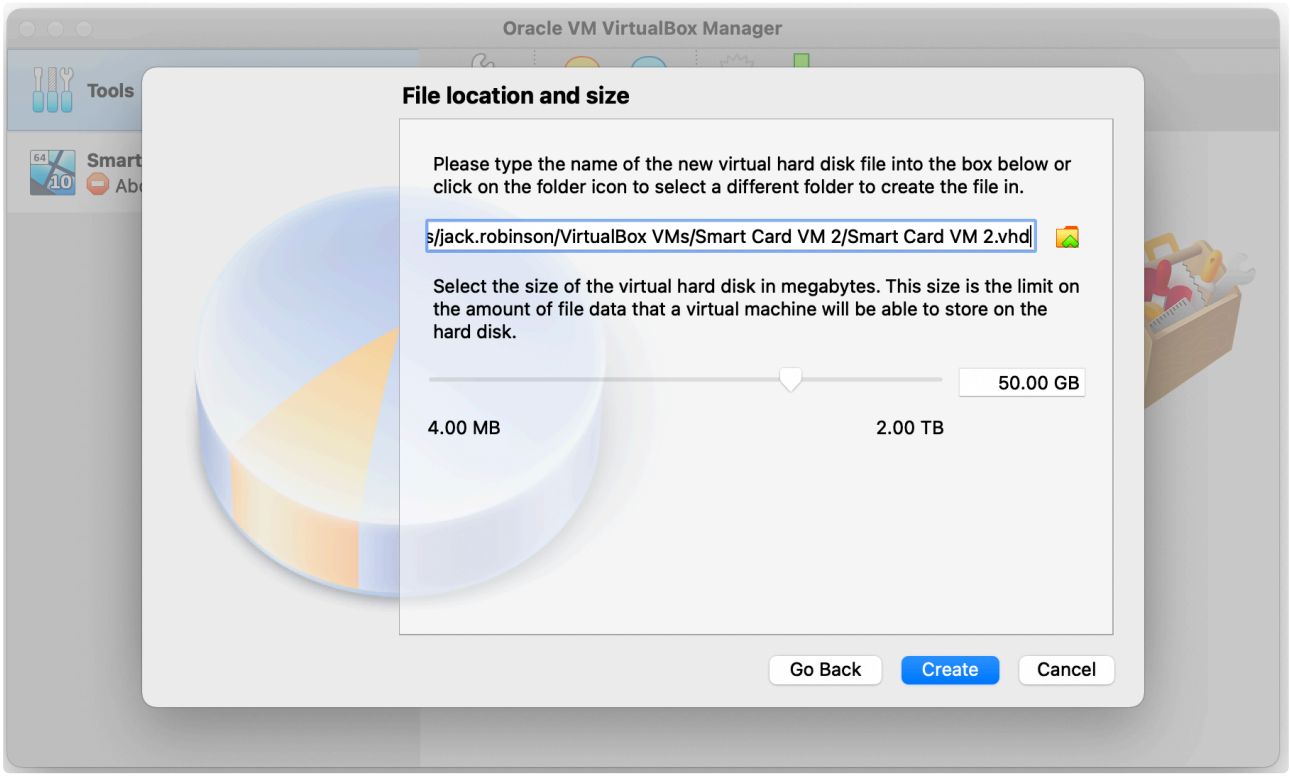
Next you will be asked to choose the memory allocation for the VM, the default value is fine but if your laptop has a reasonable amount of memory I recommend bumping it up. I have 32gb of ram on my laptop and so gave the VM 4096mb.



Add a virtual hard disk by clicking **Create**, selecting **VHD (Virtual Hard Disk)**, click **Continue** twice and finally **Create**.



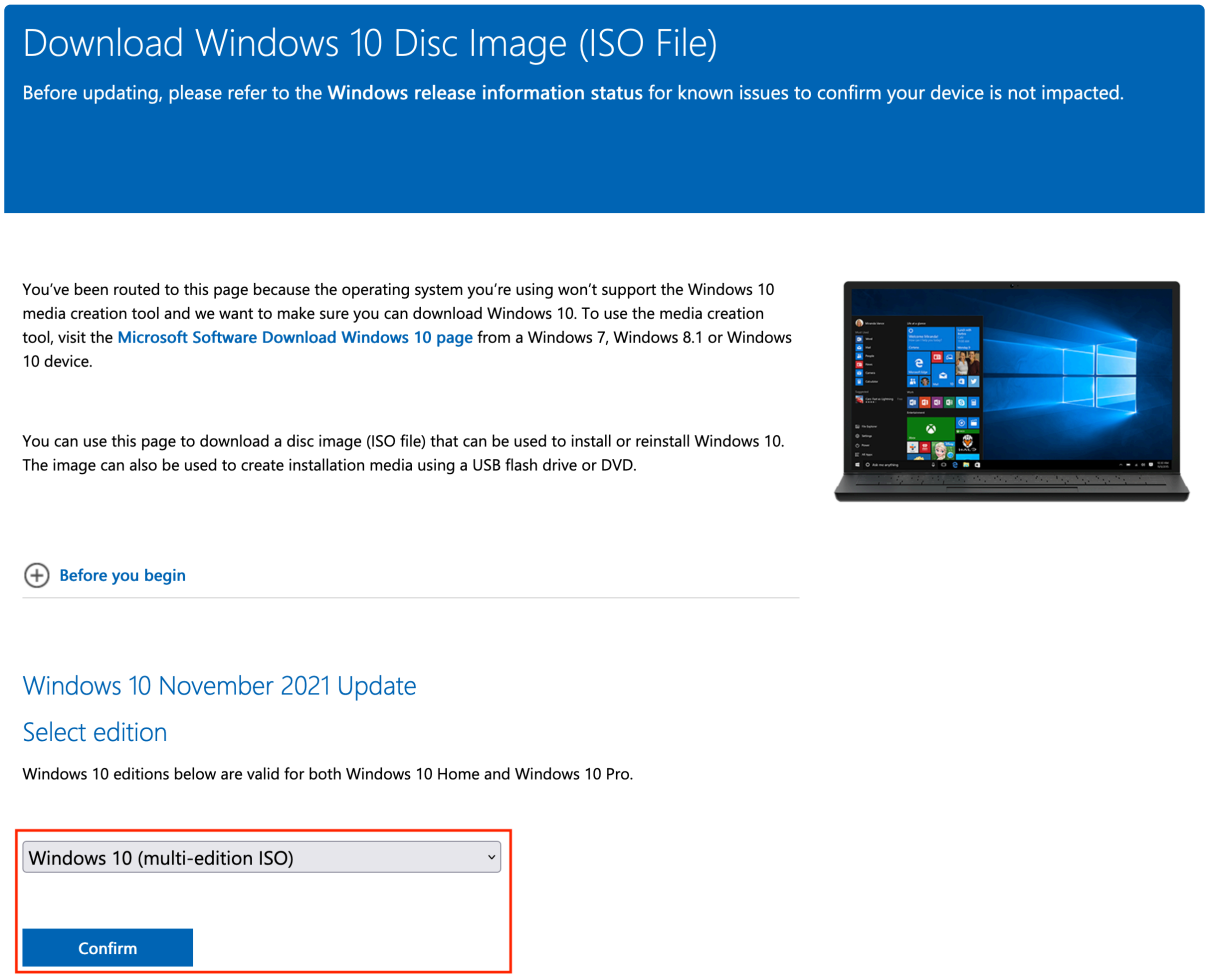




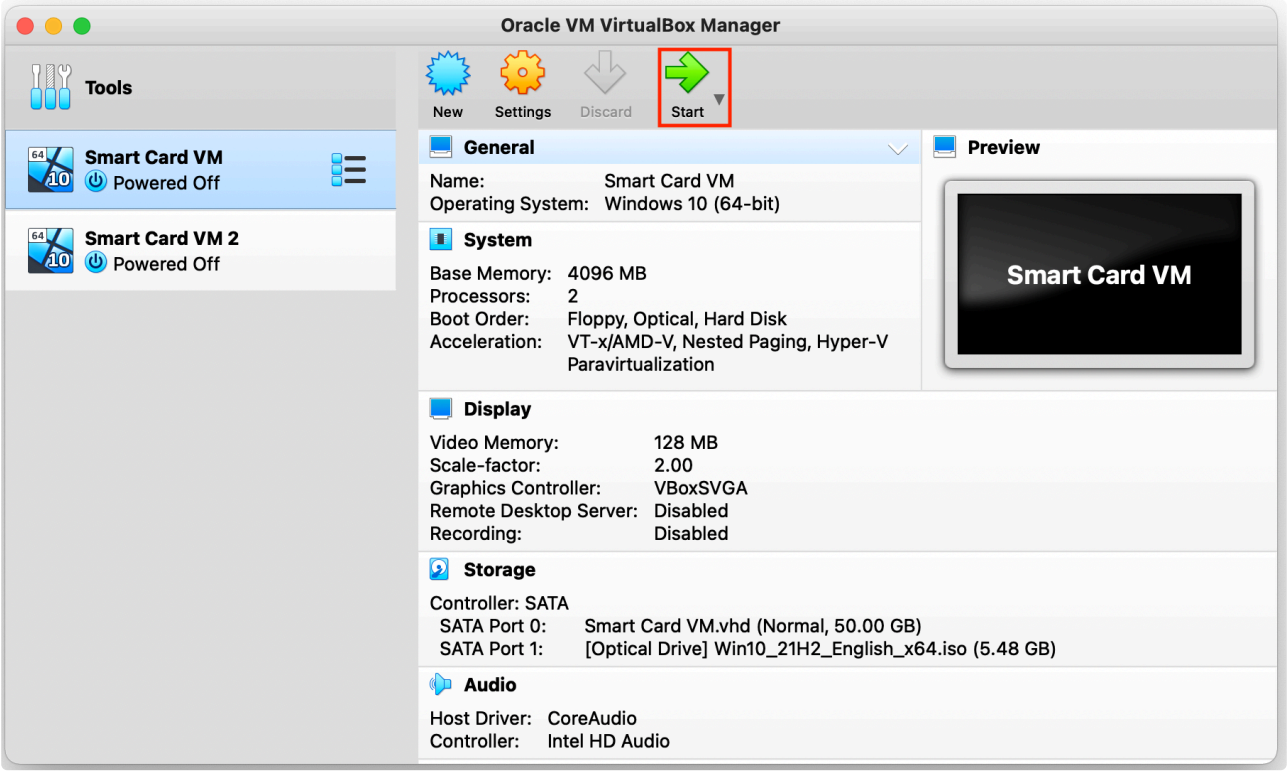
6.11.10.3.2 2.2 Installing Windows 10

Next download the Windows 10 ISO[65] by selecting the edition and clicking **Confirm.**

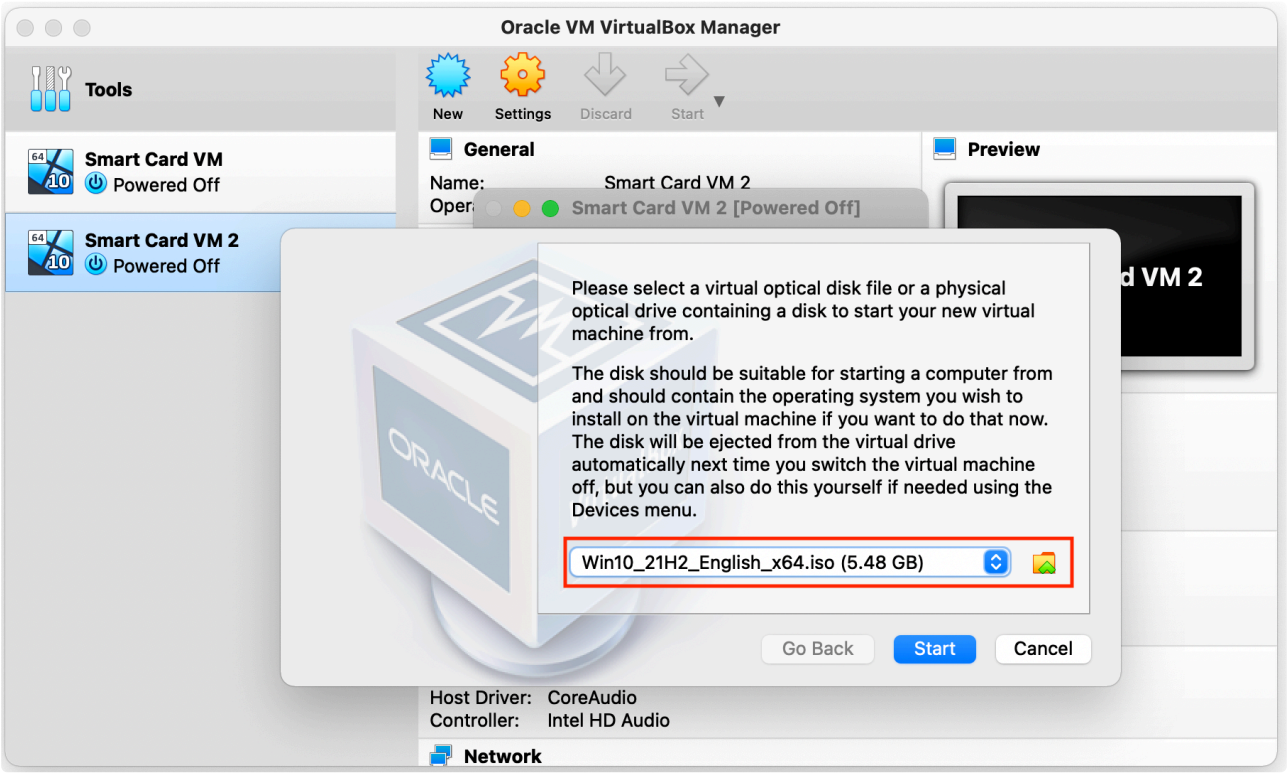
65 https://www.microsoft.com/en-us/software-download/windows10ISO



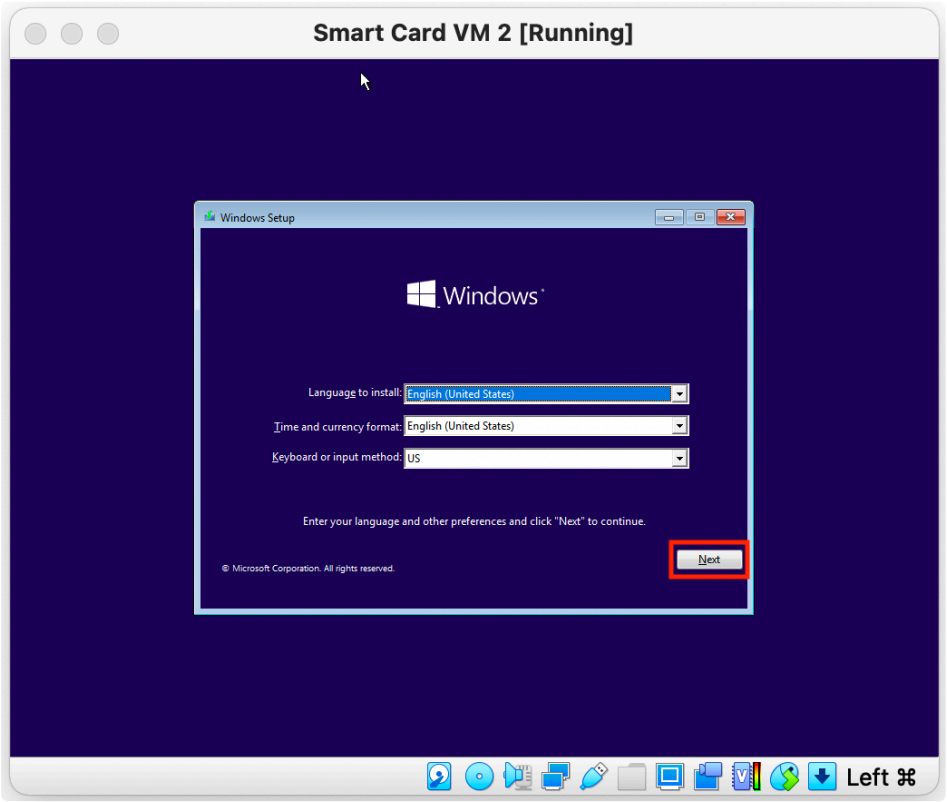
Go back to VirtualBox, select the VM and then at the top click **Start**.



A new window will open and ask you to provide a **Windows 10 Disk Image**. Click the **Folder** icon and find the **ISO** file previously downloaded and then **Start**.



You will now be presented by a **Windows 10** installation screen. Customize it as you please, I chose to use the default settings.

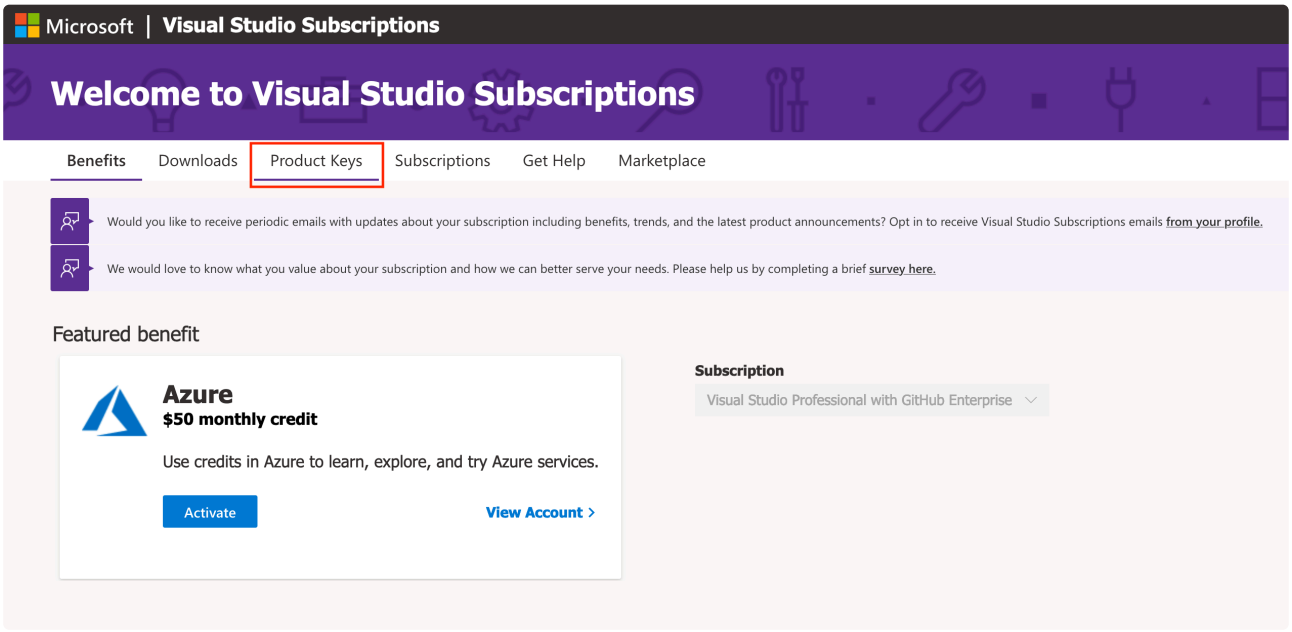


You will be asked for a Windows 10 **Product Key**.

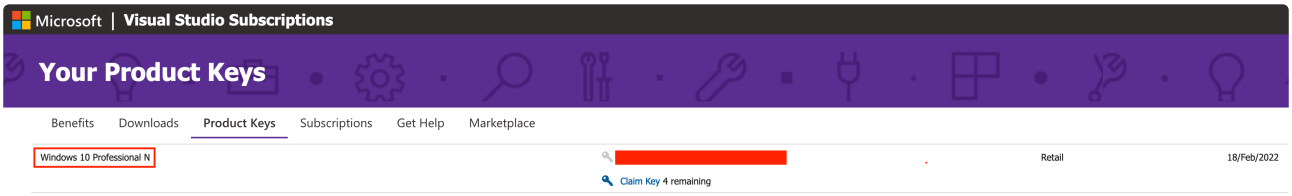
The following section covers how **InfinityWorks** employees can get access to a **Product Key**, none InfinityWorks team members will need to source one through **NHSD** or their own employer.

Go to the following link[66] when signed into your **Accenture** Microsoft account. Then click on the **Product Keys** link at the top.

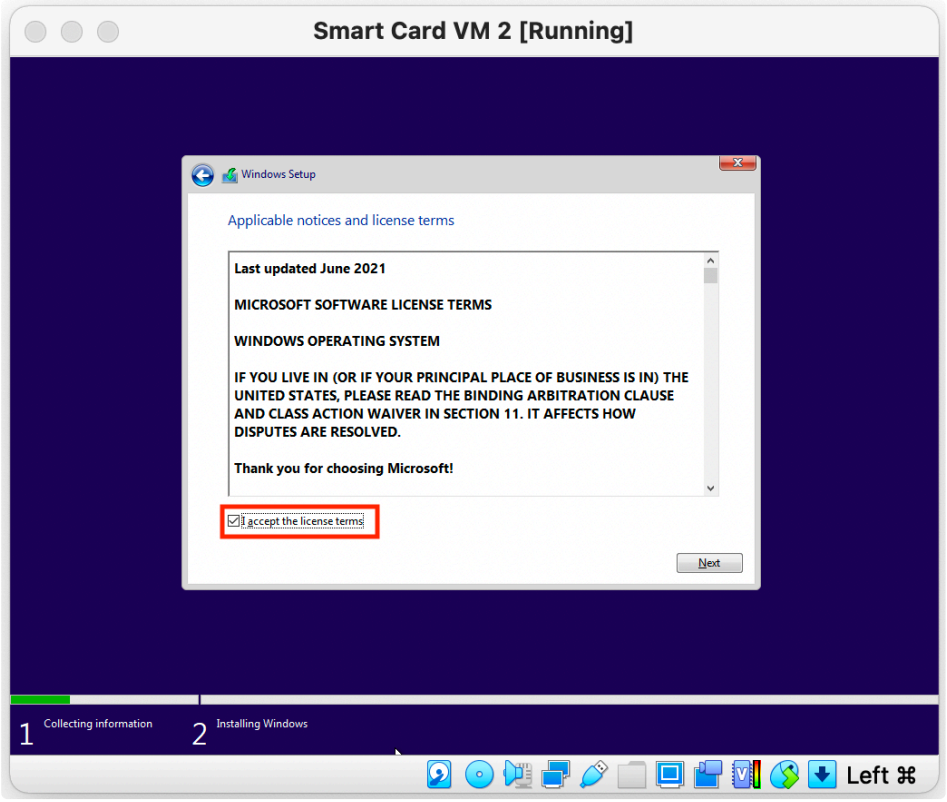
66 https://my.visualstudio.com/?auth_redirect=true



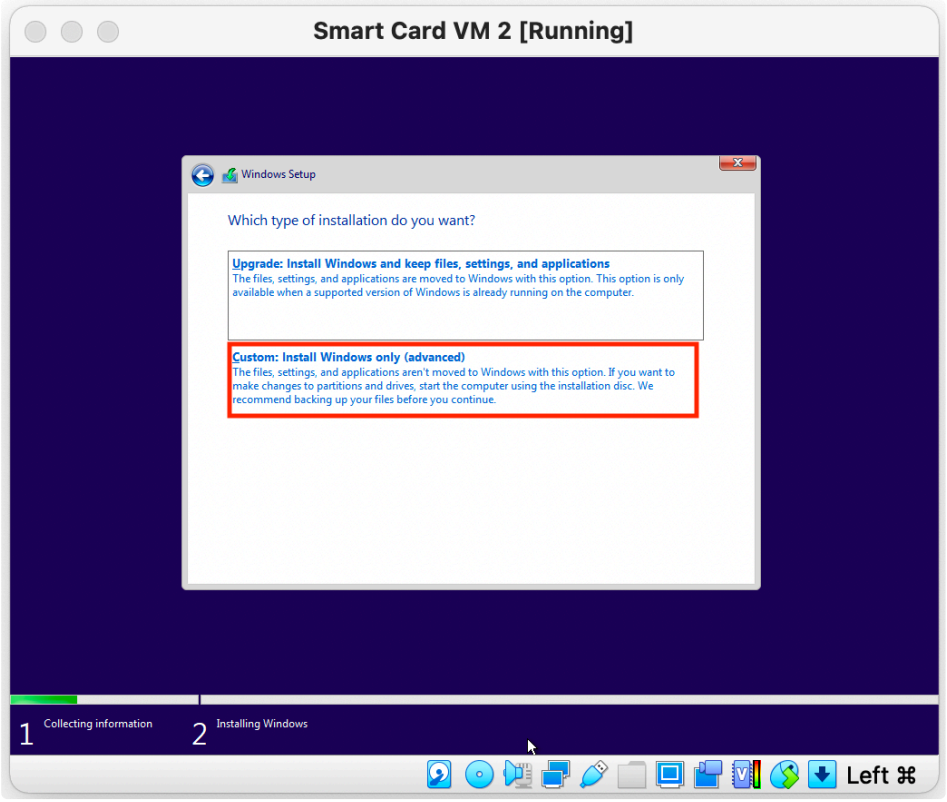
Scroll down and find the product **Windows 10 Professional N**, click **Claim Key** and then use that key to activate Windows inside the VM.

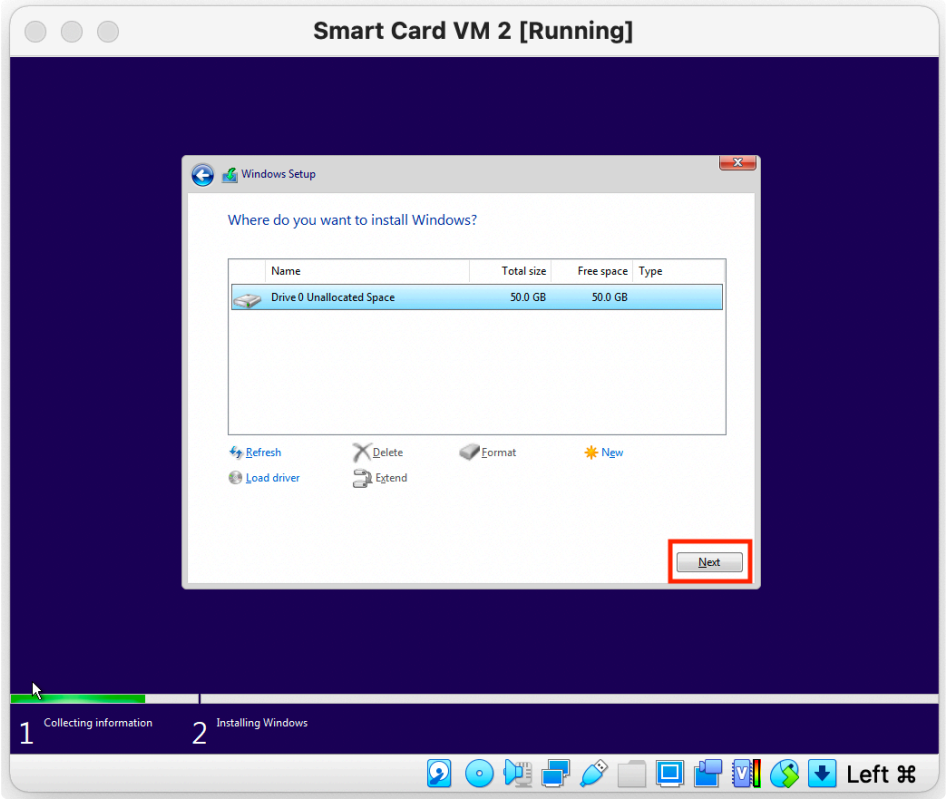


Next accept the license terms.



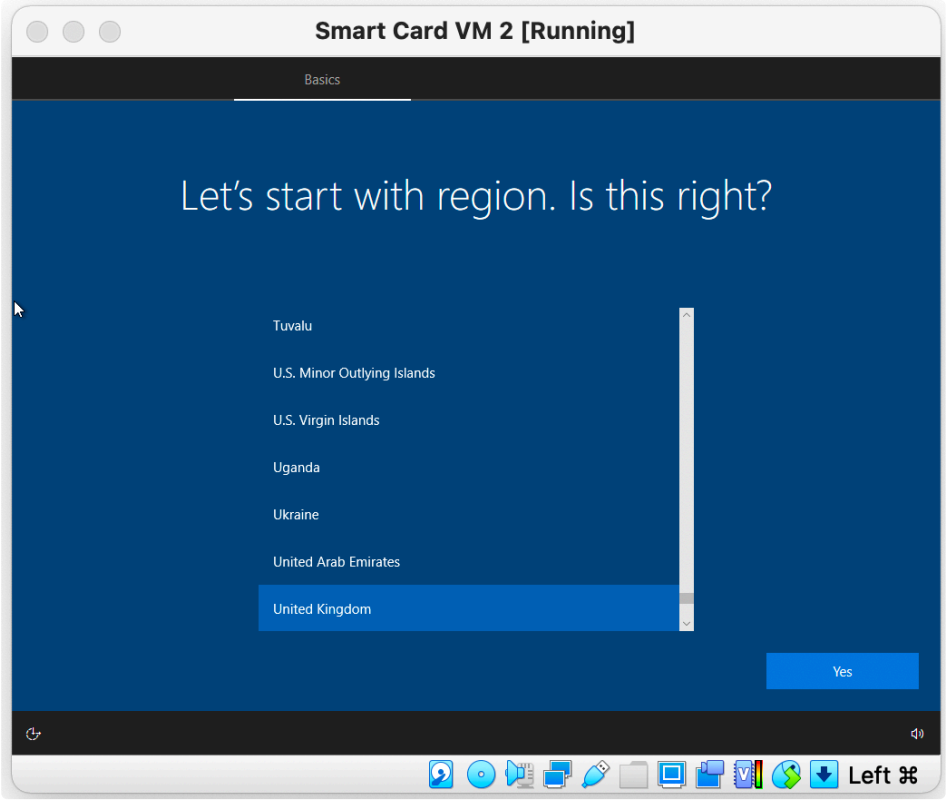
Select **Custom: Install Windows only (advanced)**, **Next** and then wait for the install to finish. (The VM may restart repeatedly)

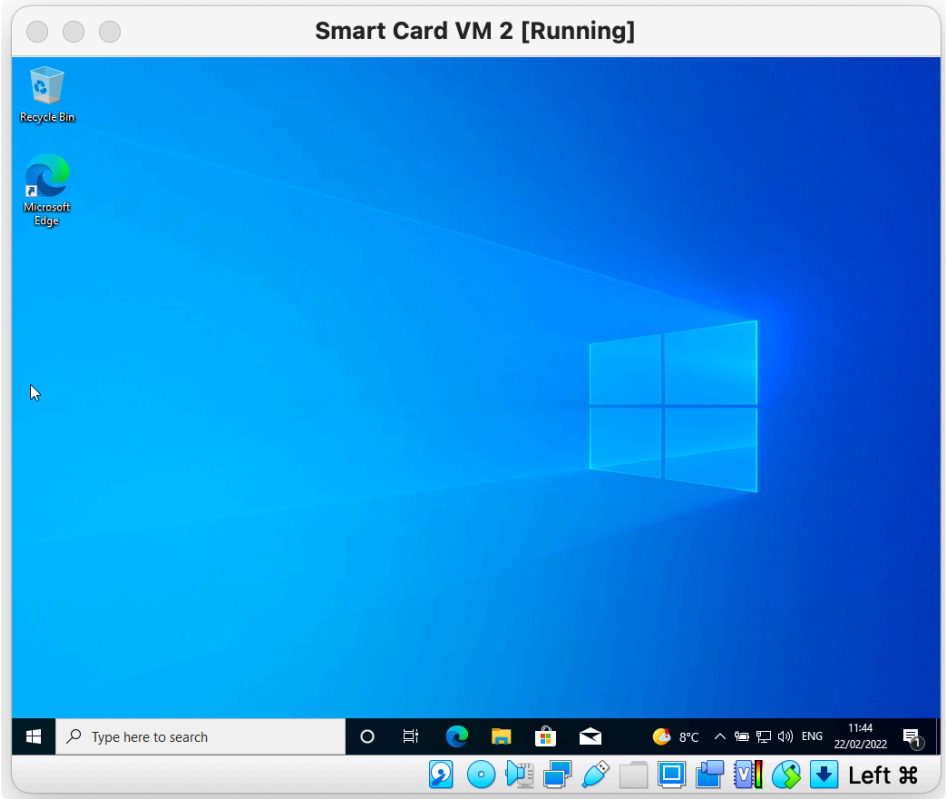

Smart Card VM 2 [Running]
Windows Setup
Which type of installation do you want?
Upgrade: Install Windows and keep files, settings, and applications
The files, settings, and applications are moved to Windows with this option. This option is only available when a supported version of Windows is already running on the computer.
Custom: Install Windows only (advanced)
The files, settings, and applications aren't moved to Windows with this option. If you want to make changes to partitions and drives, start the computer using the installation disc. We recommend backing up your files before you continue.
1 Collecting information
2 Installing Windows
Left ⌘


Smart Card VM 2 [Running]
Windows Setup
Where do you want to install Windows?
Name
Total size
Free space
Type
Drive 0 Unallocated Space
50.0 GB
50.0 GB
Refresh
Delete
Format
New
Load driver
Extend
Next
1 Collecting information
2 Installing Windows
Left ⌘

Once it has finished installing you will be presented with a region selection screen. From here you will need to complete the standard Windows account setup process.

Follow the on screen instructions until you are able to login.





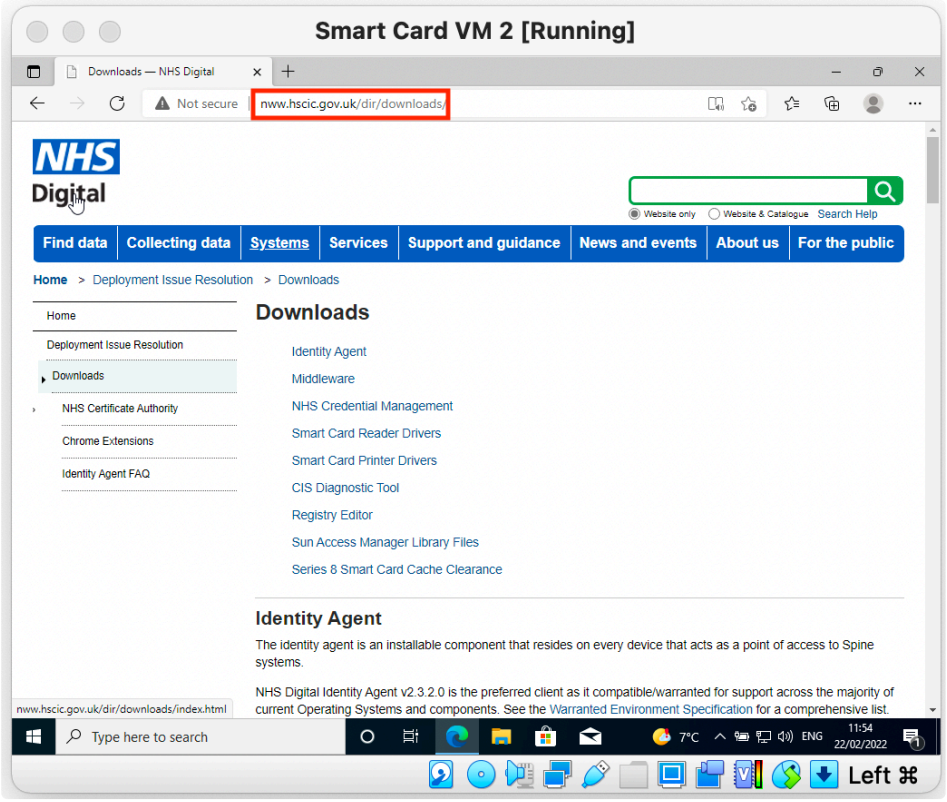
6.11.10.4 3. Setting up the smart card on the VM

6.11.10.4.1 3.1 Check you can access the HSCN VPN

In order to set up and use the smart card you need to be on the **HSCN** VPN, this should be a case of simply connecting to the VPN on your host machine (Mac) and then the VM should also use it.

To check this is working I recommend connecting on your Mac and then trying to go to http://nww.hscic.gov.uk/dir/downloads inside the VM. If this webpage loads then everything is working.

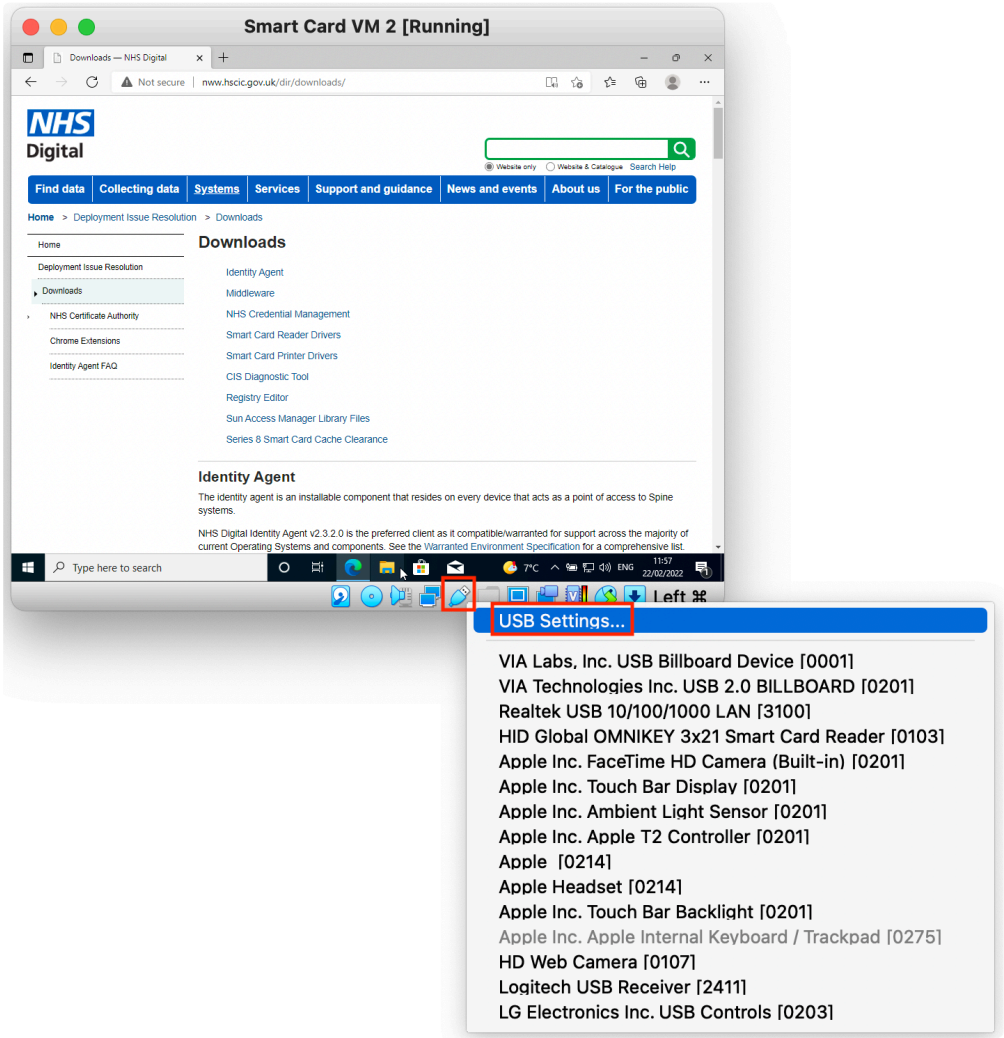
If not get in touch with Unknown User (jaro16) and we can troubleshoot the issue.



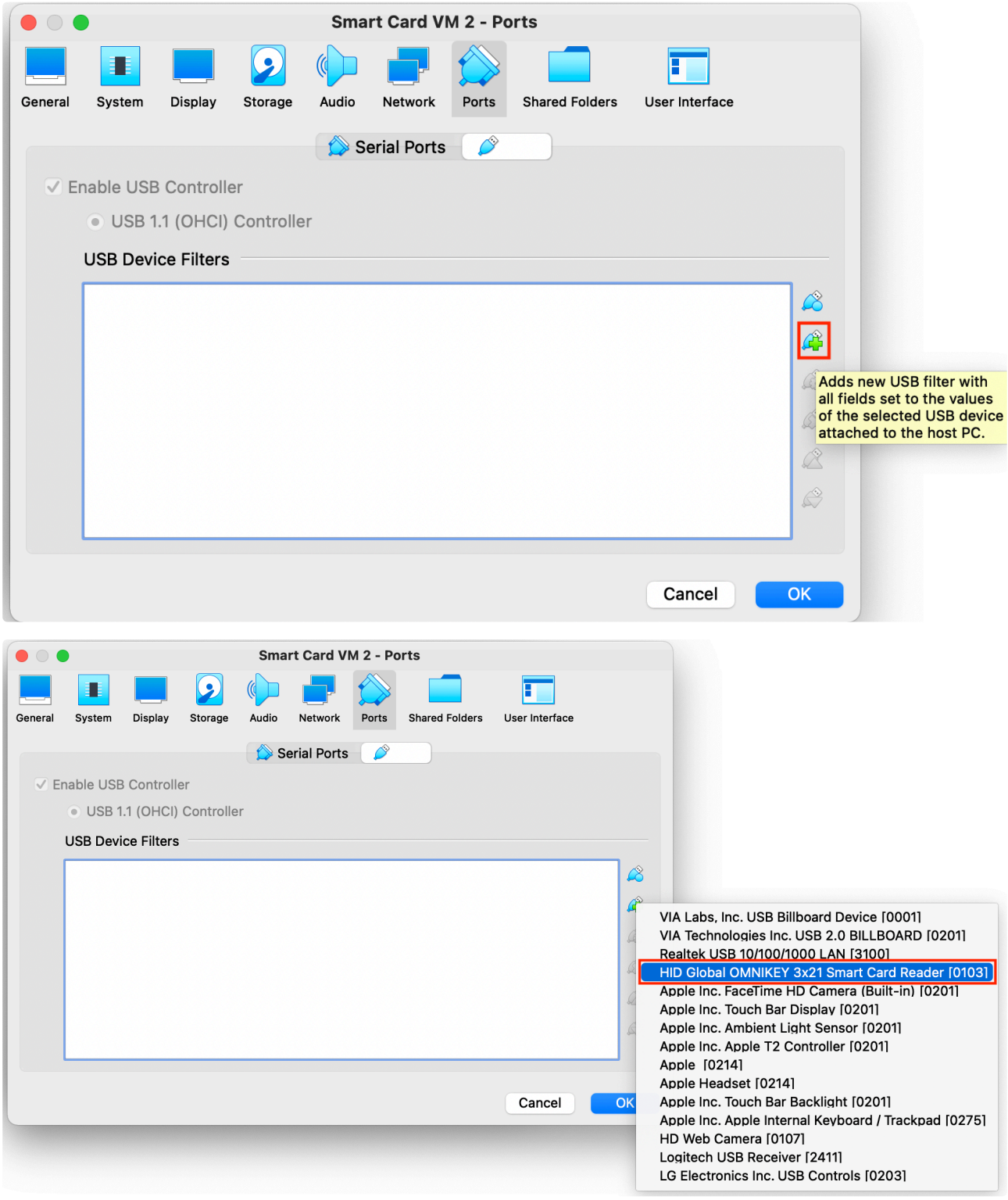
6.11.10.4.2 3.2 Set up USB pass-through

The smart card reader connects to the computer via USB, to use it inside the VM we need to pass it through.

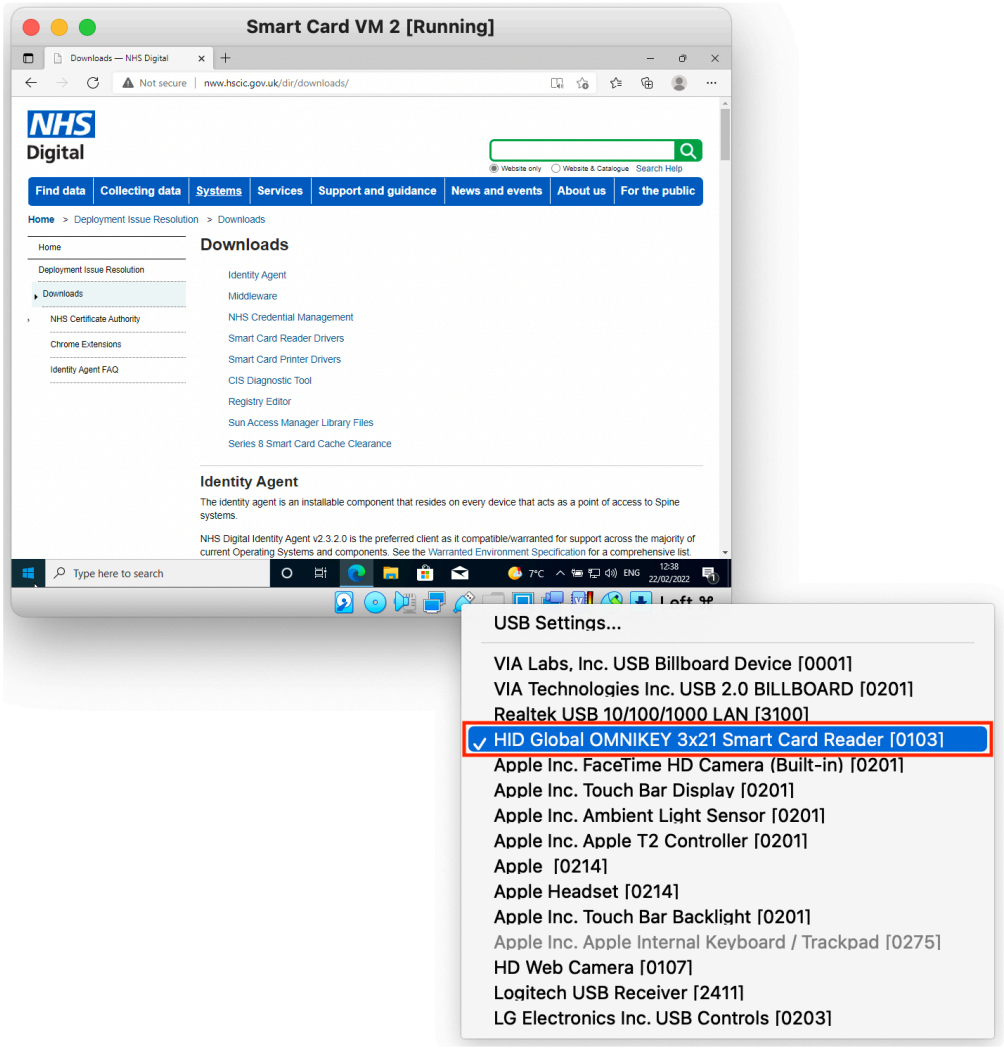
Plug the smart card reader into your Mac, click the USB icon in the bottom toolbar of the VM and then click **USB Settings...**



Click the **Add new USB filter** button, select the smart card reader from the list and click **OK**.



Then disconnect and reconnect the smart card reader. Click on the USB icon again and you should see a tick next to the smart card reader.



Now the VM can connect to the smart card reader.

6.11.10.5 4. Setting up the smart card

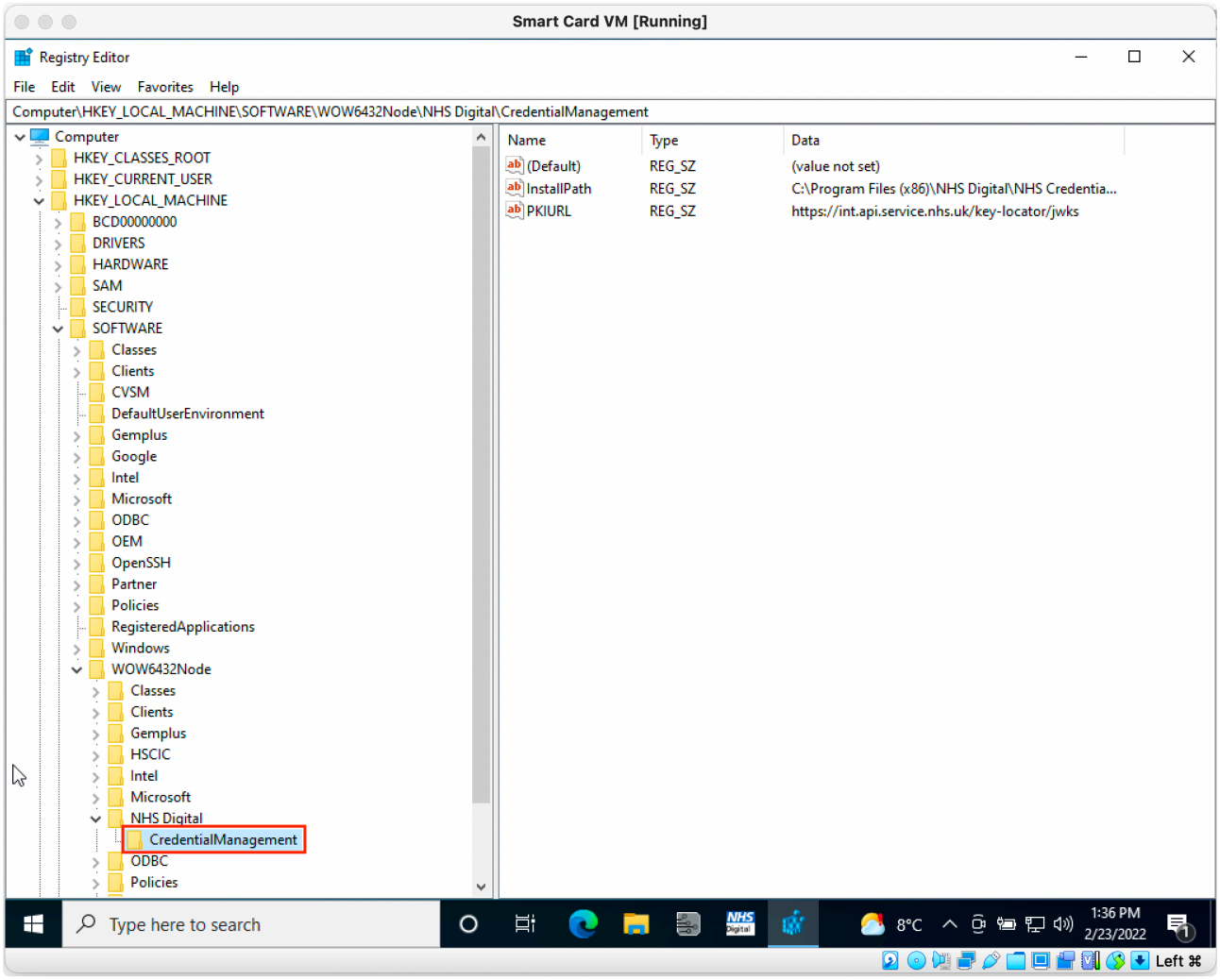
If all the above steps have worked you should now be able to set up your smart card to work inside the VM.

This should be identical to setting it up on a normal Windows computer so I will direct you towards the previous documentation here (see page 165), follow the steps from **Setting up your machine** onward inside the VM.

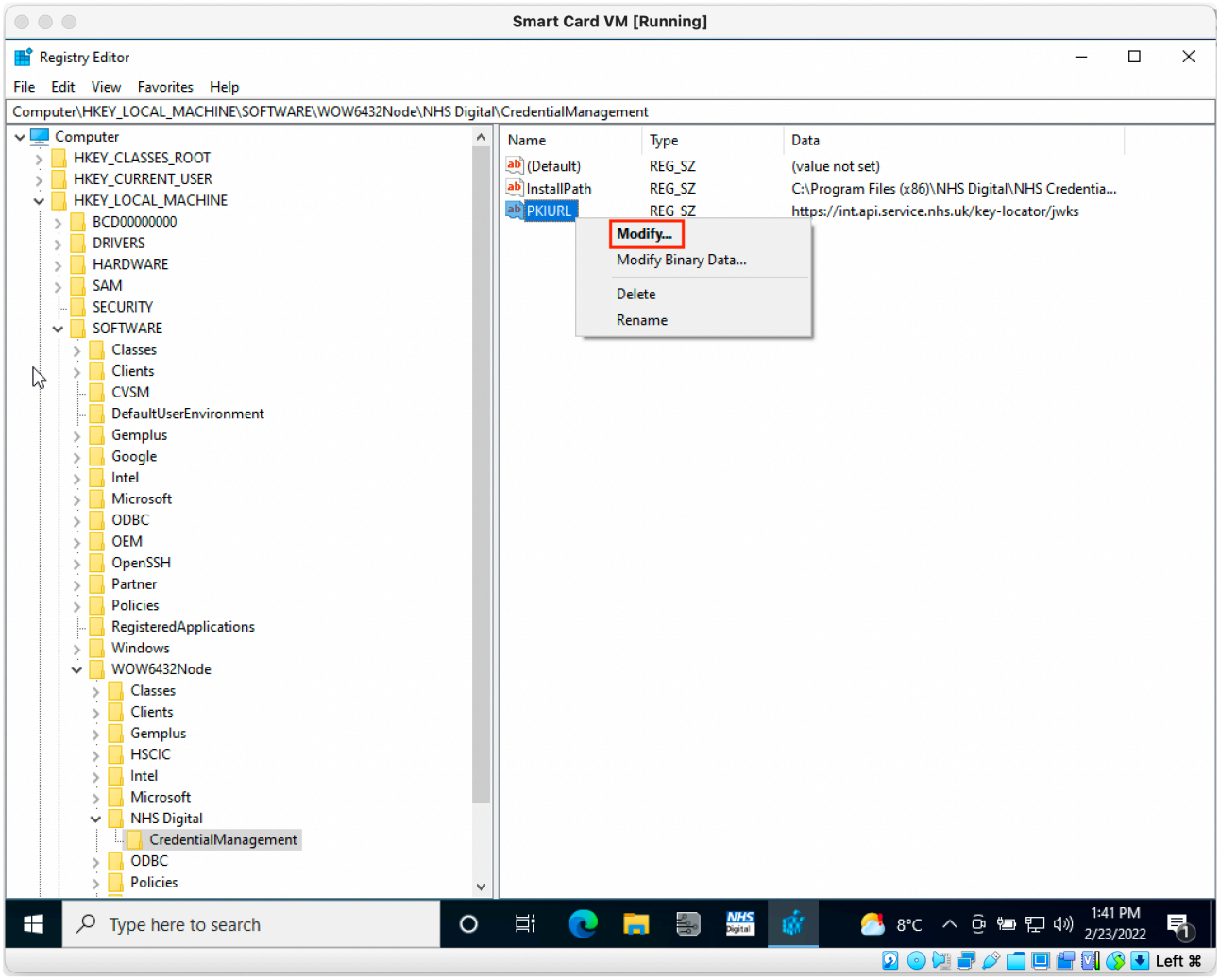
The only additional thing to call out is that once you have set up your machine you need to ensure everything is correct inside the VM's registry editor.

To do this, open the **Registry Editor** inside the VM (search for **Registry Editor** in the start search bar).

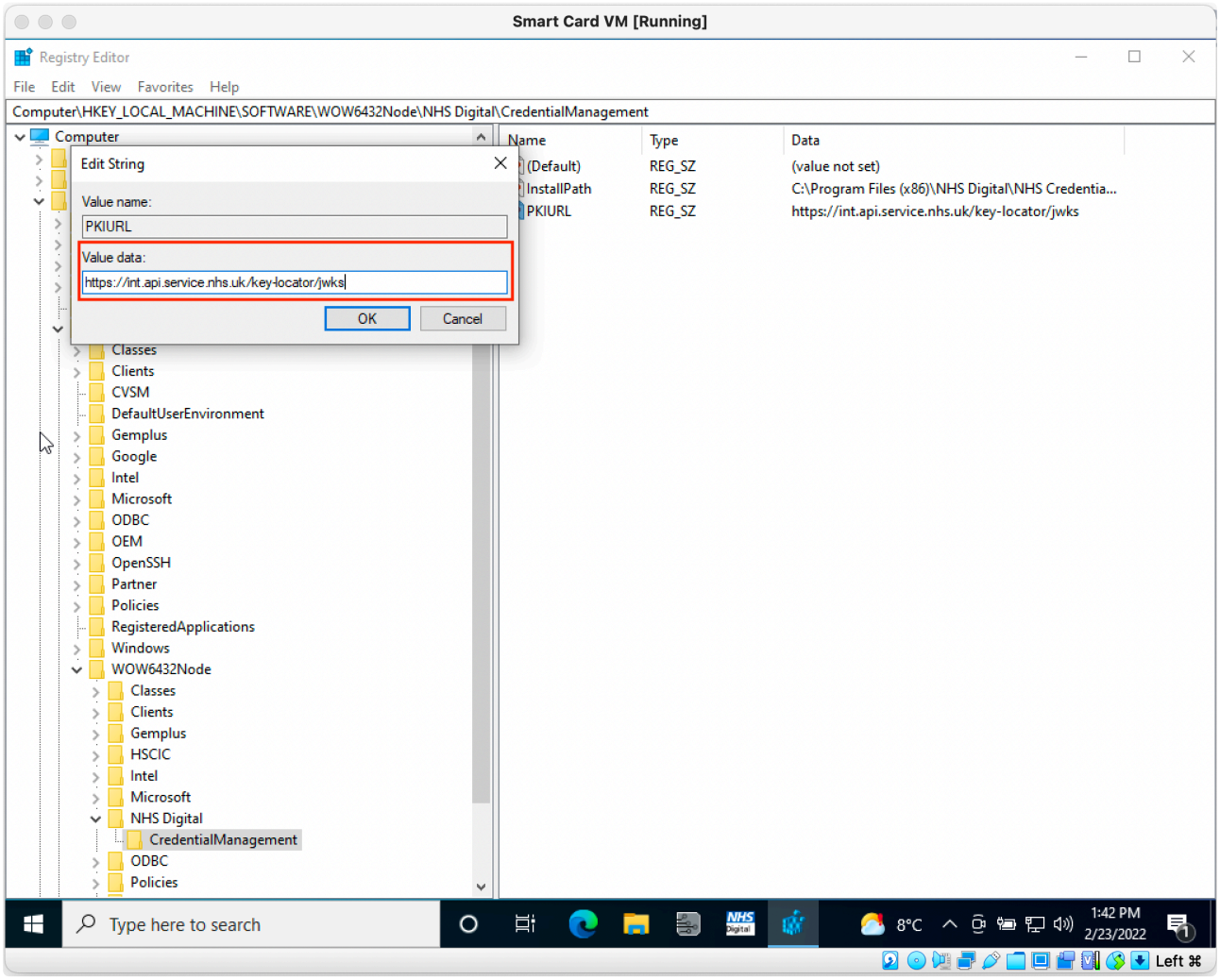
In the side navigation panel follow the path:
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\NHS Digital\CredentialManagement



Find the key **PKIURL**, right click on it and select **Modify...**



In the box that appears, set the **Value data:** field to **https://int.api.service.nhs.uk/key-locator/jwks**



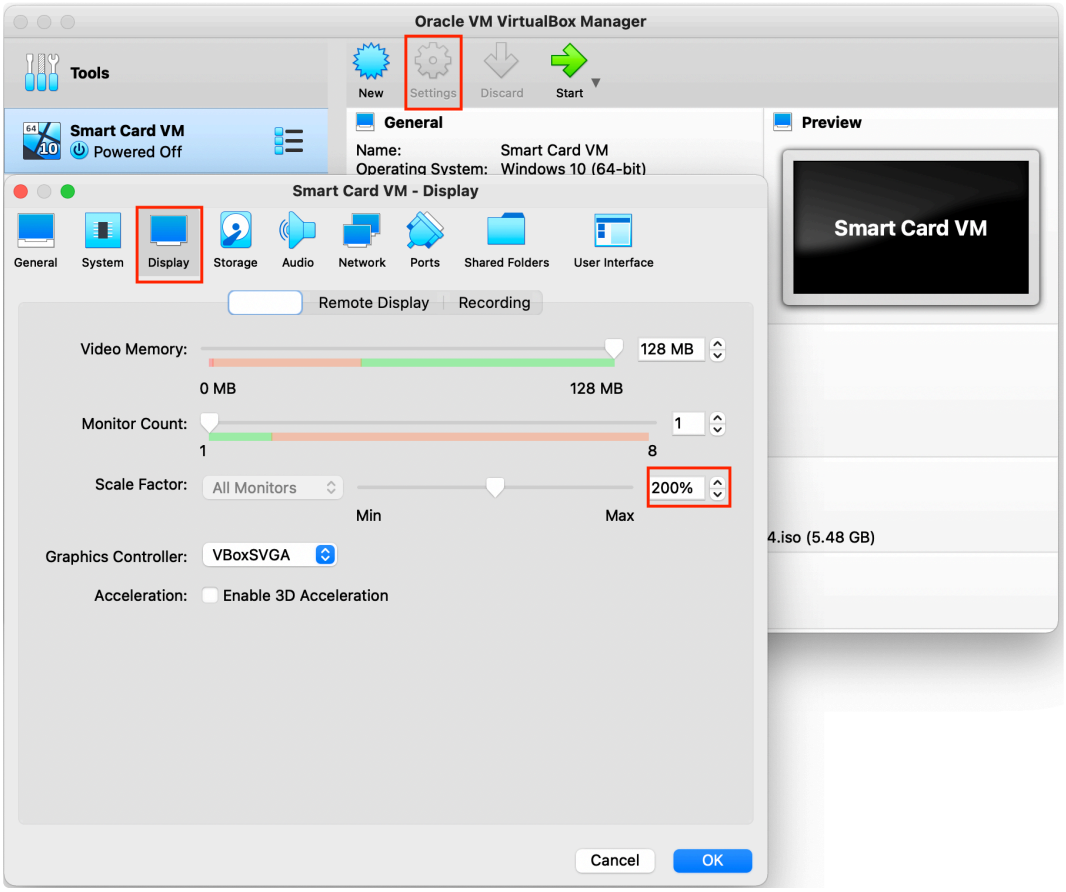
Once you have done the above you are all set to use your smart card in the VM just as you would use it on a Windows machine.

6.11.10.6 5. Tips/Misc Info

General advice about using the smart card in the VM is listed below, feel free to expand this section with any other useful information.

6.11.10.6.1 5.1 VM screen size

The screen size of the VM may be quite small, I recommend going into the **Display** tab of the VM settings and scaling up the screen. I use 200%.



6.11.10.6.2 5.2 Copying files between Windows VM and Mac host

In order to set up the smart card you may need to copy files from the host machine (Mac) to the VM.

I tried a few different methods to do this but the easiest way I found was to upload the file to **Google Drive** and then re-download it inside the VM.

VirtualBox does have a setting called **Drag'n'Drop** which should allow you to just drop files from host to VM but I couldn't get this working. If anyone manages to get this working please do let me know!

6.11.10.6.3 5.3 Smart card reader disconnects

Sometimes when using the smart card inside the VM the USB pass-through will break, when this happens unplugging and re-plugging the smart card reader should fix the problem

6.11.11 Apigee prod access - Custom producer roles

6.11.11.1 Overview

Access to the Apigee prod organisation is tightly managed by the API Platform team. A producer team can have a custom role inside of Apigee prod to enable them access to products/proxies for int, sandbox and dev (excluding prod), allowing tracing etc, which is restricted to one API.

6.11.11.2 Step 1

The first part of this process is for each desired member to follow the process for AD groups membership (see page 161) and get access to **GRP-AzureNHSD-API-Management-PROD-Producers-DL**. This must be done by the producer team, as it requires approval from line manager(s).

> Without membership of that group you will not be able to access https://nhs-digital-prod.login.apigee.com/login

6.11.11.3 Step 2

Once one of you have obtained access, raise a request through service now for Apigee Custom Role (see here[67]). In the request you will need to provide the following for each user you require adding to the role:

- Confirmation you have requested and gained access to the required AD group (above)
- API / Proxy name
- Email associated with your delve account
    - Go to your Delve profile[68], and retrieve the email associated with the account - this is the email address next to the envelope icon:
    -

67 https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_cat_item&sys_id=9f9dd4a51be0561078ac4337b04bcb6e

68 https://gbr.delve.office.com/

- This is highly **unlikely** to be your short code email, in most cases will be nhs.net[69] email but may be the first half of nhs.net[70] email appended with '@hscic.gov.uk[71]'. The email must match this to work with SAML single sign-on.

6.11.11.4 Step 3

Once the work is complete Deathstar will ask you to test you can log in and access what you expect.

Enjoy your new role!

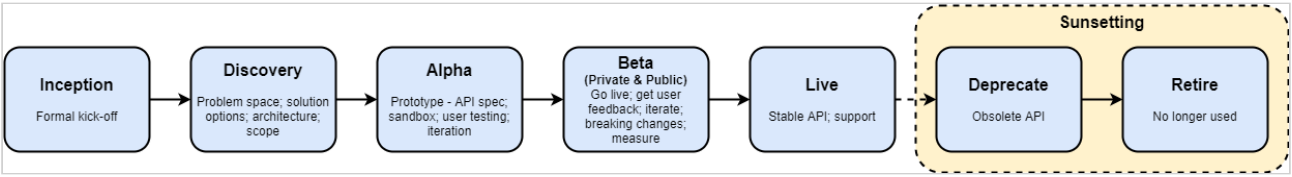
69 http://nhs.net
70 http://nhs.net
71 http://hscic.gov.uk

7 Deliver your API

7.1 Overview

This page gives a summary of our process for delivering APIs, including API specifications.

Our API delivery process is based on GDS agile delivery process[72]. At a high level, it looks like this:



7.2 Different types of API

The lifecycle can vary depending on the type of API you're delivering, such as:

- a central API service
- an intermediary API service
- a peer-to-peer API (sometimes referred to as an "API standard")
- an interoperability standard (not really an API)

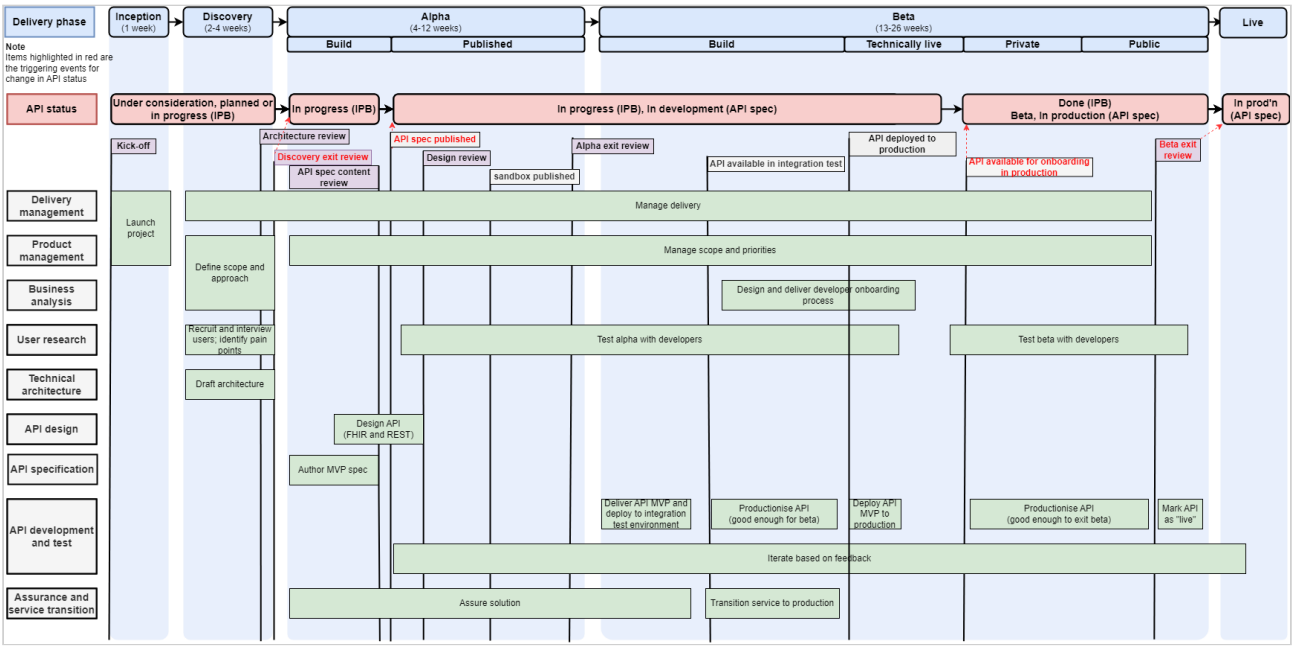
Make sure you know what type of API you are delivering - for more details, see Different types of API (see page 297).

Our delivery lifecycle is mainly focussed on delivering **central and intermediary API services**, but it can also be tailored to delivering **API standards** for peer-to-peer APIs.

7.3 A bit more detail

The following diagram gives a bit more detail:

72 https://www.gov.uk/service-manual/agile-delivery



7.4 Delivery phases

7.4.1 Inception

Inception is all about kicking off your project.

You will:

- learn about the API delivery process
- kick off your API delivery formally by
  - engaging with the API Management team
  - doing some paperwork

Start by copying the API process checklist (see page 198) into your Confluence space.

7.4.2 Discovery

Discovery is about understanding the problem that needs to be solved.

You will:

- identify your stakeholders
- map out the user journey for your API consumers
- identify solution / architecture options for your API and agree a draft architecture

7.4.3 Alpha

Alpha is about building and iterating your API design.

You will:

- design your API
- publish your API specification
- build test environments
- test the API with an API consumer
- iterate the design based on feedback

7.4.4 Beta

Beta is about moving your API into production and making it available to a wider audience. In private beta this is limited to a smaller audience, in public beta this is widened to a larger (invitation only) audience with increased volumes.

You will :

- design the onboarding process for API consumers
- get your API into production
- get a small number of external developers to use it
- iterate on their feedback
- transition your service to live

If you are delivering an API standard for a peer-to-peer API, beta is slightly different - you will not deliver anything into production yourself - rather you will work with third party early adopters to deliver their APIs into their production environment.

7.4.5 Live

The live phase is about supporting the service in a sustainable way, and continuing to iterate and make improvements

You will:

- publicise your service is live
- ramp down the team, but ensure you still have a support capability
- commit to not making breaking changes; or if we do, ensure we version the API
- continue to support your API consumers through API consumer onboarding (see page 999)

7.4.6 Sunsetting (deprecation and retirement)

Sunsetting (deprecation and retirement) is about deciding when an API is no longer needed.

To deprecate an APIs you will:

- tag the deprecated API/versions as deprecated in the API catalogue.
- discourage any further integrations with the deprecated API/version.

To retire an API you will:

- tag the retired API/version in the API catalogue.
- ensure that the API/version is no longer available for use in production.

7.5 Delivery phases versus API statuses

Delivery phases are **not** the same thing as API statuses. These exist in two places: on the API spec and also on the interactive product backlog (IPB).

Delivery phase	**API status on API specification**	**API status on IPB**
Inception	Not published	"under consideration" (sometimes "planned" if we're definitely going to do it, but haven't started yet)
Discovery	Not published	"in progress"
Alpha	Once published, "in development"	"In progress"
Private beta	"beta" + "in production"	"done"
Public beta	"beta" + "in production"	"done"
Live	"in production"	"done"

For more details on the IPB see: Getting your API on the interactive product backlog (see page 250).

7.6 GDS assessment

The Government Digital Service (GDS) requires some public services to have a service assessment[73].

GDS service assessments do not currently apply to our APIs. This is because (a) GDS service assessments don't currently apply to APIs in general and (b) we're the NHS so apparently we can choose whether or not GDS service assessments apply to any of our services, so I'm told.

This might change in the future - we are working with GDS, HMRC and others to better understand how APIs should be assessed

73 https://www.gov.uk/service-manual/service-assessments

7.7 API delivery options

There are a few options for who delivers (and supports) your API - it could be:

- your team
- our team
- a combination of the two

For more details, see API delivery and support options (see page 293).

7.8 API process checklist

To make it easy to follow our process, we provide a line-by-line API process checklist (see page 198). Specifically, in the inception phase, we ask you to make a copy of the checklist page for your API so you can fill it out as you progress.

7.9 Get started

To get started, follow the instructions on our API process checklist (see page 198). This applies even if you're planning for us to deliver your API - you need to kick it off first.

7.10 API process checklist

Instructions for using this template

Make a copy of this page, change the name to "API process checklist - <your API name>, and save it in your API's confluence space. Then delete this message (and the warning message below)
Template page version:

Version	Published	Changed By	Comment
CURRENT (see page 198) **(v. 347)**	**Mar 31, 2025 14:24**	Jonathan Eagle[74]	

Go to Page History[75]

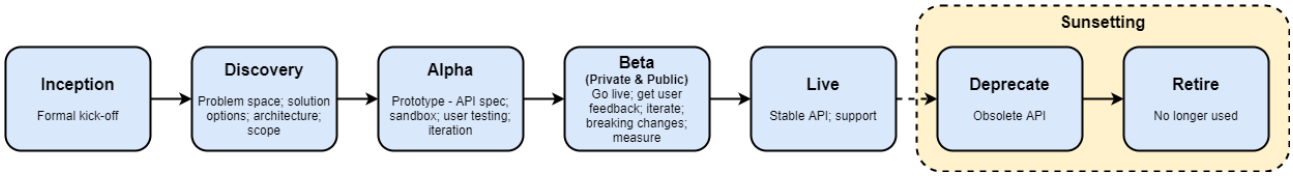
74 https://nhsd-confluence.digital.nhs.uk/display/~jonathan.eagle1@nhs.net
75 https://nhsd-confluence.digital.nhs.uk/pages/viewpreviousversions.action?pageId=97747146

Instructions for updating this centralised template

All links in this template need to be **added as web links** otherwise the links can break when a producer copies this template into their own Confluence space

7.10.1 Overview

This page provides an audit trail of how the above API has been developed in line with the NHS England API delivery process (see page 194), including links to evidence where appropriate.



7.10.1.1 Status codes key

To change the status on the steps in the process checklist below, just copy and paste the relevant option from the ones listed here :

TO DO

IN PROGRESS

DONE

N/A

Inception

7.10.1.2 Overview

Inception is all about kicking the project off - introductions and paperwork.

7.10.1.3 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Technical architecture (optional)	TBC NAME

7.10.1.4 Activities

Activity	Status	Comments / Evidence
Learn about the API platform and its benefits - and make sure it is right for you - by reading Benefits of using the API platform (see page 134).	TO DO	TBC
Learn about the API delivery process by reading Deliver your API (see page 194).	TO DO	TBC
Conduct a delivery kick-off meeting (see page 249) with the API Management team and how to Get started as an API producer (see page 125).	TO DO	TBC
Create a Confluence space and/or page for your API.	TO DO	TBC
Create a copy of the API process checklist in **your Confluence space** (see instructions at top of the page)	TO DO	TBC

Activity	Status	Comments / Evidence
Add your API to the API register (see page 1204) and include links to your Confluence space and process checklist page. Leave the JIRA column blank, as this will be completed by the APIM team. **Note: If you cannot add to the API register you will need to send all the relevant details to your API management producer liaison (currently Jonathan Eagle) who will update this Confluence page.** You won't be able to complete all columns in the API registry, as you won't have created the artefacts yet. These will need to be updated when you reach the appropriate phase in the API delivery process.	TO DO	TBC
Add your API to the interactive product backlog (see page 250) to let API consumers know what we're up to.	TO DO	TBC
Confirm with the API management producer liaison that your API has been added to our API roadmap[76]	TO DO	TBC
Decide who will do the discovery for your API. • See API delivery and support options (see page 293).	TO DO	TBC
****This service is not available from April 2025 until further notice**** If you want us to do your discovery for you, raise a New Work Request to cover our discovery costs. • See Platform costs and raising a New Work Request (NWR) (see page 305) If you are doing discovery yourself, but want us to do the build, you don't need to raise an NWR until you start Alpha.	N/A	TBC
To help guide and support you through the API delivery process, your API management liaison holds regular catch-ups with representative roles from your team your team. This starts with an an initial hour session to allow for a bit of orientation, followed by shorter focused 15 min sessions to review how you're progressing through the checklist and provide any help you might need. Contact you API management liaison (currently Jonathan Eagle) to arrange catch-up sessions	TO DO	TBC

76 https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?rapidView=2826&view=detail&selectedIssue=AMR-127

Discovery

7.10.1.5 Overview

Discovery is all about figuring out what you're going to build, and confirming that it's actually needed.

GDS guidance: how the discovery phase works[77]

7.10.1.6 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Business analysis/service design	TBC NAME
User research	TBC NAME
Technical architecture	TBC NAME
FHIR API architecture (if relevant) (this requires, at least, a review by the Interoperability Standards team)	TBC NAME

7.10.1.7 Stakeholders

(see Stakeholders for API Producer teams[78])

77 https://www.gov.uk/service-manual/agile-delivery/how-the-discovery-phase-works
78 https://nhsd-confluence.digital.nhs.uk/display/APM/Stakeholders+for+API+Producer+teams

Role	Name
Information Governance (IG)	TBC NAME
Commercial and legal	TBC NAME
Security	TBC NAME
Solution Assurance	TBC NAME
Service Management	TBC NAME
Information Asset Owner	TBC NAME
Clinical Safety	TBC NAME If not applicable include rationale and relevant links

7.10.1.8 Activities

Activity	Status	Comments / Evidence
Identify and engage with internal stakeholders (see page 253) and add them to the table above.	TO DO	TBC
Understand the context and problem / opportunity space (see page 255).	TO DO	TBC

Activity	Status	Comments / Evidence
Identify the API consumers (see page 1172) who will be consuming your API. • This might be difficult if you're planning a "build it and they will come" API - think **very** carefully before adopting this approach - you're at risk of building something that doesn't meet any user needs and that costs money to maintain.	TO DO	TBC
Map out the user journey for the API consumer. This shouldn't be hard because it's generally the same for all APIs - see the "user journey" documented in Understanding API consumers - User journey (see page 257).	TO DO	TBC
Work with API consumers to understand and map out **their** end users and their user journeys (see page 258).	TO DO	TBC
Establish a user need (for the service overall and specifically for an API).	TO DO	TBC
Understand "as-is" (technology or otherwise) - including existing back-end services that could be used / re-used / need enhancing.	TO DO	TBC
Define your scope (see page 259) and agree this with your sponsoring stakeholders.	TO DO	TBC
Arrange an architecture kick-off meeting (see page 262) with your architect so that they can guide and advise you before you go too far with your architecture.	TO DO	TBC
Identify solution / architecture options for your API. Read top-level guidance Architecting your API (see page 260), which includes key aspects such as Is my API a FHIR API? (see page 285). Although not essential, it will help you if you use the API Architecture options template (see page 262). If the pages don't answer all your architecture queries, need some further architecture guidance or need some architectural resouce, see Help and support - Architecture support (see page 2056).	TO DO	TBC
Agree a draft architecture, based on your scope - see API architecture review checklist (see page 266).	TO DO	TBC

Activity	Status	Comments / Evidence
Conduct a API architecture review meeting (see page 289) with your architect to make sure you are heading in the right direction.	TO DO	TBC
Follow up with wider architecture governance sanity check by creating an SDO (Solution Design Overview) and possibly attending TRG - more details in the Wider architecture review (see page 290) guidance.	TO DO	TBC
Do a cost / benefit analysis (see page 304) to prove that it's worth building your API / service. Cancelling your project at the end of discovery because the benefits don't justify the costs is an entirely valid decision. If need be, and if you feel brave enough, use this activity to challenge a questionable political decision. At the very least, get your project sponsor to sign off the cost/benefit case (or lack thereof).	TO DO	TBC
Define metrics for service success. See Metrics for success / KPIs for APIs (see page 292).	TO DO	TBC
Decide who will deliver your API and who will support it long term. See API delivery and support options (see page 293).	TO DO	TBC
****This service is not available from April 2025 until further notice**** If we are delivering all, or part, of your API, raise a New Work Request (NWR) to cover our delivery costs. • See Platform costs and raising a New Work Request (NWR) (see page 305).	N/A	TBC
Conduct a discovery exit review (see page 307) with the API Management team. • This includes deciding whether to proceed to alpha.	TO DO	TBC

Alpha

7.10.1.9 Overview

Alpha, for an API, is all about getting the API published - as a spec and in test environments - and getting early feedback.

GDS guidance: How the alpha phase works[79]

7.10.1.10 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Business analysis/ service design	TBC NAME
User research	TBC NAME
API design	TBC NAME
FHIR API design (if relevant) (this requires, at least, a review by the Interoperability Standards team)	TBC NAME
API development (for spec and sandbox)	TBC NAME
API test (for spec and sandbox)	TBC NAME

79 https://www.gov.uk/service-manual/agile-delivery/how-the-alpha-phase-works

Skill	Name
API documentation - API producer team member(s) (see Who should write your API documentation (see page 452))	TBC NAME
API documentation - API platform team point of contact (see Who should write your API documentation (see page 452))	TBC NAME
Technical architecture	TBC NAME

7.10.1.11 Activities

Activity	Status	Comments / Evidence
****This service is not available from April 2025 until further notice**** Raise a New Work Request (NWR). If you haven't done so already, you must raise an NWR if the API is being delivered by an NHS England Platforms team or by us. • See Platform costs and raising a New Work Request (NWR) (see page 305).	N/A	TBC
On the interactive product backlog (see page 250), change the status to "in progress".	TO DO	TBC

Activity	Status	Comments / Evidence
Revisit the list of internal stakeholders[80] you created during discovery and ensure you are still engaged with them. You are responsible for the assurance of your API, including security, IG and clinical safety. To ensure a smooth service transition into production during private beta, make sure you bring these stakeholders on the journey with you.	TO DO	TBC
Engage with legal to determine you need a connection agreement[81] for use of your API. To do this create a legal requisition form - follow the process here: Legal Team (sharepoint.com)[82]	TO DO	TBC
Get setup and get access: Building your API alpha (see page 578) **GitHub repo is being deprecated - these instructions will be removed** *If you haven't already, create a GitHub repo for your API and get the deployment pipeline up and running.* • *See* Creating a GitHub repository and deployment pipelines for your API (see page 580). • *See* Creating a "spec-only" API (see page 585)*for information about how to set up documentation only APIs.* *See* Building your API alpha (see page 578) *for more information about the engineering steps of this section.*	TO DO	TBC
Design your API. • See Designing your API (see page 322).	TO DO	TBC
Draft your API specification - mark it as "In development" (i.e. a prototype). • See Documenting your API (see page 445).	TO DO	TBC

80 https://nhsd-confluence.digital.nhs.uk/display/APM/Stakeholders+for+API+Producer+teams
81 https://digital.nhs.uk/developer/guides-and-documentation/online-connection-agreement#20-third-parties
82 https://nhs.sharepoint.com/sites/thehub/SitePages/Legal(1).aspx?xsdata=MDV8MDJ8YXVieW4uY3Jhd2ZvcmQxQG5ocy5uZXR8NTRkZDMwZjdjNTI5NDk0NmJhYTIwOGRjY2I0ZDMzZTI8MzdjMzU0YjI4NWIwNDdmNWIyMjIwN2I0OGQ3NzRlZTN8MHwwfDYzODYwODc3OTY0NDk2ODIzNnxVbmtub3dufFRXRnBiR1pzYjNkOGV5SldJam9pTUM0d0xqQXdNREFpTENKUUlqb2lWMmx1TXpJaUxDSkJUaUk2SWsxaGFXd2lMQ0pYVkNJNk1uMD18MHx8fA%3d%3d&sdata=MnlDWTdqcWRFdTBpL1Q1N0hJOXRCajVDUjJ3czJLWEVhNDZXbGtCNW5SWT0%3d

Activity	Status	Comments / Evidence
Publish your API specification (API status should be tagged as "In development" in the API specification. To make this visible on the API catalogue contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler). **Your API does not need to be complete before it is published**, as the Alpha phase is all about getting feedback and iterating your specification. • See Publishing your API documentation in Bloom reach CMS (see page 507).	TO DO	TBC
Conduct an API design review meeting (see page 375) (to primarily look at your API's physical design - endpoint names, granularity, FHIR conformance etc.) with your team.	TO DO	TBC
Conduct an API specification content review meeting (see page 557) (to look at the word-smithery of your API specification).	TO DO	TBC
Consider building a "sandbox" service for your API - a callable API that simulates the API's behaviour, either using stubs or in a path-to-live test environment. • See Setting up your API sandbox (see page 592) for more details.	TO DO	TBC
Consider building a Postman collection for your API - this makes it easier for API consumers to understand how to use your API, but you do need to make sure you keep it in sync with any changes to your API. • See the PDS FHIR API specification[83] for an example.	TO DO	TBC
Consider building an open source reference implementation for your API - in whichever language(s) your API consumers use. Python, Java and C# are the usual contenders. This is a lot of effort but you could double it up as a test harness for your own use.	TO DO	TBC

83 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-description__environments-and-testing

Activity	Status	Comments / Evidence
Provide an integration environment for API consumers, for formal integration testing. • You might not need to do this if your API is simple enough - API consumers could test against production. See Integration testing environment for more details.	TO DO	TBC
Make sure you have thought about test data. • For sandbox, check you have aligned your canned responses with other APIs, as per Standard test data - Sandbox (including specs) (see page 606) • For integration test, check you have considered providing standard test data for read-only testing, as per Path-to-live environments (externally facing) - Test data[84] • ...and that you have aligned it with other APIs, as per Standard test data - Integration test (see page 607)	TO DO	TBC
If your API calls a Spine based API, there are a few other considerations: • We don't ask API consumers to provide ASIDs, but map to them • Except for Production there are some in-built generic ASIDs (see page 418) that might need additional permissions to work in the context of your API call See Spine based APIs (see page 408)	TO DO	TBC
Test the API with real users (ideally with API consumers who actually want to use the API). • Ideally face-to-face - hackathons are also good • Micro-polls on your API doc page are also good • Be ready receive feedback If struggling to find a consumer to test with in INT or driven by project deadlines, as a minimum, some testing needs to have been performed in the INT environment (even ifs only by the API producer team).	TO DO	TBC
Where possible, engage with testers from API consumer organisations to ensure you are meeting their testing needs. • Iterate the testing service based on feedback.	TO DO	TBC

84 https://nhsd-confluence.digital.nhs.uk/display/APM/Path-to-live+environments#Pathtoliveenvironments-Testdata

Activity	Status	Comments / Evidence
Iterate the API design and test environments, based on the user feedback from real users, but bear in mind you'll get way more feedback in beta.	TO DO	TBC
Agree method for managing Dependabot alerts (see page 788) on your API proxy GitHub repo	TO DO	TBC
Conduct an alpha exit review (see page 933) with the API Management team. • This includes deciding whether to proceed to beta, and whether to start with public or private beta. Go public unless you have very strong reasons to stay private, because the more open your API is, the more feedback you'll get during beta.	TO DO	TBC

Beta (private)

7.10.1.12 Overview

In private beta you:

- get your API into production
- get a small number of API consumers to use it
- iterate on their feedback

If you are delivering an API standard for a peer-to-peer API, you work with third party early adopters to deliver their APIs, rather than delivering one yourself.

GDS guidance: How the beta phase works[85]

7.10.1.13 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Business analysis/service design	TBC NAME

85 https://www.gov.uk/service-manual/agile-delivery/how-the-beta-phase-works

Skill	Name
User research	TBC NAME
API design (iterate)	TBC NAME
API documentation (iterate)	TBC NAME
API development	TBC NAME
API testing	TBC NAME
Technical architecture (iterate)	TBC NAME
Solution assurance	TBC NAME
Service transition	TBC NAME

Note : These are needed whether you decide to transition in technical go-live only or directly into private beta

Activity	Status	Comments / Evidence
Build your API service - first iteration / MVP. • You might have already done some of this if you provided a formal integration testing environment.	TO DO	TBC

Activity	Status	Comments / Evidence
Prepare your service for production (see page 945) : • Go through the appropriate Service transition process (see page 947). **THIS CAN TAKE A LONG TIME**	TO DO	TBC
• Prepare the consumer onboarding (see page 999) process for your API, and if necessary get it approved at OGG. **THIS CAN TAKE A LONG TIME!**	TO DO	TBC
• Conduct a penetration test (see page 968). If your penetration test is low risk you may be able to do this internally (if agreed with Cyber Security).. **IF YOU NEED AN EXTERNAL PENETRATION TEST THERE IS A LEAD TIME OF 6 WEEKS**	TO DO	TBC
Consider reviewing the Api platform hazard log (see page 966) with your clinical lead	TO DO	TBC
For APIs containing Personally Identifiable Data (PID or PII), review if the Information Asset Owner requires **APIM** on-call rota staff to raise Asset Access Request (AAR) form in order to access raw logs information case during an incident. • For details, see Supporting your API (see page 964)	TO DO	TBC
Engage with the CSOC to ensure proactive security monitoring, email england.csoc.services@nhs.net[86]	TO DO	TBC
Complete the following first time into production activities (required for the first time deploying into production review (see page 1120) meeting). This is an API platform-specific checklist which is additional to the above service transition process.	TO DO	TBC

86 mailto:england.csoc.services@nhs.net

7.10.1.13.1 First time deploying into production technical activities

Category	Activity	Status	Comments / Evidence
Readme	Ensure you have an up-to-date README file in your GitHub repo.	TO DO	TBC
Rate limiting	Ensure you have set up rate limiting for your API. This might be a low rate limit for your private beta (if you haven't yet scaled and load tested your API). Or it might be a higher rate limit if you have already scaled and load tested your API. • For details, see Rate limiting (see page 1037).	TO DO	TBC
Monitoring	Ensure you have set up **and enabled** status monitoring (aka "ping and status" endpoints) for your API, to monitor the health of your API. • For details, see Adding status monitoring to a service (see page 1065) • Ensure that _ping and _status are not rate limited as non-200 responses will trigger alerts that your proxy is down.	TO DO	TBC
	Ensure you have set up and enabled status monitoring for your sandbox (if you have one). • For details see Adding status monitoring to a service - Adding status monitoring to your Sandbox environment	TO DO	TBC

Category	Activity	Status	Comments / Evidence
Logging	Ensure you have set up logging to Splunk for your API. Check that it works in your path-to-live environment. Note : You will need inform APIM of any data that needs adding to the Splunk summary/reporting indexes • For details, see Logging to Splunk (see page 1082).	TO DO	TBC
	In particular, ensure you have set up sensitive data redaction so that people who are not security cleared can view your logs via the open-access dashboards. Check that it works in your path-to-live environments. • For details, see Logging to Splunk - Set up sensitive data redaction (see page 1088)	TO DO	TBC
Spine	If your API calls a Spine based API, you should have worked through your Spine specific configuration -see Spine based APIs (see page 408) for first time into Production work	TO DO	TBC
Smoke testing	Prepare your automated tests for when you go live (if possible). • For details, see Smoke testing (see page 1145)	TO DO	TBC
Auto-approval	If appropriate, set up automated production approval for consumers of your API (the default is manual production approval). • For details, see Accessing your deployed API - manual vs auto approval for consumer applications subscribing to a product in Apigee (see page 927)	TO DO	TBC

Category	Activity	Status	Comments / Evidence
Pipelines	Ensure you have you implemented the production deploy step in the release pipeline. • For details, see Deploying to Production - Add a production step to your release pipeline (see page 1131)	TO DO	TBC
	Ensure your manifest_template.yml (or manifest.yml) file has an entry for production. • Ensure that the product approvalType set to `manual` • Confirm you are happy with the values for `ratelimit` and `quota` • Ensure the scopes on the product are correct. • For more details, see Deploying to Production - Add information about your production environment into your manifest file (see page 1131)	TO DO	TBC
	Configure self-serve production releases for your API proxy. • See Deploying to production-Self-serve releases (see page 1121) for information on self-serve deployments. • See Using manual approval on pipelines -Self-serve production deployments (see page 916) for how to set up self-serve production releases	TO DO	TBC

Category	Activity	Status	Comments / Evidence
Key value maps	If your API has its own KeyValueMap (due to Apigee limits on KVMs this design pattern is deprecated, we prefer ClearCaribou/SecretSquirrel), ensure it is present in apim-infrastructure ansible config, and has it been deployed to the Apigee production environment, • For details, see Per-environment configuration - Using your own Key Value Map	TO DO	TBC
	Not Yet Implemented If your API uses the ClearCaribou/ SecretSquirrel KeyValueMaps to store data in Apigee, ensure that any data required by your proxy is present in apim-infrastructure ansible config and deployed to Apigee prod environment. For details, see Per-environment configuration - Using the Clear and Secret common config stores (see page 600)	N/A	Not currently required. Planned for future implementation.
Plumbing	If using APIM hosted containers:		
	Does your container's IAM role need any non-standard permissions? (e.g. access to certain secrets or other AWS services). If so, has such an empowered role been created in apim-infrastructure terraform for AWS prod? • For details, see Hosted Containers - Adding extra IAM permissions[87]	TO DO	TBC

87 https://nhsd-confluence.digital.nhs.uk/display/APM/Hosted+Containers#HostedContainers-AddingextraIAMpermissions

Category	Activity	Status	Comments / Evidence
	Have all secrets/parameters required by your container been uploaded to the prod secretsmanager/parameter store? • For details, see Storing Secrets Securely (see page 667)	TO DO	TBC
	If calling out from your container to other services, has egress access been configured? This requires security groups to be created allowing outbound traffic in apim-infrastructure terraform. • For details, see Hosted Containers - Configuring egress to other services[88]	TO DO	TBC
	Have you load tested your API and checked the default scaling policy is sufficient to handle expected load? If the default is not sufficient, have you created the custom scaling policy in apim-infrastructure terraform? • For details, see Hosted Containers - Scaling hosted containers[89]	TO DO	TBC
	If **NOT** using APIM hosted containers:		

88 https://nhsd-confluence.digital.nhs.uk/display/APM/Hosted+Containers#HostedContainers-Configuringegresstootherservices
89 https://nhsd-confluence.digital.nhs.uk/display/APM/Hosted+Containers#HostedContainers-Scalinghostedcontainers

Category	Activity	Status	Comments / Evidence
	Is the TargetServer present in the prod environment? Has it been smoke-tested? • For details, see Setting up a target server - Adding an existing target server to more environments (see page 609), and Setting up a target server- Testing connectivity (see page 611)	TO DO	TBC
	Are certificates present in AWS prod secretsmanager for TLS MA? • For details, see Setting up a target server - Setting up TLSMA (aka mTLS, TLS Mutual Authentication) on your target server (see page 609)	TO DO	TBC
	Is there a corresponding entry in referenced in apim-infrastructure ansible config so that the TargetServer uses the correct certificates to authenticate with your backend? • For details, see Setting up a target server - Create a keystore and reference your key and cert (see page 610) and Setting up a target server - Associate keystore and cert with target server (see page 610)	TO DO	TBC
	Does your backend require any addition authorization from Apigee? (e.g. headers). Are these implemented for prod? • For details, see Passing information to your API backend (see page 611)	TO DO	TBC

Category	Activity	Status	Comments / Evidence
Caching	If appropriate, implement caching in your API. • For details, see Caching for your API (see page 1112)	TO DO	TBC
OAS Spec Checkpoint	Has OAS spec been published?	TO DO	TBC

Activity	Status	Comments / Evidence
Conduct a first time into production review (see page 1120) meeting. This is basically a check through of the above first time into production activities.	TO DO	TBC
Do you need approval for technical go-live ? See Service transition for APIs - Enter technical go-live (see page 948). Alternatively, if you have already completed your onboarding process, you might decide to get approval to go directly to private beta - see Service transition for APIs - Enter private beta (see page 949).	TO DO	TBC
Deploy your service to production. • For details, see Deploying to production (see page 1120)	TO DO	TBC

Activity	Status	Comments / Evidence
If you have automated smoke tests, check that they ran successfully. If you don't have any automated smoke tests, run a manual smoke test. • For details of how to test your API as a "consumer", see Smoke testing - Apigee Test Application (see page 1147) • As this is a production app, enablement is manual. The nominated onboarding lead (see page 1000) will also need to request "enablement of the Application" and any "custom attributes" via the platforms-apim-producer-support workflows. Also where applicable, request ASIDs (see page 414)	TO DO	TBC
Check your open access dashboard in Splunk is working for your API. You should be able to see your smoke test(s). This is applicable to all APIs. • For details, see Logging to Splunk - Set up production indexes	TO DO	TBC

At this point, you are technically live, but not yet in private beta.

7.10.1.14 Enter private beta activities

Activity	Status	Comments / Evidence
Make sure you are geared up to support the service: • Set up first line support for API consumers. • Make sure the business is ready. • Consider getting your technical author involved closely in first or second line support, with a view to them updating documentation on the back of user queries. If API is transitioning to another team see API handover guidance (see page 963) for some areas to consider when handing over.	TO DO	TBC
Identify a first-of-type API consumer. This can be a bit chicken and egg as you haven't yet advertised your API as being available in beta, but hopefully you've had some feedback via the interactive product backlog or you're already engaged with a keen API consumer from alpha.	TO DO	TBC
Onboarding you API consumer. Ideally you will need an onboarding lead See API consumer onboarding - Onboarding lead (see page 1000)	TO DO	TBC
Enter private beta. • You ideally need a first-of-type API consumer lined up for this. • You'll also need to agree the restrictions of your private beta with Service Management e.g. maximum number of participants, maximum TPS throughput. • See Service transition for APIs- Enter private beta.	TO DO	TBC
Out of hours support active and genuine Production data: • If your service requires any out of hours support (i.e. Gold or Platinum), the APIM on-call rota needs warning. Work with API producer liaison to inform	TO DO	TBC

Activity	Status	Comments / Evidence
the core APIM team (this includes updating the ITOC runbook[90])		
Update your API spec: • Change the API status tag to "in production + beta", to say that the API is now available in production (you can still reserve the right to turn people away). To change the tag and make this visible on the API catalogue, contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler • Add a section into the spec to state what the API's service level is, as per What goes in your API documentation (see page 453).	TO DO	TBC
On the interactive product backlog (see page 250), change the API status to "done" (we use "done" to mean "available in production"). While you're at it, check that the interactive product backlog contains any follow-on items that are on your roadmap.	TO DO	TBC
As part of our API delivery process we like to get feedback on how you have found our process. To help us improve our process, APIM runs a short producer survey with representative roles from your team. Contact your API producer liaison (currently Jonathan Eagle) to arrange a producer survey.	TO DO	TBC
Work closely with first-of-type and subsequent API consumers to get feedback on the API design (as they build) and on the API's functioning (once they put their software live): • this is often when you discover data quality issues • use their feedback to update API documentation to highlight data quality "features" and business rules that arise	TO DO	TBC

90 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/253886796/ITOC+runbook

Activity	Status	Comments / Evidence
Get feedback from real end users - work with API consumers to do this: • could be feedback questionnaires sent out to lots of end users • or face-to-face user testing with a few end users • make sure the API fits into their end-to-end journey as expected	TO DO	TBC
Aim to get a variety of API consumers of different types - larger organisations, smaller start-ups - as you will get different feedback from them.	TO DO	TBC
Iterate the beta service based on feedback from API consumers and end users.	TO DO	TBC
Inform the APIM team and Platforms Resourcing (platformsresourcing@nhs.net[91]) of any movers/ leavers since 'inception'. Use the Slack workflow in #platforms-apim-producer-support called 'Report a leaver' (for guidance on how to use workflows see the pinned message in the channel) and email Resourcing. **This should be an ongoing task.**	TO DO	TBC

Beta (public)

7.10.1.15 Overview

Public beta is a continuation of private beta, but with more participants and higher volumes.

GDS guidance: How the beta phase works[92]

7.10.1.16 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME

91 mailto:platformsresourcing@nhs.net
92 https://www.gov.uk/service-manual/agile-delivery/how-the-beta-phase-works

Skill	Name
Business analysis/service design	TBC NAME
User research	TBC NAME
API design (iterate)	TBC NAME
API documentation (iterate)	TBC NAME
API development	TBC NAME
API testing	TBC NAME
Technical architecture (iterate)	TBC NAME
Solution assurance	TBC NAME
Service transition	TBC NAME

7.10.1.17 Activities

Activity	Status	Comments / Evidence
Ensure you have a robust support capability in place that can support the service at scale. That includes having resource to handle the onboarding process.	TO DO	
Declare that you're moving into public beta. • Your API spec should already say you're in "in production + beta", so no need to change it.	TO DO	
Continue to gather feedback.	TO DO	
Continue to iterate.	TO DO	

Activity	Status	Comments / Evidence
Measure service against success metrics defined during discovery.	TO DO	
Conduct an API consumer integration survey. • See API consumer integration survey	TO DO	
Notify Live Services Board that you are exiting beta.	TO DO	
Update your API specification to "in production" (ie drop the "beta" part of the labels), to say it's not in beta. Also, change the API status tag to "in production ", to say that the API is now available in production. To change the tag and make this visible on the API catalogue, contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler	TO DO	
Conduct a beta exit review (see page 1154) with the API Management team. • This is really just to check through that you've done all of the above, although it might be a moot point if you've already been to Live Services Board	TO DO	

Live

Details to follow but likely to include:

- Publicise the fact that your service is now live (not beta).
- Ramp down the team, but ensure you still have a support capability. DO NOT punt it over the fence to a live service team that doesn't understand the thing they are running. You will need a DevOps model.
- Continue to monitor and iterate the service.
- If breaking changes have been delivered as a new version of the API, ensure older versions of the API are still available to allow API consumers time to migrate.

Deprecation and retirement

There is a separate checklist for deprecation and retirement - for details, see Deprecation and retirement phases[93].

93 https://nhsd-confluence.digital.nhs.uk/display/APM/Deprecation+and+retirement+phases

7.11 API process checklist using proxy generator

Instructions for using this template

Make a copy of this page, change the name to "API process checklist - <your API name>, and save it in your API's confluence space. Then delete this message (and the warning message below). If you need to request a new Confluence Space for your team please request via Live Service: Request a new Confluence space[94]

Version	Published	Changed By	Comment
CURRENT (see page 227) **(v. 30)**	**Mar 31, 2025 14:21**	Jonathan Eagle[95]	

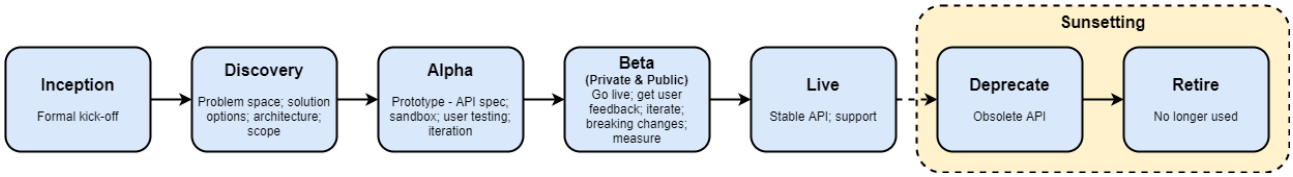
Go to Page History[96]

Instructions for updating this centralised template

All links in this template need to be **added as web links** otherwise the links can break when a producer copies this template into their own Confluence space

7.11.1 Overview

This page provides an audit trail of how the above API has been developed in line with the NHS England API delivery process (see page 194), including links to evidence where appropriat



94 https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_cat_item&sys_id=33fd05b91b680110892d4046b04bcb5e
95 https://nhsd-confluence.digital.nhs.uk/display/~jonathan.eagle1@nhs.net
96 https://nhsd-confluence.digital.nhs.uk/pages/viewpreviousversions.action?pageId=678899062

7.11.1.1 Status codes key

To change the status on the steps in the process checklist below, just copy and paste the relevant option from the ones listed here :

TO DO

IN PROGRESS

DONE

N/A

Inception

7.11.1.2 Overview

Inception is all about kicking the project off - introductions and paperwork.

7.11.1.3 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Technical architecture (optional)	TBC NAME

7.11.1.4 Activities

Activity	Status	Comments / Evidence
Learn about the API platform and its benefits - and make sure it is right for you - by reading Benefits of using the API platform (see page 134).	TO DO	TBC
Learn about the API delivery process by reading Deliver your API (see page 194).	TO DO	TBC

Activity	Status	Comments / Evidence
Conduct a delivery kick-off meeting (see page 249) with the API Management team and how to Get started as an API producer (see page 125).	TO DO	TBC
Create a Confluence space and/or page for your API.	TO DO	TBC
Create a copy of the API process checklist in **your Confluence space** (see instructions at top of the page)	TO DO	TBC
Add your API to the API register (see page 1204) and include links to your Confluence space and process checklist page. Leave the JIRA column blank, as this will be completed by the APIM team. **Note: If you cannot add to the API register you will need to send all the relevant details to your API management producer liaison (currently Jonathan Eagle) who will update this Confluence page.** You won't be able to complete all columns in the API registry, as you won't have created the artefacts yet. These will need to be updated when you reach the appropriate phase in the API delivery process.	TO DO	TBC
Add your API to the interactive product backlog (see page 250) to let API consumers know what we're up to.	TO DO	TBC
Confirm with the API management producer liaison that your API has been added to our API roadmap[97]	TO DO	TBC
Decide who will do the discovery for your API. • See API delivery and support options (see page 293).	TO DO	TBC
****This service is not available from April 2025 until further notice**** If you want us to do your discovery for you, raise a New Work Request to cover our discovery costs. • See Platform costs and raising a New Work Request (NWR) (see page 305) If you are doing discovery yourself, but want us to do the build, you don't need to raise an NWR until you start Alpha.	N/A	TBC

97 https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?rapidView=2826&view=detail&selectedIssue=AMR-127

Activity	Status	Comments / Evidence
To help guide and support you through the API delivery process, your API management liaison holds regular catch-ups with representive roles from your team your team. This starts with an an initial hour session to allow for a bit of orientation, followed by shorter focused 15 min sessions to review how you're progressing through the checklist and provide any help you might need. Contact you API management liaison (currently Jonathan Eagle) to arrange catch-up sessions	TO DO	TBC

Discovery

7.11.1.5 Overview

Discovery is all about figuring out what you're going to build, and confirming that it's actually needed.

GDS guidance: how the discovery phase works[98]

7.11.1.6 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Business analysis/service design	TBC NAME
User research	TBC NAME
Technical architecture	TBC NAME
FHIR API architecture (if relevant) (this requires, at least, a review by the Interoperability Standards team)	TBC NAME

98 https://www.gov.uk/service-manual/agile-delivery/how-the-discovery-phase-works

7.11.1.7 Stakeholders

(see Stakeholders for API Producer teams[99])

Role	Name
Information Governance (IG)	TBC NAME
Commercial and legal	TBC NAME
Security	TBC NAME
Solution Assurance	TBC NAME
Service Management	TBC NAME
Information Asset Owner	TBC NAME
Clinical Safety	TBC NAME If not applicable include rationale and relevant links

7.11.1.8 Activities

Activity	Status	Comments / Evidence
Identify and engage with internal stakeholders (see page 253) and add them to the table above.	TO DO	TBC
Understand the context and problem / opportunity space (see page 255).	TO DO	TBC

99 https://nhsd-confluence.digital.nhs.uk/display/APM/Stakeholders+for+API+Producer+teams

Activity	Status	Comments / Evidence
Identify the API consumers (see page 1172) who will be consuming your API. • This might be difficult if you're planning a "build it and they will come" API - think **very** carefully before adopting this approach - you're at risk of building something that doesn't meet any user needs and that costs money to maintain.	TO DO	TBC
Map out the user journey for the API consumer. This shouldn't be hard because it's generally the same for all APIs - see the "user journey" documented in Understanding API consumers - User journey (see page 257).	TO DO	TBC
Work with API consumers to understand and map out **their** end users and their user journeys (see page 258).	TO DO	TBC
Establish a user need (for the service overall and specifically for an API).	TO DO	TBC
Understand "as-is" (technology or otherwise) - including existing back-end services that could be used / re-used / need enhancing.	TO DO	TBC
Define your scope (see page 259) and agree this with your sponsoring stakeholders.	TO DO	TBC
Arrange an architecture kick-off meeting (see page 262) with your architect so that they can guide and advise you before you go too far with your architecture.	TO DO	TBC
Identify solution / architecture options for your API. Read top-level guidance Architecting your API (see page 260), which includes key aspects such as Is my API a FHIR API? (see page 285). Although not essential, it will help you if you use the API Architecture options template (see page 262). If the pages don't answer all your architecture queries, need some further architecture guidance or need some architectural resouce, see Help and support - Architecture support (see page 2056).	TO DO	TBC
Agree a draft architecture, based on your scope - see API architecture review checklist (see page 266).	TO DO	TBC

Activity	Status	Comments / Evidence
Conduct a API architecture review meeting (see page 289) with your architect to make sure you are heading in the right direction.	TO DO	TBC
Follow up with wider architecture governance sanity check by creating an SDO (Solution Design Overview) and possibly attending TRG - more details in the Wider architecture review (see page 290) guidance.	TO DO	TBC
Do a cost / benefit analysis (see page 304) to prove that it's worth building your API / service. Cancelling your project at the end of discovery because the benefits don't justify the costs is an entirely valid decision. If need be, and if you feel brave enough, use this activity to challenge a questionable political decision. At the very least, get your project sponsor to sign off the cost/benefit case (or lack thereof).	TO DO	TBC
Define metrics for service success. See Metrics for success / KPIs for APIs (see page 292).	TO DO	TBC
Decide who will deliver your API and who will support it long term. See API delivery and support options (see page 293).	TO DO	TBC
****This service is not available from April 2025 until further notice**** If we are delivering all, or part, of your API, raise a New Work Request (NWR) to cover our delivery costs. • See Platform costs and raising a New Work Request (NWR) (see page 305).	N/A	TBC
Conduct a discovery exit review (see page 307) with the API Management team. • This includes deciding whether to proceed to alpha.	TO DO	TBC

Alpha

7.11.1.9 Overview

Alpha, for an API, is all about getting the API published - as a spec and in test environments - and getting early feedback.

GDS guidance: How the alpha phase works[100]

7.11.1.10 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME
Business analysis/service design	TBC NAME
User research	TBC NAME
API design	TBC NAME
FHIR API design (if relevant) (this requires, at least, a review by the Interoperability Standards team)	TBC NAME
API development (for spec and sandbox)	TBC NAME
API test (for spec and sandbox)	TBC NAME
API documentation - API producer team member(s) (see Who should write your API documentation (see page 452))	TBC NAME
API documentation - API platform team point of contact (see Who should write your API documentation (see page 452))	TBC NAME
Technical architecture	TBC NAME

100 https://www.gov.uk/service-manual/agile-delivery/how-the-alpha-phase-works

7.11.1.11 Activities

Activity	Status	Comments / Evidence
****This service is not available from April 2025 until further notice**** Raise a New Work Request (NWR). If you haven't done so already, you must raise an NWR if the API is being delivered by an NHS England Platforms team or by us. • See Platform costs and raising a New Work Request (NWR) (see page 305).	N/A	TBC
On the interactive product backlog (see page 250), change the status to "in progress".	TO DO	TBC
Revisit the list of internal stakeholders you created during discovery and ensure you are still engaged with them. You are responsible for the assurance of your API, including security, IG and clinical safety. To ensure a smooth service transition into production during private beta, make sure you bring these stakeholders on the journey with you.	TO DO	TBC
Engage with your IG representative to liaise with legal, to determine if the connection agreement[101] requires updating with your new API	TO DO	TBC
Design your API. • See Designing your API (see page 322).	TO DO	
Conduct an API design review meeting (see page 375) (to primarily look at your API's physical design - endpoint names, granularity, FHIR conformance etc.) with your team.	TO DO	
Build your API proxy • If creating a specification only API proxy see Creating your spec only API specification (see page 696) • If your API is not a specification only API, see Building and deploying your API alpha (see page 680) for	TO DO	TBC

101 https://digital.nhs.uk/developer/guides-and-documentation/online-connection-agreement#top

Activity	Status	Comments / Evidence
Draft your API specification - mark it as "In development" (i.e. a prototype). • See Documenting your API (see page 445).	TO DO	TBC
Publish your API specification initially to UAT (API status should be tagged as "In development" in the API specification. To later make this visible on the API production catalogue contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler). **Your API does not need to be complete before it is published**, as the Alpha phase is all about getting feedback and iterating your specification. • See Publishing your specification to UAT (see page 571)	TO DO	TBC
Conduct an API specification content review meeting (see page 557) (to look at the word-smithery of your API specification).	TO DO	TBC
Consider building a "sandbox" service for your API - a callable API that simulates the API's behaviour, either using stubs or in a path-to-live test environment. • See Setting up your API sandbox (see page 684) for more details.	TO DO	TBC
Consider building a Postman collection for your API - this makes it easier for API consumers to understand how to use your API, but you do need to make sure you keep it in sync with any changes to your API. • See the PDS FHIR API specification[102] for an example.	TO DO	TBC
Consider building an open source reference implementation for your API - in whichever language(s) your API consumers use. Python, Java and C# are the usual contenders. This is a lot of effort but you could double it up as a test harness for your own use.	TO DO	TBC

102 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-description__environments-and-testing

Activity	Status	Comments / Evidence
Provide an integration environment for API consumers, for formal integration testing. • You might not need to do this if your API is simple enough - API consumers could test against production. See Integration testing environment for more details.	TO DO	TBC
Make sure you have thought about test data. • For sandbox, check you have aligned your canned responses with other APIs, as per Standard test data - Sandbox (including specs) (see page 606) • For integration test, check you have considered providing standard test data for read-only testing, as per Path-to-live environments (externally facing) - Test data[103] • ...and that you have aligned it with other APIs, as per Standard test data - Integration test (see page 607)	TO DO	TBC
Currently Proxy Generator doesn't support Spine APIs, so this is not yet applicable. If your API calls a Spine based API, there are a few other considerations: • We don't ask API consumers to provide ASIDs, but map to them • Except for Production there are some in-built generic ASIDs (see page 418) that might need additional permissions to work in the context of your API call See Spine based APIs (see page 408)	N/A	TBC
Test the API with real users (ideally with API consumers who actually want to use the API). • Ideally face-to-face - hackathons are also good • Micro-polls on your API doc page are also good • Be ready receive feedback If struggling to find a consumer to test with in INT or driven by project deadlines, as a minimum, some testing needs to have been performed in the INT environment (even if it's only by the API producer team).	TO DO	TBC

103 https://nhsd-confluence.digital.nhs.uk/display/APM/Path-to-live+environments#Pathtoliveenvironments-Testdata

Activity	Status	Comments / Evidence
Where possible, engage with testers from API consumer organisations to ensure you are meeting their testing needs. • Iterate the testing service based on feedback.	TO DO	TBC
Iterate the API design and test environments, based on the user feedback from real users, but bear in mind you'll get way more feedback in beta.	TO DO	TBC
Conduct an alpha exit review (see page 933) with the API Management team. • This includes deciding whether to proceed to beta, and whether to start with public or private beta. Go public unless you have very strong reasons to stay private, because the more open your API is, the more feedback you'll get during beta.	TO DO	TBC

Beta (private)

7.11.1.12 Overview

In private beta you:

- get your API into production
- get a small number of API consumers to use it
- iterate on their feedback

If you are delivering an API standard for a peer-to-peer API, you work with third party early adopters to deliver their APIs, rather than delivering one yourself.

GDS guidance: How the beta phase works[104]

7.11.1.13 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME

104 https://www.gov.uk/service-manual/agile-delivery/how-the-beta-phase-works

Skill	Name
Business analysis/service design	TBC NAME
User research	TBC NAME
API design (iterate)	TBC NAME
API documentation (iterate)	TBC NAME
API development	TBC NAME
API testing	TBC NAME
Technical architecture (iterate)	TBC NAME
Solution assurance	TBC NAME
Service transition	TBC NAME

Note : These are needed whether you decide to transition in technical go-live only or directly into private beta

Activity	Status	Comments / Evidence
Build your API service - first iteration / MVP. • You might have already done some of this if you provided a formal integration testing environment.	TO DO	TBC

Activity	Status	Comments / Evidence
Prepare your service for production (see page 945) : • Go through the appropriate Service transition process (see page 947). **THIS CAN TAKE A LONG TIME**	TO DO	TBC
• Prepare the consumer onboarding (see page 999) process for your API, and if necessary get it approved at OGG. **THIS CAN TAKE A LONG TIME!**	TO DO	TBC
• Conduct a penetration test (see page 968). If your penetration test is low risk you may be able to do this internally (if agreed with Cyber Security).. **IF YOU NEED AN EXTERNAL PENETRATION TEST THERE IS A LEAD TIME OF 6 WEEKS**	TO DO	TBC
Consider reviewing the Api platform hazard log (see page 966) with your clinical lead	TO DO	TBC
Engage with the CSOC to ensure proactive security monitoring, email england.csoc.services@nhs.net[105]	TO DO	TBC
For APIs containing Personally Identifiable Data (PID or PII), review if the Information Asset Owner requires **APIM** on-call rota staff to raise Asset Access Request (AAR) form in order to access raw logs information case during an incident. • For details, see Supporting your API (see page 964)	TO DO	TBC
Complete the following first time into production activities (required for the first time deploying into production review (see page 1120) meeting). This is an API platform-specific checklist which is additional to the above service transition process.	TO DO	TBC

105 mailto:england.csoc.services@nhs.net

7.11.1.13.1 First time deploying into production activities

Category	Activity	Status	Comments / Evidence
Rate limiting	Ensure you have set up rate limiting for your API. This might be a low rate limit for your private beta (if you haven't yet scaled and load tested your API). Or it might be a higher rate limit if you have already scaled and load tested your API. • For details, see Rate limiting (see page 1117)	TO DO	TBC
Monitoring	This is mostly automated, you need to confirm: • You have told the APIM _status endpoint where your backend health check • Your OAS file has monitoring enabled (aka "ping and status" endpoints) so that you can see the health of your API and get alerts For details, see Adding status monitoring to a service (see page 1065)	TO DO	TBC
	Ensure you have set up and enabled status monitoring for your sandbox (if you have one). • For details see Monitoring your API (see page 1115)	TO DO	TBC
Logging	Check that you can see your logs in the APIM Splunk indexes for your API in the path-to-live environments • For details, see Logging to Splunk (see page 1082).	TO DO	TBC
	You **must** request sensitive data redaction APIM, even if you are stating there is no sensitive data. This will enable people who are not security cleared can view your logs via the open-access dashboards. Check that redaction works in your path-to-live environments. • For details, see Logging to Splunk - Set up sensitive data redaction (see page 1088)	TO DO	TBC

Category	Activity	Status	Comments / Evidence
Spine	Currently Proxy Generator doesn't support Spine APIs, so this is not yet applicable If your API calls a Spine based API, you should have worked through your Spine specific configuration -see Spine based APIs (see page 408) for first time into Production work	N/A	TBC
Smoke testing	Prepare your automated tests for when you go live (if possible). • For details, see Smoke testing (see page 1145)	TO DO	TBC
Auto-approval	Not available in Digital Onboarding Service (DOS) yet	N/A	TBC
Production instance configuration	Review instance payload - e.g have the required secret been posted and available for use • See Configuring your API backend[106]	TO DO	TBC
OAS Spec Checkpoint	Has OAS spec been published?	TO DO	TBC

Activity	Status	Comments / Evidence
Conduct a first time into production review (see page 1120) meeting. This is basically a check through of the above first time into production activities.	TO DO	TBC

106 https://nhsd-confluence.digital.nhs.uk/display/APM/Configuring+your+API+backend+with+Proxy+Generator

Activity	Status	Comments / Evidence
Do you need approval for technical go-live ? See Service transition for APIs - Enter technical go-live (see page 948). Alternatively, if you have already completed your onboarding process, you might decide to get approval to go directly to private beta - see Service transition for APIs - Enter private beta (see page 949).	TO DO	TBC
Deploy your service to production. • For details, see Deploying to production (see page 1135)	TO DO	TBC
If you have automated smoke tests, check that they ran successfully. If you don't have any automated smoke tests, run a manual smoke test. • For details of how to test your API as a "consumer", see Smoke testing - Apigee Test Application (see page 1147) • As this is a production app, enablement is manual. The nominated onboarding lead (see page 1000) will also need to request "enablement of the Application" and any "custom attributes" via the platforms-apim-producer-support workflows. Also where applicable, request ASIDs (see page 414) (not supported with Proxy Generator)	TO DO	TBC
Check your open access dashboard in Splunk is working for your API. You should be able to see your smoke test(s). This is applicable to all APIs. • For details, see Logging to Splunk - Set up production indexes	TO DO	TBC

At this point, you are technically live, but not yet in private beta.

7.11.1.14 Enter private beta activities

Activity	Status	Comments / Evidence
Make sure you are geared up to support the service: • Set up first line support for API consumers. • Make sure the business is ready. • Consider getting your technical author involved closely in first or second line support, with a view to them updating documentation on the back of user queries. If API is transitioning to another team see API handover guidance (see page 963) for some areas to consider when handing over.	TO DO	TBC
Identify a first-of-type API consumer. This can be a bit chicken and egg as you haven't yet advertised your API as being available in beta, but hopefully you've had some feedback via the interactive product backlog or you're already engaged with a keen API consumer from alpha.	TO DO	TBC
Onboarding you API consumer. Ideally you will need an onboarding lead See API consumer onboarding - Onboarding lead (see page 1000)	TO DO	TBC
Enter private beta. • You ideally need a first-of-type API consumer lined up for this. • You'll also need to agree the restrictions of your private beta with Service Management e.g. maximum number of participants, maximum TPS throughput. • See Service transition for APIs- Enter private beta.	TO DO	TBC
Out of hours support active and genuine Production data: • If your service requires any out of hours support (i.e. Gold or Platinum), the APIM on-call rota needs warning. Work with API producer liaison to inform	TO DO	TBC

Activity	Status	Comments / Evidence
the core APIM team (this includes updating the ITOC runbook[107])		
Update your API spec: • Change the API status tag to "in production + beta", to say that the API is now available in production (you can still reserve the right to turn people away). To change the tag and make this visible on the API catalogue, contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler • Add a section into the spec to state what the API's service level is, as per What goes in your API documentation (see page 453).	TO DO	TBC
On the interactive product backlog (see page 250), change the API status to "done" (we use "done" to mean "available in production"). While you're at it, check that the interactive product backlog contains any follow-on items that are on your roadmap.	TO DO	TBC
As part of our API delivery process we like to get feedback on how you have found our process. To help us improve our process, APIM runs a short producer survey with representative roles from your team. Contact your API producer liaison (currently Jonathan Eagle) to arrange a producer survey.	TO DO	TBC
Work closely with first-of-type and subsequent API consumers to get feedback on the API design (as they build) and on the API's functioning (once they put their software live): • this is often when you discover data quality issues • use their feedback to update API documentation to highlight data quality "features" and business rules that arise	TO DO	TBC

107 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/253886796/ITOC+runbook

Activity	Status	Comments / Evidence
Get feedback from real end users - work with API consumers to do this: • could be feedback questionnaires sent out to lots of end users • or face-to-face user testing with a few end users • make sure the API fits into their end-to-end journey as expected	TO DO	TBC
Aim to get a variety of API consumers of different types - larger organisations, smaller start-ups - as you will get different feedback from them.	TO DO	TBC
Iterate the beta service based on feedback from API consumers and end users.	TO DO	TBC
Inform the APIM team and Platforms Resourcing (platformsresourcing@nhs.net[108]) of any movers/ leavers since 'inception'. Use the Slack workflow in #platforms-apim-producer-support called 'Report a leaver' (for guidance on how to use workflows see the pinned message in the channel) and email Resourcing. **This should be an ongoing task.**	TO DO	TBC

Beta (public)

7.11.1.15 Overview

Public beta is a continuation of private beta, but with more participants and higher volumes.

GDS guidance: How the beta phase works[109]

7.11.1.16 Team members

Skill	Name
Delivery management	TBC NAME
Product management	TBC NAME

108 mailto:platformsresourcing@nhs.net
109 https://www.gov.uk/service-manual/agile-delivery/how-the-beta-phase-works

Skill	Name
Business analysis/service design	TBC NAME
User research	TBC NAME
API design (iterate)	TBC NAME
API documentation (iterate)	TBC NAME
API development	TBC NAME
API testing	TBC NAME
Technical architecture (iterate)	TBC NAME
Solution assurance	TBC NAME
Service transition	TBC NAME

7.11.1.17 Activities

Activity	Status	Comments / Evidence
Ensure you have a robust support capability in place that can support the service at scale. That includes having resource to handle the onboarding process.	TO DO	
Declare that you're moving into public beta. • Your API spec should already say you're in "in production + beta", so no need to change it.	TO DO	
Continue to gather feedback.	TO DO	
Continue to iterate.	TO DO	

Activity	Status	Comments / Evidence
Measure service against success metrics defined during discovery.	TO DO	
Conduct an API consumer integration survey. • See API consumer integration survey	TO DO	
Notify Live Services Board that you are exiting beta.	TO DO	
Update your API specification to "in production" (ie drop the "beta" part of the labels), to say it's not in beta. Also, change the API status tag to "in production ", to say that the API is now available in production. To change the tag and make this visible on the API catalogue, contact someone in the APIM team with Bloomreach access, such as Nathan Kinsler	TO DO	
Conduct a beta exit review (see page 1154) with the API Management team. • This is really just to check through that you've done all of the above, although it might be a moot point if you've already been to Live Services Board	TO DO	

Live

Details to follow but likely to include:

- Publicise the fact that your service is now live (not beta).
- Ramp down the team, but ensure you still have a support capability. DO NOT punt it over the fence to a live service team that doesn't understand the thing they are running. You will need a DevOps model.
- Continue to monitor and iterate the service.
- If breaking changes have been delivered as a new version of the API, ensure older versions of the API are still available to allow API consumers time to migrate.

Deprecation and retirement

There is a separate checklist for deprecation and retirement - for details, see Deprecation and retirement phases[110].

110 https://nhsd-confluence.digital.nhs.uk/display/APM/Deprecation+and+retirement+phases

7.12 Inception phase

7.12.1 Overview

Inception is all about kicking off your project.

You will:

- learn about the API delivery process
- kick off your API delivery formally by
    - engaging with the API Management team
    - doing some paperwork

7.12.2 Process steps

For detailed process steps, see the API process checklist using proxy generator (see page 227) (Default checklist) or API process checklist (see page 198). if Proxygen is deemed not suitable for your API.

7.12.3 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

- Delivery kick-off meeting (see page 249)
- Getting your API on the interactive product backlog (see page 250)

7.12.4 Delivery kick-off meeting

Hold a delivery kick-off meeting during the inception phase (see page 249) to formally kick off your engagement with us.

This is to:

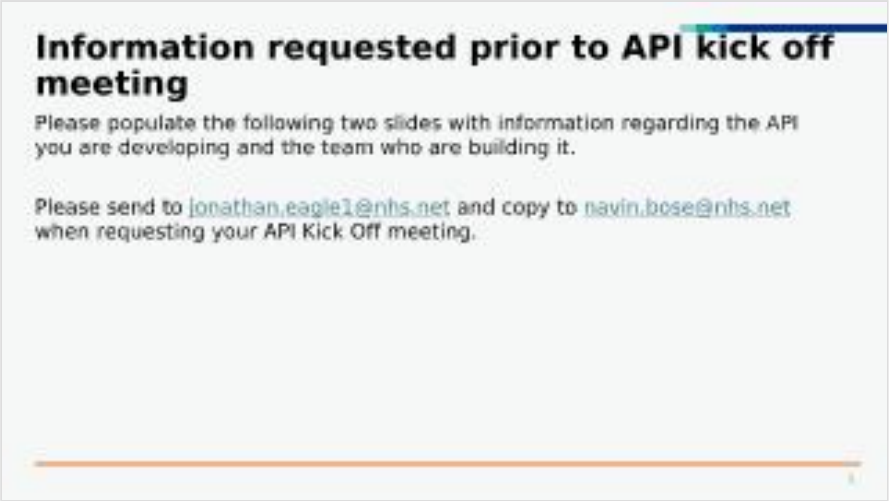
- introduce your project to the API Management team so we know what you're up to
- give an overview of the delivery process, confluence documentation and support channels
- give early advice on your plans / approach
- answer any initial questions you have

Attendees:

- API Producer team:
    - Product Owner
    - Delivery Lead
    - (optional) Technical Architect
- API Management team
    - Lead Product Owner (currently Navin Bose - navin.bose@nhs.net[111])
    - API producer liaison (currently Jonathan Eagle - jonathan.eagle1@nhs.net[112])
    - (optional) Technical Architect (Aubyn Crawford - aubyn.crawford1@nhs.net)

Duration: Up to 1 hour

Prior to the meeting to help set the context and allow us to understand a little about your planned API and your team ahead, we request that you download a copy and complete the two PowerPoint Slides and send them to Jonathan Eagle and Navin Bose when requesting your kick off meeting.



7.12.5 Getting your API on the interactive product backlog

7.12.5.1 Overview

A key part of our strategy is to engage with external software developers to get their input and feedback.

The **interactive product backlog** is part of this engagement strategy. It is an externally-visible backlog of all our epics, including:

- APIs (or API improvements)
- API platform improvements

It allows developers to:

- upvote APIs / platform improvements

111 mailto:navin.bose@nhs.net
112 mailto:daniela.simonato1@nhs.net

- comment on APIs / platform improvements (we are able to review these before publishing them)
- suggest new APIs / platform improvements (we are able to review these before publishing them)

The backlog tool also **captures email addresses** for anyone that votes for a feature or comments on it, so this is a good source for recruiting participants for your alpha/beta.

You can access it via the developer hub: https://digital.nhs.uk/developer - see the banner on the vision[113] for the link. Or go direct to: https://nhs-digital-api-management.featureupvote.com/ (you will need to be logged in to make any changes)

The interactive product backlog runs on a SaaS product called Feature Upvote (https://featureupvote.com/). We've customised it a bit (e.g. NHS Digital logo) but it's pretty much out of the box.

7.12.5.2 How to use it

As an API producer team you should:

- During inception:
  - Get an admin account for our Feature Upvote board (ask Navin Bose or your producer liaison currently Jonathan Eagle).
  - Add your API to the interactive product backlog (done by adding it as Suggestion)- this might be a single feature or, if you have phases planned, several features which constitute a roadmap.
  - Tag all these features with the "api" tag.
  - Create a new tag for your API so you can easily identify all the features for your API - for example, the PDS FHIR API features all have a tag of "pds-fhir-api"
  - Set the status of the features as appropriate e.g. "under consideration", "planned" or "in progress" - this depends on how confident you are that the API will proceed beyond discovery
- During alpha:
  - At the start of alpha, change the API status to "in progress"
  - Link to the interactive product backlog from the "API status and roadmap" section of your API spec page, using a link which pre-filters the backlog for your API features. For example, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir.
  - Add a comment to the feature to announce that the API is now in alpha, and invite alpha participants. This should automatically notify anyone who has upvoted/commented and left their email address
  - Get the emails and maybe email those people directly as well
  - Monitor the board for feedback (you should get a weekly digest)
  - If any new features are raised for your API:
    - tag them with "api" and your API-specific tag
    - alter the status accordingly - "planned" if you plan to do it, "not planned" if you don't plan to do it - in the latter case, say why
- During beta:

113 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision

- Add a comment to announce your API is in beta
- Change the status to "done" (done means available in production in beta)

7.12.5.3 Accessing Feature Upvote as a moderator or admin

To get added as a moderator or an admin, ask an existing admin, such as Navin Bose or producer liaison currently Jonathan Eagle .

Being added will trigger an email to you to set up your account.

To sign in, go to Dashboard - Feature Upvote[114].

7.13 Discovery phase

7.13.1 Overview

Discovery is about understanding the problem that needs to be solved.

You will:

- identify your stakeholders
- map out the user journey for your API consumers
- identify solution / architecture options for your API and agree a draft architecture

7.13.2 Process steps

For detailed process steps, see the API process checklist (see page 198).

7.13.3 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

- Identify and engage with internal stakeholders (see page 253)
- Understanding the Context and Problem/Opportunity Space (see page 255)
- Understanding API consumers (see page 256)
- Mapping Out the User Journey for API Consumers (see page 258)
- Define your scope and agree this with your sponsoring stakeholder (see page 259)
- Architecting your API (see page 260)

114 https://app.featureupvote.com/dashboard/

- Metrics for success / KPIs for APIs (see page 292)
- API delivery and support options (see page 293)
- Different types of API (see page 297)
- Access modes - compliance - security and simplifying running your API (see page 299)
- Cost / Benefit Analysis (see page 304)
- API platform costs and raising a New Work Request (NWR) (see page 305)
- Discovery exit review (see page 307)
- Products and Product IDs (see page 308)
- Analysis of API Migration from Non-OAS to OAS Format (see page 314)

7.13.4 Identify and engage with internal stakeholders

7.13.4.1 Why it is important to engage with Internal stakeholders

Engaging with internal stakeholders is crucial for the success of any project. Here's why:

Shared Understanding: Ensures everyone involved has a common understanding of project goals, objectives, and deliverables. Achieving this common understanding reduces misunderstandings and miscommunications

Increased Buy-in: Involving stakeholders early on builds trust and commitment, enabling them to feel more invested in the project's success.

Identification of Risks and Issues: Stakeholders can identify potential risks and challenges that might not be obvious to the project team. Gaining this early identification of issues allows for timely mitigation and potentially reduces costs.

Resource Allocation: Stakeholders can provide input on resource allocation, ensuring that the project has the necessary resources to succeed.

Decision-Making: Involving stakeholders in decision-making processes can lead to more informed and effective decisions which enables the project to build consensus and avoid conflicts.

Change Management: Engaging stakeholders early on can help manage change effectively. Engaging stakeholders in the change process, can minimise resistance and increase acceptance.

Improved Communication: Regular communication with stakeholders helps to keep them informed and engaged, which can facilitate opportunities to gain feedback and address any concerns or questions.

Actively engaging with internal stakeholders increases the likelihood of project success, improve team morale and build stronger relationships within the organisation.

7.13.4.2 Stakeholders for API producer teams

When you're building an API, you'll need to engage with a whole load of stakeholders to help you along the way. They will help and advise you. The stakeholders listed below are key contacts for the APIM and you as a producer team should have your own subject matter experts (SMEs) to liaise with the APIM SMEs below.

For an example stakeholder list see Stakeholders (see page 253).

Here's a list of some of the more common stakeholders and what they do.

Stakeholder	What they do	Contact
Information Governance (IG)	Help you make sure your service respects users' privacy. Help you ensure your service complies with GDPR. Help you to define the data processing/ sharing model and processor/control relationships for your service (to inform the legal agreements). Stop your service going live if it doesn't - they are a sign-off at the Live Services Board.	ighelplineservice@nhsdigital.nhs.uk[115]
Commercial and Legal	Help you to apply the data processing/ sharing model to the contents of the legal agreements.	Nawshad Hossain-Ibrahim (Commercial) Lexi Moffatt (Legal)
Security	Help you make sure your service is secure. Help you arrange pen testing. Stop your service going live if it isn't secure - they are a sign-off at the Live Services Board.	Mike Farrell
Solution Assurance	Help you make sure you've tested your service. Help you define what external developers will need to do to test that their software is suitable to connect to your service (known as onboarding). Stop your service going live if either of the above isn't covered - they are a sign-off at the Live Services Board.	Simon Branton
Service Management	Help you define / design a whole load of things you need to have in place before you go live (e.g. incident management) Help you pull together everything you need to go to the Live Services Board.	Gemma Lofthouse

115 mailto:ighelplineservice@nhsdigital.nhs.uk

Stakeholder	What they do	Contact
Information Asset Owner	Every service that manages data should have an Information Asset Owner (IAO), who is ultimately responsible for the data being managed. For example, there's an IAO for PDS. (Stephen Smith) The IAO is likely to have a team of people working for them. The IAO and their team are likely to understand the service your API is giving access to and will likely need to approve the API. They should also be in a good place to give help and guidance when designing the API.	Depends on the service. Ask around!
TRG	Make sure your service is architecturally sound.	Aubyn Crawford
Clinical Safety	Help make sure your API is clinically safe.	Raman Behl Sean White
API Management Team (that's us!)	Give help and support to deliver your API. Help you make sure you've built a good service by checking you've followed the API delivery process.	api.management@nhs.net[116]

7.13.5 Understanding the Context and Problem/Opportunity Space

For API development this refers to gaining a deep understanding of the specific environment, goals, and challenges that the API will address. It involves:

Contextual Understanding

- Business Objectives: Clearly defining the business goals and objectives that the API is intended to support.
- Target Audience: Identifying the primary users or applications that will interact with the API.
- Industry Trends: Understanding relevant industry trends, regulations, and best practices.
- Competitive Landscape: Analysing competing APIs or similar solutions.

Problem/Opportunity Identification

- Pain Points: Identifying the specific problems or challenges that the API aims to solve.

116 mailto:api.management@nhs.net

- Value Proposition: Articulating the unique value that the API will provide to users.
- Opportunities: Recognising potential opportunities for growth and innovation.

Key Questions to Consider:

- What is the business need or problem the API is trying to solve?
- Who are the primary users or target audience for the API?
- What are the key requirements and constraints for the API?
- How will the API create value for the organisation or its customers?
- What are the potential risks and challenges associated with developing and deploying the API?

By thoroughly understanding the context and problem/opportunity space, API developers can ensure that the API is designed and built to meet the specific needs of its users and deliver maximum value. This understanding will also help guide decision-making throughout the API development process.

7.13.6 Understanding API consumers

7.13.6.1 Overview

This page describes the different types of API consumers, their needs and their user journey (delivering healthcare tech software).

More help and guidance in identifying wider users of APIs, user research methods and recruiting API consumers can be found at User research for APIs (see page 139)

7.13.6.2 Archetypes

We originally identified three API consumer archetypes:

1. The Establishment - larger orgs, well-established in the UK healthcare market
2. The Analysts - larger orgs looking to break into the UK healthcare market from other sectors or territories
3. The New Order - small start-ups looking to disrupt the UK healthcare market

We subsequently expanded that into six areas:

1. The Establishment
2. Established Disrupters
3. Connected Evolvers
4. Connected Sustainers
5. Tech Start-Ups
6. Niche Small Sustainers

7.13.6.3 Roles within development organisations

From other user research and our own knowledge of healthcare software development, we know there are multiple roles within each archetype:

- Programmer
- Architect
- Product Owner
- Tester
- Project Manager / Scrum Master
- Security SME
- Information Governance SME
- Clinical Safety SME
- ...and more

We aspire to cater for them all. We especially hope to remember that, as API consumers, **not all API consumers are just like us**. For example, they might be:

Easy Target	**Harder Target**
From large, established organisations	From small start-ups
Seasoned professionals	Have-a-go heroes
Textual thinkers	Visual thinkers
Architectural purists	Pragmatists
Knowledgeable about the UK healthcare market / IT / terminology	Ignorant of the UK healthcare market / IT / terminology
Just like us	Not at all like us

7.13.6.4 User journey

API consumers deliver health and care tech software.

A good way to understand the user journey is to look at our externally-facing "getting started" page at https://digital.nhs.uk/developer/getting-started.

High level user journey available on Mural:

https://app.mural.co/t/nhsdigital8118/m/nhsdigital8118/1727877745194/58242ac3caaaba8d12b908d0e5a1058355ce4f64?wid=0-1744628393390

7.13.7 Mapping Out the User Journey for API Consumers

Understanding the user journey for your API consumers is crucial for designing an effective and user-friendly API. Here are some key steps to consider:

Identify Your Target Audience:

- Who are the primary users of your API?
- What are their technical skills and experience levels?
- What are their goals and expectations when using the API?

Define Key Use Cases:

- Identify the most common scenarios or tasks that users will perform with your API.
- Consider different user roles and their specific needs.

Create User Personas:

- Develop fictional user profiles representing your target audience.
- Consider their goals, motivations, and challenges.

Map the User Journey:

- Visualise the steps a user will take to achieve their goals using your API.
- Consider factors like:
- Onboarding: How easy is it for new users to get started?
- Authentication: What authentication methods are required?
- API Usage: What API calls and parameters are needed?
- Error Handling: How are errors and exceptions handled?
- Documentation: Is the documentation clear and helpful?

Gather Feedback:

- Conduct user interviews or surveys to gather feedback on the user journey.
- Identify pain points and areas for improvement.

Iterate and Improve:

- Continuously refine the user journey based on feedback and evolving needs.
- Make necessary changes to your API design or documentation to enhance the user experience.

Key Considerations:

- Simplicity: Make the API as simple and intuitive as possible.
- Consistency: Ensure consistent naming conventions and error messages.
- Clear Documentation: Provide comprehensive and easy-to-understand documentation.
- Security: Implement robust security measures to protect user data.
- Performance: Optimize the API for speed and efficiency.

By carefully mapping out the user journey, you can create an API that is easy to use, well-documented, and meets the needs of your target audience.

7.13.8 Define your scope and agree this with your sponsoring stakeholder

Gaining Stakeholder Agreement for Your API Scope

Securing agreement from stakeholders on your API scope is crucial for successful development and deployment. Here are some strategies to achieve buy-in:

Clear Communication and Education

- Explain the Benefits: Clearly articulate the value that the API will bring to the organisation/service/users and how it aligns with business objectives.
- Address Concerns: Anticipate and address potential concerns or objections from stakeholders.
- Provide Examples: Use real-world examples or case studies to illustrate the API's potential impact.

Involve Stakeholders Early

- Seek Input: Involve stakeholders from the beginning of the API development process to ensure their needs and perspectives are considered.
- Collaborative Workshops: Conduct workshops or brainstorming sessions to gather input and build consensus.

Prioritise and Justify

- Prioritise Features: Clearly articulate the rationale for including or excluding specific features based on business value and technical feasibility.
- Justify Decisions: Provide evidence and data to support your decisions and address stakeholder concerns.

Iterative Approach

- Be Flexible: Be open to feedback and adjust the scope as needed based on evolving requirements or constraints.
- Iterative Development: Consider an iterative development approach to allow stakeholders to review and provide input at different stages of the process.

Leverage Data and Evidence

- Use Metrics: Support your decisions with data and metrics, such as market research, user feedback, or technical feasibility assessments.
- Demonstrate Value: Highlight the potential benefits and ROI of the API to gain stakeholder support.

Address Concerns and Objections

- Actively Listen: Pay attention to stakeholder concerns and questions.
- Provide Clear Explanations: Address concerns with clear and concise explanations.
- Offer Alternatives: If necessary, propose alternative solutions or compromises to address objections.

Obtain Formal Approval

- Document Agreement: Once consensus is reached, document the agreed-upon scope in a formal document.

- Seek Formal Approval: Obtain formal approval from key stakeholders or decision-makers.

By following these strategies, you can effectively communicate the value of your API, address stakeholder concerns, and gain the necessary buy-in to proceed with development.

7.13.9 Architecting your API

Understanding Architecture and Design in Our API Delivery Process

While our API delivery process separates architecture and design into distinct phases, we recognise that these activities often overlap. However, this distinction helps us make informed decisions throughout the development lifecycle.

Architecture: The Foundation

In the discovery phase, we focus on making high-level architectural decisions. These include:

- Interaction Style: Choosing between RESTful (default) or Messaging (if necessary).
- FHIR Compliance: Determining if the API should adhere to the FHIR standard.
- Backend Integration: Identifying the location of backend services and how the API proxy will connect to them.
- Security: Defining the necessary security measures and protocols.
- Service Level: Within NHS Digital, there are number of defined service levels (bronze, silver, gold, platinum). You need to decide which service level is appropriate for your API. Choose the lowest service level that is "good enough" - higher service levels cost more to run.
- Note that:
    - the API platform itself is a PLATINUM service, so can support APIs at any level
    - NHS CIS2 (which the API platform uses for healthcare worker authentication/authorisation) is also a platinum service
    - NHS login (which the API platform uses for patient authentication/authorisation) is TBC

Key Steps for Architecture:

- Learn Our Integration Practices: Familiarise yourself with NHS Digital's integration patterns by reading the "Integration Patterns Book[117]."
- Assess FHIR Compliance: Determine if your API aligns with the FHIR standard by reading "Is my API a FHIR API? (see page 285)"
- Schedule an Architecture Kick-off (see page 262): Meet with your architect to get initial guidance and direction.
- Explore Options: Identify and document potential architecture options using our API architecture options template (see page 262).
- Draft Your Architecture: Write up a draft architecture, referencing our API architecture review checklist (see page 266).
- Seek Feedback: Conduct an architecture review meeting (see page 289) with your architect to ensure you're on the right track.

117 https://digital.nhs.uk/developer/architecture/integration-patterns-book

- Obtain Wider Approval (see page 290): Present your architecture to a wider audience for feedback and validation.

Architecture design principles

The following are an API Producer focused subset of the Platform Architecture design principles[118]:

- The API Platform (Proxy) is loosely coupled with the API backend
- The APIs should be RESTful and synchronous
- The API end points exposed should all be consistent across the platform
- The API Management Platform will reduce the assurance burden for API Consumers
- The APIM coordinated authentication process uses national identity providers and will be seamless between different APIs
- All configuration is stored and managed in a GitHub repo for that API

In the future coarse grained authorisation based on the National RBAC model will be incorporated into the Platform.

Design: The Building Blocks

In the alpha phase, we shift our focus to detailed design decisions. This includes:

- Finalising the Base URL: Determining the main address for accessing your API.
- Defining Endpoints and Methods: Specifying the available API endpoints and the HTTP methods (GET, POST, PUT, DELETE, etc.) associated with each.
- Structuring Requests and Responses: Defining the format and content of API requests and responses, including headers, parameters, and bodies.
- Defining Business Rules: Establishing the logic and conditions that govern API behaviour.
- Extending FHIR (if applicable): Collaborating with the IOPS team to design any necessary extensions for FHIR-based APIs.
- Minimising Data Requirements: Identifying opportunities to reduce the amount of information required from API consumers by leveraging external data sources like the NHS Digital Terminology Server[119].

By carefully considering these design elements, we can create a well-structured and user-friendly API that meets our organisation's needs.

For more details, see Designing your API (see page 322).

Getting started

To get started with architecting your API, refer to the architecture steps in the discovery section of our API process checklist (see page 198).

For help and support, see Help and support - architecture support (see page 2056).

Supporting information

Use the following pages to help inform your architecture work:

118 https://nhsd-confluence.digital.nhs.uk/display/APM/Architecture
119 https://digital.nhs.uk/services/terminology-server

- Architecture kick-off meeting (see page 262)
- API architecture options template (see page 262)
- API architecture review checklist (see page 266)
- Routing requests to multiple backends (see page 280)
- Is my API a FHIR API? (see page 285)
- Service levels for APIs (see page 288)
- API architecture review (see page 289)
- Wider architecture review (see page 290)
- Large payloads (see page 292)

Results of the API / FHIR Survey : 28/02/2023 - Developer FHIR / Restful API Survey Results[120]

7.13.9.1 Architecture kick-off meeting

Hold an architecture kick-off meeting with your architect during discovery phase (see page 252) - before you go too far with your architecture efforts.

This is to:

- Explain the architecture process to you
- Ensure you're adhering to the API standards NHSE have defined
- Discuss architecture options with you
- Answer any questions you have
- Give you a point of contact for further questions

Attendees:

- A selection of your API producer team members
- API producer team architect
- IOPS tech lead (if you think you might need a FHIR API - see Is my API a FHIR API? Who are IOPS (see page 288)?)

7.13.9.2 API architecture options template

Consciously considering solution / architecture options is a really good idea - it avoids the "tunnel" vision that sometimes occurs when a team is presented with a fait accompli solution to a problem. You can use it to challenge what you've been asked to deliver.
This template gives you a structure to organise your options analysis. You don't have to use it, but we've found it helpful.
To use this page:

- Create a copy of it in an appropriate place in your own Confluence space
- Rename it to "API architecture options" (i.e. remove the word "template" from the page title!)

120 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=608904062

- Fill out the page - deleting anything you don't need, including this tip. When you edit this page there are number of hints and further help.

7.13.9.2.1 Overview

This page explains the solution / architecture options considered for the above API, and states which option has been selected, and why.

Consciously considering options avoids the "tunnel vision" that sometimes occurs when a team is presented with a fait accompli solution to a problem.

Here are some of the things we have considered:

Consideration	Notes
Range of options	Make sure a good range of options are worked through. Brainstorm options within the team.
Architecture patterns	The Integration Pattern Book[121] helps in the early stage of architectural decision making, guiding you in the choice of architectural pattern which best fits your business use case. For most scenarios, this advice will guide you towards implementing a synchronous API through API Management. It would be a good idea though to confirm, using the pattern book, that there is not a more appropriate architectural pattern for your use case.
REST and FHIR	If the pattern book is driving you towards and API - then it will use the RESTful design pattern. However, you need to make an initial proposal of whether it is FHIR based or not - see Is my API a FHIR API? (see page 285)
Authorisation	The API platform offers the following options: https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation. A discussion with Cyber Security and Privacy, Transparency and Ethics groups will ultimately be need to finalise this.

121 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

Consideration	Notes
API granularity	The definition of an "API" is quite broad, but we want to focus that down to help with the "cookie cutter" approach of the Platform. If you are in the position that all the answers below are yes - there is probably nothing further to discuss, it is clearly one service. • Is this an established "service" known publicly by a single title? • When developers come to the API Catalogue on digital.nhs.uk[122] - do they expect information to come under one heading? • Do all the API endpoints get deployed from one source repository? • Stakeholders who want statistics and reporting, do they consider this as a unit?

7.13.9.2.2 Context and user needs

<describe the context (as-is situation) and the user needs or opportunity you're hoping to achieve by delivering the API>

7.13.9.2.3 Option 1

7.13.9.2.3.1 Details

Summary	<include a written summary of the option>
Context diagram	<include a diagram showing the key components and how information flows amongst them>
Integration pattern	<state the integration pattern being used with reference to the the Integration Pattern Book[123] e.g. request / response; registry / repository; information push; publish / subscribe; stared repository>
Type of API	<delete this row if this option does not involve an API> <with reference to Is my API a FHIR API? (see page 285), state the type of API - is it RESTful; is it FHIR; which version of FHIR>

122 http://digital.nhs.uk
123 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

API consumer authorisation	<delete this row if this option does not involve an API> <with reference to https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation, state the authorisation pattern>
API granularity	<delete this row if this option does not involve an API> <explain whether this will be one API or many (in this context, "API" means a single entry in our API catalogue - possibly with many endpoints>

7.13.9.2.3.2 Pros

<explain the pros for this option>

7.13.9.2.3.3 Cons

<explain the cons for this option>

7.13.9.2.4 Option 2

7.13.9.2.4.1 Details

Summary	<include a written summary of the option>
Context diagram	<include a diagram showing the key components and how information flows amongst them>
Integration pattern	<state the integration pattern being used with reference to the the Integration Pattern Book[124] e.g. request / response; registry / repository; information push; publish / subscribe; stared repository>
Type of API	<delete this row if this option does not involve an API> <with reference to Is my API a FHIR API? (see page 285), state the type of API - is it RESTful; is it FHIR; which version of FHIR>
API consumer authorisation	<delete this row if this option does not involve an API> <with reference to https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation, state the authorisation pattern>

124 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

API granularity	<delete this row if this option does not involve an API> <explain whether this will be one API or many (in this context, "API" means a single entry in our API catalogue - possibly with many endpoints>

7.13.9.2.4.2 Pros

<explain the pros for this option>

7.13.9.2.4.3 Cons

<explain the cons for this option>

7.13.9.2.5 Option 3

<add further options if you have any - brainstorming options with the whole team or with us is a good way to identify further options>

7.13.9.2.6 Selected option

<state which option has been selected>

For more details on the selected option, see <insert a hyperlink to your detailed architecture page>.

7.13.9.3 API architecture review checklist

To use this page:

- Create a copy of it in an appropriate place in your own Confluence space
- Rename it to "<My API name> API architecture review"
- Add a link to it in API register (see page 1204)
- Fill out the page - deleting anything you don't need, such as this tip. When you edit this page there are number of hints and further help.

This template page is to:

- give advice on how to architect an API on the NHS Digital API Platform
- provide a structure for basic meta data about your API that the Platform team will use to populate / manage the platform
- challenge you early on about the consistency of your API with other NHS Digital APIs

CONSIDER time-boxing filling in this page to a max of TWO hours prior to an API architecture review

7.13.9.3.1 Overview

This page is the API architecture review checklist for the above API.

It captures (or provides links to) details of key architectural decisions for the API.

Links might be to documentation on other Confluence pages, or possibly code artefacts, such as the API's OAS file.

As per the API delivery process (see page 194), the decisions are split into two sections:

1. in the discovery phase: **architecture** - big architectural decisions
2. in the alpha phase: **design** - detailed design decisions

7.13.9.3.2 Status codes key

TO DO

IN PROGRESS

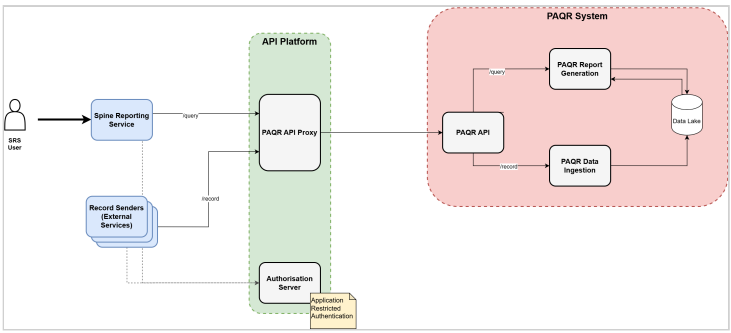
DONE

N/A

7.13.9.3.3 Architecture checklist (discovery phase)

If this API is going to TRG for review, it might make sense to complete an Solution Design Options (SDO) and / or Key Architectural Decision (KAD) document and link out to that.

Activity	Status	Comments / Evidence
Explain the context / problem / opportunity space for your API.	TODO	Sharing Clinical Data between different organisations.

Activity	Status	Comments / Evidence
Consider solution / architecture options to address the problem / opportunity space, and select a preferred option. Everything below this activity is based on the selected option, and assumes the selected option includes an API.	TODO	<provide evidence of your options, for example by completing a copy of our API architecture options template (see page 262) and linking out to that>
Identify the key use cases / functions for your API.	TODO	<complete as appropriate or link out to details elsewhere> This API will allow external systems to: • TBC function 1 • TBC function 2 • etc
Identify the end user - healthcare worker, patient or not present.	TODO	<complete as appropriate or link out to details elsewhere> The end user is a **healthcare worker** / **patient** / **not present**
Determine the end-to-end architecture - how data flows and where processing occurs. Illustrate it using a context diagram.	TODO	<edit the diagram below or link out to details elsewhere>  <if necessary, add a narrative>
If your API routes requests to multiple backends, figure out how it will do this. • See Routing requests to multiple backends (see page 280).	TODO	<incorporate your decisions into the above diagram> See above diagram.

Activity	Status	Comments / Evidence
Determine where the various components of your architecture (including your back-end, if new) will be hosted. • We have some hosting patterns - we can help you with this. https://nhsd-confluence.digital.nhs.uk/display/APM/Transforming+data+on+the+APIM+platform+and+containers[125]	TODO	<provide details, or draw a diagram, or link out to details elsewhere>

125https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsd-confluence.digital.nhs.uk%2Fdisplay%2FAPM%2FTransforming%2Bdata%2Bon%2Bthe%2BAPIM%2Bplatform%2Band%2Bcontainers&data=05%7C01%7Cpaul.fisher18%40nhs.net%7C576a3f84242849e79d3708db14cfca35%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638126654897068216%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=fmfLXuxADYr8tJHeq39tTxppBJAf2B2Ltq8Jpl5omeM%3D&reserved=0

Activity	Status	Comments / Evidence
Decide on the API consumer security pattern. • See API consumer security[126]	TODO	<edit as appropriate or link out to details elsewhere> This API uses the following security pattern(s): <delete as applicable> • application-restricted access - using API key authentication (as per API consumer security - Application restricted- API key authentication[127]) • application-restricted access - using signed JWT authentication (as per API consumer security - Application restricted API- signed JWT authentication[128]) • user-restricted access - using NHS Identity for end user authentication (end user is a healthcare professional) (as per API consumer security- User-restricted-NHS CIS2[129]) • user-restricted access - using NHS Login for end user authentication (end user is a citizen) (as per API consumer security- User-restricted-NHS login[130]) This pattern is appropriate because: • <justify the decision>
Decide how to do backend security. Security for API Backend (see page 778)	TODO	<edit as appropriate or link out to details elsewhere> The API proxy connection to the backend is secured using **TLS-MA** / **<something else>**.
Decide on the integration pattern for the API. • See Integration Pattern Book[131] • Your APIs should be synchronous under the majority of situations	TODO	<edit as appropriate or link out to details elsewhere> This API uses the following integration pattern(s): <delete, amend or add as appropriate> • REST - synchronous **<recommend>** • REST - asynchronous short polling • some form of messaging pattern

126 https://nhsd-confluence.digital.nhs.uk/display/APM/API+consumer+security
127 https://nhsd-confluence.digital.nhs.uk/display/APM/API+consumer+security#APIconsumersecurity-Application-restrictedAPI-APIkeyauthentication
128 https://nhsd-confluence.digital.nhs.uk/display/APM/API+consumer+security#APIconsumersecurity-Application-restrictedAPI-signedJWTauthentication
129 https://nhsd-confluence.digital.nhs.uk/display/APM/API+consumer+security#APIconsumersecurity-User-restricted-NHSCIS2
130 https://nhsd-confluence.digital.nhs.uk/display/APM/API+consumer+security#APIconsumersecurity-User-restricted-NHSlogin
131 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

Activity	Status	Comments / Evidence
Decide whether to use FHIR. • See Is my API a FHIR API? (see page 285). • See API Landscape[132].	TODO	<edit as appropriate or link out to details elsewhere> This API is / is not FHIR-conformant. In particular: • It uses **FHIR R4** • It uses the following FHIR R4 global base resources: **TBC** • It has the following local profiles in FHIR UK Core: **TBC**
If you're not using FHIR but you should be, or if you're using an older version fo FHIR, explain why. You'd better have a damn good reason.	TODO	<complete as appropriate or link out to details elsewhere>
Decide on the payload type(s). • See https://developer.nhs.uk/apis/fhir-policy/serialisation.html. • Consider large payloads - see Large payloads (see page 292).	TODO	<edit as appropriate or link out to details elsewhere> The API payload is **JSON** / **XML**. Will any payload exceed 10MB in size? **YES** / **NO**
Decide the API's service level. These specify SLAs around things like availability and incident fix times. • See Service levels for APIs (see page 288).	TODO	<edit as appropriate or link out to details elsewhere> During private beta, the service level for this API is **BRONZE**. From public beta and beyond, the service level for this API is **PLATINUM**.

132 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Landscape

<table>
<tr><th>Activity</th><th>Status</th><th>Comments / Evidence</th></tr>
<tr><td>Determine what path-to-live environments you'll need to meet your API consumers' testing needs.</td><td>TODO</td><td><edit as appropriate or link out to details elsewhere>
The API includes the following path-to-live environments:
<table>
<tr><th>Environment</th><th>Purpose</th><th>API consumer security pattern(s)</th><th>Implementation notes</th></tr>
<tr><td>Integration test</td><td>Developer initial work
Formal integration testing</td><td>As per production</td><td>TBC which backend environment does this connect to?
TBC how will test data be managed?</td></tr>
<tr><td>Sandbox</td><td>Early developer investigation</td><td>Open access</td><td>Canned responses to a small number of scenarios.
Response fragments shared with API specification.
Deployed to AWS as per API platform standard approach.
Stateless.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Activity</th><th>Status</th><th>Comments / Evidence</th></tr>
<tr><td>Decide the granularity and name(s) for your API(s).
See API naming and grouping endpoints (see page 359)</td><td>TODO</td><td><edit as appropriate or link out to details elsewhere>
The functionality described on this page is grouped into coherent APIs as follows:<table><tr><th>Plain English name (as per the API catalogue[133])</th><th>Physical name (URL)</th><th>Functional scope</th></tr><tr><td>Do A Thing Service - FHIR API</td><td>https://api.service.nhs.uk/do-a-thing/FHIR/R4</td><td>All functions</td></tr></table></td></tr>
<tr><td>Check that your API doesn't overlap with another API owned by another team.
The above naming exercise, and a scan of the API catalogue[134], will help with this.
Silo teams building overlapping APIs has historically been one of our biggest issues.</td><td>TODO</td><td><confirm or discuss as appropriate></td></tr>
</table>

133 https://digital.nhs.uk/developer/api-catalogue
134 https://digital.nhs.uk/developer/api-catalogue

<table>
<tr><th>Activity</th><th>Status</th><th>Comments / Evidence</th></tr>
<tr><td>Estimate the expected volumes for your API.</td><td>TODO</td><td><edit as appropriate or link out to details elsewhere>
We estimate that the API will experience the following volumes:
<table><tr><th>Metric</th><th>Private beta</th><th>Long term</th></tr><tr><td>Number of connected systems</td><td>TBC</td><td>TBC</td></tr><tr><td>Peak load transactions per second (TPS) per connected system</td><td></td><td></td></tr><tr><td>Peak load transactions per second (TPS) overall</td><td></td><td></td></tr><tr><td>Transactions per month per connected system</td><td></td><td></td></tr><tr><td>Transactions per month overall</td><td></td><td></td></tr></table></td></tr>
<tr><td>Do you have PID or sensitive data in your URL Path or Query parameters?
As key pieces of data, such as NHS number, is a key element of any RESTful design, the platform is geared up to have sensitive data in this.
However, this should be kept to a minimum because of systems (e.g. client browsers) that have access to the full values could log such data</td><td>TODO</td><td><discuss as appropriate - there are a number of pages in the Producer Zone, for instance Privacy considerations in your design (see page 406) ></td></tr>
</table>

7.13.9.3.4 Design checklist (alpha phase)

7.13.9.3.4.1 Consistency checklist

The API Management Platform is going to be a key entry point into NHS Digital, and as a consequence will highlight potential inconsistency between different approaches. Some of these consistencies can be hidden behind the Platform façade, others simply won't be cost effective to change. See the page on Consistency[135] for a run down of the wider view of this issue.

The following checklist is designed to help you identify inconsistencies early on.

Primary considerations

For detailed guidance on what is meant in these areas see: Designing your API (see page 322)

Area	Consideration		If No:
API Name	Does it fit into the wider APIM Namespace / domain model?	Y/N	*For guidance on naming see:* API naming and grouping endpoints (see page 359)
Session	Can Apigee manage the session with your API Consumer? Does the API Consumer call other APIs outside of Apigee?	Y/N	This has a number of small complications, so if your existing API has a custom call to start the session - or lots of data is pre-configured during authentication or authorisation this might need some extra design work
Update	PUT is for updating entire FHIR resources PATCH is being used where there are small changes to a large FHIR resource	Y/N	
Atomic operations	Always use small atomic (stateless) requests - batching multiple items should be a last resort. Batching encourages poor design, reduces scalability and increases the 'blast radius' of any failures	Y/N	• Are the requests related by a transaction? This could (would) then still be atomic - but it depends on the size of the batch. • Is there a cost to enabling atomic over batch?

135 https://nhsd-confluence.digital.nhs.uk/display/APM/Consistency

Area	Consideration		If No:
Data minimisation	Do your endpoints return the minimum amount of data possible? (To align with GDPR and other design principles)	Y/N	If not, could you: • Default to a smaller subset, and allow parameter to expand for specific requests? • Could you have different access modes, so the lower security connections receive less data
Testing	Are your current unit and integration tests all automated?	Y/N	
	Can you perform the testing via Apigee and your proxy?	Y/N	This is important to ensure the end to end service is fully tested.

Secondary considerations

These are the next level down of detail, and ideal for exploring in Alpha and Private Beta.

Area	Consideration		Response
Terminology	Are there key terms in your API that don't fit with other NHS Digital services?		
URL Path	Are there terms that might be confusing, stand out from other APIs? Is the path very long? See policy APIM-LAND-05 on API Landscape[136] for details of requirements for FHIR API URL Paths.		
Endpoint names	Use nouns instead of verbs (RESTful pattern) Are they intuitive? For FHIR APIs, endpoint names must align with the FHIR resource model as described at https://www.hl7.org/fhir/http.htmland https://www.hl7.org/fhir/resourcelist.html (the endpoint name must be capitalised) For non-FHIR APIs use spinal-case		*If there are inconsistencies here, mapping of these in Apigee is easy. However, what impacts are there on your existing users?*

136 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Landscape

Area	Consideration		Response
Query Parameters	Are these consistent with other API Platform query parameters? FHIR APIs must be conformant with https://www.hl7.org/fhir/search.html		*If there are inconsistencies here, mapping of these in Apigee is easy. However, what impacts are there on your existing users?*
Field Names	Are the field names in the **payload** datasets the same as FHIR?		*Even for non-FHIR APIs, should developers have to deal with a different field name for basic data such as first name and last name?*
REST Conventions	Does your HTTP Verb usage align with RESTful principles? For FHIR APIs, does this also align with https://www.hl7.org/fhir/http.html		
	Do your HTTP Status Codes align with RESTful principles?		
	Do your FHIR APIs align with guidance found in the FHIR specification for HTTP status codes		*tbc guidance on this side*
Headers	For API Consumers - are they consistent with other API Management Headers?		*You decide as an API Publisher which platform headers are mandatory, see* HTTP headers for your API (see page 395) *for detailed guidance*
	For your API backend (less important as you are managing this)		
Environments	Does the way you use environments align with the API Management platform, and the Path To Live?		See https://confluence.digital.nhs.uk/display/APM/Architecture#Architecture-environments
API Versioning	The private beta hasn't needed to deal with versioning, but we are looking to use accept-headers to support this in the future, though the current NHS Digital policies include the API major version in the URL path.		

Area	Consideration		Response
Error Messages	How you return error messages should be consistent with other interfaces (e.g. FHIR APIs will return OperationOutcome resources to express error detail)		See additional information pages here Handling errors (see page 624) Consistency between APIs (see page 362) RESTful HTTP design points (see page 365)
Test Data	What is your approach to test data for External Developers?	-	
S Flag	Does your API need to consider the Sensitive flag status[137]	Y/N	

Detailed design considerations

The following are very detailed elements about your APIs, but as with the other points above, they can introduce breaking changes into your design. The design pattern in place on the Platform has come out of a combination between the FHIR standard, Apigee platform design, Spine as a key backend across the platform, and input from across the Tribe and NHS Digital.

Area	Consideration		
ETag	FHIR standard mandates the use of the ETag for version checking		See Does my API need to use ETag Headers (see page 1109)
Postman collections	There are other Postman collections across the platform, does yours follow the same conventions ?		
Datetime	Date and time field values should follow the ISO-8601[138] standard, read the details on how to do this: RESTful HTTP design points (see page 365)		

7.13.9.3.5 Governance

This section includes information related to the architecture and design reviews that occur as part of the API delivery process (see page 194).

137 https://digital.nhs.uk/services/national-back-office-for-the-personal-demographics-service/management-of-nhs-numbers-and-pds-records
138 https://en.wikipedia.org/wiki/ISO_8601

7.13.9.3.5.1 API architecture review

Date	TO DO
Attendees	TO DO
Actions	TO DO

7.13.9.3.5.2 API design review

Update this page with the latest, could be important points for the later TRG / LSB governance steps

Date	TO DO
Attendees	TO DO
Actions	TO DO

7.13.9.3.5.3 Post implementation review

Update with any final changes, and flag up any key future work

Date	TO DO
Attendees	TO DO
Actions	TO DO

7.13.9.4 Routing requests to multiple backends

7.13.9.4.1 Overview

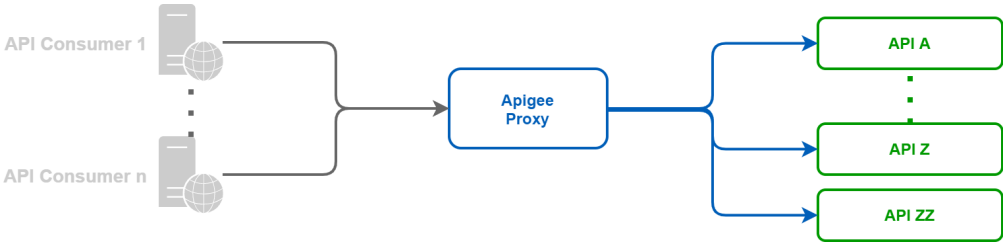
As an API producer, Apigee can deliver the ability to route an incoming request to multiple backends - key scenarios:

- A small number of known targets e.g. separate backends for each major version of your API
- Request by request forwarding, replacing current solutions for GP Connect and UEC which use the SSP to route

The first one is very simply setup in the Apigee flows. Determine what incoming piece of information is going to determine the routing (e.g. version number in the content accept header, or "FHIR" in the URL path) and select a different target server.

The second now has multiple APIs / proxies that use this pattern: GP Connect with the new Patient Facing Services APIs and Bookings and Referrals Standard (BaRS) FHIR API. As it develops, along with other similar programmes, the intent is to support multiple design patterns:

- Forward proxy
- In the EA Pattern book there is "Information Push Pattern", and this has two patterns that could be met by this model:
    - Hub & spoke
    - Messaging Channel
- Spine Secure Proxy (SSP) follows this pattern
- Synchronous HTTP broker model
- Conceptually similar to HTTP Tunnel



When operating at scale, the following key benefits of this model are that:

- API Consumers only have to manage one connection to Apigee
- API Producers only have to manage one connection from Apigee

7.13.9.4.1.1 Spine Secure Proxy (SSP) future direction

As part of the NHS Digital strategy for APIs, all API Consumers should connect to APIM, coming through a standard on-boarding model and connection mechanisms. APIM has set out to provide a forward proxy alternative that does not require a new version of SSP to be implemented by APIM (or another team). The approach is that:

- Each new API (or group of endpoints) will have it's own proxy, so a single (monolith) proxy will not be built
- APIM will create an implementation pattern to do the forward proxy work, so that each proxy can simply re-use this
- This will isolate services, helping to make each one more agile and reduce risks in Production settings
- The current SSP performs functions that are appropriate to de-scope, so does not need to be as complex as previously

To see how this interacts with the current functionality of the SSP see:

- D064 - Spine Secure Proxy - SSP - Strategy[139]
- D042 - Spine Secure Proxy - SSP - and future direction[140]
- GP Connect PFS implementation: GP Connect Patient Facing Services (PFS) API[141]

TODO

Formally ratify this strategy by following up on:

- *the other design decisions in the Platform Design Group, update* D064[142] *&* D042[143]
- *updating this page with the results*

A number of pages moving into the details of what this looks like is here: API Forward Proxy[144]

7.13.9.4.2 Concept design for forward proxy

There are a number design issues that need to be addressed:

- How to look up the destination address
- How to inform the proxy of that address
- If there is an authorisation model then apply checks
- Perform the routing action
- Alongside the runtime processing, this needs to be managed:

139 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/597341343/D064+-+Spine+Secure+Proxy+-+SSP+-+Strategy
140 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/188195260/D042+-+Spine+Secure+Proxy+-+SSP+-+and+future+direction
141 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/358582937/GP+Connect+Patient+Facing+Services+PFS+API
142 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/597341343/D064+-+Spine+Secure+Proxy+-+SSP+-+Strategy
143 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/188195260/D042+-+Spine+Secure+Proxy+-+SSP+-+and+future+direction
144 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/488440182/API+Forward+Proxy

- Automate the on-boarding and management of API backend connection
- TLS MA is harder to manage that Private Key JWT, but easier to use for a connection
- Store the address in a way that Apigee can use at run time

The initial use case for 999 to 111 triage could look like this (see UEC 9s to 1s API (see page 280)):

draw.io

Diagram attachment access error: cannot display diagram

Proposed solution elements:

Element	Solution	Implementation
Remove the need to look up an address	Looking to make integration easier, the proxy should look up the address rather than requiring the API Consumer to do a second call.	For small numbers of addresses, or simple use cases, then an Apigee KVM can be pre-loaded from GitHub with a mapping. If an external service is provided, then a Service Callout can be used, which should, ideally cache the lookup result if the Proxy is a high volume one.
What data is needed, instead of an address?	ODS code should form the core aspect, however, there are many situations where this is not enough information and the ASID code is needed. The strategy is to hide the ASIDs and associated complexities - this type of use case is the likely the largest hurdle to achieving that.	tbc

Element	Solution	Implementation
How to pass the data, in the request, to lookup the address?	There is not a standard model for this passing in the destination address, the first use case should explore options with the API Consumers.	• Payload - which might work well for FHIR, as some STU3 FHIR payloads already have this. However, it conflates routing and payload data leading to confusion and complications further downstream • Custom header - likely the best option with a JSON payload • Unique identifier in URL - could interfere with FHIR specification • Include destination URL as a parameter for the request - this is the mechanism that SSP use, but means you can't just make a call to an API • It might be possible to re-use the HTTP Tunnel / CONNECT concept in this context
Authorisation model	Some solutions will require a check that a specific consumer can (or cannot) connect to certain API backend.	For simple use cases, then an Apigee KVM can be loaded from GitHub. If an external service is provided then a Service Callout can be used, which should, ideally cache the lookup if the Proxy is a high volume For complex scenarios an APIM AWS Hosted Container should be considered, but this might then move outside the definition of a "lightweight" translation layer
Perform the routing action	With the address known, the Apigee flow can specify and Target Server	
Auditing transactions	90 day logs for all proxies are available in NHS D Splunk	There are analytics logs with data retention of years. If the requirements are simple these logs might meet this need - but, key items such as IP addresses are not stored. If a solution requires APIM to audit this, then an addition is required to the NHS D Splunk implementation to accommodate.

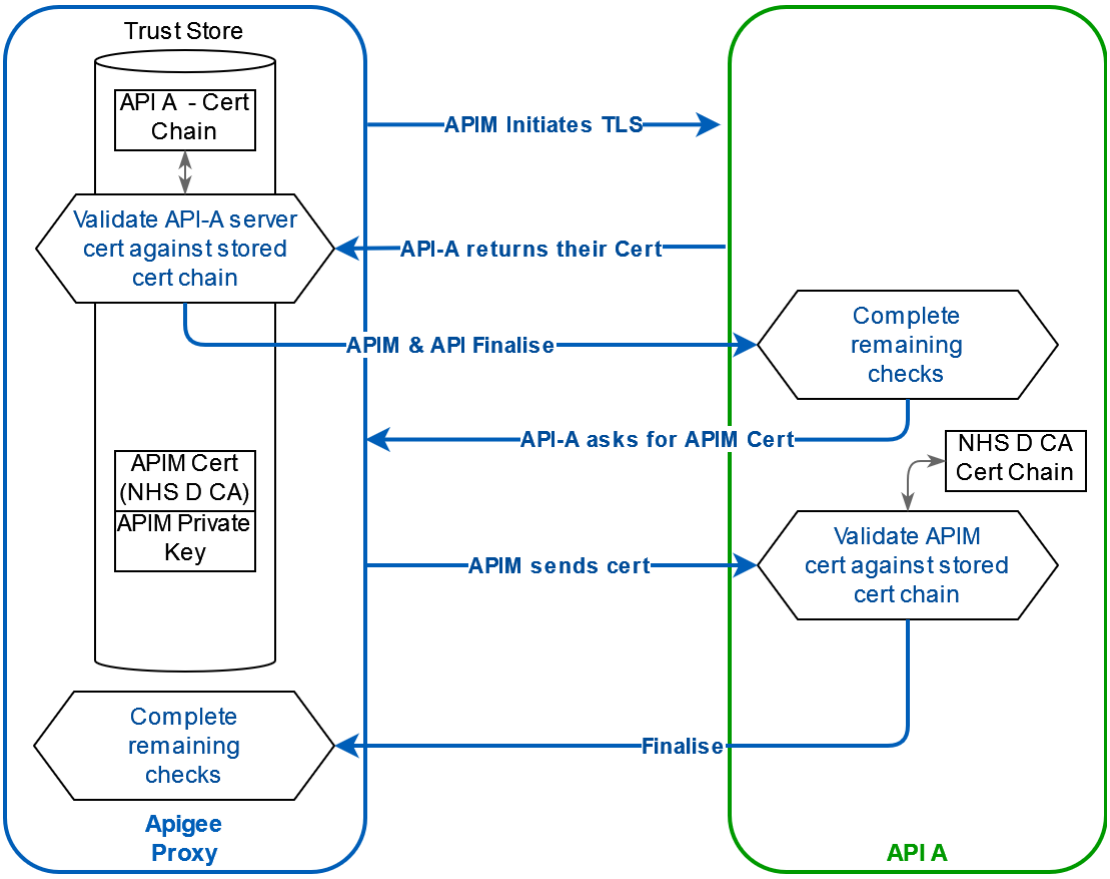
Element	Solution	Implementation
Managing API backends	This would be APIM as a platform providing this, but we would work with the first API Producer to do an MVP	APIM Considerations: • Priority is to provide a format / structure in GitHub to store API Backed meta data • Add to CI/CD a way to update Apigee based on this • Investigate OAuth and Private Key JWT integrations (can be done, but target servers do not have native support) • Longer term add in ability for automated on-boarding of API Producers including and managing their own connections

7.13.9.4.3 Securing the connection between Apigee and the backend

TLS MA will be used initially as it is well supported and secure. In the longer term, the solution should move to using OAuth, but in the mean time the solution will be to replicate the existing Spine PKI based model:

- The backend owner (receiver / producer) will obtain a standard endpoint certificate from the DIR team (or the use of the EndPoint Registration service):
    - There will be one certificate per installation (reflecting the API consumer deployment topologies)
    - An aggregator (tenanted) / service provider will have one deployment for all the receivers they support
    - GP Connect currently incorporates the ODS code of the receiver into the FQDN, the proposal is to continue this
        - This has **no** direct impact on authorisation / configuration - but is there to make it easier to manage
        - For BaRS (and other forward proxy pattern), for aggregators the Supplier ODS code will be used
        - The FQDN will be a sub-domain under the receivers control to route traffic to their endpoint
    - APIM will already have the NHS / Spine PKI cert chain - and will be common to all receivers (**is that correct?**)
    - SHOULD this pattern remain, or should we move to using Public CAs?
- There will be a common client certificate created to identify that the source of the request is API Management. This will be a single cert issued by the NHS D CA (DIR team), for the BaRS team:
    - The receiver API team will download and store (if they haven't already) the NHS D CA chain
    - The FQDN for this certificate will be virtual, and will not resolve

Summary of the mutual TLS sequence:



APIM will create the NHS D Cert by:

- APIM create a private key for each environment (and associated FQDN) and CSR (Certificate Signing Request)
    - Note that the FQDN is "virtual" it is never called in this scenario
- CSR sent to DIR team to issue certificate
- APIM upload key and certificate into Apigee

7.13.9.5 Is my API a FHIR API?

7.13.9.5.1 Overview

This is one of the most common questions early on in the design process.

HL7 FHIR (Fast Healthcare Interoperability Resources) is an international standard which sets out how healthcare data can be shared between organisations in an way which promotes system-to-system interoperability. The key element of the FHIR standard is the resource model which describes familiar data entities which are encountered in a healthcare context, such as patients, practitioners, organisations, medications and care plans. See https://www.hl7.org/fhir/resourcelist.html.

The current version of FHIR is R4.

Different countries are expected to hone this data model to make it more specific to their own setting. NHS Digital has carried out this work in collaboration with HL7.uk[145]. The result of this is FHIR UK Core[146], which is the standard data model for exchange of healthcare resources in the UK.

7.13.9.5.2 How do I decide if I need to use FHIR

Considering the following questions will help you decide, the first question being the most important:

- **Are my data entities found in the FHIR model?**
- Do the FHIR entities need a lot of extension to fit my use case?
- How does my API fit into a wider business process?

Most of the time your answer will be "My API is a FHIR API" because most NHS API are used to convey healthcare domain information between systems. This is acknowledged in the following NHS Digital architectural notice: API Landscape policy[147] which states:

> New NHS Digital APIs **MUST** be based on HL7 FHIR R4 and FHIR UK Core where data conveyed through the API can reasonably be expressed using the FHIR R4 and, if necessary, FHIR UK Core.

7.13.9.5.2.1 Are my data entities found in the FHIR model

1. List out the data entities which are involved in your API. Think about not just the entities in your MVP, but how you expect your API to be used in future.
    - E.g. "Patient", "Employee", "Medication", "Salary Scale", "Album"
2. Compare this list with the https://www.hl7.org/fhir/resourcelist.html. While you are looking to match up your entities with those in the FHIR data model, think about synonyms. E.g. "drug" is expressed in FHIR as Medication.

If you can you find all of the data entities which will be conveyed in your API within the FHIR resource model then this is a strong indication that your API must be implemented using the FHIR standard.

For a more in-depth analysis, the FHIR resource model gives a further indication. Each of the FHIR resources is associated with a "maturity level[148]". If you find that all your data entities are found in the FHIR resource model, but that the maturity levels of these are very low, this may indicate that the case for a FHIR API is not as strong.

Some example use cases are given below to help you get an idea of the process.

Use case: my API is FHIR (data entities clearly match)

- I have the following data entities:

145 http://HL7.uk
146 https://digital.nhs.uk/services/fhir-uk-core
147 https://aalto.digital.nhs.uk/#/document/viewer/33c81219-d1db-4a3b-a63d-c79d8ef5e055?library=5464c07f-daf1-4eee-b9b6-22e6c4dfbbd0
148 https://www.hl7.org/fhir/versions.html#maturity

- Patient
- Organisation
- Practitioner
- Medication

All of these entities are in the FHIR model with a high maturity level. The API should be implemented as FHIR.

Use case: my API is not FHIR (data entities clearly *don't* match)

- I have the following data entities:
  - Album
  - Genre
  - Artist
  - Label

None of these entities are in the FHIR resource model - so the API should not use the FHIR standard.

Use case: my API is not FHIR (FHIR data entities are not mature, or don't match)

- I have the following data entities
  - Invoice
  - Charge Item
  - Organisation
  - Person
  - Address

Most of these entities are found in the FHIR resource model, but they all have low maturity levels (0 or 1). The address entity is not found in the FHIR resource model, but only exists as a FHIR data type used in entities such as "Organization".

Is it not clear from this analysis that this API must use FHIR.

7.13.9.5.2.2 Do the FHIR entities need a lot of extension to fit my use case

It is possible that you can data entities which match, but the specialised nature of your data is such that these entities, though present in the FHIR model, would need to be changed alot in order to work for you. For example, you may have mostly domain specific data attributes which are important for your API, but which are not found in the corresponding FHIR data entities. FHIR deals with these scenarios through the use of "extension", and UK Core has a list of approved extensions at https://simplifier.net/guide/UKCoreDevelopment/ExtensionLibrary

Sometimes, if it looks like you will need a lot of extensions, this is a sign that you have chosen the wrong resource, so it may be worth considering whether a better fit from the FHIR resource model is available.

If your conclusion is that, though the data entities are present in the FHIR, they would need radical change, this would be an indication that the case for FHIR is not as strong.

7.13.9.5.2.3 How does my API fit into a wider business process

Usually, your API will be used by consumers in the context of a business process which will be supported through the use of multiple system interactions. Does your API fit into such a scenario where the other API calls required to complete the end-to-end business process are based on FHIR? NHS Digital policy is to promote an API ecosystem which is easy to navigate - where there is a single standard data model (FHIR) which is used across many APIs.

7.13.9.5.3 What if my API is clearly not a FHIR API

In this case, NHS Digital still promotes the use of standards to convey data through APIs. API designers are encouraged to make use of existing data standards (other than FHIR) which would be most appropriate for their use case.

7.13.9.5.4 What if I'm still not sure

The API architecture review[149], part of your discovery phase, will cover the question of whether your API is a FHIR API. Representatives from APIM and IOPS there to guide you to make this call before you commit to more detailed API design.

7.13.9.5.5 Who are IOPS

IOPS is the **old name** for the Interoperability Standards team. They sit within the Technology Strategy, Architecture and Standards (TSAS) directorate and are responsible for:

- Building and maintaining FHIR UK Core
- Supporting teams building FHIR-based integrations, including
    - FHIR APIs
    - FHIR message integrations

If you are building a FHIR API, you can ask the Interoperability Standards team to help. If you need a significant amount of help, they might ask you to raise an NWR to cover their costs.

Note that they are more focussed on FHIR data modelling than RESTful APIs design - so they would likely help you design the **payload** of your FHIR APIs rather than the **endpoints**.

To contact the team, email interoperabilityteam@nhs.net[150].

7.13.9.6 Service levels for APIs

Within NHS Digital, there are number of defined service levels (bronze, silver, gold, platinum). The following document details what Service Level Agreements (SLAs) are recognised:

149 https://nhsd-confluence.digital.nhs.uk/display/APM/Governance#Governance-APIarchitecturereview(Discovery)
150 mailto:interoperabilityteam@nhs.net

You need to decide which service level is appropriate for your API. Choose the lowest service level that is "good enough" - higher service levels cost more to run.

Note that:

- the API platform itself is a PLATINUM service, so can support APIs at any level
- NHS CIS2 (which the API platform uses for healthcare worker authentication/authorisation) is also a PLATINUM service
- NHS login (which the API platform uses for patient authentication/authorisation) is TBC

7.13.9.7 API architecture review

An API architecture review is a process to evaluate the design and structure of your API to ensure it meets the desired requirements and is well-suited for its intended use.

It involves a thorough examination of various aspects of the API, to do this, you need to complete our API architecture review checklist. (see page 266)

During an API architecture review, you will examine your API's design using the questions outlined in our API architecture review checklist. To ensure a productive meeting, it is recommended spending no more than two hours preparing for the review using the template.

By conducting a thorough review, you can identify areas for improvement and ensure that your API meets our high standards.

Suggested Meeting Attendees:

- Your API producer team
    - Product Owner
    - Architect / Tech lead
- API platform team
    - IOPS rep: contact england.interoperabilityteam@nhs.net[151]

151 mailto:england.interoperabilityteam@nhs.net

Duration: between 30 minutes and an hour.

7.13.9.8 Wider architecture review

7.13.9.8.1 Overview

The Technology Policy and Architecture function has a focus on keeping wider governance faster and lightweight, by focusing on:

- Early engagement
- Logging decisions
- Updating Aalto with the solution model
- Only going to governance boards (e.g. Technical Review Group - TRG), where there is divergence from standard patterns

The **API Producer** squad is responsible for this work, and the APIM Tribe will support you in this with things like example documents, define the scope of the work - i.e. what is responsibility of the API Platform vs. what your project / programme is delivering. There are multiple governance boards, but Architects will focus on the these boards: TPA - Governance Boards Overview[152].

The vast majority of API Producers will be creating APIs that require a new or updated Solution Design - and possibly an appearance at TRG.

Once you've undertaken the API Architecture review - it is an ideal time to **start the governance process - set out here: https://architecture.digital.nhs.uk/**:

- Find out who your lead architect is (Aubyn will help here)
- Are you adding to an existing solution?
    - No - create a new Solution Design Overview (SDO)
    - Yes - then create a KAD (Key Architectural Decision) which you will attach to your SDO
- Will your API backend be on AWS or Azure?
    - If you don't have access to AWS or Azure then you will need to submit a Cloud Hosting Request (CHR) to TRG
    - The CHR allows the AWS or Azure teams to provision you a landing zone account, giving you a dedicated account just for your API Producer team
    - To create a CHR you have to have an SDO in the system (although it doesn't have to be approved if you are in Discovery or Alpha phases)

7.13.9.8.2 Creating a Solution Design Overview (SDO)

If you need to create a Solution Design Overview, there are some key points to follow / keep in mind:

- You will **not** know all the answers at this stage, but the questions are broad and generally easy to answer - go for worst case scenarios

152 https://hscic365.sharepoint.com/architecture/Pages/Governance/Governance%20Top%20Level%20Page/Governance-Home.aspx

- The hardest part is that it includes two key assessments that help to determine whether the project will be high enough profile to warrant extra scrutiny or wider governance :
    - Risk Classification
    - Prioritisation model
- If you don't have any other Architecture pages - simply link to the one you've done for APIM
- Before starting your SDO you must create your solution in Aalto[153]:
    - You need to create a Layer Model
    - You need to have at least one Application Component
        - If you have more than one you will need to decide which one the SDO is held against
    - Once the Application Component is created, it is sent to the TRG Apps overnight.
- Follow the process set out here: https://architecture.digital.nhs.uk/trg
    - The TRG process uses Microsoft Power Apps to manage this
    - Creating your SDO involves selecting an Application / Service - and that list is populated (overnight) from Aalto Application Components
    - Aalto training material is here: NHS Digital Sharepoint[154]
- When you submit your SDO, you will discuss with your Lead Architect on whether you need to present to TRG or not

This does tend to be an Architect doing this work, but Senior Tech Leads or Product Owners can do this as well (*there be dragons...* 😉):

- If you do need to go to TRG:
    - You will have already created an SDO and/or KAD
    - For the SDO / KAD it is recommended you have a page on Confluence with the details and reasoning behind your solution
    - Creating a power point is useful (but not essential). The PPT is there to enable you to talk TRG through the situation at a high level, not delve into the nitty gritty
    - Example PPT: TRG Ambulance Analytics API-v0.2.pptx[155]

Any work with Live Services Board will require support from the Lead Architect.

Talk to Aubyn Crawford for more details about the above. As you move through the project stages and near service transition (see page 947) into Private Beta (and later full general availability) there are likely questions that will be asked by compliance teams (see page 951).

---

153 https://aalto.digital.nhs.uk/

154 https://hscic365.sharepoint.com/architecture/Pages/Tooling/Tooling%20Content/Aalto-Guidance-Training.aspx

155 https://nhsd-confluence.digital.nhs.uk/download/attachments/97747076/TRG%20Ambulance%20Analytics%20API-v0.2.pptx?api=v2&modificationDate=1604073860815&version=1

7.13.9.9 Large payloads

7.13.9.9.1 Overview

Our API platform, which is based on Apigee Edge, has a hard limit of 10MB on payload size.

If you need to handle payloads larger than 10MB, the recommended pattern is to use durable storage (e.g. S3) for the files, and just pass pointers to the files via APIM. E-RS is using this pattern, and also MESH (although that doesn't use Apigee).

For more details on this approach, see https://nhsd-confluence.digital.nhs.uk/display/APM/20220209+-+WDR+Meeting+Minutes[156].

For information about all of our patterns, see https://digital.nhs.uk/developer/nhs-digital-architecture/integration-patterns-book

7.13.10 Metrics for success / KPIs for APIs

Key Performance Indicators (KPIs) for API Success

Understanding how successful your API is can be a challenging task to quantify but by using the appropriate indicators can help you validate the success of your API and

present a compelling story to your stakeholders and consumers, as well as helping developers understand where additional benefits could be achieved.

To measure the success of your API, consider tracking these key performance indicators (KPIs):

Usage Metrics

- API Calls: Total number of API calls per unit of time (e.g., per day, per month)
- Unique Users: Number of distinct users or applications accessing the API
- Most Used Endpoints: Identification of frequently accessed API endpoints
- Integration Success Rate: Percentage of successful integrations with other systems

Performance Metrics

- Response Time: Average time taken for the API to respond to a request
- Latency: Time taken for data to travel between the client and server
- Throughput: Number of requests the API can handle per unit of time
- Error Rates: Percentage of API requests resulting in errors

Business Metrics

- Customer Satisfaction: Feedback from API consumers on their experience

156 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsd-confluence.digital.nhs.uk%2Fdisplay%2FAPM%2F20220209%2B-%2BWDR%2BMeeting%2BMinutes&data=05%7C01%7Ctony.heap1%40nhs.net%7C9786748697ea466efd6808dacb93b237%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638046132447655262%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=Iwi8O%2Bgj3XfN6lPvuvSvd16BbvxeXT5D2kStLHWKoTY%3D&reserved=0

- Time to Market: Time taken to develop and launch new API features or versions
- API Adoption Rate: How quickly new users or applications are adopting the API

Security Metrics

- Security Breaches: Number of security incidents or breaches
- Unauthorised Access: Instances of unauthorised access to the API
- Data Exfiltration: Attempts to steal or copy data from the API

Developer Experience Metrics

- Documentation Usage: Usage of API documentation and support resources
- Developer Community Engagement: Participation in developer forums and communities
- Time to First API Call: Time it takes for developers to make their first successful API call

KPIs for health outcomes

- Number of unique end users using the API
    - Healthcare workers
    - Citizens
- Availability/uptime (as a percentage)
- Number of live incidents (as low as possible)

By tracking these KPIs, you can gain valuable insights into your API's performance, identify areas for improvement, and make data-driven decisions to optimise its success

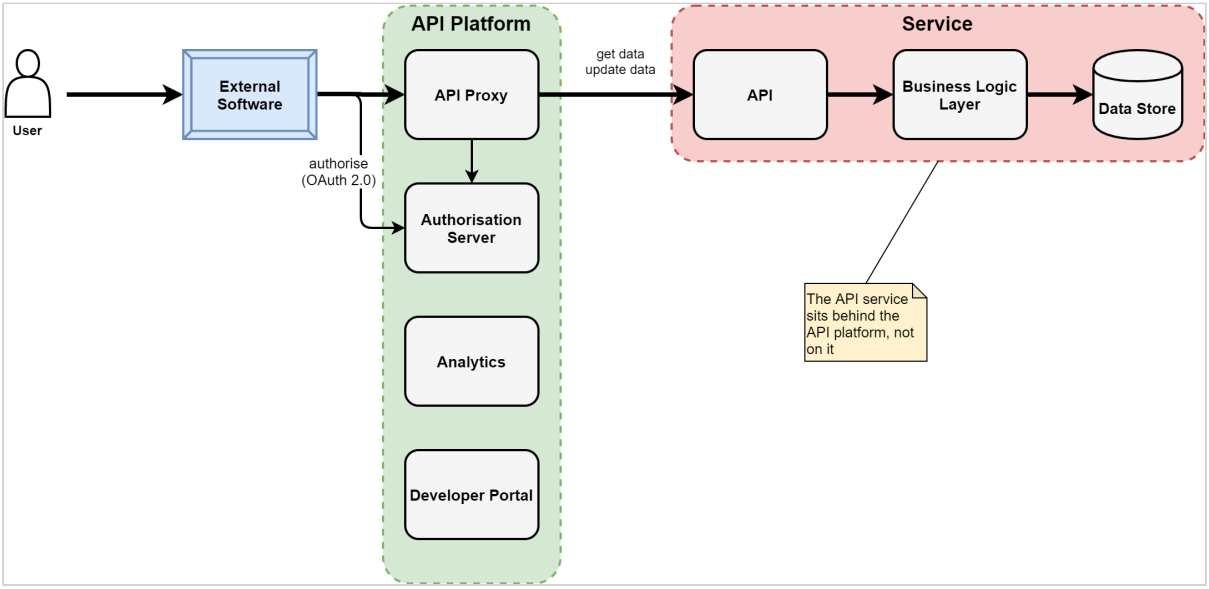
and build a comprehensive story around the benefits of your API.

7.13.11 API delivery and support options

7.13.11.1 Overview

This page explains your options for delivering an API and for longer term support. The two are related - the best approach is a "devops" approach - the same team delivers and runs the service.

There's a common misconception about APIs and the API platform. This diagram helps to explain:



The key point is that when we talk about APIs, we are usually actually talking about **services** that have business logic and a data store. The service exposes an API, and the API platform is the "front door" for that API - it provides security, analytics, a developer portal and more.

You do not build an API **on** the API platform - you build a service **behind** the API platform.

The API Management team can help you get your API service connected to the API platform. We can even help you design the shape of the API for your service. But we can't **own** your API service long term.

7.13.11.1.1 Intermediary APIs

Some APIs don't have a back-end service owned by NHS Digital - they are intermediary APIs, orchestrating API requests from multiple API consumers to multiple API providers.

Examples of this are:

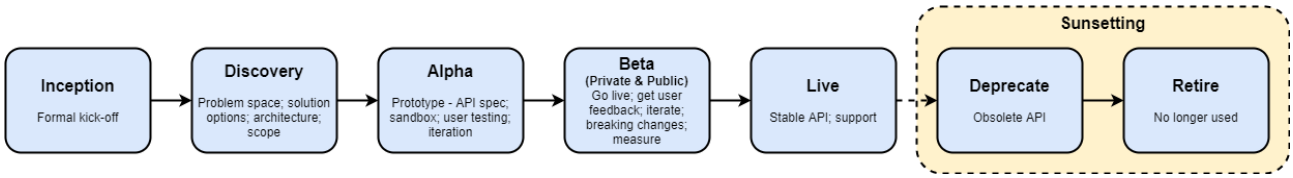
- the GP Connect APIs
- the Booking and Referral API

Often, the teams delivering these APIs don't have their own engineering team. Historically they have used the Spine Secure Proxy (SSP).

For intermediary APIs, API Management **are** able to build and run the API if required. Please speak to us to discuss your options.

7.13.11.2 API delivery process

Whoever delivers your API, it will always be delivered using our API delivery process as explained on Deliver your API: (see page 194)



7.13.11.3 API delivery options

If you want to deliver an API on the NHS Digital API Platform, there are a few options.

7.13.11.3.1 Option 1: Do it all yourself

The API Platform is designed to be self-service. If you have a delivery team you can deliver the API yourself, from start to finish, with support from us.

Pros:

- No waiting for us to have capacity
- You know your service best

Cons:

- You might not have API Platform expertise in your team

This is our preferred model, as it scales well.

7.13.11.3.2 Option 2: Ask us to build it

Within the API Management team we have an API Factory, a couple of agile squads who know how to deliver APIs. Subject to capacity (as ever), we can build the API for you and them hand it back to you to run once we've done it. We don't like this model, but it is an option

Pros:

- You don't need a team, although you will need to find a team to own the API longer term

Cons:

- We have limited capacity
- We don't know your service domain well
- There is a cost associated

7.13.11.3.3 Option 3: Other hybrid / blended approach

In this option we work together to help you build your API.

For example, we might pair with you initially to accelerate getting you started.

7.13.11.3.4 Option 4: We build and run it (intermediary APIs only)

As mentioned above, we are able to build and run intermediary APIs (see page 294).

7.13.11.3.5 Option 5: Ask us to do the discovery

This is a specific variant of the hybrid approach - if you don't have the right skills to do the discovery phase, maybe we could do that for you, and then hand delivery back to you afterwards i.e. for the actual build through alpha and beta.

7.13.11.4 Long term support options

API Management is a platform team. Whilst we can help you deliver your API, we cannot support it in the longer term (unless it is an intermediary API - see above).

You need to work out who will run your API long term.

In general, ownership of an API should reside with the team that runs the back-end service that sits behind it - so that the service team has autonomy over their whole service, including the API.

If you need help figuring out your long term support approach, get in touch.

7.13.11.5 API delivery costs and timescales

Delivering a good API costs money and takes time. It's not just about the software development work to build an API proxy - that's actually pretty quick and easy. And it's not about the cost of using the API platform - that's free and it actually saves you money.

It's the "other stuff" that takes time, such as:

- discovery - working out the problem you're trying to solve
- writing really good API documentation
- providing an open access sandbox
- providing an integration test environment
- designing and delivering an onboarding process for API consumers
- engaging with potential API consumers, getting their feedback and iterating your tech, docs and onboarding process
- getting approval to put the API live - including clinical safety, information governance, security etc
- actually onboarding API consumers
- comms and marketing to encourage adoption

Here are some finger-in-air costs and timescales to help you with project budgeting:

API complexity	**Meaning**	**Using API platform**		**Not using API platform***	
		Budgeting cost	**Elapsed time****	**Budgeting cost**	**Elapsed time****
Low	API with a single read-only endpoint Low security needs e.g. no personal or sensitive data	£200,000	3 months	£400,000	5 months
Medium	API with five read-only endpoints Medium security needs e.g. personal or sensitive data	£400,000	6 months	£600,000	8 months

API complexity	Meaning	Using API platform		Not using API platform*	
		Budgeting cost	Elapsed time**	Budgeting cost	Elapsed time**
High	API with update endpoints High security needs e.g. updating personal or sensitive data	£800,000	12 months	£1,000,000	14 months

***** This accounts for all the extra work you'll have to do that the API platform gives you "for free" such as logging, monitoring, alerting, threat detection, API consumer access management and more.

****** This is the elapsed time from starting discovery to getting your API into production in beta.

7.13.11.5.1 Running and maintenance costs

The costs above are costs to **deliver** an API. You also need to factor in running and maintenance costs. Actually, if you are running and maintaining a back-end application, the overhead cost of running and maintaining the API are very low - they very rarely go wrong, but you do need engineers in your team to be able to maintain it.

7.13.12 Different types of API

7.13.12.1 Overview

This page explains the different types of APIs and how the lifecycle differs for each.

7.13.12.2 Central API services

These are where we have a "back end" service of our own that the API connects to.

For example:

- PDS
- EPS
- e-RS

For more details, see https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/integration-and-apis#api-services.

These follow our normal API status model: see https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#api-status.

7.13.12.3 Peer to peer APIs (aka API standards)

This is where third party systems talk directly to one another, and all we do is publish an API standard / spec that tells them what shape the APIs. There is no central back-end, and we do not provide any tech for proxying our routing traffic.

For example:

- Eyecare e-Referral Service

For more details, see https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/integration-and-apis#peer-to-peer-apis

The status model has similar names to API service statuses but the meanings are different:

API standard status	Meaning
Alpha	Spec written and published - feedback loop in progress. We might supply a sandbox or other test harness. We might not. Subject to significant breaking changes.
Private beta	Working with a small number (sometimes just one) of early adopters to implement the spec. More stable than alpha but still subject to breaking changes.
Public beta	Spec has been used successfully by early adopters, who are now fully integrated and running in live. Spec now open for use by all comers. More stable than private beta but still subject to minimal breaking changes.
Stable	API spec has been baselined and there will be no more breaking changes. Any breaking changes would be issued as a new major version, which is subject to its own status lifecycle.

7.13.12.4 Intermediary API services

This is where we provide some tech to route requests (sync) or messages (async) from a sender to a receiver. Sometimes called a broker.

For example:

- GP Connect

- BaRS
- Anything that runs over MESH

For more details, see https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/integration-and-apis#api-services.

These follow either an API service stats model or an API standard status model:

- If the interaction is via an unchanging general broker, such as MESH, it's more like an API standrad for the message payload
- If the interaction is via a strongly-defined sync API, then it's more like an API service

7.13.12.5 Interoperability standards

Some of the things in our API catalogue are more general interop standards - they don't define the shape of a specific API interaction, they are "building blocks" for building APIs services or API standards.

For example:

- Care Connect
- FHIR UK Core

These might follow a different status model.

For example, FHIR UK Core follows the HL7 status model:

HL7 status
Draft Standard for Trial User (DSTU)
Standard for Trial Use (STU)
Normative

7.13.13 Access modes - compliance - security and simplifying running your API

7.13.13.1 Overview

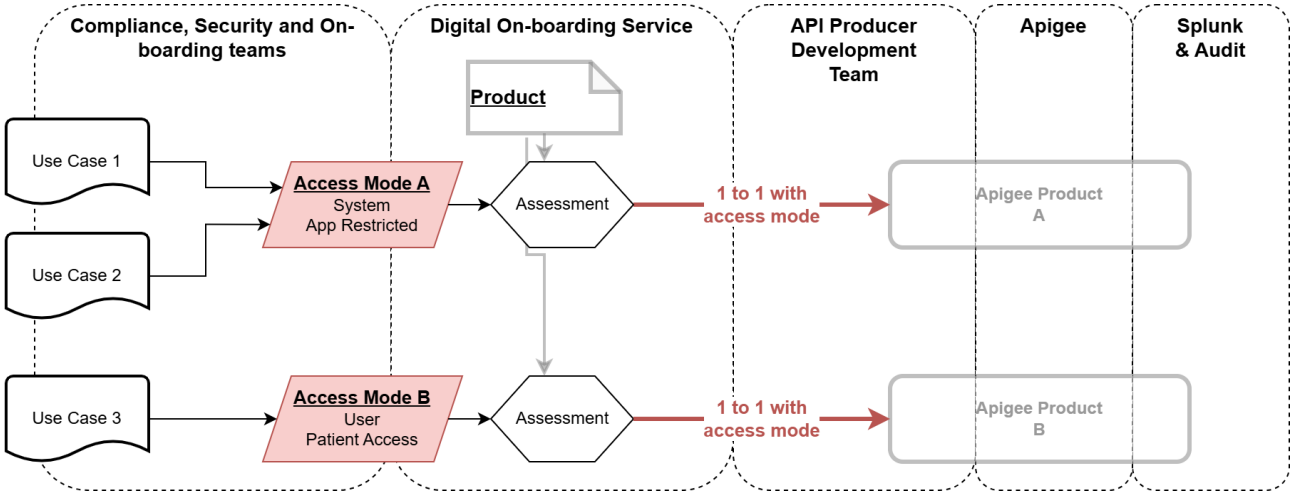
Discovery is too early to get into details around how you are going to manage the running of your API. The guidance on this page is to help you embed some concepts early to keep things simple:

Principles

- Define a few, simple, and generic **use cases** for your service

        - Assign **access modes** to the use cases
    - Use the term "access mode" during:
        - Onboarding API Producers
        - Service readiness processes, including security and compliance
        - Your technical implementation
        - Monitoring and auditing your service
        - Dashboards whilst running your service
    - Minimise the data sensitivity, which will allow more groups to use your service
    - Involve your Information Asset Owner (IAO); Privacy, Transparency and Ethics group; security early on
        - Does your governance team require visibility of End User Organisation (EUOs)?

    By following those principles you will not only reduce the amount of time spent working with various governance groups - but you leverage how Digital Onboarding Service and API Management Platform operate, giving you high confidence on how your API Consumers are continuing to comply with your compliance requirements



Note - a more demonstrative split might be, EPS, who split their API access / compliance into:

- Dispensing
- Prescribing
- Tracking

Spine considerations:

- Create one message set (see page 408)per "access mode", this will make requesting ASIDs and instructions around ASIDs simpler

7.13.13.1.1 Access modes

Access modes are designed to help simplify assurance, and bridge the gap between:

- Onboarding teams and technical teams
- Technical authorisation rules and what it means for data security

The most common access modes are:

- application-restricted
- healthcare worker
- patient

When developing your service, you need to balance the number of access modes. Some example consequences:

- Too many - it then gets confusing, e.g. 20 is way too many
- Too few - in a data breach situation, it is unclear to the non-technical teams what access a specific API consumer has

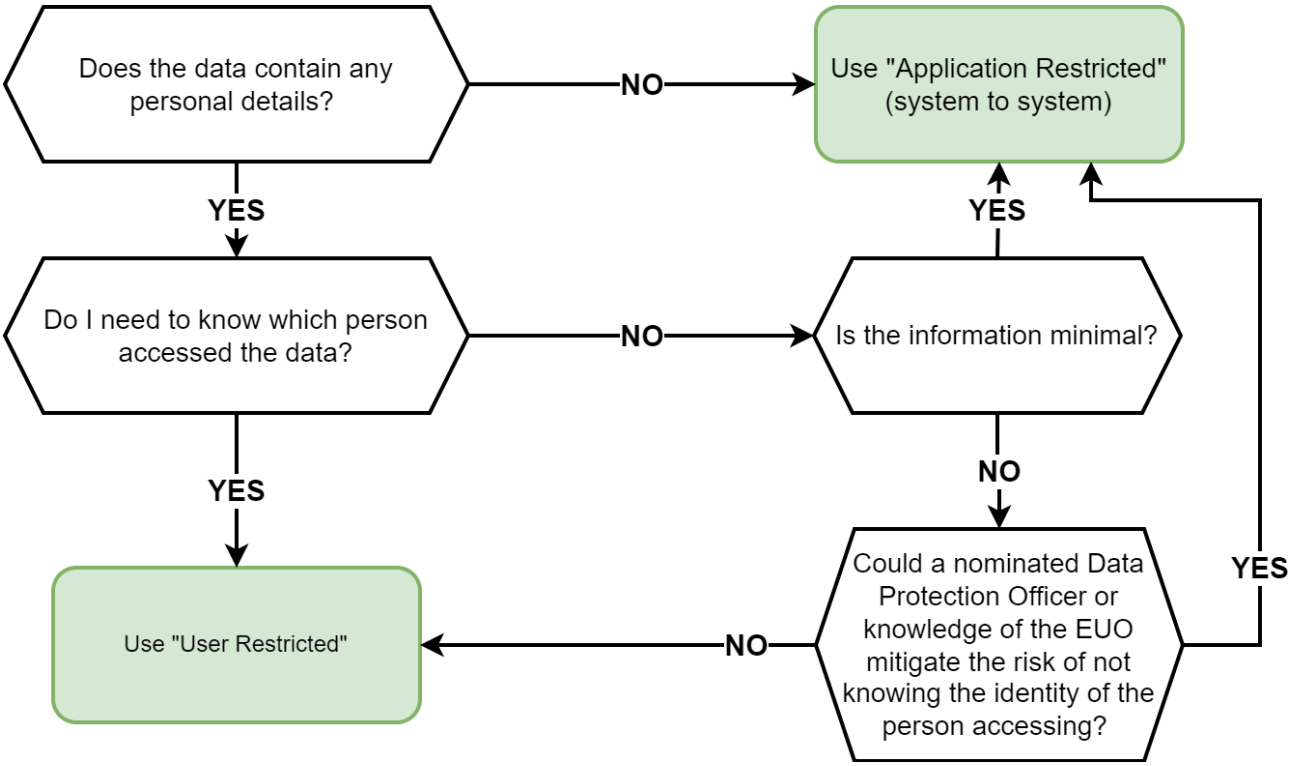
The access modes are listed in the Security and Authorisation section of your API Catalogue entry (OAS file).

7.13.13.2 Approach

Keeping the principles in mind, you need to work through and define the use cases, such as:

Use Case	**Actor**
Using PDS for data integrity checks	System
Doctor signing a prescription	Healthcare worker

You can then determine the type of access modes, based on the data **returned** (or type of data to be changed):



For a more detailed set of guidance see: Guidance for making API security decisions (see page 437).

7.13.13.3 Implications

If you choose a specific **access mode** there are some implications that should be aligned to:

Application Restricted	The API Consumer (or Connecting Party) will always be known / traceable There will be no record if a person was involved.

I want a record of which End User Organisations (EUO) accessed the data or service	• Application restricted and ask connecting parties to supply the EUO with each request
	• For services that have healthcare worker access, you could also choose record the EUO:
	  • The preferred model is to use the NHSD-End-User-Organisation-ODS header in your API
	  • The CIS2 identity, used in healthcare worker mode, allows the person to select their current role. You can derive the EUO from this, but has a known set of caveats which some governance teams are not comfortable with

Audit of users connecting	Currently, from an onboarding / legal perspective, there is a focus on the API Consumer keeping audit data about which users accessed. This can make integration harder for some suppliers. • There is possibility (if in demand from API Consumers & agreed) that APIM can take on that audit responsibility

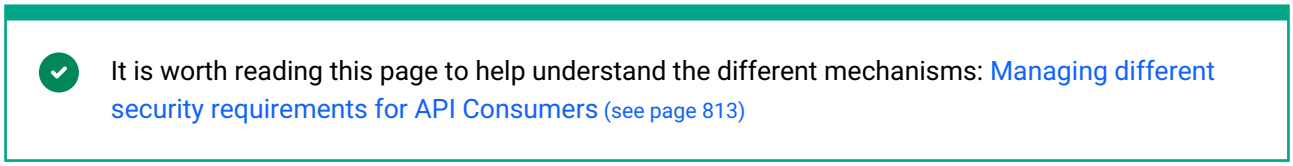
7.13.13.3.1 More complex authorisation

There will be some APIs where access modes do not give enough control over API consumers:

- An APIM design principle is that coarse grained (simple logic / high level) authorisation should be applied in Apigee
  - This aligned with the access modes concept
  - Access modes and their assurance should be managed in the Digital Onboarding Service (DOS)
- Fine grained authorisation should be applied in your backend
  - This might be permissions down to user level (e.g. cleaner can't sign a prescription)
  - But, you (as the API Producer) need to manage this authorisation data. APIM has not implemented a full authorisation data model, although there are some options being considered that might make it easier for API Producers to manage this with the API Consumer (who should ideally be able to self-serve this)
  - You need to:
    - Have somewhere to store this data, and for not very complex scenarios:
      - APIM has a shared flow for this: Configuring extended attributes for API Consumer Apps (see page 806)

    - In the future, a UI will be added to the Digital Onboarding Service that will allow API Producers and API Consumers to mange key value pairs
- Have a mechanism to transmit key API Consumer data to your backend so that your backend can apply the relevant logic - see Passing information to your API backend (see page 611)

Note that you might also have API specific configuration values, not just authorisation based ones.



7.13.13.3.1.1 Thoughts to put down:

- Involve your Data Protection Officer or IAO - the technical view of what is sensitive or not is not always aligned
    - Even if Application Restricted is chosen, the Connecting Party is **still** under a contractual obligation to protect any personal data or secure access to the service
- Looking at a specific API Consumer, who has been granted a certain level of access - the various teams will be able to infer:
    - How sensitive the data is that they have access to
    - Default audit and logging requirements there are
- Even if your backend API service / data store is Class V, it does not mean all your API Endpoints are rated at that level. Your API endpoints should constraint access based on the type of authentication

7.13.14 Cost / Benefit Analysis

A cost-benefit analysis (CBA) is a valuable tool for evaluating the potential benefits and costs of developing an API. It helps you make informed decisions about resource allocation and prioritise projects.

Here's a guide to conducting a CBA for your API:

Identify Costs:

- Development Costs: Estimate the costs of development, testing, and deployment
- Maintenance Costs: Consider ongoing costs for maintenance, monitoring, and support
- Operational Costs: Factor in costs related to hosting, infrastructure, and bandwidth

Identify Benefits:

- Cost Savings: Identify areas where the API can reduce costs (e.g., improved efficiency, reduced manual processes)

- Improved Customer Experience: Evaluate how the API will enhance the customer experience and drive loyalty
- Competitive Advantage: Assess how the API can differentiate your business and provide a competitive edge

Quantify Benefits:

- Monetary Value: Assign monetary values to tangible benefits (e.g., increased revenue, cost savings)
- Intangible Benefits: Consider intangible benefits (e.g., improved customer satisfaction, enhanced brand reputation) and try to quantify them using qualitative data or expert opinions

Conduct Financial Analysis:

- Return on Investment (ROI): Measure the expected return on investment compared to the initial costs
- Payback Period: Estimate the time it will take to recoup the initial investment

Consider Risks and Uncertainties:

- Identify Risks: Assess potential risks and challenges that could impact the project's success
- Develop Mitigation Strategies: Outline strategies to mitigate or address identified risks

Present Findings:

- Clear and Concise Report: Prepare a well-structured report that summarises your findings and recommendations
- Visualisations: Use charts and graphs to illustrate key points and make the analysis easier to understand
- Sensitivity Analysis: Conduct sensitivity analysis to assess the impact of changes in assumptions or variables on the results
- Remember to involve stakeholders throughout the CBA process to ensure that their perspectives and concerns are considered

By conducting a thorough cost-benefit analysis, you can make informed decisions about whether to proceed with your API development project and effectively justify your investment.

7.13.15 API platform costs and raising a New Work Request (NWR)

7.13.15.1 Overview

The API platform is centrally funded.

Our preferred delivery model is self-serve - you build and own your own API. See API delivery and support options (see page 293).

If you build and run your own API, there is no charge to use the API platform. We only charge you if you need is to do delivery work specifically for you.

This table hopefully makes that position clear:

Item	Chargeable	Notes
Supporting you to deliver your API yourselves	No	This includes both tech support and delivery process support
Use of the API platform path-to-live environments	No	
Use of the API platform production environment	No	Regardless of transaction volumes
Doing your API discovery for you	Yes	Cost determined via NWR process - see below
Delivering your API for you	Yes	Cost determined via NWR process - see below
Adding features to the API platform specifically for your API	Depends	We will absorb the cost of small improvements. We might charge for significant work items.

7.13.15.2 API delivery costs and timescales

Use of the API platform is free but it costs money (and takes time) to deliver an API. For more details, see API delivery and support options- API delivery costs and timescales (see page 296).

7.13.15.3 How to raise a New Work Request (NWR)

If you need us to the discovery work and/or the build the API you will need to raise separate NWR's for the discovery phase and the alpha build phase.

The first step in the New Work Request (NWR) process is to raise a PILS PMO NWR eStore form to ensure that your request goes to the correct area(s) in the organisation. After PILS PMO approval is given, you will need to then complete a Platforms NWR describing the request in more detail so that it can be effectively assessed and prioritised.

Raise a PILS New Work Request (NWR)

a. Get the latest version of the form from Sharepoint. Use the link that is embedded in the paragraph called 'New Work Request' (not the link beneath the Quick Links section) - https://hscic365.sharepoint.com/Platforms_Infrastructure_Live_Services/Pages/Platforms/Platforms%20New-Work-Requests.aspx.
b. Mark it as destined for "DDC" which is the old name for NHSD Platforms.

c. Send the form to pilsnwr@nhs.net[157].

Note: You will get a confirmation email when the PILS NWR is approved. The confirmation email suggests that you will be sent the platforms NWR. This is generic message for all of platforms, ignore the advice and follow the instructions detailed below to "Raise a Platforms NWR"

Raise a Platforms NWR.

a. Get the latest version of the 'Platforms New Request' form from Sharepoint, located under the quick links section - https://hscic365.sharepoint.com/Platforms_Infrastructure_Live_Services/Pages/Platforms/Platforms%20New-Work-Requests.aspx.
b. Send the form to nwr@nhs.net.[158]

The Platforms PMO team will log your NWR and assign it a reference in the form "NWR-XXXX". We will review your NWR and respond with costs and timescales.

If you have any issues or cannot access any of the linked documentation please contact api.management@nhs.net[159].

7.13.15.4 How we prioritise work

If we're doing work for you, we'll need to prioritise it against other work. To do this we will perform a comparative assessment of value for all the work on our backlog and prioritise accordingly.

If we have a lot of work on, we might increase our team size using people from our delivery partners.

7.13.16 Discovery exit review

Finalising Discovery: A Peer Review with Your Producer Liaison

Before moving to the next phase, we'll conduct a review to ensure we've covered everything necessary during the Discovery phase. This meeting will likely take place during our regular catch-up.

We'll focus on:

- Reviewing the activities you've completed according to the discovery checklist.
- Assessing whether we're ready to move forward to the alpha phase.

This review will help us confirm that we have a solid foundation for the project and are prepared to proceed to the next stage.

157 mailto:pilsnwr@nhs.net
158 mailto:nwr@nhs.net.
159 mailto:api.management@nhs.net

7.13.17 Products and Product IDs

7.13.17.1 Overview

Managing API Consumers and their access to data and APIs is complex, and involves many teams. The term "Product" is used to help multiple teams, both technical and non-technical to work together and help everyone understand who has access to what system and data.

A Product is:

- An application or solution to be assured by NHS England
- Systems uses all sort of terms for this - this introduces consistency
- Loosely defined - the application owner's definition, and what it means to NHS England will change over time
- A way to talk about granted authorisation between stakeholders

At present, ProductIDs cannot be issued until further work is completed by CPM. A manual workaround has been provided for certain early adopters but is not sustainable for a wider release. If this causes any inconvenience, reach out to us via the platforms-apim-producer-support[160] channel for further discussions.

7.13.17.2 Global Product ID

Each Product will be issued a Product ID, and is defined as:

- Human readable - easy to spot matches
- Randomly generated
- Available instantly
- Stays the same in all environments
- Stays the same from inception to decommissioning
- A way to audit and support traceability for compliance - you should log it everywhere
- Loosely coupled, but acts as unique key across system boundaries
- Use to unambiguously attach:
    - Authorisation settings
    - Configuration settings
- Centrally manage settings at a “coarse” grain – simple rules:
    - Locally (e.g. API Producer) manages complex settings
    - Use the identifier to link between central and local

160 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY

- For more complex solutions – you will have to manage your own unique identifier, but must always have a traceable route to the Product ID

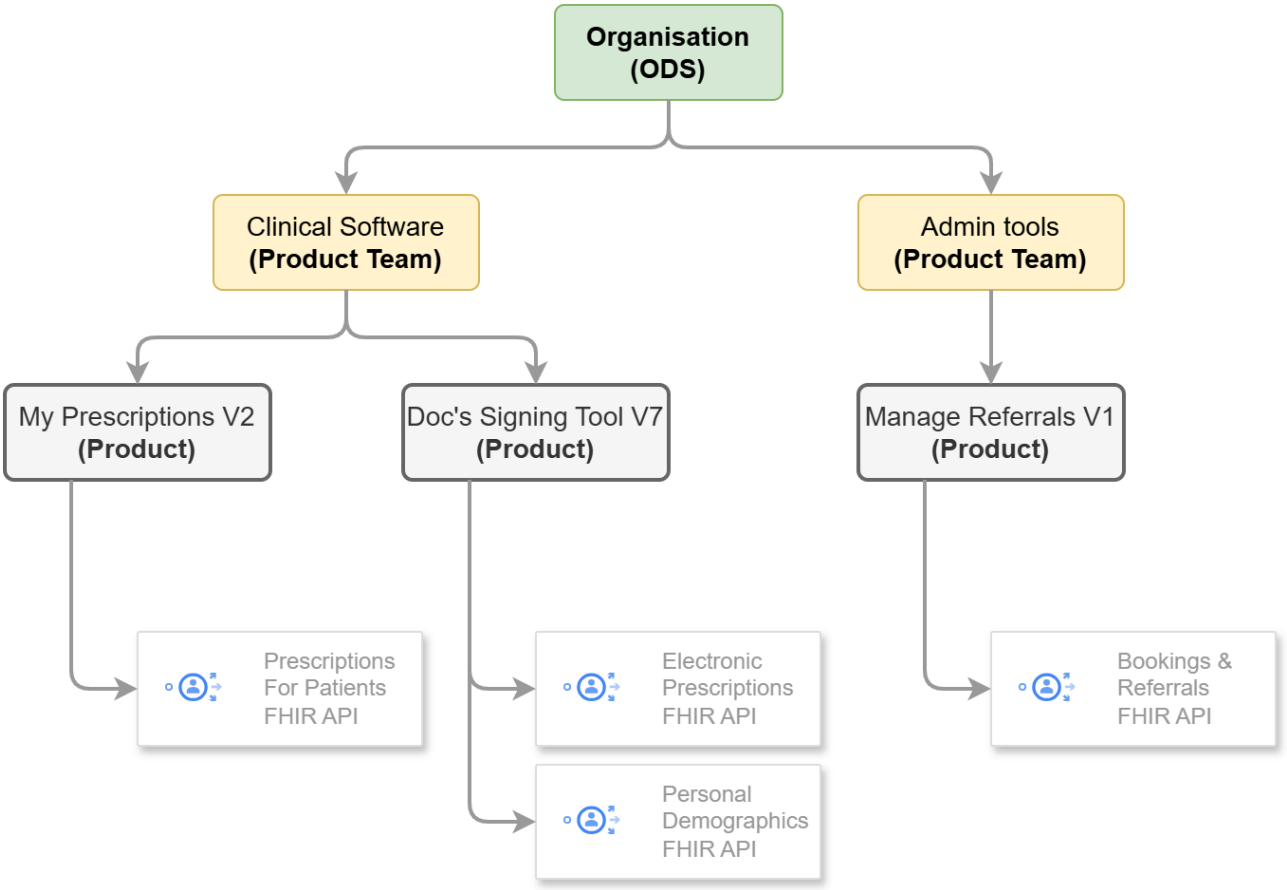
7.13.17.2.1 Identifier format

- Six alpha numeric digits - format: "P.GH7-4TY"
  - "P." is fixed suffix, just for when the ID is being used without context (P = Product)
  - "-" to visually split up the digits
- Case insensitive, but the default is to display as all caps
- No confusing characters e.g. only allow either letter "O" or number 0 (probably allow digits)
- Can be used in a URL without encoding - see RFC3986[161]

7.13.17.3 Organisations, product teams and products

Developing software is complex, especially in large organisations where ownership can change over time - even early on. The relationships are:

161 https://datatracker.ietf.org/doc/html/rfc3986#section-2.3



The Product Team is:

- Always owned by one organisation / ODS code
- Can me moved to a different organisation
- Informal grouping / membership - it can include members from other organisations

7.13.17.4 Products, assurance and connection agreements

The term "product" has been used for many years in NHS England, mostly in the onboarding and legal teams.

All electronic connections to NHS England must be traceable to a legal entity, and this is the entity that signs a "connection agreement" - see the public facing onboarding documentation: https://digital.nhs.uk/services/partner-onboarding#how-this-service-works

7.13.17.5 Versions of products

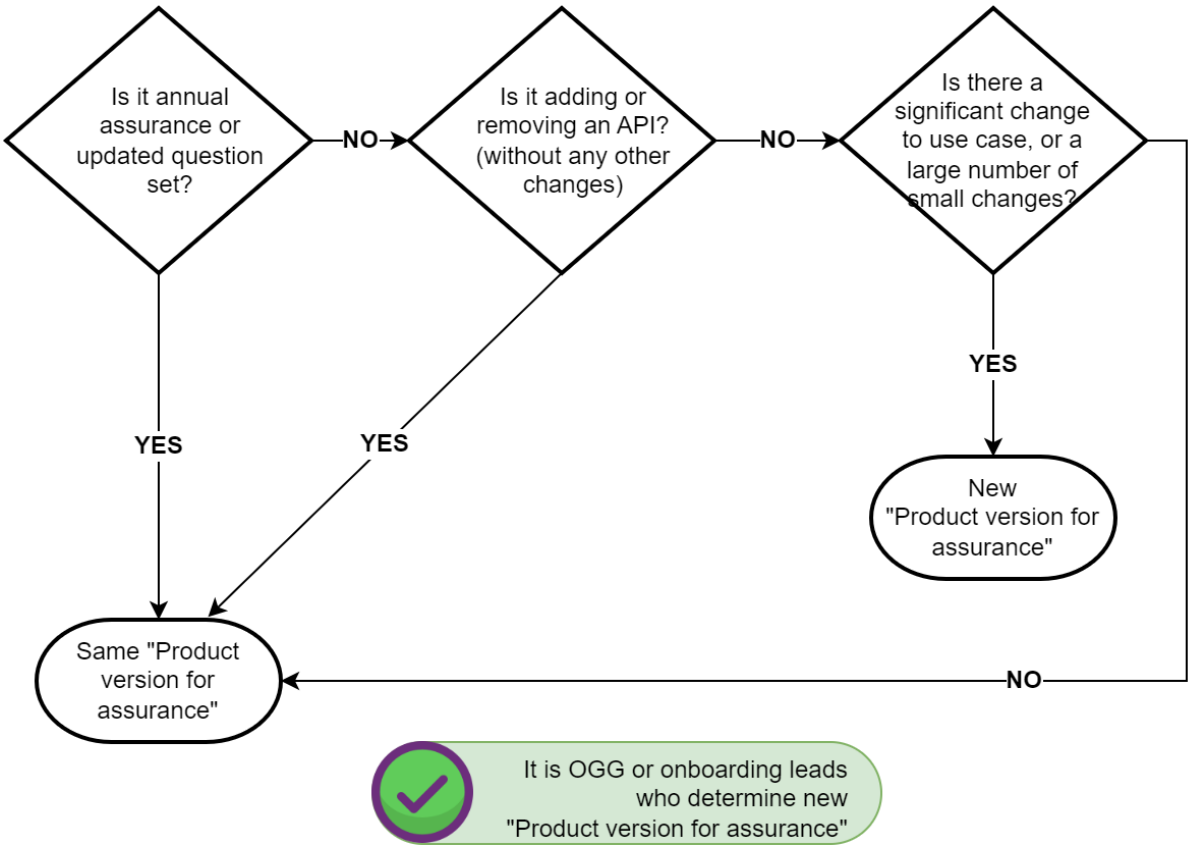
There are many complexities around this, and to make that more complex NHS England doesn't own many of the products. Every time a third party changes their version, they won't want to update NHS England.

The decision was made that a Product would be loosely connected to versions. For some products it would be one version, others it might be many.

The **critical factor** was the impact on NHS England assurance, did we need to do more:

- Changes that impact assurance:
  - Use case changes
  - Legal entity changes (e.g. taken over by another company)
  - Moving processing outside of the UK, or a change to an existing exception
- Adding / Removing APIs will require additional assurance, but it does not change the existing assurance

That makes the decision arbitrary, so it was agreed with the Onboarding governance group (OGG) that a decision process they and onboarding leads own will decide if a new Product is needed:



*** "Product version for assurance" is the short hand definition of "Product"*

7.13.17.6 API Producer pages

Access modes - compliance - security and simplifying running your API (see page 299)

Passing information to your API backend (see page 611)

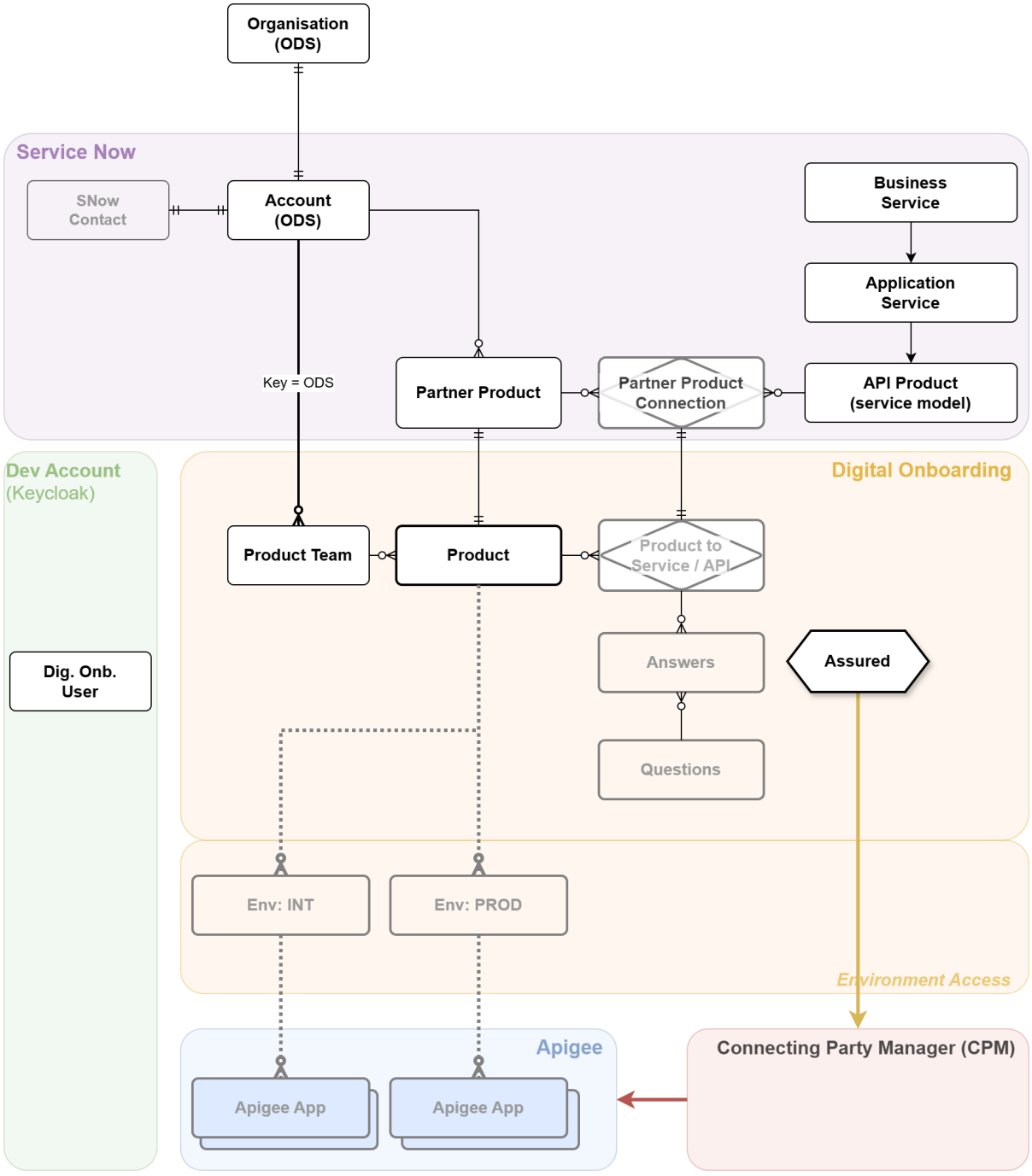
Configuring target attributes (see page 708)

HTTP headers for your API (see page 395)

7.13.17.7 Bigger picture

The Product ID is just one aspect of the wider NHS England ecosystem to onboarding teams, assurance and managing the security of the data we hold.

What the bigger picture looks like has **not** been agreed, however this picture should provide a feel for what that eco system **might** look like.



If you want to see some background, and more technical definitions of how this ecosystem data model **might** work:

- Data model for Product assurance ecosystem[162]

162 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/394533250/Data+model+for+Product+assurance+ecosystem

- Lifecycle - Product and Product Version[163]
- Lifecycle - Organisation[164]
- Lifecycle - APIs and services[165]

7.13.17.7.1 Approvals and agreements for this approach

The global product ID has been agreed by a number of groups Onboarding governance group (OGG), Sarah Wraith - key onboarding lead, Service Now (SNow) team, Digital Onboarding Service (DOS).

Technical Review Group (TRG) approved KAD-10012 21st July 2023 - for links and other approvals see D063 - Identifying and managing connecting applications[166]

7.13.18 Analysis of API Migration from Non-OAS to OAS Format

As part of the Content & Catalogue application, we are starting a project to migrate non-OAS APIs to OAS-compliant JSON specifications. To assist API owners with this transition, we are providing guidance on how to generate OAS spec JSON files and how to publish them to Bloomreach using Proxygen or Apigee. This document outlines the end-to-end process.

7.13.18.1 A comprehensive list of all identified non-OAS APIs.

Attached the comprehensive list of Non-OAS and OAS based API documents.

https://nhs-my.sharepoint.com/:x:/r/personal/vinodkumar_narayanan1_nhs_net/_layouts/15/Doc.aspx?sourcedoc=%7BE8BB52BC-847C-4420-951E-95CF260B84B4%7D&file=OASify-Tracker.xlsx&action=default&mobileredirect=true

S.No	API Catalogue Documents	Count
1	**OAS based API documents**	**58**
2	**Non OAS based API documents** (Valid API with - No Operations section/ No Valid End points)	**28**

163 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/684558838/Lifecycle+-+Product+and+Product+Version

164 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/678892833/Lifecycle+-+Organisation

165 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/700352088/Lifecycle+-+APIs+and+services

166 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/571325748/D063+-+Identifying+and+managing+connecting+applications#D063Identifyingandmanagingconnectingapplications-Reviewsandapprovals

S.No	API Catalogue Documents	Count
3	**Others - Non-OAS based documents** (Non Valid API - *Integrations, API Standards, Retired, Deprecated, Adaptors, Events etc.)* *Valid API - Y/N*	**87**
	Y - but no Operations section and deprecated	2
	Y - but no operations section and retired	9
	Y - but no Operations section and under review for deprecation	*1*
	Y - but retired	*2*
	Y- but third party API	*1*
	Y-but deprecated	*2*
	Y-but no operations section and deprecated	*1*
	Y-but no operations section and under review for deprecation	*2*
	N - Adaptor	*1*
	N - Event	*2*
	N - links to the OAS specs	*1*
	N - API standard	*18*
	N - API standard and retired	*1*
	N - Integration	*38*
	N - Integration (deprecated)	*1*

S.No	API Catalogue Documents	Count
	No - Replaced with someother	1
	Total API Catalogue Documents	**171**

7.13.18.2 A detailed outline of the steps involved in migrating a Non OAS document into OAS spec JSON.

7.13.18.2.1 Steps to Migrate a Non-OAS API Catalogue Document to an OAS-based JSON Document

1. **Generate the OAS Specification JSON**
    - The API team is responsible for creating the OpenAPI Specification (OAS) in YAML or JSON format using the appropriate framework for the respective API service.
2. **Push the OAS JSON to Proxygen/Apigee**
    - The generated OAS specification must be uploaded to **Proxygen** or **Apigee**.
    - Upon successful upload, a **Specification ID** will be generated from the respective platform.
3. **Create an API Specification Document in Bloomreach CMS**
    - In Bloomreach CMS, create a new document using the `APISpecification` document type.
    - Ensure that the "**Show as result in API catalogue**" checkbox is enabled.
    - Associate the document with the corresponding **Specification ID** to allow retrieval of the OAS JSON from backend sources.
4. **Bloomreach Batch Job – JSON Fetch and Document Update**
    - Keep the new API document in **draft status** initially.
    - A scheduled batch job will:
        - Fetch the latest OAS specification JSON from the backend using the Specification ID.
        - Update the document with the retrieved JSON content.
        - Automatically change the document's status to **Published**.
5. **End-to-End Validation**
    - Conduct full end-to-end testing to verify that:
        - All API endpoints, requests, responses, parameters, and schemas are correctly rendered.
        - The API documentation is displayed as expected on the website via Rapidoc or equivalent UI rendering tools.

7.13.18.3 An estimated development time for migrating a single API to OAS.

S. No	Action Item	Owner	Time
1	Generate the OAS Specification JSON	API Team	**TBC**
2	Push the OAS JSON to Proxygen/Apigee	API Team	1 hour
3	Create an API Specification Document in BR CMS	C&C Team	1 hour
4	BR Batch Job – JSON Fetch and Document Update	C&C Team	1 hour
5	End-to-End Validation & Testing	C&C and API Team	1 Day
	Estimated total time to migrate single API in to OAS		

7.13.18.4 An estimated time for migrating all APIs to OAS

Since the criteria for identifying which API documents require migration to the OAS format are not yet defined, it is currently not possible to estimate the time required to complete the migration for all APIs.

7.13.18.5 A summary of any potential challenges or dependencies identified during the analysis.

- Many API services have child documents nested within the API Catalogue. However, it may not be feasible to generate OAS JSON specifications for these child documents.
- After discussing with the APIM team and sharing the names of a few non-OAS documents, it was found that these are not linked to any API services deployed in the Apigee environment. We are facing challenges in identifying the respective owners of these APIs, and the first step is to determine whether they are indeed valid API services.

- We may need to coordinate with the API producer team to generate the OAS JSON specifications and ensure they are pushed to the Apigee or Proxygen environment. So, there is a dependency with API producer team.

7.13.18.6 API Migration from Non-OAS to OAS Format

7.13.18.6.1 Overview

As part of the Content & Catalogue application, we are starting a project to migrate non-OAS APIs to OAS-compliant JSON specifications. To assist API owners with this transition, we are providing guidance on how to generate OAS spec JSON files and how to publish them to Bloomreach using Proxygen or Apigee. This document outlines the end-to-end process.

In Content & Catalogue application, we have hosted the API documentation in Bloomreach. There are two ways to create documentation for an API:

1. **Manual Method**: Creating a document using the 'General' document type and manually adding content.
2. **Automated Method**: Creating a document using the 'Apispecification' document type and generating documentation using an OpenAPI Specification (OAS) JSON file through mapping the correct specification ID.

To support the automated approach, we have implemented a batch job in Bloomreach that retrieves the OAS spec JSON files from our backend platforms, Apigee and Proxygen.

7.13.18.6.2 Why use OpenAPI Specification (OAS) JSON for documentation?

Using OAS JSON for API documentation offers several key advantages:

- **Standardization**: OAS provides a widely adopted, language-agnostic standard for describing RESTful APIs. This ensures consistency across all API documentation.
- **Automation**: With OAS, documentation can be generated automatically, reducing manual effort and human error. Tools like Swagger UI or Redoc can render interactive documentation directly from the spec.
- **Developer Experience**: Consumers of the API benefit from clear, structured, and interactive documentation that's easier to understand and explore.
- **Integration Friendly**: OAS is supported by many platforms (including Apigee, Proxygen, and Bloomreach), enabling smoother integration and faster onboarding of APIs.
- **Maintainability**: Changes to the API can be reflected in the documentation by simply updating the spec file—no need to manually edit documentation pages.
- **Validation & Testing**: OAS can be used to validate API behavior against the spec and to auto-generate client SDKs or test cases.

Overall, using OAS spec JSON helps us maintain high-quality, up-to-date, and scalable API documentation with minimal manual intervention.

7.13.18.6.3 OAS Migration Implementation - Environment:

The OAS migration and creation of Bloomreach documentation should initially be carried out in the **Training** environment. Once successfully validated and tested, the implementation should then be promoted to the **Production** environment.

7.13.18.6.4 Steps to Migrate a Non-OAS API Catalogue Document to an OAS-based JSON Documents

1. **Generate the OAS Specification JSON**
    - The API team is need to create the OpenAPI Specification (OAS) in YAML or JSON format using the appropriate framework for the respective API service.
2. **Push the OAS JSON to Proxygen/Apigee**
    - The generated OAS specification must be uploaded to **Proxygen** or **Apigee.** (Apigee is highly recommended)
    - [To publish the spec JSON into Proxygen, follow this link - Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571) ; To push the spec JSON into APigee, follow this link - Publishing your API documentation in Bloomreach CMS (see page 507)]
    - Upon successful upload, a **Specification ID** will be generated from the respective platform.
    - [For any support with pushing the OAS specification JSON to Proxygen or Apigee, please reach out via the producer support Slack channel: #platforms-apim-producer-support[167]]
3. **Create an API Specification Document in Bloomreach CMS**
    - In Bloomreach CMS, create a new document using the `APISpecification` document type. [To create a Bloomreach CMS document - Publishing your API documentation in Bloomreach CMS (see page 507)]
    - Ensure that the "**Show as result in API catalogue**" checkbox is enabled.
    - Associate the document with the corresponding **Specification ID** to allow retrieval of the OAS JSON from backend sources.
    - [For any support with creating a bloomreach document, please reach out to the Content and Community[168] team (C+C)]
4. **Bloomreach Batch Job – JSON Fetch and Document Update**
    - Keep the new API document in **draft status** initially.
    - A scheduled batch job will:
        - Fetch the latest OAS specification JSON from the backend using the Specification ID.
        - Update the document with the retrieved JSON content.
        - Automatically change the document's status to **Published**.

167 https://nhsdigital-platforms.slack.com/archives/G016JRWN6AY
168 mailto:england.contentandcommunity@nhs.net

5. **End-to-End Validation**
    - Conduct full end-to-end testing to verify that:
        - All API endpoints, requests, responses, parameters, and schemas are correctly rendered.
        - The API documentation is displayed as expected on the website via Rapidoc or equivalent UI rendering tools.

7.13.18.6.5 Estimated development time and required steps for migrating a single API to an OAS-compliant spec JSON:

S. No	Action Item	Owner	Time
1	Generate the OAS Specification JSON	API Team	2-4 days
2	Push the OAS JSON to Proxygen/Apigee	API Team	1 hour
3	Create an API Specification Document in BR CMS	C&C Team	1 hour
4	BR Batch Job – JSON Fetch and Document Update	C&C Team	1 hour
5	End-to-End Validation & Testing	C&C and API Team	1 Day
	Estimated total time to migrate single API in to OAS		**1-2 Weeks**

7.13.18.6.6 Contact:

For support with migrating API documentation into OAS spec JSON, please reach out through the following channel:

- **Email:** england.contentandcommunity@nhs.net (see page 318)
- **Slack Channel:** #platforms-core-services-content-and-community[169]
- **Team:** Content & Community[170]

169 https://nhsdigital-platforms.slack.com/archives/C02PLAUN6G6
170 https://nhsd-confluence.digital.nhs.uk/spaces/CONCOM/pages/1031458347/Content+and+Community+Home

7.14 Alpha phase

- Designing your API (see page 322)
- Building your API alpha (see page 578)
- Building and deploying your API alpha with Proxy Generator (see page 680)
- Testing your API (see page 715)
- Securing your API (see page 774)
- Securing your API with proxy generator (see page 856)
- Deploying your API (see page 882)
- Alpha exit review (see page 933)
- Proxygen CLI user guide (see page 933)

7.14.1 Overview

Alpha is about building and iterating your API design.

You will:

- design your API
- publish your API specification
- build test environments
- test the API with an API consumer
- iterate the design based on feedback

7.14.2 Process steps

For detailed process steps, see API process checklist (see page 198). For APIs not created via the proxy generator, or API process checklist using proxy generator (see page 227).

7.14.3 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

- Designing your API (see page 322)
- Building your API alpha (see page 578)
- Building and deploying your API alpha with Proxy Generator (see page 680)
- Testing your API (see page 715)

- Securing your API (see page 774)
- Securing your API with proxy generator (see page 856)
- Deploying your API (see page 882)
- Alpha exit review (see page 933)
- Proxygen CLI user guide (see page 933)

7.14.4 Designing your API

7.14.4.1 Overview

API Design is the process of defining how an API will expose data and functionality to its consumers. It involves making intentional decisions about the API's structure, behavior, and interactions.

During the discovery phase, you will have made some big decisions about your API (as per Architecting your API (see page 260)):

- Is a RESTful API the correct interaction pattern (it will be in most cases)?
- Will it be a FHIR API?
- Where will the backend (or backends) be, and how will the API proxy connect to it/them?
- What security wrap does it need?

During the alpha phase, you need to make some more detailed decisions - we call this "designing your API". This page splits out some of these key topics into pages of specific help and advice - main areas:

- Are there a key set of design principles to help guide the detailed design?
- What logic should be in my proxy vs my backend?
- Detailed API design of your endpoints, making sure (if appropriate) all the details are FHIR compatible
- Setting the scope of the sandbox
- If I need to do transformation or mediation of requests, what are the options

Following this advice should:

- make your life easier because decisions have already been made for you
- make sure your API is more consistent with other APIs on the API platform

As ever, there are always exceptions to the rule - if you want to deviate from this advice, get in touch with us to discuss.

During this phase (Alpha), you conduct an API design review - for more details, see Deliver your API (see page 194).

7.14.4.2 Guiding design principles

The following are an API Producer focused subset of the Platform Architecture design principles[171]:

- Adopt an API consumer viewpoint
- The APIs should be RESTful and synchronous
- The solution should be a FHIR Compliant API where the API conveys healthcare data
- The Platform is loosely coupled with the API backend

171 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97747483/Architecture

- The API end points exposed should all be consistent across the platform
- The API Management Platform will reduce the assurance burden for API Consumers
- The APIM coordinated authentication process uses national identity providers and will be seamless between different APIs
- Minimise the information requested from the API Consumer
- All configuration is stored and managed in a GitHub repo for that API

In the future, coarse grained authorisation based on the National RBAC model might be incorporated into the Platform.

POTENTIAL additional principle: Where the logic is simple consider what is the "right tool for the job", Apigee offers a resilient HA central point for traffic to progress through, allowing some types of decisions to be made here. It might, also, be much more complex to implement in the backend.

The Architecture pages in the Discovery Phase (see page 260)are good to run over (again) some of the top level design concepts, prior to jumping into some of the detailed implementation design considerations below.

7.14.4.2.1 Adopting an API consumer viewpoint

In making API design decisions, we encourage you to take a consumer viewpoint, known as "API first design". Taking this approach you will:

- Consider what would be the simplest API design for a consumer to understand and implement
- If the API is not a greenfield (i.e. an existing back-end exists), intentionally forgetting the complexity of the back-end in order to start with an "ideal" API design.

For an example of how we adopted this approach for the Summary Care Record API, have a look at Lessons from the coal face of API first design[172].

7.14.4.3 Moving into the detailed design

Overview areas of how to refine your API design:

- Proxies in Apigee and what is an API Producer responsible for (see page 402)
- Key elements of any API published on the platform (see page 404)
- Access modes - compliance - security and simplifying running your API (see page 299)
- Managing different security requirements for API Consumers (see page 813)
- Transforming data on the APIM platform and containers (see page 371)

7.14.4.3.1 Considerations for designing your interface

Many of the API interface decisions are easy to make, but as the work progresses some more detailed considerations need to be looked at:

- API naming and grouping endpoints (see page 359)
- Consistency between APIs (see page 362)
- Proxy base path vs API base path (see page 364)
- RESTful HTTP design points (see page 365)
- HTTP headers for your API (see page 395)

172 https://digital.nhs.uk/blog/tech-talk/2021/lessons-from-the-coalface-of-api-first-design

- Privacy considerations in your design (see page 406)

FHIR and building an API that is FHIR conformant has many aspects, these pages take you through that:

- Designing your FHIR API (see page 324)
- Getting Started with HL7 FHIR (see page 334)
- HL7 in NHS Digital and wider NHS (see page 340)
- CareConnect FHIR STU3 to UKCore FHIR R4(+) Conversion (see page 345)
- FHIR Testing (Validation) (see page 350)
- FHIR Servers (see page 356)

7.14.4.3.2 Other areas of design

- Setting up your sandbox (see page 592)
- Passing information to your API backend (see page 611)
- Spine based APIs (see page 408)
- OAS - you specify your API design in an OAS file see Documenting your API (see page 445).
- Version control - APIM has automated this to a large degree see Version control (see page 380)

7.14.4.3.3 Results of the FHIR survey

Results of the API / FHIR Survey : 28/02/2023 - Developer FHIR / Restful API Survey Results[173]

7.14.4.4 FHIR

- Designing your FHIR API (see page 324)
- Getting Started with HL7 FHIR (see page 334)
- HL7 in NHS Digital and wider NHS (see page 340)
- CareConnect FHIR STU3 to UKCore FHIR R4(+) Conversion (see page 345)
- FHIR Testing (Validation) (see page 350)
- FHIR Servers (see page 356)

7.14.4.4.1 Designing your FHIR API

7.14.4.4.1.1 Use of FHIR

NHS Digital has a policy of using FHIR for new APIs where this is appropriate. In most cases, APIs will use FHIR as they will convey data entities which are clearly healthcare related. But sometimes FHIR may not fit well - see Is my API a FHIR API? (see page 285).

- See API Landscape[174] for the policies guiding this

173 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=608904062
174 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Landscape

- During the Discovery phase of your APIM project, you will go through an API architecture review
    - We will include a member of the IOPS (Interoperability Standards) Team, who has expertise in designing FHIR-compliant APIs
    - See Governance[175] for the different GDS project phases and details of the review

For an introduction to FHIR see Getting Started with HL7 FHIR (see page 334).

Also, see API naming and grouping endpoints (see page 359) for a definition of terms such as "Endpoint" and "API" as used in the API producer zone.

7.14.4.4.1.2 FHIR API design patterns

The FHIR standard defines different ways in which FHIR data entities can be exchanged - as defined at http://hl7.org/fhir/exchange-module.html. The most common means of exchanging FHIR information are summarised below:

Exchange Paradigm	Usage
RESTful API[176] (**preferred**)	Implementation of RESTful FHIR APIs where CRUD operations are defined on resource endpoints (e.g. GET /Patient/123456) This also includes the operations framework for scenarios where you want to enable a CRUD API doesn't fit your use case. (e.g. POST /$register_patient)
Messaging [177]	Used when exchanging data asynchronously, for example using MESH, and therefore not typically used to implemented a FHIR API.
Documents[178]	A (html) statement of healthcare information in the form of a FHIR composition - effectively a self-contained document. Often used when you need to share patient care information in a fixed human readable format. Often used in conjunction with the Messaging paradigm

In most cases you will be using creating a synchronous FHIR API, so the RESTful API approach is strongly encouraged.

The FHIR RESTful API specification[179] appears fairly complex at first, but there are FHIR reference server implementations which you can use to do much of the hard for of creating a FHIR-compliant server implementation. See FHIR Servers (see page 356).

Reasons that FHIR Messaging should be avoided where possible:

175 https://nhsd-confluence.digital.nhs.uk/display/APM/Governance
176 https://www.hl7.org/fhir/http.html
177 https://www.hl7.org/fhir/messaging.html
178 https://www.hl7.org/fhir/documents.html
179 https://www.hl7.org/fhir/http.html

- The messages are "hidden" under one endpoint the $process_message, this make it less intuitive
- Each message requires it's own definition / schema - the OAS format provides a simple way of making that consistently visible, under a well named endpoint
- In almost all web and API frameworks there is no cost to adding an endpoint name
- Where actual message routing is part of the design, having the routing information in the payload isn't automatically the simplest approach
- Overall it is not as intuitive as REST, quote from Lloyd McKenzie "*The general rule is that REST will give you greater interoperability with more systems. With REST, the data structures (the resources) and the allowed operations (create, search, update, etc.) are all nailed down. Systems only need to decide which they're going to support. With messaging, there's a need to negotiate and agree...*" (See FHIR Community Messaging vs RESTful[180])

Also, to get some further sense of what you are designing, have a look at the PDS FHIR API[181] which is an exemplar of a RESTful FHIR API which provides search, retrieval and update options for consumers.

7.14.4.4.1.3 Your design approach - use references to provide context

Your API will have a set of API endpoints which provide operations on single data entities such as Patient[182] or AllergyIntolerance[183]. These individual resources contain references[184] which provide the wider context and meaning of the resource. For example, the AllergyIntolerance resource contains references to:

- The patient who has the allergy
- The healthcare encounter during which the allergy or intolerance was asserted

This set of supporting information is vital in order to ensure that the healthcare information made available by the API is used safely. In previous implementations of healthcare standards, a common practice was to include all this supporting information within API responses. The FHIR http standard no longer encourages this approach to maintaining clinical context:

> **Lloyd Mckenzie, HL7 international**
>
> Maintaining context in REST is generally achieved by passing the references. If you've got a pointer to the patient and are able to retrieve it if/when you want it, then you have the context. Having the context is distinct from having all of the data.

So for the example above, a search on the /AllergyIntolerance endpoint would return a set of AllergyIntolerance resources only. Each of these resources would contain:

- A reference to the patient in the AllergyIntolerance.patient field

180 http://community.fhir.org/t/considerations-for-messaging-vs-restful-exchange-paradigm/1401
181 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
182 https://simplifier.net/guide/HL7FHIRUKCoreR4Release1/Home/ProfilesandExtensions/ProfileUKCore-Patient
183 https://simplifier.net/guide/HL7FHIRUKCoreR4Release1/Home/ProfilesandExtensions/ProfileUKCore-AllergyIntolerance
184 http://www.hl7.org/fhir/references.html#Reference

```
 "patient": {
    "reference": "https://api.service.nhs.uk/personal-demographics/FHIR/
R4/9000000009",
    "identifier": {
        "system": "https://fhir.nhs.uk/Id/nhs-number",
        "value": "9000000009"
    },
    "display": "Jane Smith"
  }
```

- A reference to the encounter

```
"encounter": {
    "reference": "Encounter/972f29f9-5020-4b91-9301-5177a20f4208",
    "identifier": "972f29f9-5020-4b91-9301-5177a20f4208"
}
```

NHS Digital policy on use of references, FHIR-IDENT-04[185] outlines how references in RESTFul FHIR APIs are defined:

- Where possible you should include a resolvable URL in the "reference" attribute
- Include a logical identifier, using a naming system[186] where possible, in the "identifier" attribute.

Ideally, both a resolvable reference and a logical identifier should be provided.

Where only a logical identifier is provided, your API documentation must include guidance on how to access the full information associated with this identifier. This guidance is particularly important where there is a clinical safety driver which requires this supporting information to be accessible, and could refer to another API, or other means of accessing this information.

Providing "display" information in references

In the example above, the patient resource reference includes a "display" attribute containing the patient's name.

Often the API client is only interested in a single data attribute in a referenced resource - a piece of information to be used in the user interface. The "display" attribute of a resource can be very helpful for clients in many cases as this often removes the need for an additional API call. Where you can provide this textual alternative to the referenced resource, you are encouraged to do so.

185 https://simplifier.net/guide/NHSDigital/Home/Policy/Identifiers.guide.md
186 https://simplifier.net/guide/NHSDigital/Home/FHIRAssets/NamingSystems.guide.md

For APIs that accept incoming data, this might requires some design thinking. NHSE, in the centre, is often taking in data to then make it available to others. The incoming payload should not ask the API Consumer to attach / insert arbitrary "display" information into the payload. When the data is retrieved the API Consumer should benefit from common use cases for the API (e.g. holding the patient's name).
This is about NHSE doing the hard work to make integration easy.

7.14.4.4.1.4 Identifying your API endpoints

The approach below will help you to define your RESTful FHIR API endpoints:

1. Consider first the set of data entities which the API will expose to consumers.
    - Refer to any analysis which has already been done during an earlier phase, particularly when you were seeking to answer the question Is my API a FHIR API? (see page 285)
    - For example, if your API will enable consumers to search for information about NHS organisations and patients, you will have two API endpoints - Organization and Practitioner
    - There might be more than one suitable resource - it is really important to look at your decision from the perspective of an API Consumer and how systems inter-operate in everyday situations. The choice can impact user journeys and make the API significantly easier or harder to integrate with
2. For each of these API endpoints, define what your consumers need to be able to do with these FHIR resources. You can express this in CRUD terms as follows:
    - Create - use POST and follow the FHIR guidance as described https://www.hl7.org/fhir/http.html#create
    - Read. There are two options here:
        i. Searches against your resource endpoints for a matching set of resources of the same type.
        ii. Read of a single resource based on a supplied identifier - often referred to as a resource retrieval
    - Updates. There are two types of updates
        i. Partial updates, which should be expressed using a PATCH http verb - see below
        ii. Full resource update, which should use a PUT http verb.

The following sections provide best practice in implementing RESTful CRUD operations in FHIR:

Create operations in FHIR

The Information Push pattern is an example of an integration pattern[187] where you RESTful create operations[188]are recommended.

It is often necessary to create more than one FHIR resource to convey all information associated with an event which has occurred.

For example, the details of a medical device which has been implanted into a patient during a procedure will involve two resources to provide the information on the event - the Procedure[189] and the implanted Device[190]. This would be implemented in the following API calls:

- POST /Device
  The API client captures the logical ID of the created Device resource in the response Location: header
- POST /Procedure
  The API client populates the Procedure.focalDevice attribute with the logicalID of the Device previously captured

Update this section & page with:

- Reminder about references above
- Is a reference, just a reference? Or are there IG or business rules that require repeating information about that reference in the payload?
- Does the scope of API actually include being the master owner of that reference? If that is the case then an endpoint for that referenced resource might be appropriate for that specific api

When is an information push API "too chatty"?

RESTFul APIs are chatty - that's not a bad thing.

But for complex events which involve a graph of a large number of different FHIR resources, there comes a point where the level of chattiness becomes a complexity problem for your API consumers. When more than 5-10 resource types are involved in such a scenario, it may be better to divert from an RESTFul approach, and simplify things using a custom operation (see page 332). In this scenario, defining a custom operation which creates multiple resources as a single atomic transaction is preferred to the use of Create operations on the Bundle resource - known as "transactional Bundles"

Read operations in FHIR

187 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book
188 https://www.hl7.org/fhir/http.html#create
189 https://www.hl7.org/fhir/procedure.html
190 https://www.hl7.org/fhir/device.html

Implement a search interface which meets the FHIR search specification - https://www.hl7.org/fhir/http.html#search. Each of the FHIR resources has a defined set of possible search parameters - For example, for Patient see http://hl7.org/fhir/patient.html#search

The IOPS team maintain the NHS FHIR Implementation Guide which gives you more information about how FHIR R4 APIs should be defined. For example, see https://simplifier.net/guide/NHSDigital/CapabilityStatement for information about supported search parameters. If your search parameter is not listed, this can be discuss during the API Design Review[191]

For an idea of what a FHIR API looks like which implements searches, the CareConnect API specification is worth a look. This is based on FHIR STU3 however searches are implemented in very much the same way in FHIR R4. For example, https://nhsconnect.github.io/CareConnectAPI/api_workflow_encounter.html#2-search describes a standard search interface for Encounters.

Defining resource retrievals on endpoints

There are two types of resource retrieval - by logical or business identifier.

Retrieval by logical identifier

They enable your consumers to retrieve a resource by an ID which your server is responsible for. E.g. the UUID of a patient record in your database:

```
GET [baseUrl]/Patient/5a872303-99b5-446a-9746-4d9bf4083cd5
```

Retrieval by business Identifier

Where identifiers have meaning outside your server, and there is another service which maintains the those identifiers, this would be defined as a business identifier.

For example, the Personal Demographics Service maintains the NHS number identifier. Therefore to retrieve a patient by NHS Number in your API, you would define the retrieval as follows:

```
GET [baseUrl]/Patient?identifier=https://fhir.nhs.uk/Id/nhs-number|9876543210
```

Returning more than one resource type in responses

Search operations on resource based endpoint should normally return resources only of the type associated with the endpoint. I.e. an /Encounter endpoint will only return Encounter resources within the result Bundle. As mentioned above, context is maintained normally through the use of references[192]. It may be useful to additional resources with the result in scenarios where in most cases clients will would otherwise need to make an additional API call or set of calls in order to retrieve data needed to display views in a user interface. The FHIR constructs _include and _revinclude[193] provide a RESTFul means of doing this.

Returning resources referenced from another resource

191 https://nhsd-confluence.digital.nhs.uk/display/APM/Governance#Governance-APIdesignreview
192 http://nhsd-confluence.digital.nhs.uk#use_references
193 https://www.hl7.org/FHIR/search.html#revinclude

Each resource may belong to one or more logical compartments. A compartment is a logical grouping of resources which share a common property.

Compartments have two principal roles:

- Function as an access mechanism for finding a set of related resources quickly
- Provide a definitional basis for applying access control to resources quickly

Variant Searches, allows searches on compartments for either all possible resources or for a particular resource type, respectively:

GET [baseUrl]/[Compartment]/[id]/*{?[parameters]{&_format=[mime-type]}}
GET [baseUrl]/[Compartment]/[id]/[type]{?[parameters]{&_format=[mime-type]}}

In the first URL the character " `*` " appears in the URL as a literal to mean 'all types'. So, for example, to retrieve all the observation resources for a particular LOINC code associated with a particular encounter:

```
GET [baseUrl]/Encounter/23423445/Observation?code=2951-2  {&_format=[mime-type]}
```

Note that there are specific operations defined to support fetching an entire patient record[194] or all record for an encounter[195].

Update operations in FHIR

Synchronous updates using the FHIR http paradigm[196] fit best when the update operation can be completed without the involvement of a human actor.

If a human actor must be involved, refer to the Integration Patterns Book[197] for guidance.

Implementing partial updates to FHIR resources

Example API use case: PDS FHIR Update to sections of a Patient resource.

Where the consumer will be updating a small portion of a FHIR resource, rather than significant or full update of an existing record, the following approach is suggested:

- HTTP PATCH verb should be used, rather than an HTTP PUT - https://www.hl7.org/fhir/http.html#patch
- A FHIR JSON Patch document should be supplied to the API with content type of "application/json-patch+json", meeting the JSON Patch specification: https://tools.ietf.org/html/rfc6902
- The API should support the presence of multiple JSON Patch operations within a single HTTP PATCH verb instance.

7.14.4.4.1.5 Use of custom FHIR operations

Sometimes, your business use case cannot easily be described as a set of CRUD operations. For example, you may need to invoke business rules across multiple different resources, and/or change the state of many

194 https://www.hl7.org/fhir/patient-operation-everything.html
195 https://www.hl7.org/fhir/encounter-operation-everything.html
196 https://www.hl7.org/fhir/http.html#update
197 https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-architecture/integration-patterns-book

resources in a single atomic call. The FHIR operations framework[198] enables you to do this. You are really defining an RPC (Remote Procedure Call) where your API consumer sends a set of parameters to a function. You define a custom result such as a bundle of FHIR resources describing the outcome, which you return to the consumer as the outcome.

Because we want to encourage a RESTful style where possible, it's good to avoid use of FHIR operations if possible. It is often possible to express these in RESTful terms.

7.14.4.4.1.6 Using FHIR profiles and extensions

FHIR is a global standard. FHIR profiles and extensions are used to extend FHIR to deal with specific setting or use case, often for a specific geographical region.

Some definitions:

- A FHIR profile[199] customises a FHIR resource by adding, removing or changing resource elements for a given setting.
- A FHIR extension[200] defines the shape of an additional. resource element

When you are defining the requests and responses of your API endpoints (see page 328), you may find that the base global FHIR resources don't meet your needs. Help is at hand:

- FHIR UK Core[201] is a suite of profiles and extensions designed for use across the UK.
- The NHS Digital FHIR profiles and extensions are specifically for use in the NHS in England.

These profiles and extensions are owned and maintained by the IOPS team. If they don't meet your needs, contact them to discuss.

When not to use FHIR profiles and extensions

In some circumstances, profiles and extensions are not the right choice. For example:

1. When the data attribute is found as a core attribute belonging naturally only to another resource. For example, it would be incorrect to define a "Patient Name" extension to the MedicationRequest UK Core profile[202] because the Patient UK Core profile[203] has Patient.address as a core element.

Version control for FHIR profiles and extensions

The UK Core and NHS Digital FHIR profiles and extensions change from time to time. Sometimes, the changes are breaking changes. The FHIR profiles and extensions are version controlled - each version is known as a "package".

When first built, a given API will be built against a specific version/package of the UK Core and/or NHS Digital FHIR extensions.

---

198 http://hl7.org/fhir/operations.html
199 https://www.hl7.org/fhir/profiling.html
200 https://www.hl7.org/fhir/extensibility.html
201 https://digital.nhs.uk/developer/api-catalogue/fhir-uk-core-standards
202 https://simplifier.net/guide/HL7FHIRUKCoreR4Release1/Home/ProfilesandExtensions/ProfileUKCore-MedicationRequest
203 https://simplifier.net/guide/HL7FHIRUKCoreR4Release1/Home/ProfilesandExtensions/ProfileUKCore-Patient

The API specification must clearly state which package it is built against. For an example, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-description__technology.

New versions of UK Core are generally announced at https://simplifier.net/organization/hl7uk.

API producer teams are responsible for:

- Keeping an eye out for new packages (versions) having been published form time to time.
- Deciding whether and when to uplift their API to use newer packages. Uplifting an API is optional, not mandatory - especially if the change is a breaking change.

7.14.4.4.1.7 Using contained resources

In the context of your RESTful FHIR API, contained resources enable you to include another FHIR resource *contained within* resource associated with the endpoint.

Contained resources should be used with caution.

When to use a contained resource

There are some specific scenarios however where using a contained resource information is appropriate:

- **When implementing a RESTful operation on resource based endpoints where you must convey information within the request which could be expressed as an additional resource, *but where this additional resource does not have an identity of its own and there is no expectation that this additional resource will subsequently be resolvable.***

  For example, there are legal requirements that prescription information conveyed in APIs in England must include the name of the patient. The requirement dates back to the time before the presence NHS numbers could be assumed. In this case it is acceptable to use a minimal contained Patient resource within the MedicationRequest.

7.14.4.4.1.8 FHIR capability statements

A key requirement for a FHIR API is to have a "capability statement" endpoint. The physical name of is endpoint is actually `metadata`, but the FHIR documentation refers to it logically as a capability statement (see https://www.hl7.org/fhir/capabilitystatement.html).

They are designed to be machine readable and are supposed to help with automated discovery and plug & play type automation.

In practice we don't see that happening a lot. We tend to see developers reading the API specification (perhaps downloading the OAS file) and building against that. We don't have a lot of evidence that people find them useful. Building a capability statement endpoint is an overhead, both in terms of delivery and maintenance - it needs keeping in line with any additions or changes to the API itself.

For these reasons, very few of our RESTful APIs have capability statements. Technically speaking, they are therefore not FHIR-compliant. But we haven't really had much feedback asking us to add them.

We recommend **not** building a capability statement (metadata) endpoint for your API, but do consider adding one (and let us know) if you get user feedback saying it would be useful.

If you follow this guidance, be sure to call it out in your API specification. For example, see the FHIR conformance statement in https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#overview--technology.

7.14.4.4.1.9 Getting help

If you need help designing your FHIR API:

- API Management can advise on RESTful API design - see Help and support-Architecture support (see page 2056)
- IOPS can advise on FHIR payload design, including use of FHIR UK Core - see Is my API a FHIR API? Who are IOPS (see page 288)?

7.14.4.4.2 Getting Started with HL7 FHIR

7.14.4.4.2.1 Introduction

HL7 FHIR is the mandated standard for healthcare domain API's and Messages in the NHS. It is an international standard from Health Level 7 (HL7), the number seven refers to the application layer, which is "layer 7" in the OSI model[204].

For background on HL7 and it's use in the wider NHS and NHS Digital see HL7 in NHS Digital and wider NHS (see page 340)

Overview for developers[205]

7.14.4.4.2.2 Schema and 'rules' profiles

The FHIR schema is available in XML, JSON and RDF formats from https://www.hl7.org/fhir/downloads.html

Each country may apply specific rules on top the base schemas, these are called **FHIR Profiles** and the UK set is known as UKCore[206]

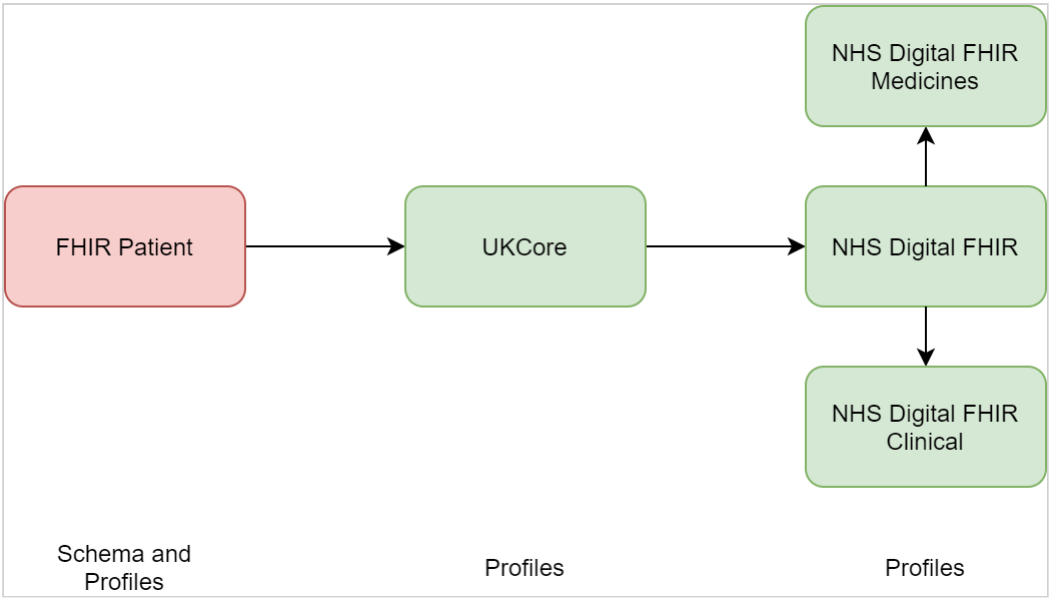
UKCore also contains clinical and organisational data model rules, such as how NHS Number is represented or what are the valid codes and codesystems+valuesets for clinical specialty. CodeSystems, valuesets tend to based on NHS Data Dictionary and SNOMED CT (clinical terms codesystem).

NHS Digital has extra rules on top of the UK Core and these are found in NHS Digital FHIR Implementation Guide[207] which focuses on core Patient Administration resources with extensions for Medicines (NHS Digital FHIR Medicines Implementation Guid[208]e) and Clinical (NHS Digital FHIR Medicines Implementation Guide[209])

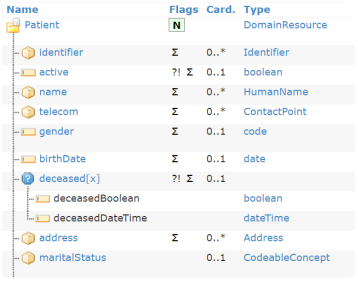
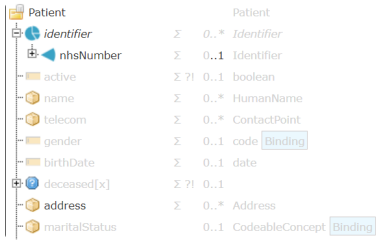
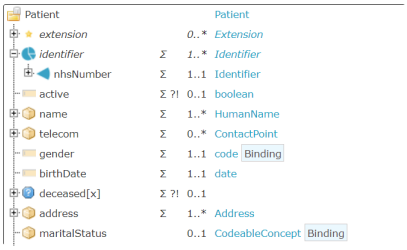
204 https://en.wikipedia.org/wiki/OSI_model
205 https://www.hl7.org/fhir/overview-dev.html
206 https://simplifier.net/guide/ukcoredevelopment2/home
207 https://simplifier.net/guide/nhsdigital/home
208 https://simplifier.net/guide/digitalmedicines/home
209 https://simplifier.net/guide/nhsdigitalclinical/home

The diagram below illustrates that:

- UKCore profiles are derived from the HL7 "top level" resource defintion, and thus the UKCore Patient includes an NHS Number element.
- Further derived profiles in this hierarchy can be defined. Here, there is an NHSDigital Patient profile which is a futher specialisation of the UKCore Patient profile.



Profiles are typically expressed in a tree structure within a Implementation Guide.:

FHIR Patient	UKCore-Patient	NHSDigital-Patient
		
https://www.hl7.org/fhir/patient.html	https://simplifier.net/guide/UKCoreDevelopment2/ProfileUKCore-Patient	https://simplifier.net/guide/nhsdigital/NHSDigital-Patient

FHIR Patient	UKCore-Patient	NHSDigital-Patient
Note **identifier** is optional	**identifier** is still optional but contains guidance on nhsNumber	**identifier, name, gender, birthDate and address** are now mandatory. nhsNumber is also mandatory

NHS Digital Implementation Guides

The aim of the NHSD IG's to ensure the data models across API's is consistent. For example:

PDS is a restful API and is the main source of the Patient resource. EPS is predominantly a messaging API and requires the consumer to create a Patient resource, to make it easier for consumers they will be encouraged to source the Patient from PDS. Both PDS and EPS should conform the the NHSDigital-Patient[210] *profile.*

NHS Digital IG	Covers	Example API
nhsdigital[211]	Individuals Entities Other	PDS SDS ODS
nhsdigital.medicines[212]	Medications Workflow Billing	EPS COVID Vaccination History
nhsdigital.clinical[213]	Diagnostics Request/Response	COVID Test Results NHS App Patient Notifications

FHIR Validation

Conformance to the profiles is performed using FHIR Validation[214] and not schema validation (as the schema does not contain local rules)

FHIR Validation will check:

210 https://simplifier.net/guide/nhsdigital/NHSDigital-Patient
211 https://simplifier.net/guide/nhsdigital/home
212 https://simplifier.net/guide/digitalmedicines/home
213 https://simplifier.net/guide/nhsdigitalclinical/home
214 https://www.hl7.org/fhir/validation.html

- **Structure**: Check that all the content in the resource is described by the specification, and nothing extra is present
- **Cardinality**: Check that the cardinality of all properties is correct (min & max)
- **Value Domains**: Check that the values of all properties conform to the rules for the specified types (including checking that enumerated codes are valid)
- **Coding/CodeableConcept bindings**: Check that codes/displays provided in the Coding[215]/ CodeableConcept[216] types are valid
- **Invariants**: Check that the invariants[217] (co-occurrence rules, etc.) have been followed correctly
- **Profiles**: Check that any rules in profiles have been followed[218] (including those listed in the Resource.meta.profile[219], or in CapabilityStatement[220], or in an ImplementationGuide[221], or otherwise required by context)
- **Questionnaires**: Check that a QuestionnaireResponse[222] is valid against its matching Questionnaire[223]
- **Business Rules**: Business rules are made outside the specification, such as checking for duplicates, checking that references resolve, checking that a user is authorized to do what they want to do, etc.

7.14.4.4.2.3 Consuming FHIR Resources from an API

As FHIR is typically in JSON or XML format, support is included in the majority of development languages.

Many languages have packages which will have classes and client libs to provide extra support (note: these will be based on the core FHIR schema not any localisation).

Language	Name	Url
java and kotlin	HAPI FHIR	https://hapifhir.io/
c#	firely-net-sdk	https://github.com/FirelyTeam/firely-net-sdk https://www.nuget.org/packages/Hl7.Fhir
typescript and javascript	fhir - npm	https://www.npmjs.com/package/fhir

215 https://www.hl7.org/fhir/datatypes.html#Coding
216 https://www.hl7.org/fhir/datatypes.html#CodeableConcept
217 https://www.hl7.org/fhir/conformance-rules.html#constraints
218 https://www.hl7.org/fhir/profiling.html
219 https://www.hl7.org/fhir/resource.html#Meta
220 https://www.hl7.org/fhir/profiling.html#profile-uses
221 https://www.hl7.org/fhir/implementationguide.html
222 https://www.hl7.org/fhir/questionnaireresponse.html
223 https://www.hl7.org/fhir/questionnaire.html

Language	Name	Url
swift	Swift-FHIR	https://github.com/smart-on-fhir/Swift-FHIR
python		See https://confluence.hl7.org/pages/viewpage.action?pageId=35718838

7.14.4.4.2.4 Producing HL7 FHIR

It is strongly recommended to use a language specific library to produce the FHIR resources.

RESTful API - Search Parameters

Each resource has a defined list of search parameters e.g. for patient this is Search Parameters[224]. Documentation on searching can be found here https://www.hl7.org/fhir/search.html

This guidance is recommended, not mandatory

These lists can be quite large and API producers are encouraged to support the most common queries for each resource. We do not have UK R4 guidance on this but the STU3 CareConnect guidance would apply with only minimal changes. This can be found here https://nhsconnect.github.io/CareConnectAPI/explore.html and an extract for Patient is shown below:

224 https://simplifier.net/guide/nhsdigitalclinical/home

Name	Type	Description	Conformance	Path
address-postalcode	string	A postalCode specified in an address	MAY	Patient.address.postalCode
birthdate	date	The patient's date of birth	SHALL	Patient.birthDate
email	token	A value in an email contact	MAY	Patient.telecom (system=email)
family	string	A portion of the family name of the patient	SHALL	Patient.name.family
gender	token	Gender of the patient	SHALL	Patient.gender
given	string	A portion of the given name of the patient	SHALL	Patient.name.given
identifier	token	A patient identifier (NHS Number, Hospital Number, etc)	SHALL	Patient.identifier
name	string	A portion of either family or given name of the patient	SHALL	Patient.name
phone	token	A value in a phone contact	MAY	Patient.telecom(system=phone)

Client systems SHALL provide at least two parameters of differing types, unless searching on identifier where one parameter is permitted. Systems SHALL support the following search combinations:

- name + gender
- name + birthdate
- family + gender
- given + gender

It is expected UKCore will provide a list of recommended search parameters for all NHS restful API's. This is aligned with US Core which can be found here http://www.hl7.org/fhir/us/core/searchparameters.html

The main difference between US and UK will often be around the support for searching by NHS Number (patient:identifier)

Messaging

This guidance is recommended, not mandatory

Producers supporting messaging will typically support one or a combination of transports including

- http
- MESH

http producers should support the **$process-message**[225] endpoint. ← FHIR Policy is this a **MUST**?

225 https://www.hl7.org/fhir/messaging.html#process

Messaging is a well established interaction style within the wider NHS, it follows clear patterns and HL7 v2 Messaging is the predominant standard. It follows that NHS trusts and suppliers migrating to FHIR will be converting from HL7v2, not HL7v3 as is often the case within NHS Digital.

It is recommended FHIR R4 messaging leans on NHS HL7v2 rather than NHS Digital HL7v3 / FHIR STU3. ← *Needs discussing in FHIR Policy*

Message metadata profile can be found here https://simplifier.net/guide/nhsdigital/NHSDigital-MessageHeader

NHSDigital-MessageHeader has been profiled as a union of prior STU3 from
- **NEMS (pub/sub)**
- **ITK3 / Transfer of Care (Message channel)**

Plus 'point to point channel' EPS

It is not designed for any transport, so can be used in MESH or plain http.

It is recommend each message is defined in a FHIR MessageDefinition (← Needs discussing in FHIR Policy). EPS has several examples which can be found https://simplifier.net/guide/DigitalMedicines/MessageDefinition

Within EPS the MessageDefinition is also used to control the behaviour of FHIR Validation.

Documents

No API is currently using this in FHIR R4. It is anticipated that this would decouple the transport and documentation standards into

- messaging (see previous section)
- documents

This will avoid coupling documents to individual programs (e.g. ITK3/Transfer of Care and Digital Medicines, which were all flows to GP's) and be aimed as a generic NHS standard.

7.14.4.4.3 HL7 in NHS Digital and wider NHS

FHIR or Fast Healthcare Interoperable Resources is the latest standard from HL7. HL7 in use within the NHS includes:

HL7 Standard	NHS Usage	Interaction Style	Notes

v2	NHS Trusts	TCP/IP pipe and hat **messages**	'pipe and hat', similar to CSV but with more rules. The separators are '|' - pipe and '^' - hat, e.g.
			`PID|1||3333333333^^^NHS||SMITH^FREDRICA^J^^MRS^^L|`
			`SCHMIDT^HELGAR^Y|196512131515|2|||29`
			`WEST AVENUE^BURYTHORPE^MALTON^NORTH YORKSHIRE^YO32 5TT^GBR^H|`
			`|+441234567890||EN|M|C22||||A|Berlin|||GBR||DEU`
			
			Although format of payload is **message** they are often focused have **resource** focus. E.g. the ADT_A31 patient update **event** focuses on the PID/PD1 (FHIR Patient) and have CRUD properties (ADT_A28 is the create and ADT_A31 is the update)

HL7 Standard	NHS Usage	Interaction Style	Notes
v3	mostly NHS Digital	SOAP/XML **messages**	XML
CDA	mostly NHS Digital	SOAP/XML **documents**	A specialisation of HL7 v3

HL7 Standard	**NHS Usage**	**InteractionStyle**	**Notes**
FHIR	NHS Digital and Trusts	Restful/http **resource** json	json (R4) and xml (Stu3) Although FHIR is resource focused, it is quite common especially in NHSD STU3 to build message/document interactions. In R4 (APIM) we seeing far more RESTful resource API's and resource focused event messages (like Hl7 v2 not v3). In addition we are seeing a decoupling in FHIR from SDS to ODS(/NHS Data Dictionary). Security, auditing and technical metadata has moved into Apigee. **A significant refactor?**

FHIR provides an alternative to document-centric approaches by directly exposing discrete data elements as services (These were common in both HL7v3 and CDA as NHS Digital ITK Standards)

One of its goals is to facilitate interoperability between legacy health care systems, to make it easy to provide health care information to health care providers and individuals on a wide variety of devices from computers to tablets to cell phones, and to allow third-party application developers to provide medical applications which can be easily integrated into existing systems.

FHIR can support the message[226] and document[227] styles of the previous standards but it is more common for FHIR implementations to use *restful*[228]. (it is recommended to follow restful unless the business process or technical considerations indicate messaging or document are more appropriate).

7.14.4.4.3.1 RESTful Resource API's, Event Messages, Documents and Operations

HL7 FHIR has mainly been driven by RESTful Resource API interactions (**synchronous** FHIR RESTful API[229]) but it also supports traditional messaging (asynchronous FHIR Messaging[230]) and a concept called clinical document architecture (known in FHIR as FHIR Documents[231], in HL7v3 this was called CDA). Clinical documents allow the mix of unstructured (html) and structured information, in NHS Digital this is normally used in conjunction with FHIR Messaging and MESH.

Also FHIR has the ability to support remote procedure calls, this is called FHIR Operations[232].

We (NHS Digital) currently have a mix of all four interaction action styles and from a consumer point of view it's possible to encounter a mix on the same workflow (e.g. Immunizations appears in three of the columns below and COVID Vaccination feeds use CSV (which does follow a FHIR data model)). This is not ideal.

FHIR RESTful	FHIR Messaging	FHIR Documents	FHIR Operations
Appointment Booking (STU3) PDS (R4) COVID Vaccination History (R4) COVID Test Results (R4) National Record Locator Service (STU3) ODS API (STU3)	NEMS (PDS Events, Digital Child Health, Immunisations) Transfer Of Care (STU3) Digital Medicines STU3 (Immunisations and Emergency Supply) EPS (R4)	Transfer Of Care (STU3) Digital Medicines STU3 (Immunisations and Emergency Supply)	GP Connect (STU3)

Query (FHIR RESTful - synchronous)

The RESTful queries FHIR supports is extensive. In order to establish a common base line standard NHS Digital and INTEROPen created the CareConnectAPI (STU3)[233] this was based on the USofA Argonaut STU3 standard. Currently UKCore has no interaction standards but it is recommended guidelines from the US Core[234] is followed (NHSDigital focused recommendations can found in CapabilityStatements in the NHS

226 https://www.hl7.org/fhir/messaging.html
227 https://www.hl7.org/fhir/documents.html
228 https://www.hl7.org/fhir/http.html
229 https://www.hl7.org/fhir/http.html
230 https://www.hl7.org/fhir/messaging.html
231 https://www.hl7.org/fhir/documents.html
232 https://www.hl7.org/fhir/operations.html
233 https://nhsconnect.github.io/CareConnectAPI/explore.html
234 https://www.hl7.org/fhir/us/core/searchparameters.html

Digital Implementation Guides e.g. https://simplifier.net/guide/NHSDigital/CapabilityStatement but these are not mandated or yet recommended)

Event (FHIR Message - asynchronous)

In FHIR STU3 we have several standards for events the two main ones being NEMS (pub/sub) and TransferOfCare (message channel) but the message metadata rules are different, so an event of TOC will not work on MESH.

FHIR R4 EPS (point to point) has created a message data standard which incorporates the rules from both NEMS and TOC https://simplifier.net/guide/NHSDigital/NHSDigital-MessageHeader it is strongly recommended this is followed for all R4 events.

7.14.4.4.3.2 Differences between NHS Digital and the wider NHS

The main difference is the NHS does not focus as much on specific programmes and will focus on specific areas.

Event Messages (asynchronous)	Queries (asynchronous)	Documents (mostly unstructured)
HL7v2 (especially ADT and pathology)	HL7 FHIR RESTful	IHE XDS (mostly at LHCRE/ regional level)
FHIR Message - unknown. Probably medications (EPS and EPMA)		

The main overlap between NHS and NHS Digital is on queries (FHIR RESTful). For events, the NHS didn't move to HL7v3 like NHS Digital

7.14.4.4.4 CareConnect FHIR STU3 to UKCore FHIR R4(+) Conversion

7.14.4.4.4.1 Structural Conversion

This page does not describe the physical differences or conversion between core FHIR STU3 and R4 resources, it does describe the differences between Care Connect and UK Core.

For information on conversion between STU3 and R4 see https://www.hl7.org/fhir/diff.html and https://www.hl7.org/fhir/r3maps.html and . Developers are encouraged to make use of code libraries in particular the version converters in HAPI FHIR, see example below.

```
VersionConvertor_30_40 convertor = new VersionConvertor_30_40();

MedicationRequest r4medicationRequest = (MedicationRequest) convertor.convert
Resource(stu3medicationRequest,true);
```

This page assumes this conversion has taken place. The code and structure conversions only apply to UK Codes and Extensions, the HAPI FHIR also covers core coding changes..

7.14.4.4.4.2 Conversion Types

Coding Changes: between versions (CodeSystems and Codes)

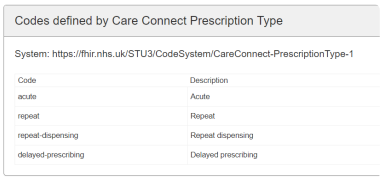
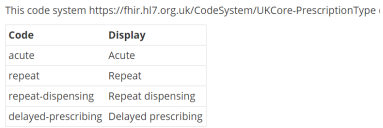
UKCore and CareConnect use different CodeSystems, in many cases this is just a simple renaming of the CodeSystem.

> **Warning: Although in many cases the CodeSystems uri has just changed, it is still technically a different CodeSystem and may contain differences.**

For example

https://fhir.nhs.uk/STU3/CodeSystem/CareConnect-PrescriptionType-1 is the same as

https://fhir.hl7.org.uk/CodeSystem/UKCore-PrescriptionType[235]

CodeSystem	
CareConnect-PrescriptionType-1	
UKCore-PrescriptionType-1	

There is no difference between the contents of both CodeSystems, they are technically different but for practical purposes they are the same.

The convention used in FHIR R4 is purposely FHIR version independent to prevent this problem in future FHIR versions. I.e. CodeSystems will not vary between FHIR versions in future.

Extensions

235 https://simplifier.net/ukcore-v2/ukcore-prescriptiontype

This is similar to the issue described above with CodeSystems and for example to convert the extension https://fhir.nhs.uk/STU3/StructureDefinition/Extension-CareConnect-GPC-PrescriptionType-1 or https://fhir.hl7.org.uk/STU3/StructureDefinition/Extension-CareConnect-PrescriptionType-1 this becomes :

https://fhir.hl7.org.uk/StructureDefinition/Extension-UKCore-PrescriptionType[236]

An *on the wire* example would look like:

Extension	

Extension-CareConnect-PrescriptionType-1[237]
Extension-CareConnect-GPC-PrescriptionType-1[238]
	"extension": [
	{
	"url": "https://fhir.nhs.uk/STU3/StructureDefinition/Extension-CareConnect-GPC-PrescriptionType-1",
	"valueCodeableConcept": {
	"coding": [
	{
	"code": "acute",
	"display": "Acute",
	"system": "https://fhir.nhs.uk/STU3/CodeSystem/CareConnect-PrescriptionType-1"
	}
	]
	}
	}
	]

Extension-UKCore-PrescriptionType[239]	"extension": [
	{
	"url": "https://fhir.hl7.org.uk/StructureDefinition/Extension-UKCore-PrescriptionType",
	"valueCodeableConcept": {
	"coding": [
	{
	"code": "acute",
	"display": "Acute",
	"system": "https://fhir.hl7.org.uk/CodeSystem/UKCore-PrescriptionType"
	}
	]
	}
	}
	]

236 https://simplifier.net/guide/UKCoreDevelopment2/ExtensionUKCore-PrescriptionType
237 https://fhir.hl7.org.uk/STU3/StructureDefinition/Extension-CareConnect-PrescriptionType-1
238 https://fhir.nhs.uk/STU3/StructureDefinition/Extension-CareConnect-GPC-PrescriptionType-1
239 https://simplifier.net/guide/UKCoreDevelopment2/ExtensionUKCore-PrescriptionType

Note in the example above the url has been transformed and the CodeSystem has also been changed (as a result of CodeSystem change described in the previous section)

Structural Changes: UK Extensions to core FHIR Changes

Some extensions may be adopted as core FHIR in subsequent versions of FHIR. For example the extension in CareConnect STU3 Extension-CareConnect-PrescriptionType-1[240] and Extension-CareConnect-GPC-PrescriptionType-1[241]

is now courseOfTherapyType[242] with a ValueSet of

> http://hl7.org/fhir/ValueSet/medicationrequest-course-of-therapy

which contains the follow codes.

All codes from system `http://terminology.hl7.org/CodeSystem/medicationrequest-course-of-therapy`

Code	Display	Definition
continuous	Continuous long term therapy	A medication which is expected to be continued beyond the present order and which the patient should be assumed to be taking unless explicitly stopped.
acute	Short course (acute) therapy	A medication which the patient is only expected to consume for the duration of the current order and which is not expected to be renewed.
seasonal	Seasonal	A medication which is expected to be used on a part time basis at certain times of the year

From a practical point of view. this means the extension in UKCore (or CareConnect) will not be recognised. The extension would need to be removed and the code added to courseOfTherapyType[243] e.g.

```
"courseOfTherapyType": {
  "coding":  [
      {
          "system": "http://terminology.hl7.org/CodeSystem/medicationrequest-
course-of-therapy",
          "code": "acute",
          "display": "Short course (acute) therapy"
      }
  ]
},
```

Note that although the **code** is the same the **system** and **display** have changed. We have also done a code conversion which is described in the next section.

240 https://fhir.hl7.org.uk/STU3/StructureDefinition/Extension-CareConnect-PrescriptionType-1
241 https://fhir.nhs.uk/STU3/StructureDefinition/Extension-CareConnect-GPC-PrescriptionType-1
242 https://www.hl7.org/fhir/medicationrequest-definitions.html#MedicationRequest.courseOfTherapyType
243 https://www.hl7.org/fhir/medicationrequest-definitions.html#MedicationRequest.courseOfTherapyType

Code Conversion

Some of the changes will not be as simple as changing uri's (described above), some will involve code conversions. As as been shown in the previous sections, **prescription type** codes are in R4 called **medicationrequest-course-of-therapy** codes

The conversion is

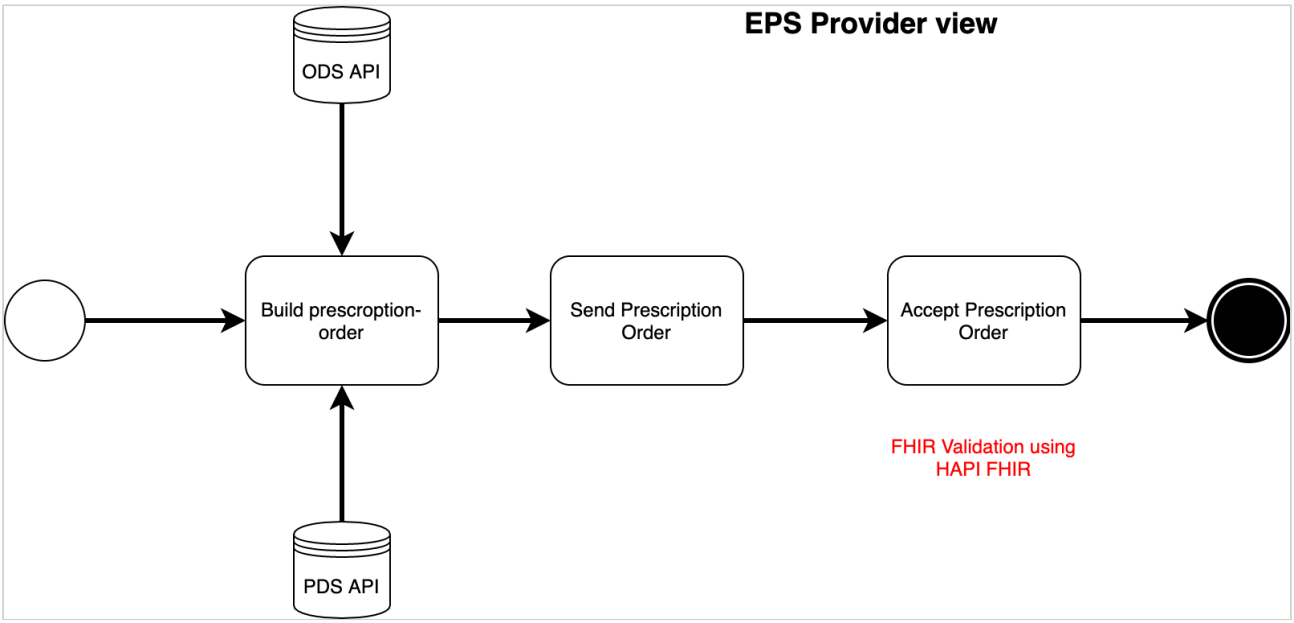
STU3 CodeSystem	code	display	R4 CodeSystem	code	display
https://fhir.nhs.uk/STU3/CodeSystem/CareConnect-PrescriptionType-1	acute	Acute	http://terminology.hl7.org/CodeSystem/medicationrequest-course-of-therapy	acute	Short course (acute) therapy
	repeat	Repeat		continuous	Continuous long term therapy
	delayed-prescribing	Delayed prescribing	(no conversion - not used in R4 EPS or EPMA)		
	repeat-dispensing	Repeat dispensing	https://fhir.nhs.uk/CodeSystem/medicationrequest-course-of-therapy	continuous-repeat-dispensing	Continuous long term (repeat dispensing)

This ConceptMap is defined in a FHIR **ConceptMap** resource - https://simplifier.net/guide/DigitalMedicines/UKCore-PrescriptionTypetoR4PrescriptionTherapyTypeMap

(Note the rendering in this link is incorrect, please refer to the JSON or XML examples for a cleaner description).

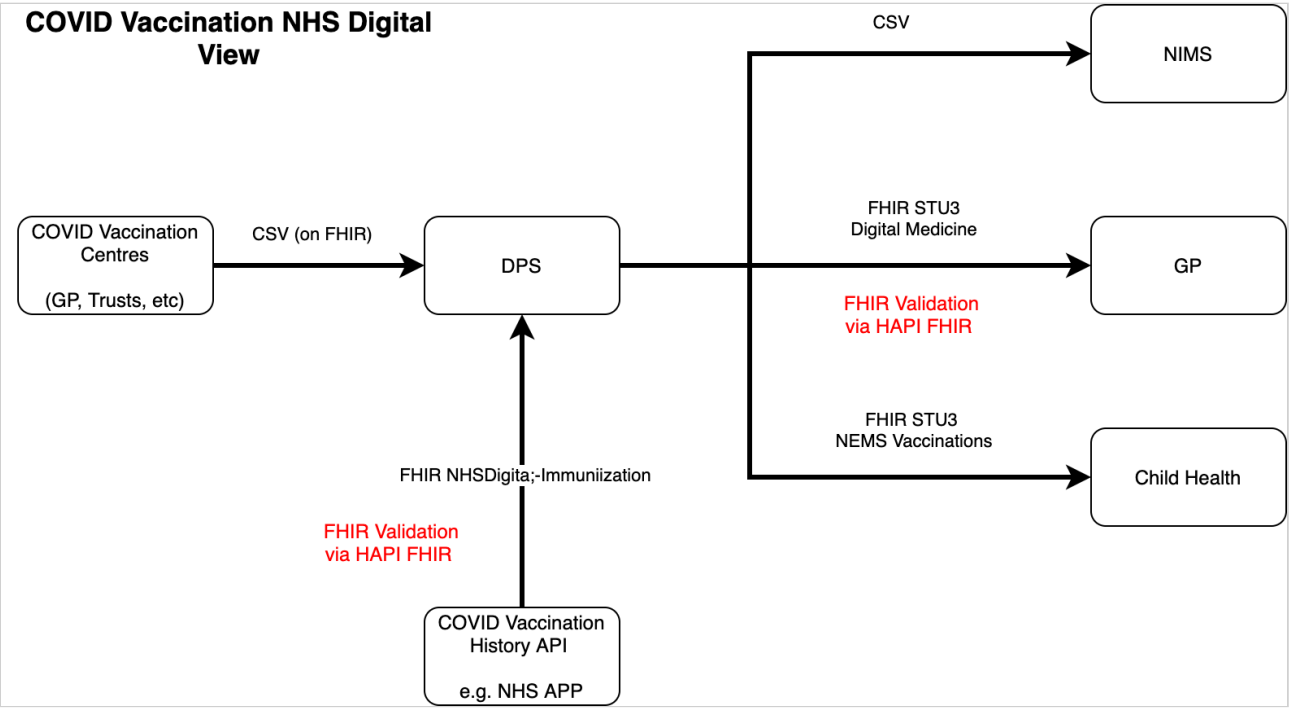
7.14.4.4.5 FHIR Testing (Validation)

7.14.4.4.5.1 Current Situation



EPS recommends assembling the prescription-order, PDS API is a source of the Patient resource and ODS API is source of Organization resources.
Both currently need conversion from FHIR STU3 or R4 UKCore1 to NHS Digital FHIR R4 definition.

We are starting to see issues reported caused by EPS Validation being more thorough and FHIR compliant than source systems which the *consumer is having to correc*t.



All providers are sending in daily vaccinations records to DPS via CSV, this aligns with the data/business model used by EPS and the data model of UKCore-Immunization (and also CareConnect-Immunization). The FHIR Validation used by Vaccination History API follows the same rules.

Internal Developer feedback:

It is very difficult to find a FHIR Validator compatible with FHIR STU3 Digital Medicines

We have some data quality issues on the CSV feed, these don't appear to be huge. Main concern at various stages is the quality of SNOMED coding which is difficult to verify.

Issues in other areas:

1. Care Connect Reference Implementation (CCRI) - Examples in other API specifications would be rejected by the server. Cause - Examples were validated against with weaker FHIR coverage.
2. GP Connect - The specification had a number of FHIR non conformances and errors in examples. GP Suppliers would notice errors in the specification when they used FHIR Validation (FHIR validation was not possible at times due to errors in the conformance resources)

7.14.4.4.5.2 FHIR Validation

FHIR API's are currently tested using a variety of methods which tend to look at business rules, structure and cardinality of the FHIR resource payloads.

FHIR Validation tools reduce the amount of testing code that needs to be written to ensure FHIR conformance (this approach has been followed by Solutions Assurance FHIR STU3 and EPS FHIR R4)

FHIR Validation covers the following (from https://www.hl7.org/fhir/validation.htm[244]

7.5 Validating Resources

FHIR Infrastructure Work Group	Maturity Level: N/A	Standards Status: Informative

This page provides an overview of how the FHIR specification supports validation of resources.

Validating a resource means, checking that the following aspects of the resource are valid:

- **Structure**: Check that all the content in the resource is described by the specification, and nothing extra is present
- **Cardinality**: Check that the cardinality of all properties is correct (min & max)
- **Value Domains**: Check that the values of all properties conform to the rules for the specified types (including checking that enumerated codes are valid)
- **Coding/CodeableConcept bindings**: Check that codes/displays provided in the Coding/CodeableConcept types are valid
- **Invariants**: Check that the invariants (co-occurrence rules, etc.) have been followed correctly
- **Profiles**: Check that any rules in profiles have been followed (including those listed in the Resource.meta.profile, or in CapabilityStatement, or in an ImplementationGuide, or otherwise required by context)
- **Questionnaires**: Check that a QuestionnaireResponse is valid against its matching Questionnaire
- **Business Rules**: Business rules are made outside the specification, such as checking for duplicates, checking that references resolve, checking that a user is authorized to do what they want to do, etc.

There are multiple ways to validate resources. This table summarizes the options described in this specification, and which of the aspects above they can validate:

Method	XML	JSON	RDF	Structure	Cardinality	Values	Bindings	Invariants	Profiles	Questionnaires	Business Rules
XML Schema	✓			✓	✓	✓					
XML Schema + Schematron	✓			✓	✓	✓		✓	✓[1]		
JSON Schema		✓		✓	✓	✓			✓[2]		
ShEx			✓	✓	✓	✓	✓[3]				
Validator	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Validation Operation[4]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	

7.14.4.4.5.3 FHIR Validation within NHS Digital

FHIR Validation is part of several API's and Testing Solutions

API / Testing	FHIR Version	Type	Software	Notes
Solutions Assurance	STU3	Validation Operation	HAPI FHIR (4.x.x)	HAPI is extended to perform FHIR Message validation
EPS	R4 (capable of STU3)	Validation Operation	HAPI FHIR (5.1.0)	HAPI is extended to perform FHIR Message validation, this is an enhancement to the SA message validation
COVID API's (IOPS)	R4	Validation Operation + Validation	HAPI FHIR (5.3.0) and HL7 Command Line	We will be able to use FHIR Validation to verify compatibility with EU profiles (https://build.fhir.org/ig/hl7-eu/dgc/index.html)

244 https://www.hl7.org/fhir/validation.html

API / Testing	FHIR Version	Type	Software	Notes
IOPS	R4 and STU3	Validation Operation	Vonk/Simiplifier	
CareConnect Reference Implementation	STU3	CI Build (like Validation Operation)	HAPI FHIR (4.x.x)	
Consumers	STU3/R4	Mixed	Mostly HAPI FHIR	

Majority now use Implementation Guides NPM Packages as test packs. For example EPS uses UKCore2 (uk.core.r4.v2[245]), NHSDigital IG (**uk.nhsdigital.r4**[246]) and NHSDigitalMedicines IG (uk.nhsdigital.medicines.r4).

Note HL7 Command Line and HAPI FHIR share a common code base.

We do have a number of issues which tend to apply to all of these solutions.

Exceptions	Notes
Coding/CodeableConcept bindings	This requires a UK FHIR Terminology Server - this is currently going live
Invariants	The profiles can contain constraints,

To help offset these issues it is proposed that

All APIM FHIR API's MUST perform FHIR Validation as defined in https://www.hl7.org/fhir/validation.html **during development (not production)**

No method of Validation will be mandated, the recommendation will be to use the command line FHIR Validator, the ask a server (e.g. EPS https://github.com/NHSDigital/validation-service-fhir-r4), Solutions Assurance TKW Validator (STU3 only) or Simplifier https://fire.ly/products/simplifier-net/

To help this rule being enforced a number of exclusions are recommended (these should be reviewed frequently with the aim of removing them to **increase the quality of testing**):

245 https://simplifier.net/packages/uk.core.r4.v2/2.0.8
246 https://simplifier.net/packages/uk.nhsdigital.r4/2.0.14-dev

Exclusion	Notes
Coding/CodeableConcept bindings	Requires the use of a Terminology Server. The new NHS Digital server has security features which make this difficult. Introduction of terminology validation is likely to be a breaking change due to increased level of testing.
Invariants	Only a few of the validation products support this and has some complexity issues (the constraints are often expressed as FHIRPaths (like JSONPath))
Questionnaires	Not used in APIM at present. Need to check coverage in validation tools
Business Rules	To be handled via normal testing.
MessageDefinitions	Is not part of HL7 Validation tools. It is present in Solutions Assurance and EPS Validation tools. ***Is this an exclusion?***

A corollary is

IOPS will provide all FHIR Conformance material as a standard FHIR FHIR IG NPM Package

(This is already agreed by IOPS and Solutions Assurance)

Note Matt Mercer has had some uses using TKW with Digital Medicines. Suggest IOPS follow up with Solutions Assurance and the new packages.

7.14.4.4.5.4 FHIR Conformance within APIM

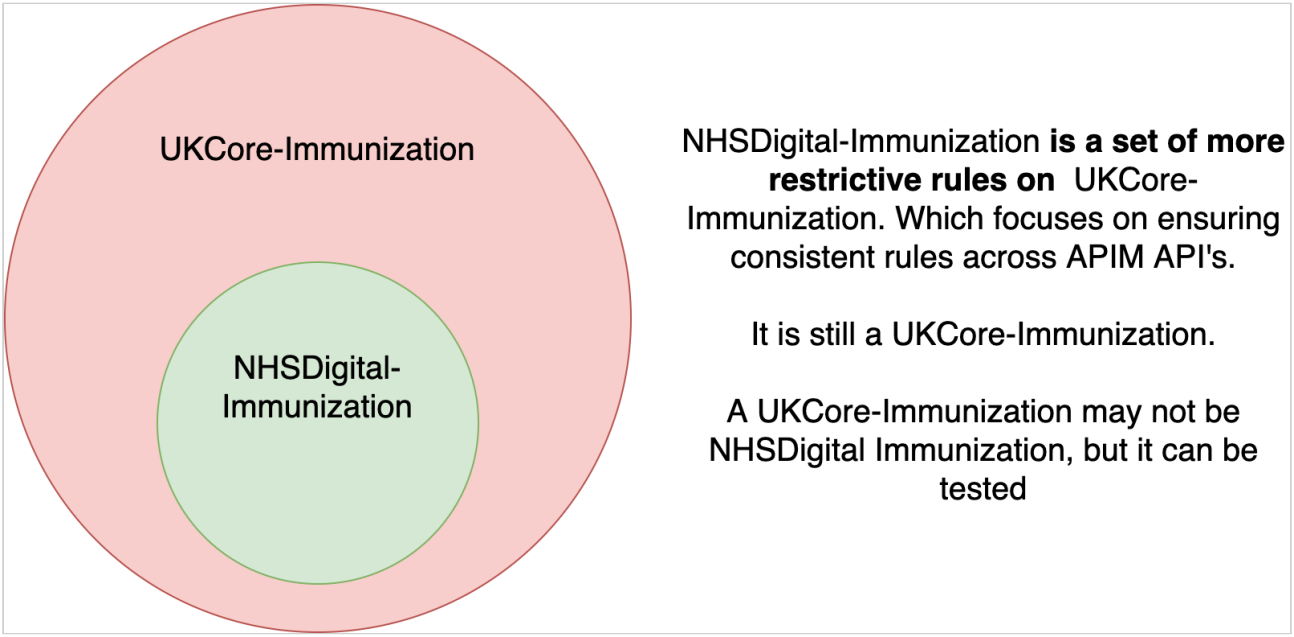
To ensure consistency across APIM API's, IOPS have started a NHS Digital FHIR Implementation Guide. This tends to add both technical and business rules to the core UKCore profiles, UKCore tends to focus on clinical rules. These profiles are still UKCore conformant.

Typical changes include forcing/encouraging the use of reference identifiers (NHS Numbers, ODS Codes, professional codes) to make lookups easier for consumers (and creation of FHIR Messages easier). Some of these changes may flow into UKCore.

Recommended is

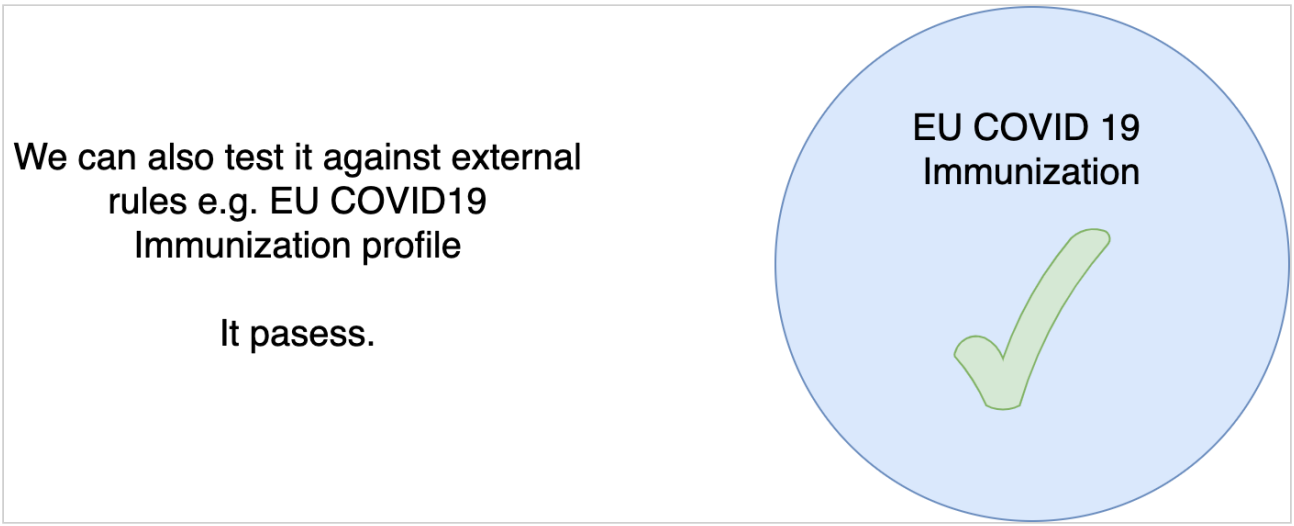
All APIM APIs MUST/SHOULD conform to the NHS Digital FHIR Implementation Guide[247]

247 https://simplifier.net/guide/nhsdigital/home



We have some concerns on the interpretation of profiles especially around schemas. Each FHIR version has only one set of schemas, UKCore and NHSDigital IG do not have schemas they are a series of rules and constraints in addition to the core schemas. E.g. UKCore-Immunization and NHSDigital-Immunization can be tested against EU-COVID19 Immunization profiles (at present it appears they do pass. The COVID Vaccination History API is currently for domestic use but it is very likely going to be used internationally, in particular the EU Vaccination Certificates https://chat.fhir.org/#narrow/stream/242580-europe/topic/vaccination.20certificates)

Do we need to be more careful of labelling schema fragments as schema?



7.14.4.4.5.5 Other Issues

HAPI FHIR is used to provide Validation support in several places and would be ideal candidate to be used as a central service. We are also seeing it used to provide different types of testing: unit (as in care connect and command line). integration (EPS) and acceptance testing (Solution Assurance). We should coordinate these

better to ensure developments to support UK/NHS Digital aspects such as security interceptors to work with NHS Digital Ontology Servers or UK specific business rules classes are shared. It's also likely these libraries would be useful outside of NHS Digital. A shared code base may be more practical than a common service (especially as each service may have a different focus e.g. APIM may want to check structure, cardinality and business rules, IOPS may wish to concentrate on coding (e.g. SNOMED).

UKCore profiles have a clinical+terminology focus. This is desirable as supporting clinical/patient care is our primary role but this does lead into issues. A common problem is the over complication of models which results in complex API's or difficulties in supporting as the model does not reflect practical use. **How do we manage this?** An area of current concern is the international/EU use of target disease and dose number - this probably reflects practical use in the UK and the UKCore suggestion of combining these into a single valueset https://simplifier.net/nhsdigitalspine/nhsdigital-vaccinationprocedure is not easy to use from either provider or consumer point of view. **We are likely to be the focal point of this feedback.**

7.14.4.4.6 FHIR Servers

7.14.4.4.6.1 Building a FHIR Server API

Consider using FHIR Server frameworks

Language	
java / kotlin	HAPI FHIR[248]
c#	Firely FHIR Facade[249]
typescript	AWS FHIR Works[250]

7.14.4.4.6.2 HAPI FHIR Restful Server

HAPI FHIR is an open source project which is known for its ability to quickly provide a FHIR Server with backend database (Test server http://hapi.fhir.org/).

It can be used without the server to provide a **plain server** where HAPI FHIR will take care of:

- HTTP Processing
- Parsing / Serialization
- FHIR REST semantics

248 https://hapifhir.io/
249 https://fire.ly/products/fhir-facade/
250 https://aws.amazon.com/about-aws/whats-new/2020/12/introducing-fhir-works-on-aws/

The REST behaviour can be enhanced/configured with interceptors which can control security, CSRF, etc.

In NHS Digital we have four projects which have used HAPI FHIR.

Project	Language	Description	Git
EPS Validation (R4)	Kotlin	Tests incoming messages against FHIR conformance packages (also works with non EPS). Uses HAPI FHIR Validation libraries	https://github.com/NHSDigital/validation-service-fhir-r4
Care Connect Reference Implementation (STU3)	Java	A public facing reference implementation which is similar to the HAPI JPA server, using Hibernate to access patient data in a traditional RDBMS SQL database.	https://github.com/nhsconnect/careconnect-reference-implementation
Solution Assurance Testing (STU3)	Java	Tests incoming Transfer of Care messages against FHIR conformance packages (also works with non TOC). Uses HAPI FHIR Validation libraries	Not public
GP Connect FHIR facade (R4)	Java	A hack for INTEROpen hack event, was an update to a proof of concept by DWP Digital. Converted the existing STU3 RPC/Operation into a RESTful FHIR R4. Used HAPI FHIR to perform the initial conversion to R4 and then applies a java based transform.	https://github.com/project-wildfyre/gpconnect-adaptor

The structure of these projects was roughly.

HAPI Server	Transformation and/or Action
Mostly configuration such as property files, environment variables. Configuration of the REST endpoints was a limited amount of java/kotlin Security and other server behaviour required some development but with APIM this would be only need to be done once and exposed a libraries.	All the transforms made use of java libraries using kotlin/java. This is where the majority of development would take place. It is believed other (scripting) languages such as javascript or python can be used for this purpose. This was the approach used n the popular Mirth Connect (https://www.nextgen.com/products-and-services/integration-engine) with HL7v2 interfaces, this is based on HAPI HL7v2 (and FHIR) using javascript for the transformations. (Would the HAPI FHIR classes be available in these languages - that would make working with FHIR structures easier)

Pro/Con

Main con is the use of JVM and kotlin/java languages. It is believe this can be reduced to configuration and minimal java/kotlin coding (if the transforms can be in other languages). It should produce a large time saving as both API (inc Apigee) and FHIR behaviour is pre built.

7.14.4.4.6.3 AWS FHIR Works

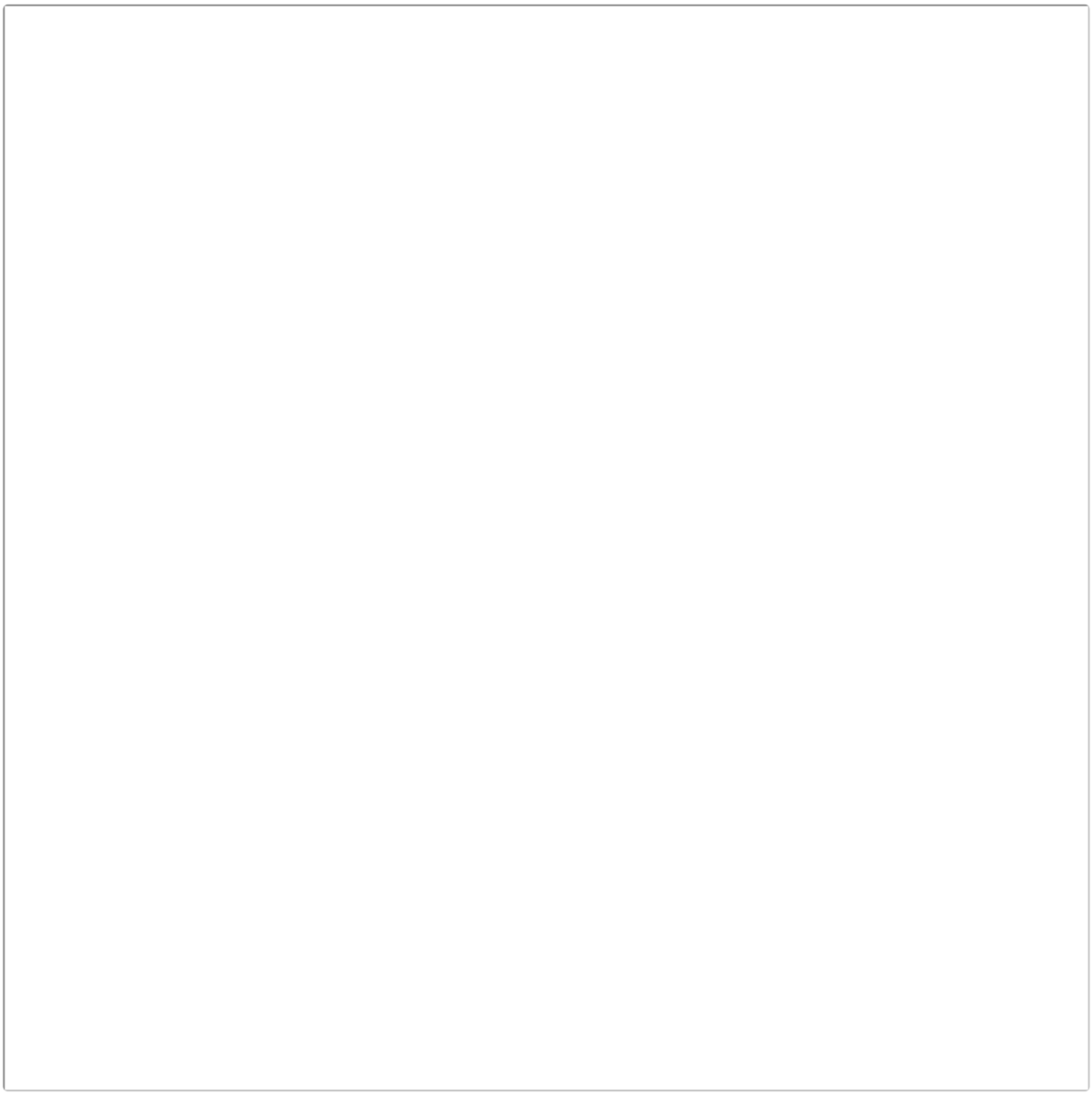
Believe this is capable of acting as a facade but the current documentation is focused on a building a server with DynamoDb and ElasticSearch storage.

7.14.4.4.6.4 Firely FHIR Facade

May be limited to .Net environments. The transformation engine may be useful, this is a new feature which may make transformations configuration rather than code based.

7.14.4.4.6.5 Traditional Health Middleware (aka Trust Integration Engine)

The two main TIE's are from Intersystems and Orion. These are focused on transforming HL7v2 messages (i.e. traditional messaging) not RESTful resources. They are designed to be run by less technical staff with particular emphasis on monitoring and configuration. Transforms can be built with UI, e.g.



Both have support for FHIR and is likely the would make **Events** easier to implement and support.

7.14.4.5 API naming and grouping endpoints

7.14.4.5.1 APIs versus endpoints

At NHS Digital we use the following language:

- an **endpoint** is an individual callable function, with a URL and an HTTP method
    - for example, "Get patient details" - GET https://api.service.nhs.uk/personal-demographics/FHIR/R4/Patient/{NHS number}
- an **API** is a group of related endpoints that provide a coherent overall service

    - for example, "Personal Demographics Service FHIR API"

The API catalogue[251] is a catalogue of APIs. The individual endpoints for each API are listed in the API specification, for example the PDS FHIR API specification[252].

At a technical level, we tend to have **one proxy per API** - all the endpoints for a given API go through the same proxy.

7.14.4.5.2 API naming

Each of your "named APIs" needs two names:

- A human-readable name
    - for example "Summary Care Record - FHIR API"
- A physical name that appears in its callable URI
    - for example "summary-care-record"

Guidelines for choosing a name:

- Use the "Ronseal" approach - choose a name describes what it is, or does
- Try to use language that will make sense to external developers (who don't know NHS Digital terminology)
- All APIs will be under a single domain (api.service.nhs.uk[253]), so your name should be unique and unambiguous
- Avoid words such as "service" or "data" in the name
- This name should be made up of lowercase letters and hyphens only ("spine-case")
- Don't "brand" your API
- Don't name your API after the programme that's delivering it - it will exist long after your programme has been closed down
- Versioning for HL7 FHIR is part of the URL after you API Name - see policy FHIR-LAND-05 and 08 on the API Landscape[254]
- The guidance above is based on the Government Digital Service guidelines: https://www.gov.uk/service-manual/design/naming-your-service
- Are there key terms in your API that don't fit with other NHS Digital services?

Independent of the other naming instances above there is a "service name" for an API - it acts as a unique key for GitHub, Azure, AWS and Apigee. Currently this is immutable. Not ideal as during Alpha the API name might change regularly, so:

- Choose a "service name" that is concise
- Not too generic - trying to avoid namespace clashes

251 https://digital.nhs.uk/developer/api-catalogue
252 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
253 http://api.service.nhs.uk
254 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/123930121/API+Landscape

- Likely to survive multiple API name changes

7.14.4.5.3 API grouping of endpoints

How granular do you want API to be, how many endpoints will be grouped together under it?

In alpha you might start with one API, but you might need to split out the **presentation** of your endpoints into multiple "API"s. Each of these APIs (proxies) can route to the same API backend.

- What endpoints will this API expose?
  - Because REST is our default approach, we favour more endpoints (each with a narrower data scope) over fewer endpoints with multiple purposes
  - Data entities in your business domain would usually map onto API endpoint names (HTTP resources)
- How will those endpoints be grouped into named APIs?
  - Each named API should represent a cohesive scope
  - If there are many endpoints it might indicate that the scope isn't clear enough or small enough - so grouping endpoints into multiple APIs helps
  - The entry for your API, in the API catalogue (https://digital.nhs.uk/developer/api-catalogue), is something that is going to be referred to on a regular basis. Look at existing APIs in the catalogue could help you group your endpoints more intuitively

7.14.4.5.3.1 What is a proxy base path?

To actually create a proxy you must set the Apigee basepath, for instance this would be "immunisation-history" for https://api.service.nhs.uk/immunisation-history/. To look at this more closely see Proxy base path vs API base path (see page 364).

7.14.4.5.3.2 Grouping multiple APIs under a higher level concept / name

As the estate grows, there will be some API Producers that have a number of related APIs. Whilst APIM strongly discourages naming your API after a service or brand, occasionally it might be appropriate. The pattern here would become:

- https://{domain / host}/{**group api-name**}/{**api-name**}

For instance the new GP Connect APIs will look similar to the following:

- https://api.service.nhs.uk/gp-connect/permissions
- https://api.service.nhs.uk/gp-connect/appointments

The {**group api-name**} is "gp-connect", and it should be noted that the two examples will be implemented as two proxies (i.e. two repos) with the following base-paths:

- gp-connect/permissions
- gp-connect/appointments

7.14.4.6 Consistency between APIs

There are many aspects of making integration easier for both API Consumers and API Producers - however, one of the most visible to API Consumers is consistency between APIs. A simple example is if one API uses "surname" instead of "family" (which is the FHIR field name for last name) - an API Consumer who calls two separate APIs, they need to ensure their code takes that into account. From their perspective they are calling an "NHS" API, why isn't the NHS consistent?

Other pages in this Designing your API (see page 322) section cover many areas of consistency, this page re-covers some points - highlighting what API consumers will see / focus on.

Area	Consideration
API Name	Does it fit into the wider APIM Namespace / domain model? *For guidance on naming see:* API naming and grouping endpoints (see page 359)
Paths and endpoint names	RESTful HTTP design points (see page 365)
Terminology	Are there key terms in your API that don't fit with other NHS Digital services?
Headers	Headers carry information related to the session or security that is not related to the payload. Don't create new headers without checking the platform wide ones - HTTP headers for your API (see page 395) Remember, where you use a platform definition, do not "customise" it to your definition otherwise API Consumers will be confused and it could create edge case complications you are not expecting

Area	Consideration
Minimise data required	Every piece of information asked for adds complexity to any API, even if it seems trivial it requires more code, more error handling and more unit tests. The move to FHIR R4 is pushing a design pattern that only asks for unique identifiers, rather than an identifier and associated meta data. Designing your FHIR API (see page 324) Example areas: • An ODS code is already unique and immutable (relatively), why ask the API Consumer to lookup the name of the ODS code and send it back to the NHS (who happen to be the holders of the master values)? • NHS Digital now has a Terminology Server[255], available to both Consumers and Producers to use - which will allow easy lookup of coding systems such as SNOMED • If a payload requires (e.g. for legal reasons) a unique ID and some specific meta-data, consider how the API Consumer can get hold of it. Can they re-use the payload from another API that provide a lightweight way to provide this information?
Error Messages	Always use the platform common error codes in the first instance, but create your own to keep them unique and small in scope - see Handling errors (see page 624)
Session	Is there anything special about your session management? This comes out in very detailed design, but keeping an eye on it upfront can simplify the API design. There are multiple elements: • Is there any session configuration or data that is fixed for a period of time? • Is there any data that needs to be passed on every request? • If applicable, do the timeouts (end to end) align with CIS 2 or NHS login?
Onboarding	Whilst on-boarding new API Consumers is likely a long way off - there are API design decisions that could make onboarding harder or non-automatable. Some areas to keep in mind: • Do you have a concept of End User Organisations? (It might not be called this currently) • Have you thought about your grouping access into "Access Modes"? • Is your API making a call to Spine? See Spine based APIs (see page 408) for considerations

255 https://digital.nhs.uk/services/terminology-servers

Area	Consideration

Environments	Does the way you carry out assurance, onboarding and the resulting environments align with the API Management platform, and the very short Path To Live (PTL)?
	API Management based their environments on Spine Path To Live, but for external API Consumers we have focused on providing just the one PTL INT (integration). This is proving successful, and whilst a few APIs have multiple external PTL environments - this is not a generic requirement.
	Aspects to consider:
	• Do you need to use CIS2 or NHS login? APIM mock services will (should) cover all your initial scenarios and help support automated testing (for you and your API Consumers)
	  • What users do you need?
	  • To see how environments are connected see Environments (see page 758)
	• Common APIs that need to be called with your API - these all need to be connected to the same backends

Test Data	What is your approach to test data for External Developers? See Standard test data (see page 606)

There is a case that sometimes consistency can create more work for the API Consumer, when these situations arise it is worth asking questions on API Producer Support channel and/ or your IOPS contact.

7.14.4.7 Proxy base path vs API base path

As an API Producer you have an **important choice** about when your proxy takes control of incoming traffic, Apigee splits up a URL in the following way:

- https://{domain / host}/{**base-path**}/{**resource-path**}

Some examples in APIM (first two have no resource paths):

- https://api.service.nhs.uk/immunisation-history/
- https://api.service.nhs.uk/immunisation-history/FHIR/R4/
- https://api.service.nhs.uk/nhs-app/communication/notification/FHIR/R4/CommunicationRequest
- https://api.service.nhs.uk/nhs-app/communication/in-app/FHIR/R4/CommunicationRequest

The choice is how much of the URL is the base-path, as this must be unique and effectively defines your proxy. You can set the **Apigee basepath** as either **immunisation-history/** or **immunisation-history/FHIR/R4/**. As an API Producer you should set the proxy base path as that top level "API name". The slightly longer path immunisation-history/FHIR/R4/ is then your API or **FHIR base path**. for (potentially) a FHIR Server. There might, occasionally, be good reasons to have a longer base path.

Always keep your Apigee basepath as short as possible

By having your Apigee base-path set as just the name of your API, it gives you the following benefits:

- You see and manage all the traffic under that base path in your proxy config (the "boundary")
- In support situations, it is logical to trouble shoot any path under your "API name" (namespace)
    - For instance maybe an API consumer missing out the R in R4 sending: **immunisation-history/FHIR/4/**
    - You would **not** see this request in your API Splunk logs
- Future - will you need to support parallel versions in the future? You can route traffic based on the URL under that top level, rather than deploying a new API
- _ping & _status should be at the top level, especially if you have parallel versions, or have a multiple FHIR servers model, or multiple endpoints targeting different backends

It is important to note that, if the overall url doesn't change (i.e. you are just changing which segments make up the base path), then the proxy base path change is not a large task, and can be done transparently to the API Consumer.

7.14.4.7.1 Grouping proxies under a common path

See API naming and grouping endpoints (see page 359)

7.14.4.8 RESTful HTTP design points

The following table covers different aspects of the HTTP standards, and how to apply them in context of NHS Digital and the API Management Platform:

Paths	• Keep the number of levels to a minimum (excluding segments such as FHIR/R4), e.g. anything over 2 is not recommended • Are there terms that might be confusing, stand out from other APIs? • See policy APIM-LAND-05 on API Landscape[256] for details of requirements for FHIR API URL Paths • You can see more guidance here: API naming and grouping endpoints (see page 359)

256 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Landscape

Endpoint names	• Use nouns instead of verbs (RESTful pattern) • Are they intuitive? • Sometimes referred to as HTTP resources • For FHIR APIs, endpoint names must align with the FHIR resource model as described at https://www.hl7.org/fhir/http.htmland https://www.hl7.org/fhir/resourcelist.html (the endpoint name must be capitalised and singular e.g. "Patient") • For non-FHIR APIs use spinal-case and plural e.g. "patients" • Endpoints are nested to show relationships. See RESTful APIs good practice (see page 368)
Query Parameters	• Are these consistent with other API Platform query parameters? • FHIR defines key terms for each resource (e.g. see Patient resource parameters[257]) • For non-FHIR APIs, use names that match the names you use in your response payload. So, for example, if your response payload is camel case with fields like "birthDate" then use camel case query parameters. Where possible, use language that appears in FHIR in preference to creating new field names, but don't feel like you have to stick to their non-standard use of kebab-case. So, for example, instead of "death-date" (FHIR standard) you could use "deathDate" to match the field names in your response payload. • For existing APIs you might find there are inconsistencies. Mapping of these in Apigee is easy. However, what impacts are there on your existing users? Might you need to support both?
Headers	• Try not to add custom Headers at all • If they are added, they must not contain any data that should be in the payload • Are they consistent with other API Management Headers? • Should be namespaced with "NHSD-" (see RFC 6648[258] for further guidance, e.g. the "X-" prefix was deprecated in 2012) See HTTP headers for your API (see page 395) for detailed guidance
Field Names	Are the field names in the **payload** datasets the same as FHIR? Even for non-FHIR APIs, developers should not have to deal with different field names, for different APIs, for basic data such as first name and last name?
Verbs	Does your HTTP Verb usage align with RESTful principles? For FHIR APIs, does this also align with https://www.hl7.org/fhir/http.html

257 http://www.hl7.org/fhir/patient.html#search
258 https://tools.ietf.org/html/rfc6648

Update operations	PUT is for updating an entire resource PATCH is being used where there are small changes to a resource
Status Codes	• Do your HTTP Status Codes align with RESTful principles? • FHIR also influences the choice of which status codes to use
Datetime	Whilst the date and time format for values is a very detailed point, it can cause a lot of confusion and re-work as data values are sent as strings. Date and time field values must follow the ISO-8601[259] standard, allowing any standard 8601 parser to work correctly. However, if creating strings with date times / time stamps in, you must consider the following: • Use the format defined in RFC 3339[260], which is a profile of ISO-8601 • If the libraries in use don't provide an RFC 3339 option, check the FHIR Datetime datatype[261] definition to ensure compatibility • ISO-8601[262] not only aligns to FHIR, the Government also uses this in open data standards, for instance: Record information about data sets you share with others[263]
Content of the response for errors	• For FHIR APIs all errors should be returned in FHIR JSON format, this includes errors from Apigee. A pattern for this can be re-used - see API Producer Zone - Handling Errors (see page 624). • All other APIs will have a format relevant to that API (this includes the error message format, so some APIs might be XML and an XML payload content is appropriate) • Platform wide generic error format (for **non FHIR** instances) is to re-use the FHIR OperationOutcome as the JSON template - agreement: D040 - Non FHIR Platform error format[264]

259 https://en.wikipedia.org/wiki/ISO_8601
260 https://www.ietf.org/rfc/rfc3339.html
261 https://build.fhir.org/datatypes.html
262 https://en.wikipedia.org/wiki/ISO_8601
263 https://www.gov.uk/guidance/record-information-about-data-sets-you-share-with-others
264 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182981712/D040+-+Non+FHIR+Platform+error+format

4xx/ 5xx errors	These error codes **will always** be accompanied by a JSON payload in a format relevant to that API: • This aides self-diagnosis by third parties • FHIR APIs should follow this pattern anyway • Where there are security sensitive responses, consideration must be made around what is returned in the payload • Preferably including a unique reference (e.g. the Apigee request messageId) for contacting support
Backend Errors - should they be 4xx or 5xx?	In scenarios where the API Backend returns 4xx errors that are service and integration related **edge cases**, the API Producer should (in the Apigee Proxy), convert to a **503** with a suggested retry-after header. • The primary examples are where Sync Wrap timeout situations or rate limiting (**429**s) occur and the API consumer **has no** control • The retry-after header will generally be a short retry period (e.g. 10 seconds), but the API Producer might be aware of more complex scenarios and wish to suggest a longer period • You also might need to look at some types of **401** / **403** coming from the backend indicating that the authentication and authorisation between Apigee and the backend is blocked. This is always separate to the incoming security segment and would be out of the API Consumer control • Some scenarios might warrant a 504 (Gateway Timeout), rather than 503 (Service Unavailable). However, 504 does not officially support the Retry-After header ⚠ This design decision has some history to it, and good reasons for the approach especially relevant for Sync Wrap. Read more: • 20210505 - WDR Meeting Minutes[265] • Async to sync conversion (sync wrap) (see page 647)

7.14.4.9 RESTful APIs good practice

A primary advantage of REST over HTTP is that it uses open standards, and does not bind the implementation of the API or the client applications to any specific implementation. For example, a REST web service could be written in ASP.NET, and client applications can use any language or toolset that can generate HTTP requests and parse HTTP responses.

Here are some of the main design principles of RESTful APIs using HTTP:

265 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/214140108/20210505+-+WDR+Meeting+Minutes

7.14.4.9.1 Summary

- Meet or better the Richardson maturity model level 2 REST https://martinfowler.com/articles/richardsonMaturityModel.html
- Resource names should generally be plural not singular, so **/customers** not **/customer** - this reflects the fact that they are collections of resources
- Resource names should represent the resource
- API Paths should represent the relationship between resources i.e. **customers/{customer id}/orders**

7.14.4.9.2 Good practice

REST APIs are designed around *resources*, which are any kind of object, data, or service that can be accessed by the client.

A resource has an *identifier*, which is a URI that uniquely identifies that resource. For example, the URI for a particular patient medication might be

```
https://my-api.com/medications/1
```

Clients interact with a service by exchanging *representations* of resources. Many web APIs use JSON as the exchange format.
For example, a GET request to the URI listed above might return this response body:

```
{"medicationId":1,"description":"Paracetamol","productId":1,"quantity":20, "date":"12/11/2022"}
```

REST APIs use a uniform interface, which helps to decouple the client and service implementations.

For REST APIs built on HTTP, the uniform interface includes using standard HTTP verbs to perform operations on resources.

The most common operations are GET, POST, PUT, PATCH, and DELETE.

7.14.4.9.3 Organise the API design around resources

Focus on the business entities that the web API exposes. For example, in an pharmacy system, the primary entities might be medication and tasks.

Creating a medication can be achieved by sending an HTTP POST request that contains the medication information.

The HTTP response indicates whether the order was placed successfully or not.

When possible, resource URIs should be based on nouns (the resource) and not verbs (the operations on the resource).

```
https://my-api.com/medications // Good

https://my-api.com/create-medication // Avoid
```

A resource doesn't have to be based on a single physical data item. For example, an order resource might be implemented internally as several tables in a relational database, but presented to the client as a single entity.

Avoid creating APIs that simply mirror the internal structure of a database.

The purpose of REST is to model entities and the operations that an application can perform on those entities. A

client should not be exposed to the internal implementation.

7.14.4.9.4 Nesting on endpoints to show relationships

Often different endpoints can be linked, so you should nest them so it's easier to understand the relationships.

Example

If a Medication resource can be prescribed by different prescribers this would make a valid nesting case.

```
https://my-api.com/medications/{medication_id}/prescribers
```

A GET on the above would bring back all prescriber resources for a specific medication resource.

Nesting allows users the ability to work with resources that are linked without having to use complex queries, and enables the user to understand the relationships between resources.

```
https://my-api.com/medications/{medication_id}/prescribers
https://my-api.com/medications/{medication_id}/tasks
https://my-api.com/medications/{medication_id}/comments
https://my-api.com/medications/{medication_id}/authorisers
```

A GET on any of the above URIs should return a list of the resources linked to the medication identified by the medication_id.

Nesting also allows you to target singular resources easier.

```
https://my-api.com/medications/{medication_id}/tasks/{task_id}
```

A GET on the above URI will should return the task resource identified by both the medication_id and the task_id.

7.14.4.9.4.1 Nesting of resources is not recommended past 3 levels as it can sometimes make the API difficult to read and use.

7.14.4.9.5 Filtering, sorting, and pagination

Sometimes, an API's database can get incredibly large. If this happens, retrieving data from such a database could be very slow.

Filtering, sorting, and pagination are all actions that can be performed on the collection of a REST API.

This lets it only retrieve, sort, and arrange the necessary data into pages so the server doesn't get too occupied with requests.

An example of a filtered endpoint is the one below:

```
https://my-api.com/medications?patient={NHSNumber}
```

A GET on this endpoint will retrieve all Medication resources where the patient identifier = NHSNumber.

Implementing Sorting and Pagination allows the consumer to filter the number and range of data retrieved.

```
https://my-api.com/medications?patient={NHSNumber}&limit=25&offset=50
```

A GET on this endpoint will retrieve all Medication resources where the patient identifier = NHSNumber, limit the number of returned results to 25 and only return results after the offset ID of 50.

7.14.4.10 Transforming data on the APIM platform and containers

APIM is looking to deprecate this capability from the APIM Platform

7.14.4.10.1 Overview

The platform Architecture principles (see page 322)discourage any processing as part of the Apigee processing, however there are some scenarios where it is valid to do some lightweight transformations (e.g. extract parameter from an incoming payload and place into a header for your API backend). Alternative approaches here could be referred to as mediation layers; orchestration engines; enterprise service bus (ESB). Apigee was chosen specifically as a thin layer, to ensure the business logic stays in the application user interfaces or the API backend - not separated between those two places.

If it is agreed a transformation is needed, then the design options are:

1. Deal with any transformation in your backend, keeping Apigee lightweight (this is the preferred option 🙂)
2. If it is ultra-simple, you can simply extract data from JSON directly in your proxy config
3. If you can't, or it is not right to include in you backend - then use an APIM AWS Hosted Container

We have ruled out using the following options:

- Apigee Hosted Targets
- Python callouts
- JavaScript callouts

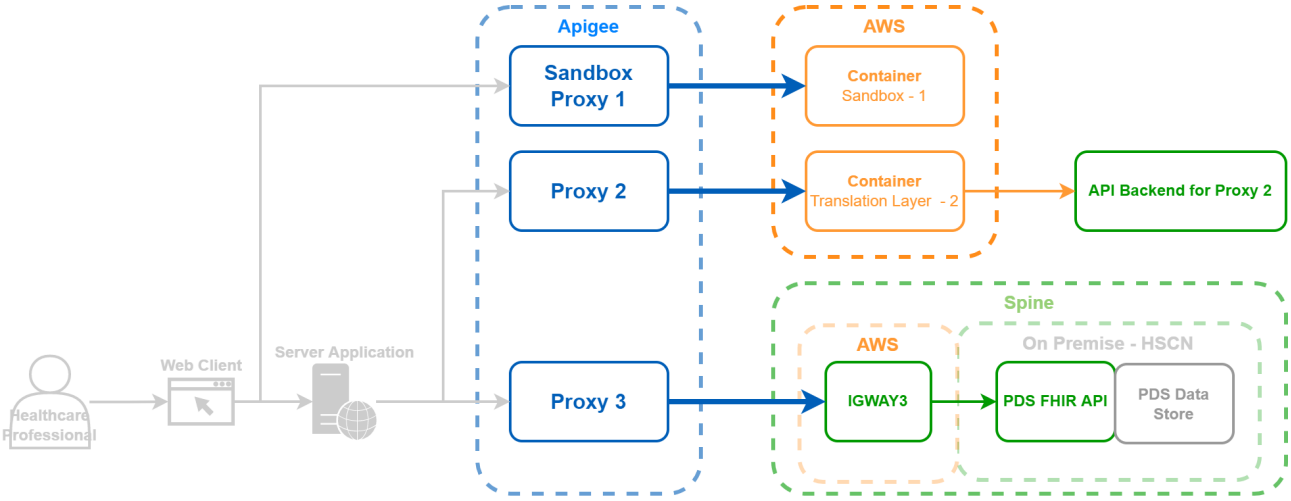
7.14.4.10.2 APIM AWS Hosted Containers

If there is an agreed scenario to do transformation processing - then APIM has a mechanism where you provide a Docker image and it gets connected and deployed between your Apigee Proxy and your backend. This is fully automated by the APIM CI/CD pipeline.

These are the following principles of the APIM design:

- The containers are a logical extensions of the Proxy
- They could be **either**:
  - Self contained processing (e.g. a Sandbox)
  - Acting as a more complex proxy, with traffic "leaving" from AWS to the API Backend instead of Apigee
- The containers are (currently) Docker images, with few restrictions of contents - you can choose the language / technologies to use
- The containers should **not** be:
  - Stateful - beyond an in memory transient stack that has no loss implications
  - Heavy on business logic
- Existing automation synchronises deployments of proxies and containers

Visual version of the different type of traffic flow through AWS:



Delving into individual design considerations that will shape any Docker solution scope and design:

Design Area	Target	Considerations

Stateless	All services to be stateless	• There are **no** backup and persistence mechanisms in the APIM AWS architecture
		• Requests should always be short-lived
		The definition of "Stateless" is broad, and at this stage APIM is looking to understand if the Apigee ecosystem should look to allow:
		• In-memory persistence between calls:
		  • However, this encourages business logic to be incorporated, which is not a wanted outcome
		  • Apigee Cache is a scalable and high availability solution that can already do this
		  • Adding AWS services into the APIM Platform can take time and add complexity that APIM does not want to maintain (e.g. use Texas instead 🙂)

Coordination	Simple logic that helps simplify your overall solution	There are benefits to lightweight coordination / mediation activity capabilities:
		• Risks:
		  • API Producers evolve their solution into full orchestration
		  • Contradicts the "thin" layer design principle
		  • Such orchestration often involves business logic
		• Initial business case and high level requirements during the procurement of Apigee, however, did include requirements to look at this type of logic to hide complexity from consumers
		• Speeding up delivering of a solution is possible because standing up and managing new infrastructure is not a simple task

Limit complexity		

Design Area	Target	Considerations
Orchestration IPaaS	APIM Platform is not an orchestration solution nor Integration Platform as a Service solution	The technology choice of Apigee does not support such a strategy. Mulesoft provides aspects, and, in part, the benefits of such an approach were not clear enough to justify the extra costs. Such a design choice also pushes back on a "stateless" design choice
Transformation		e.g. FHIR to HL7 V3 transformation
Hosting anti-pattern	Hosted solutions should not "expand by stealth" to be large complex services	The aim is to make it quick and easy to host and run a Docker image as part of a proxy. With this ease of use, API Producers could start to build everything inside the Docker service - which then changes the remit of APIM beyond an API Management Platform
IaaS	APIM are not supporting a generic hosting (Infrastructure as a Service)	• There is no desire to provide a Texas like solution • Mulesoft would have been a better • The broad NHS Digital AWS approach of empowering Tribes and squads to build out solutions with their preferred technologies does not align with APIM stepping in and providing a "one size fits all"
Private Beta Transition	- to be confirmed -	SCR and Signing Service have both been sped up by allowing at least Private Beta usage of the Platform AWS Hosted Containers

7.14.4.10.3 Further Reading

- How to actually deploy them: Developing ECS proxies (see page 638)
- *For a wider perspective and build up to this solution see:* D025 - APIM Platform offering a hosting service[266]

266 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119761702/D025+-+APIM+Platform+offering+a+hosting+service

7.14.4.11 API design review

7.14.4.11.1 Overview

Doing this design review well, as a team, is really hard - there is **always a really good** reason why you made your original decisions. To get the most out of this review, you need to make sure you re-assess all design aspects.

The review is to:

- Take a step back (from the rush 😉 to get to this point)
- Make integration easier[267] - what is the impact on the API Consumer of your design decisions?
- Consistency, consistency, consistency - apply it

Attendees:

- API designer from your team
- IOPS FHIR designer (tech lead or tech modeller) for a FHIR API.

Before the meeting:

- Take a copy of the checklist below in appropriate place in your own Confluence space and:
    - Create a copy of it in an appropriate place in your own Confluence space
    - Rename it to "<My API name> API design review"
    - Add a link to it in API register (see page 1204)
    - Fill out the page - deleting anything you don't need, such as this tip. When you edit this page there are number of hints and further help.
- For a FHIR API (or where you have just a FHIR payload), ensure the API design has been reviewed at the IOPS Design Authority

Then complete the checklist (or as much as possible) during your meeting. Use the checklist to track any tickets on your backlog you may create for any gaps.

7.14.4.11.2 Design review checklist

Description	Status	Comments / evidence
Re-read the API Landscape[268] policy	Y/N	

267 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission
268 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/123930121/API+Landscape

Description	Status	Comments / evidence
Re-read Designing your API (see page 322) and the key linked pages	Y/N	
Is the name of the API still valid? • As you've talked to others about the API, has there been confusion about the name? • Is the scope still the same? • Is the name intuitive?	Y/N	
Have you adhered to the API naming and grouping endpoints (see page 359) guidance?	Y/N	*This is the right time to do a major rename. You can't rename our repo, but you can change the API Catalogue entry and the base path*
Have you completed the Consistency checklist on your API architecture review checklist (see page 266)?	Y/N	*Re-read what you've decided. If you haven't completed it, **don't** update it (unless you find it useful) - some of the key questions are repeated below, so just focus on them*
In the OAS file, are the access modes[269] clear on each endpoint?	Y/N	*Normally access modes are listed in the "Security and authorisation" section of your OAS API specification - however, if the access mode applies to different endpoints this needs to be made clear*
<This is N/A for Proxygen> Is there a one to one mapping between access modes and the Apigee Products?	Y/N	*If this is not the case then your API won't be compatible with the Digital Onboarding Service - see* Access modes - compliance - security and simplifying running your API (see page 299)
Do your endpoints return the minimum amount of data possible?	Y/N	*If not, could you:* • *Default to a smaller subset, and allow parameter to expand for specific requests?* • *Could you have different access modes, so the lower security connections receive less data*

269 https://nhsd-confluence.digital.nhs.uk/display/APM/Privacy+-+compliance+-+security+and+simplifying+running+your+API#PrivacycompliancesecurityandsimplifyingrunningyourAPI-Accessmodes

7.14.4.11.2.1 For a FHIR API

<Delete if not a FHIR API>

Area	Description	Status
Design	Re-read Designing your FHIR API (see page 324) and key linked pages	Y/N
	Are you using the FHIR REST paradigm?	Y/N
	Have you picked the right FHIR resources?	Y/N
	Do your payloads use references rather than (see page 326) including all the data (unless it is for common display purposes)?	Y/N
	Have you implemented a search interface (see page 330)?	Y/N
	Have you & IOPS looked at FHIR UK Core[270] ?	Y/N
Endpoints	Are you endpoints aligned with the FHIR resource model? Definitions are here: FHIR HTTP[271] and FHIR Resource List[272]	Y/N
Headers	• Try not to add custom Headers at all • If they are added, they must not contain any data that should be in the payload • Should be namespaced with "NHSD-" Do your headers align with: HTTP headers for your API (see page 395) ?	Y/N or N/A
	Are your headers consistent with other API Management Headers? For instance: • Requiring a GUID in the **X-Correlation-ID** could break some API Consumers • Putting an ODS code that is not an End User Org (EUO) will confuse (potentially) or break other APIs	Y/N

270 https://digital.nhs.uk/developer/api-catalogue/fhir-uk-core-standards
271 https://www.hl7.org/fhir/http.html
272 https://www.hl7.org/fhir/resourcelist.html

Area	Description	Status
Verbs	Does your HTTP Verb usage align with the FHIR REST principles - see FHIR HTTP[273]?	Y/N
Errors	Do all your errors, include a payload? See RESTful HTTP design points (see page 365) for details	Y/N

7.14.4.11.2.2 For a non-FHIR API

<Delete if a FHIR API>

Area	Description	Status
Paths	• Do you keep the number of levels in your path to a minimum? • Are there terms that might be confusing when looking at the endpoint names? • Have you grouped your endpoints correctly? See API naming and grouping endpoints (see page 359)	Y/N
	Have you looked at the differences between: Proxy base path vs API base path (see page 364)?	Y/N
Endpoints	• Have you used nouns? • Are they intuitive? • Have you used spinal-case?	Y/N
	Have you had to nest your endpoints? See RESTful APIs good practice (see page 368)	Y/N
Query parameters	• Are these consistent with other API Platform query parameters? • If the parameter is not used elsewhere, have you looked at the FHIR terms, and is there a matching definition?	Y/N or N/A
	If you are migrating an existing API, can you allow both the old and new terms?	Y/N or N/A

273 https://www.hl7.org/fhir/http.html

Area	Description	Status
Field Names	Are the field names in the **payload** datasets the same as FHIR? Even for non-FHIR APIs, developers should not have to deal with different field names, for different APIs, for basic data such as first name and last name?	Y/N
Headers (Incoming)	• Try not to add custom Headers at all • If they are added, they must not contain any data that should be in the payload • Should be namespaced with "NHSE-" Do your headers align with: HTTP headers for your API (see page 395) ?	Y/N or N/A
	Are they consistent with other API Management Headers? For instance: • Requiring a GUID in the **X-Correlation-ID** could break some API Consumers • Putting an ODS code that is not an End User Org (EUO) will confuse (potentially) break other use of other APIs	Y/N
Verbs	Does your HTTP Verb usage align with the FHIR REST principles - see FHIR HTTP[274]?	Y/N
Errors	If the API does use an existing standard - do you use the generic platform error format (see page 624)?	Y/N or N/A

7.14.4.11.2.3 Other considerations

This final checklist does drill into the detail, but it is still important. Consistency between APIs takes the pain out of calling multiple NHSE APIs. For instance, having to use a different header for the same data between APIs creates confusions, adds code complexity and makes integration harder.

Area	Description	Status
Update operations	PUT is for updating an entire resource PATCH is being used where there are small changes to a resource	Y/N or N/A
Status Codes	Do your HTTP Status Codes align with RESTful principles? • FHIR also influences the choice of which status codes to use	Y/N

274 https://www.hl7.org/fhir/http.html

Area	Description	Status
Datetime	Do date and time field values follow the ISO-8601[275] standard? More details on RESTful HTTP design points (see page 365)	Y/N
ETag	Does your API need to use ETag Headers (see page 1109)?	Y/N

7.14.4.12 Version control

7.14.4.12.1 Overview

When thinking about version control for our APIs, we distinguish between two broad areas:

- The industry standard, and the version of that standard, that an API is published under (such as FHIR STU3, FHIR R4 etc)
- Different versions of the logic of the API, which has the following aspects:
  - the backend API code
  - the API proxy

The approach we use is:
- Use the URL path to indicate any industry standard and version, e.g. /my-api/FHIR/R4
- For selecting the logic version, use the "version" parameter in the accept header, e.g. "**application/fhir+json; version=2**"
  - For your first version, you do not need to use a version at all - only add it when you move to version 2
  - The logic version is for **major** versions only. As we define this based on SemVer, a minor version is non-breaking and therefore it does not need to be specified

There is no clear developer consensus on the best way to manage versioning - if there is significant push back from the API consumer community then the approach should be re-assessed. Apigee can be used to run multiple versioning mechanisms in parallel.
To find out more about the other options considered see:
- API versioning analysis (see page 387)

275 https://en.wikipedia.org/wiki/ISO_8601

- APIM Weekly Design Reviews that included many discussions around versioning[276] *options*

When looking at the options, the vendor specific content option was marginally the preferred one because:

- *Used by GitHub and HMRC*
- *Allows specific endpoints to be versioned under the OAS file schema*

***However**, this wasn't quite compatible with the FHIR standard and extra discovery showed the new OAS renderer supported accept header parameters.*

7.14.4.12.1.1 Industry standard version

This can often indicate a fundamentally different API, that requires its own life-cycle to manage. Based on that and some existing standards, APIM are using path name-spacing to separate standards (e.g. FHIR vs. non-FHIR). This has been ratified and you can see the full definition in:

- API Platform Landscape Policy[277]
- EA Architectural Notice AN0006[278].

7.14.4.12.1.2 Version of the API logic

This is more complex, and it is up to the API Producer how this is managed. This is made up of:

- Proxy deployed in Apigee
- Backend API version

The proxy version numbering doesn't have to have numbering that matches the backend - they are independent entities. Ideally (for supportability), you should surface your backend API version in the _status endpoint.

When working with APIM the focus is on the proxy version, which you can always get from the _ping endpoint or on the Grafana API Status Dashboard[279].

How API Consumers select the API logic version

When, as an API Producer, you have breaking changes and decide to introduce versioning of the backend API, then use the accept header and the parameter:

- application/fhir+json; version=<**major version**>

---

276 https://nhsd-confluence.digital.nhs.uk/dosearchsite.action?cql=siteSearch+%7E+%22ancestorIds%3A114912326+AND+versioning%22+and+space+%3D+%22APM%22+and+type+%3D+%22page%22&queryString=ancestorIds%3A114912326+AND+versioning

277 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/123930121/API+Landscape

278 https://aalto.digital.nhs.uk/#/search?noQuery=true&pageNumber=1&pageSize=50&searchIn=1-4-2&searchContentType=2-3&viewMode=1&location=b712be73-e685-43da-93ca-10993cd4c1fa&browseMode=true&library=5464c07f-daf1-4eee-b9b6-22e6c4dfbbd0

279 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=5m&from=now-1m&to=now

As an API Producer, you might already have another versioning strategy. However, to maximise consistency between APIs please use this as your primary versioning. You can continue to support your other mechanism in parallel and you can do that logic in Apigee (or your backend).

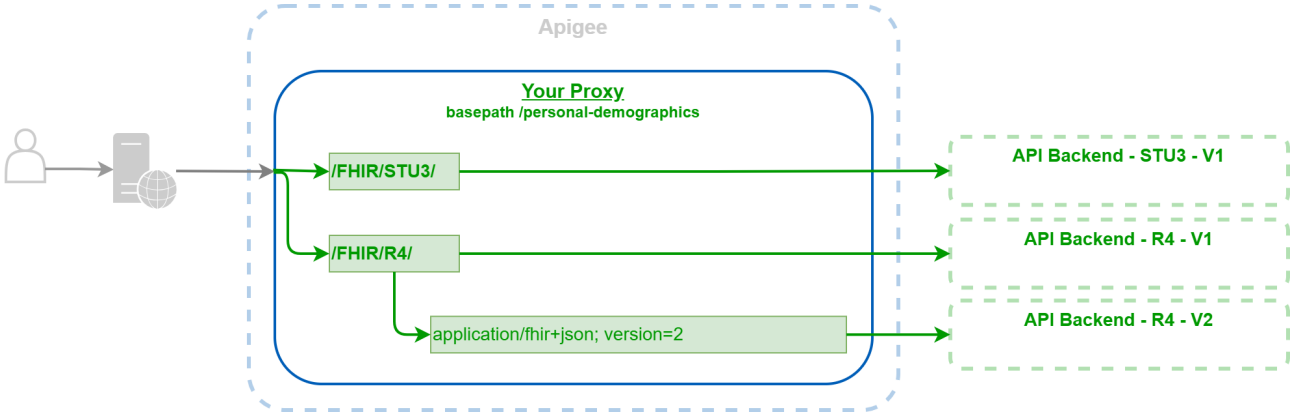
Public facing summary: https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#version-control

Key technical aspects:

- Default version used:
  - Most APIs, where no Accept header is initially required - the proxy behaviour must operate in a non-breaking way for existing API Consumers
  - For example, passing no Accept header will always select V1, and to use V2 (or beyond) the API Consumer must send an Accept Header
- For FHIR compatibility the Accept header must include "application/fhir+json", hence the example: "application/fhir+json; version=2"
- The response Content-Type header needs to be considered:
  - For all FHIR payloads this must be set (according to https://build.fhir.org/http.html#mime-type) to `application/fhir+json` (for JSON payloads)
- As soon as you introduce a second version, running in parallel, you must then include the version parameter in all requests including the original V1. The "default" version Content-Type header field should be added and specify the first major version (i.e. version=1)
- See https://httpwg.org/specs/rfc9110.html#field.accept for the wider specification

What does this mean in practice?

There are still a number of choices on how you implement a solution, but Apigee is good place to route versioning logic. It works especially well if you have chosen to deploy different target backends, keeping all version logic out of your code base. You might find that your API routing looks something like:



APIM is looking to write some guidance on how to do this see:
APM-3474[280] - API Producer supporting multiple versions WITHDRAWN

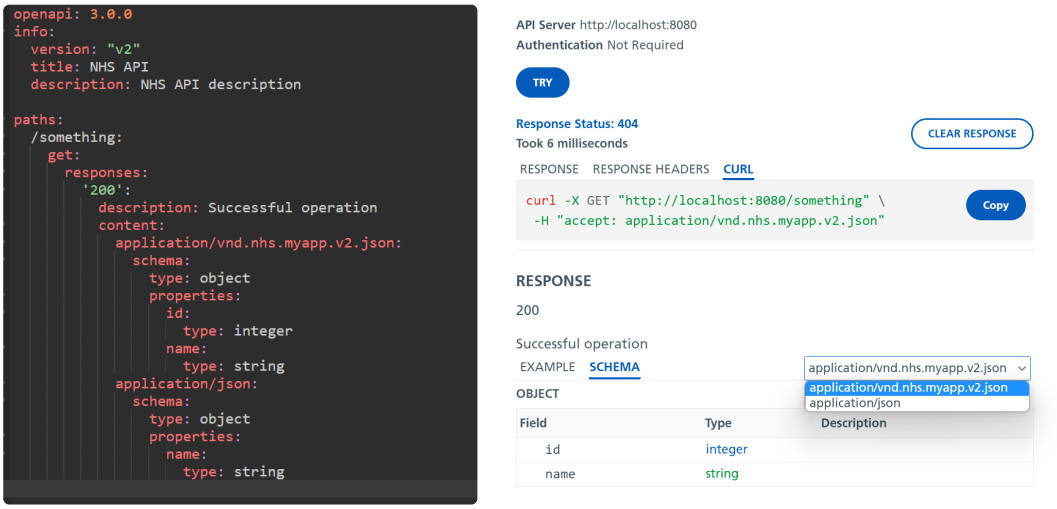
280 https://nhsd-jira.digital.nhs.uk/browse/APM-3474?src=confmacro

7.14.4.12.1.3 Using OAS content types for versioning

> ⚠ *An approach is needed to be agreed on how (if) one OAS file can contain multiple versions, since OAS support is not strong in this area*

The underlying schema for OAS does support different values based on content-type, so this **can be** used to provide some limited version dependent content. An example using RapiDoc (new OAS renderer):

```yaml
openapi: 3.0.0
info:
  version: "v2"
  title: NHS API
  description: NHS API description

paths:
  /something:
    get:
      responses:
        '200':
          description: Successful operation
          content:
            application/vnd.nhs.myapp.v2.json:
              schema:
                type: object
                properties:
                  id:
                    type: integer
                  name:
                    type: string
            application/json:
              schema:
                type: object
                properties:
                  name:
                    type: string
```



What does this cover:

- This works for both requestBody and response parts of the schema
- It does **not** support:
  - Different parameters per version
  - The endpoint existence - e.g. if a new version adds or removes an endpoint
  - Higher level commentary
- Can lead to namespace collisions in the schemas, which must also be versioned

7.14.4.12.1.4 Managing one or more OAS files

Your OAS file will also have a version, but (just as the main API version selection / routing is split) consideration of how many API specifications is needed. The recommendation here is that you have **one APIM repo** (and therefore **one proxy**) which contains:

- One OAS file per standard / standard version (e.g. one for STU3 and one for R4)
  - To enable multiple spec deployments for a single repo see Multi spec deployments (see page 907)
- If your API itself has relatively small changes between versions (even major / breaking changes):

- Are differences easy to understand by the reader?
- Does the use (**tbc how**) of the inbuilt OAS content type handling help render the page clearly?
- Is the functionality conceptually the same still after the major change?
- If any of the answers to the above are "**no**", then a new OAS should be considered to support that major version

The version of the OAS file will change the most, and gets deployed separately to your proxy. For example, the wording and examples on the page are most likely to change - with no impact on the functionality / logic of the API.

For an example of what multiple APIs might look at see: https://digital.nhs.uk/developer/api-catalogue/nhs-website-content (Versions section).

7.14.4.12.2 Automated versioning of your proxy

You need to use version numbering to keep track of changes to your API. There are the following aspects to the version control:

1. Major version number
2. Project status (e.g. alpha or beta)
3. Minor version
4. Automated increment of revision

The first three are controlled by the API Producer, and then the automation takes over to add the revision, formatting for SemVer (semantic versioning[281]) and deliver to the correct files and aspects to the CI/ CD Pipelines.

7.14.4.12.2.1 Beta phase versioning approach

Your first version of your proxy / API is version 1.

While you're in beta, you **can** make breaking changes to your API without bumping the major version number. However, this does need to considered carefully if you've ended up with many API consumers at this stage.

Once you're out of beta and in particular if your API has users, if you make a breaking change you need to have a new version.

7.14.4.12.3 Proxy and OAS specification version number

On a build, the version tag is built and applied to files relating to your proxy deployment **and** to the OAS spec itself. We use semantic versioning[282] as follows:

- <major>.<minor>.<revision>[-<status>]+<build>

281 https://semver.org/
282 https://semver.org/

- <major> is the major version number and should match the major version number of your API, which is almost certainly 1 (not 0).
- <minor> is the minor version number - see below.
- <revision> is the revision number - incremented every time you make any changes.
- <status> is "alpha" if your API is in alpha, "beta" if your API is in beta and missing if your API is out of beta.
- <build> is the build number.

For example:

- 1.0.101-alpha
- 1.0.285-beta
- 1.0.387

7.14.4.12.3.1 Viewing the current version

The current version of a branch can be viewed by running the following from the project root:

```
$ poetry run python scripts/calculate_version.py
```

7.14.4.12.3.2 Incrementing the revision number

The revision is incremented automatically on each commit (except on merge commits).

```
$ poetry run python scripts/calculate_version.py
v1.0.0-alpha
$ git add . & git commit -m "APM-123 Some changes"
$ poetry run python scripts/calculate_version.py
v1.0.1-alpha
```

7.14.4.12.3.3 Incrementing the minor version

The minor version should be upgraded whenever a significant new feature is introduced that does not contain breaking changes.

To signal a minor version increase, include "+minor" in a commit message.

Increasing the minor version will reset the revision to 0.

```
$ poetry run python scripts/calculate_version.py
v1.0.1-alpha
$ git add . & git commit -m "+minor APM-123 A new feature"
$ poetry run python scripts/calculate_version.py
v1.1.0-alpha
```

You can also use an empty commit to accomplish this:

```
$ poetry run python scripts/calculate_version.py
v1.1.0-alpha
$ git commit --allow-empty -m "+minor"
$ poetry run python scripts/calculate_version.py
v1.2.0-alpha
```

7.14.4.12.3.4 Incrementing the major version

The major version should be upgraded whenever a breaking change is introduced.

To signal a major version increase, include "+major" in a commit message.

Increasing the major version will reset the minor and revision numbers to 0.

```
$ poetry run python scripts/calculate_version.py
v1.2.0-alpha
$ git add . & git commit -m "+major APM-123 A breaking change"
$ poetry run python scripts/calculate_version.py
v2.0.0-alpha
```

You can also use an empty commit to accomplish this:

```
$ poetry run python scripts/calculate_version.py
v2.0.0-alpha
$ git commit --allow-empty -m "+major"
$ poetry run python scripts/calculate_version.py
v3.0.0-alpha
```

7.14.4.12.3.5 Setting or clearing the 'status' or prerelease indicator

As we move from alpha to beta and onward, we will want to change the tag at the end of the version to indicate this.

These can only be done as empty commits or as the first words of a commit message.

```
$ poetry run python scripts/calculate_version.py
v3.0.0-alpha
$ git commit --allow-empty -m "+setstatus beta"
$ poetry run python scripts/calculate_version.py
v3.0.0-beta
$ git commit --allow-empty -m "+clearstatus"
$ poetry run python scripts/calculate_version.py
v3.0.0
```

7.14.4.12.3.6 Releases

CI will automatically tag the latest version in our Github repository on a successful master branch run.

7.14.4.12.3.7 Including build numbers

Any code that is deployed to an API proxy, running service or anything else should also include a build number from the CI pipeline. This should be appended to the end of the version like this (given a build-id of 1234):

```
v3.0.0-alpha+1234
```

7.14.4.12.4 API versioning analysis

User	Edits	Comments	Labels
Aubyn Crawford[283]	5	0	0

7.14.4.12.4.1 TL;DR

1. Version your API for FHIR **AND** Logic !!!
2. For FHIR use Path versioning
3. For Logic use Content Negotiation

283 https://nhsd-confluence.digital.nhs.uk/display/~aubyn.crawford1@nhs.net

The following example shows FHIR at R4 and logic at 2.0

Example

```
curl -H 'Accept: application/fhir+json; verson=2.0' https://example.org/organization-data-service/FHIR/R4/Organization/RJ11
```

7.14.4.12.4.2 What are we versioning?

The APIs we produce are required to version two things:

- FHIR
  Chance are you are only going to need to support a single version of FHIR, but support should be explicit. Also we should understand what the strategy will be for introducing new versions of FHIR. Are you going to extend the existing software or completely rewrite it.
- Logic
  When writing an API it will be important to introduce changes to the logic, whether this is to introduce new features or fix defects.

7.14.4.12.4.3 Why should we introduce versioning?

We must respect the integrity of the software that our consumers write, and any change we introduce can have potentially disastrous effects on our consumers. Therefore we cannot force breaking changes onto them. This applies to Bugs as well as Improvements. Anything that causes a change to the API, which would mean it is not backward compatible cannot be forced onto consumers. We use the distinction of MAJOR and MINOR changes from semver.org[284]. Any change that is backward compatible is a MINOR change. Any change that is not backward compatible is a MAJOR change.

Therefore, when adding new features or bugfixes it should be carefully considered whether it is a MAJOR or MINOR change.

Clear Examples:

- MAJOR changes
  - Renaming a field/property (including fixing spelling errors!)
  - Removing a field/property
  - Changing the type of a field/property
  - Changing the conditions when a field is present/absent
- MINOR changes
  - Introducing a new field/property

Ambiguous examples:

- Widening the scope for your dataset
- Introducing new values for fields that are used in branching logic (e.g. enums)

284 https://semver.org/

7.14.4.12.4.4 Options

1. Query String
2. PATH
3. Custom Header
4. Content Negotiation
5. Vendor Specific Content Negotiation

7.14.4.12.4.5 Measures

1. Simplicity
   How difficult is it for developers and consumers to understand and implement?
2. REST
   Does the API adhere to the definition of what is considered "RESTful"?
3. CDN / Caching
   Are there any special considerations needed when caching an API that uses this versioning method?
4. CORS
   Are there any special considerations needed with regard to Cross Origin Request Sharing security within browsers?
5. OAS / Swagger
   Are there any special considerations needed when thinking about how you write and structure OAS / Swagger documents?
6. Referenced Data
   When records/links are referenced inside the document, do those records/links need to be version aware? Where a record references another internal record this is less of an issue, but when the reference is to another system you are coupling systems together. This may be expressed as the RESTful concept of HATEOS (Hypermedia As The Engine of Application State)
   The following example shows a dataset returned by a system, which references three things: 1) internal revising document id; 2) external Personal Demographics Service FHIR R4 patient record; 3) external Organization Data Service FHIR STU3 organization record. Our system is now coupled to PDS FHIR R4 and ODS FHIR STU3, rather than referencing a Patient or Organization it's referencing a version of the patient or organization.

Versioned URLs

```
{
  "document_id": "1234567890",
  "revising_document_id": "0987654321",
  "links": {
```

```
    "original": "/R4/DocumentReference/0987654321",
    "revise": "/R4/DocumentReference/1234567890/revise",
    "deprecate": "/R4/DocumentReference/1234567890/deprecate",
    "history": "/R4/DocumentReference/1234567890/history"
    "subject": "https://api.service.nhs.uk/personal-demographics/FHIR/R4/
Patient/1234567890",
    "custodian": "https://directory.spineservices.nhs.uk/FHIR/STU3/
Organization/RJ11",
   }
}
```

The following example is not versioned, and it is therefore up to the consuming client to decide which version of FHIR or API logic it wants to consume.

Unversioned URLs

```
{
  "document_id": "1234567890",
  "revising_document_id": "0987654321",
  "links": {
    "original": "/DocumentReference/0987654321",
    "revise": "/DocumentReference/1234567890/revise",
    "deprecate": "/DocumentReference/1234567890/deprecate",
    "history": "/DocumentReference/1234567890/history"
    "subject": "https://api.service.nhs.uk/personal-demographics/Patient/
1234567890",
    "custodian": "https://directory.spineservices.nhs.uk/Organization/RJ11",
   }
}
```

Query String

Another comment method is to use a query string parameter.

```
curl /Organization/RJ11?version=1.0
```

Measure	Measure	Description
Simplicity	⊕	Easy to implement, understand and can be posted directly into a browser.
REST	⊖	Violates REST, by referencing a version of the Resource rather than the Resource

Measure	Measure	Description
CDN / Caching	⛔	CDNs can strip query string parameters
CORS	✅	No impact on CORS
OAS / Swagger	⛔	Requires multiple OAS / Swagger
Referenced Data	⛔	Server is responsible for handling references to versioned entities within the response

Conclusion: Only use this as a last resort.

Path

The simplest and most common example you see is embedding the version into the http path of the resource.

```
curl /v1/Organization/RJ11
```

Measure	Measure	Description
Simplicity	✅	Easy to implement, understand and can be posted directly into a browser. Different versions can be deployed as different applications, if needed.
REST	⛔	Violates REST, by referencing a version of the Resource rather than the Resource
CDN / Caching	✅	Little to no impact on CDNs/Caching
CORS	✅	No impact on CORS
OAS / Swagger	⛔	Requires multiple OAS/Swagger definitions
Referenced Data	⛔	Server is responsible for handling references to versioned entities within the response

Conclusion: This is the simplest method, and is widely used across NHSD.

Custom HTTP Headers

The version can be specified using custom HTTP headers.

```
curl -H 'apiVersion: 1.0' /R4/Organization/RJ11
```

Measure	Measure	Description
Simplicity	⛔	Is more complex to implement and understand. Single deployed system supports multiple versions.
REST	➕	Adheres to REST, by referencing the Resource directly
CDN / Caching	⛔	CDN/Caching needs configuring to support
CORS	⛔	Non-standard request headers are more likely to be blocked by CORS
OAS / Swagger	❓	
Referenced Data	➕	Client is responsible for handling references to versioned entities within the response

Conclusion: HTTP headers are preferable to Query String and Path parameters, but can be blocked by CORS/ CDNs.

Content Negotiation Parameters

Using the Accept Header is an extension of the HTTP Headers solution above, except that it uses the existing `Accept` headers, which is already used for content negotiation. To this we add `<parameter>=<value>` separated by `;`.

https://restfulapi.net/versioning

API Version Only

```
curl -H 'application/fhir+json; version=1.0' /R4/Organization/RJ11
```

https://docs.fire.ly/projects/Firely-Server/en/latest/features/multiversion.html

API & FHIR Version

```
curl -H 'application/fhir+json; fhirVersion=4.0' /Organization/RJ11
```

Measure	Measure	Description
Simplicity	⛔	Is more complex to implement and understand. Single deployed system supports multiple versions.
REST	➕	Adheres to REST, by referencing the Resource directly
CDN / Caching	⛔	CDN/Caching needs configuring to support
CORS	➕	`Accept` header is normally not blocked by CORS because it is in general use
OAS / Swagger	Part	Allows each endpoint to define a content type within a single document, but can lead to namespace collisions in the schemas, which must also be versioned.
Referenced Data	➕	Client is responsible for handling references to versioned entities within the response
FHIR compatibility	➕	

Conclusion: This allows you to support multiple request body/response versions in a single OAS/swagger document, but that does not allow for parameter or narrative versioning.

Vendor Specific Content Negotiation

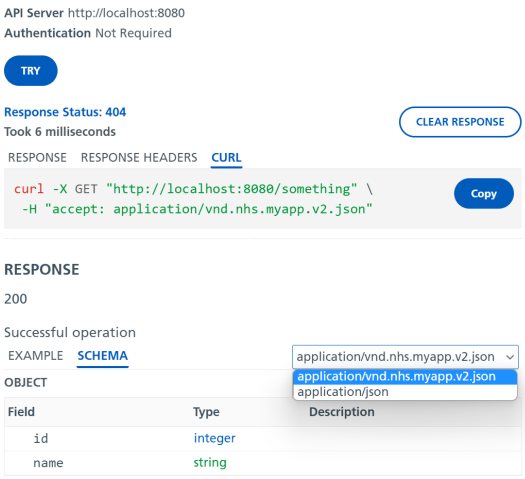
Vendor specific content negotiation is when we make use of custom mime types, such as `"application/vnd.nhsd.patient+fhir+json"`. We can extend these content types to include versioning.

```
curl -H 'application/vnd.myapp.v2.json' /R4/Organization/RJ11
```

Measure	Measure	Description
Simplicity	⛔	Is more complex to implement and understand. Single deployed system supports multiple versions.
REST	✅	Adheres to REST, by referencing the Resource directly
CDN / Caching	⛔	CDN/Caching needs configuring to support
CORS	✅	`Accept` header is normally not blocked by CORS because it is in general use
OAS / Swagger	Part	Allows each endpoint to define a content type within a single document, but can lead to namespace collisions in the schemas, which must also be versioned.
Referenced Data	✅	Client is responsible for handling references to versioned entities within the response
FHIR compatibility	Part	Accept header is compatible. but the returned Content-Type header does not allow multiple values. tbc - a parameter could be used here. e.g. application/fhir+json; version=2

Conclusion: This allows you to support multiple request body/response versions in a single OAS/swagger document, but that does not allow for parameter or narrative versioning.

Confirmed that the OAS Rendering in RapiDoc works with both types of Content Type (parameters and vendor specific)



7.14.4.12.4.6 References

1. https://restfulapi.net/versioning/
2. https://httpwg.org/specs/rfc7231.html#header.accept
3. https://docs.fire.ly/projects/Firely-Server/en/latest/features/multiversion.html

7.14.4.13 HTTP headers for your API

IN BETA

This design pattern is in "beta" - we've had initial feedback, but still have not really had feedback from API Consumers on the impact on integration. It will also take a while before this design pattern is consistent across APIs.

See D052 - Header usage design pattern[285] for background

7.14.4.13.1 Overview

Ideally, all the data for a request would be in the payload body. However, the design requirements and circumstances add complexity very quickly to any API - HTTP headers will be required on most APIs.

285 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/343675823/D052+-+Header+usage+design+pattern

7.14.4.13.2 Avoid HTTP headers where possible

The aim is **minimise** the use of HTTP **request** headers, following the principles:

- Should the data be in the payload?
    - The question "does it make the API consumer integration easier by doing this?" should be included in that decision process
- Would a header duplicate any data in the payload?
- Is it a field recognised only by NHS Digital? If so, it might be important to keep it out of the payload
- Will the data need to be removed or added in the journey of the request? If so, it might become important to place it in a header to avoid payload manipulation

Those principles above discourage using a header - always focus on keeping the data in the payload where possible. Headers tend to contain authorisation, identity, request routing or compliance data not directly related to the payload.

It is important to note that performance should only be considered in edge case conditions. Modern infrastructure stacks can often parse the payloads quickly and efficiently.

7.14.4.13.2.1 If headers are required - follow this design pattern

- One header for each field / piece of data
- Always look to reuse APIM Platform existing headers
    - Make integration easier - if your definition of an existing header does not quite match, re-factor your API to align
- Loosely couple with the backend - don't make a hard dependency between what the API Consumer uses and the backend requirements
- Simplify the data structure for the API Consumer, consider:
    - Will a single value do? If the backend or audit log want a specific format (e.g. FHIR), then the single value can be expanded in the proxy
    - FHIR JSON should only be used as a last resort, where it benefits the API Consumer
- Data formats supported, two levels:
    - Simple value (where there is no risk of illegal HTTP header characters)
    - Base64 encoded JSON - all other data
        - Should the JSON always be a JWT?
            - No - use the JSON format that works best for your solution
            - JWTs are excellent where you need a tamper proof set of claims (fields), but does introduce potentially unnecessary complexity

7.14.4.13.2.2 Naming your headers

- The field name should be name spaced with "NHSD-" (see RFC 6648[286] for further guidance, e.g. the "X-" prefix was deprecated in 2012)
    - For headers that are specific to your API, build on that name spacing and insert an acronym e.g. NHSD-ERS-
- Use "-" between words
    - Title Case - whilst HTTP Header field names are case insensitive (RFC 9110 header fields[287]) the original X- header names tended to be shown as capitals. Unknown User (misc4) - is there a Tardis guide on how Header field names should be displayed?
    - Identity field names should end with "ID"
- Balance brevity with clarity - avoid acronyms past the top level ones focused on namespace (e.g. APIM should have really named NHSD-Session-URID to NHSD-Session-User-Role-ID)
- Using name spacing to group related values, e.g. group session related headers by having session closer to the start of the field name
- UK English

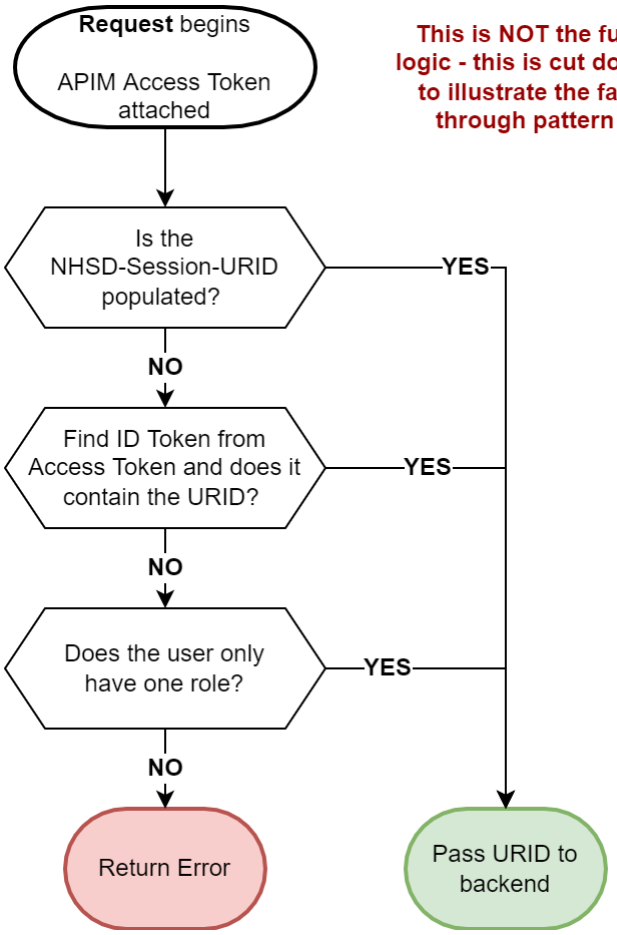
7.14.4.13.3 Making integration easier - value fall through

Every extra piece of information that is required from the API consumer makes the integration harder. When designing your API, follow the following principles:

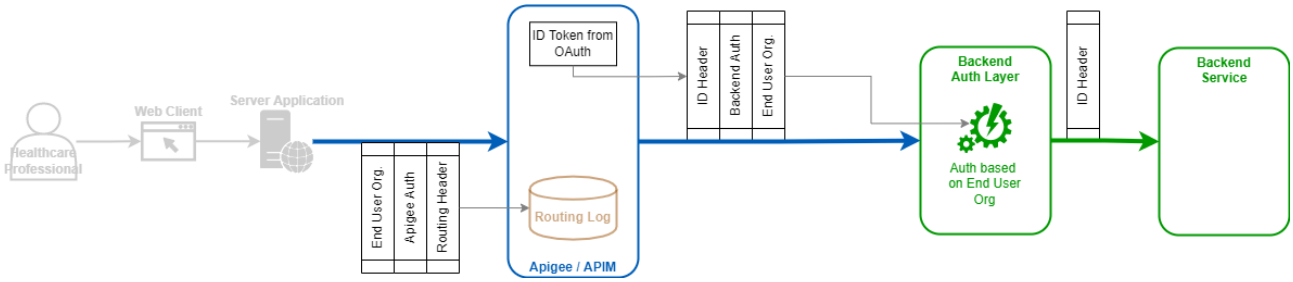
- Only ask for the specific piece of information once
- If the backend API needs that information twice (e.g. backwards compatibility) - use Apigee to duplicate it
- Options to provide the same data in multiple ways - this will make integration easier.
    - This generally makes integration easier, however, too many options can start to make it harder
    - Use Apigee to allow the value to be picked up in multiple ways

An example - there is a requirement (for many APIs) to provide the User Role ID (URID) of the logged in NHS CIS2 user. The logic (or a request once the OAuth access token has been obtained) might be shown as:

286 https://tools.ietf.org/html/rfc6648
287 https://www.rfc-editor.org/rfc/rfc9110.html#name-header-fields



Example header addition and removal:



7.14.4.13.4 APIM platform request standard headers

There are currently not many standard headers, this is expected to grow, but not my a large number. They are:

Header	Status	Notes
NHSD-Session-URID	Active	The user role ID (URID) for the current session. Also known as a user role profile ID (URPID). Already has a shared flow to help obtain this:
NHSD-End-User-Organisation-ODS	Active	End User Organisation is important for audit logging and compliance. It is a common concept on existing APIs across. This header accepts an ODS string
NHSD-End-User-Organisation	**Proposed**	For BaRS (and future) APIs the coding system for an End User Organisation (EUO) is not just based on ODS code. This allows a JSON string containing system and unique identifier to be sent over If NHSD-End-User-Organisation-ODS is populated then this field does not need to be sent (and should be ignored)
NHSD-End-User-Identity	**Future Potential**	Some system to system solutions will still have local authentication, this is a potential header name to store a copy of that identity information for audit - even if NHSD does not have access to the definition of the user
NHSD-End-User-Identity-CIS	**Proposed**	e-RS allow app restricted, but are still expecting the API Consumer to have local auth. The requirement is to map that local user to a CIS identity. This header is for the API Consumer to send the SDS User ID with each request.
NHSE-Product-ID	**Proposed**	Under certain conditions it allows an API Consumer, that is acting as a routing type solution, to pass the calling App's Product ID. Proposed for use on NRL by BaRS
~~**NHSD-Requesting-Software**~~	Remove proposal	Deprecated as the use of User-Agent is fully agreed.

Header	Status	Notes
User-Agent	Active	All API Consumers should be encouraged to set the User Agent field (defined in RFC 7231) to record the calling software. The minimum entry should be a "product", which is defined as: • `token`[288] `["/" product-version`[289]`]` • For example: MyProductName / 1.2 Bookings and Referrals (BaRS) is the first API to mandate this usage. Where a browser is making the call to APIM (currently only the Authorisation Service would fall in this category), the browser user agent string is expected - ideally with the actual web app (product) added to that string. *TBC what this means with the introduction of the Product ID - as that will be **known** by the Apigee and already logged. The User Agent string is to help debug and tracing, often the string name of software application is a powerful way of tracing new problems.* For the decision behind this see D054 - Use User-Agent header instead of an NHS D specific custom header[290]
apikey	Active	Commonly used header name for the lowest level security APIs - see Securing your API (see page 774)
X-Request-ID	Active	
X-Correlation-ID	Active	
Accept	Active	Optional to be supported, needed if more than one MIME content type is available, or if the API has multiple functional versions - see Version control (see page 380)

Please note:

- **Authorization** header is a standard for securing APIs
- There is **no identity** field - the identity is attached to the access token and the API Producer must retrieve the details from there
    - tbc - how to approach locally authenticated apps that use Application Restricted access mode

288 https://httpwg.org/specs/rfc7231.html#imported.abnf
289 https://httpwg.org/specs/rfc7231.html#header.user-agent
290 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/390955924/D054+-+Use+User-Agent+header+instead+of+an+NHS+D+specific+custom+header

7.14.4.13.4.1 Response headers

Response headers are not the primary focus for API consumers, however, there are a number of them that need to be considered. They are:

Header	Status	Notes
Content-Type	Active	FHIR requires Content-Type to be populated, and for JSON payloads this must be: application/fhir+json Where multiple functional versions is supported the API Producer should also include the vendor specific media type (as was MIME type), so the returned value might look like: application/fhir+json; version=2 For more details see Version control (see page 380)
X-Correlation-ID & X-Request-ID	Active	See Supporting API consumers with tracing unique identifiers (see page 1104)
Cross-origin resource sharing (CORS)	Active	See Cross-origin resource sharing (CORS) (see page 786)
Retry-After	Active	For situations where a 503 is the correct response code, this header must be populated with a suggest period to retry the request - see RESTful HTTP design points (see page 365)

7.14.4.13.4.2 Outbound headers - going from Apigee to backend

Ideally, all outbound request headers from Apigee should match if they have the same data. However, this is less important as the **API Producer** is responsible for the Proxy and the Backend.

For the full breakdown see Passing information to your API backend (see page 611), but these ones on PDS could be used more widely:

Header	Example	Notes
NHSD-Identity-UUID	910000000001	Extracted from ID Token subject ("sub") claim. For NHS ID this is the UUID in Spine

NHSD-Session-URID	98765400091	Supplied by the API Consumer, where the Clinician picks the role they are undertaking see Authorisation[291]
NHSD-ASID	200000000421	Sometimes referred to as the API Consumer ASID or FromASID. See Spine and IGWAY3 security[292] for an overview.
NHSD-Correlation-ID	14d68733-bd4f-4aae-b5ca-a3f783301b91..rrt-936033110107781752-b-geu2-18919-10347056-2	See Supporting API Consumers with tracing unique identifiers (see page 1104)
NHSD-Request-ID	14d68733-bd4f-4aae-b5ca-a3f783301b91	See Supporting API Consumers with tracing unique identifiers (see page 1104) supplied by API Consumer
NHSD-Identity-IdP	https://am.nhsspit-2.ptl.nhsd-esa.net:443/openam/oauth2/realms/root/realms/oidc[293]	Extracted from ID Token JWT issuer ("iss") claim. Not used within Spine

7.14.4.14 Proxies in Apigee and what is an API Producer responsible for

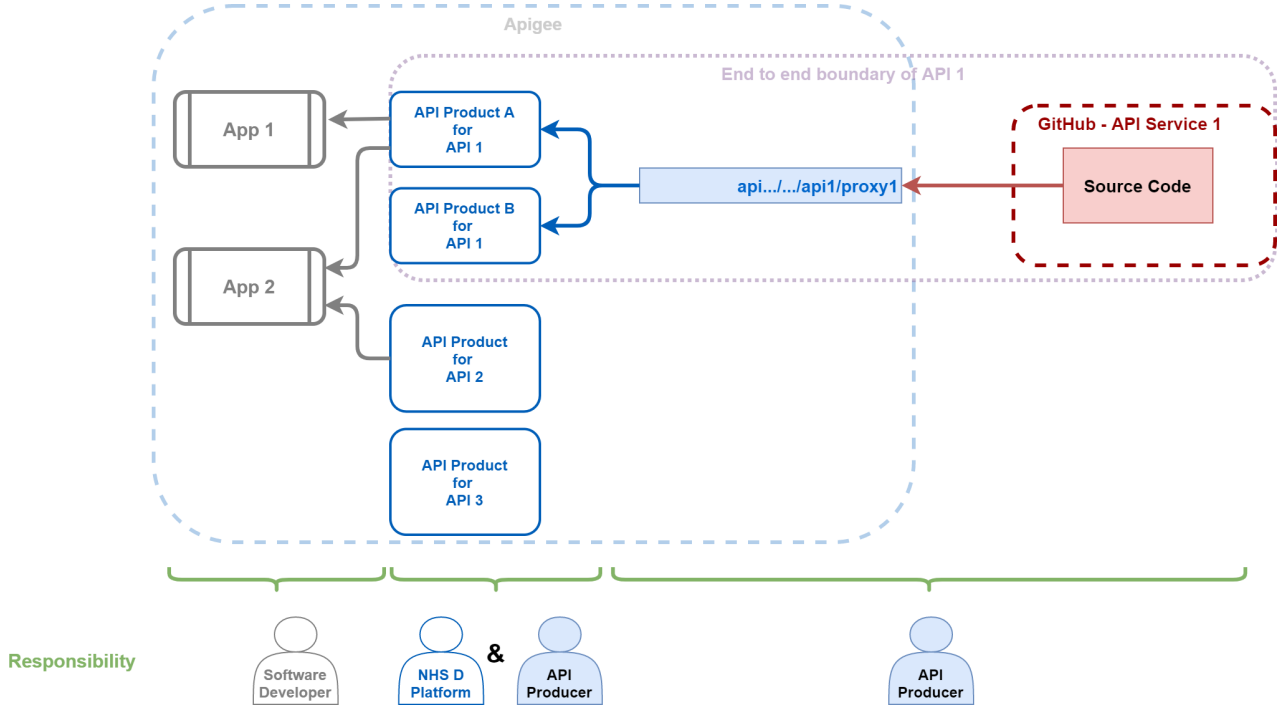
A Proxy is defined as one or more endpoints under a unique base URL. The work of the proxy is to support any Platform functionality and pass through the remaining items.

Just thinking about a "proxy" isn't enough to help publish an API. To address this, Apigee has three key concepts that help us (as NHS D) manage the platform. These same concepts also help Software Developers (API Consumers) manage their own applications:

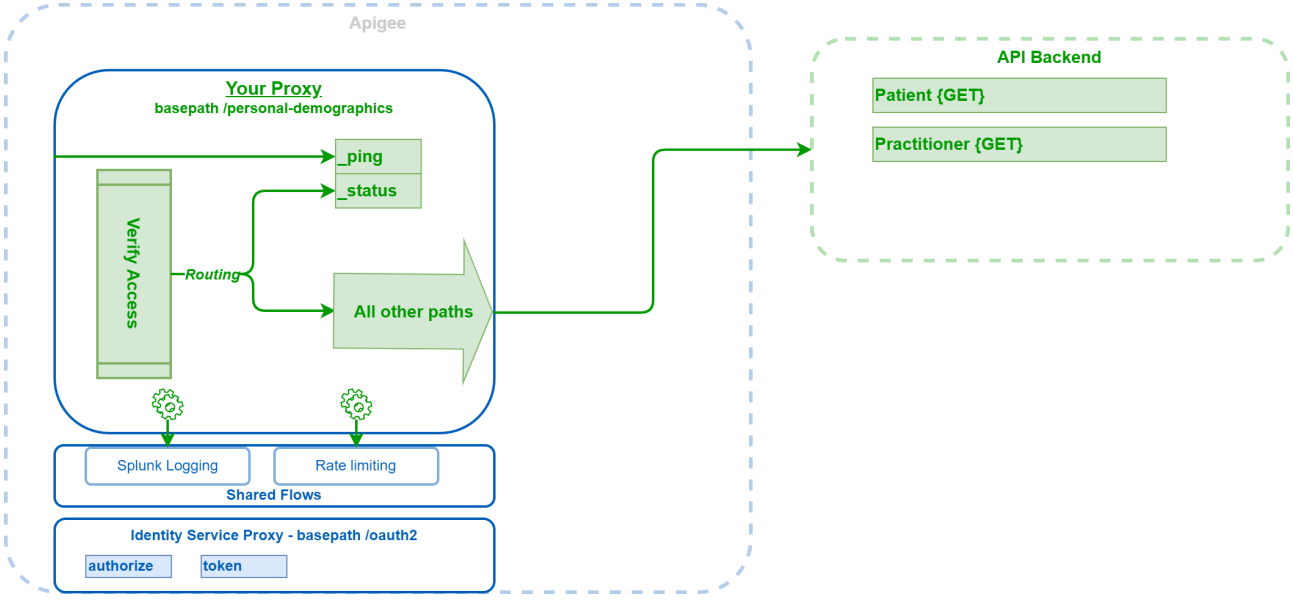
- **Proxies** - lowest level of control. This is the core entity where an API Producer manages detailed security and simple routing logic. Apigee enforces a uniqueness at the URL path level
- **Products** - "products" group proxies together to make it easier to manage, note the following:
  - The same proxy can appear in multiple products
  - We are using Products to help align technical access with onboarding and compliance assurance
  - Products are currently, primarily, dedicated to a specific API
- **Apps** - third party developers select one or more products for a specific application. App access can be managed by the NHS D team

Putting these in context, this diagram splits our who is responsible for which parts:

291 https://digital.nhs.uk/developer/user-restricted-restful-apis#user-completes-authentication-and-authorisation
292 https://nhsd-confluence.digital.nhs.uk/display/APM/Spine+and+IGWAY3+security
293 https://am.nhsspit-2.ptl.nhsd-esa.net/openam/oauth2/realms/root/realms/oidc



As with everything on the platform, API Producers control this via the config / code in GitHub. Focusing on a Proxy itself, the following diagram shows what an API Producer is responsible for from a configuration perspective - all the items in green:



Note: we recommend that an API Producer manages everything under that top level API name, by setting the actual proxy base path at that level - see Proxy base path vs API base path (see page 364)

7.14.4.15 Key elements of any API published on the platform

7.14.4.15.1 Overview

In order to provide the best possible experience for developers integrating with our API ecosystem as a whole, we would like to see all APIs exhibit the following attributes:

Consistent

With **other** APIs on the platform (see API Landscape[294] for our policies in this area).

This includes consistency in:

- Naming
- URL paths & query parameters
- Updates - use of POST vs PATCH, requirements around ETags
- Avoiding Asynchronous patterns

Firebreak (façade)

The API Management Platform manages the relationship with the API Consumer, which means that:

- Consider all the meta data / authentication being split between what you need as an API Producer, and the API Consumer
- The API Consumer will want to supply common information in a consistent way across **all** APIs
- You as an API Producer decide what information you need, you then inject that into your request from Apigee to your backend
- Apigee has a paper on the façade design pattern: https://pages.apigee.com/rs/apigee/images/api-facade-pattern-ebook-2012-06.pdf

Ownership

You as the API Producer will be self-sufficient.

The platform has extensive automation (which will be supported by us) to enable you to manage your proxy. Your team are responsible for the runtime once everything is set up.

Cookie Cutter

We want to help ensure all the APIs on the platform are similar enough to use the standard CI/CD pipelines, and to do that we have structured the relationship between key items:

- One Service (consisting of multiple endpoints - also termed "API")
- One Repo
- One OAS Specification (which has one API Catalogue entry, one sandbox / learning area)

294 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/123930121/API+Landscape

- Single owner of an Apigee API product (you can have multiple products but there should be no joint ownership with other APIs)
- *For a wider perspective on how these are related see:* APIs, proxies, repos and dashboards[295]

Hosting

How do you secure and route traffic from Apigee to your API backend?

Security of your API

APIM manages this on behalf of the API Producer, through the use of OAuth. To make integration easier, there is a multi-tiered approach to securing an API and the associated data:

- Public interfaces - minimal on-boarding and security
- Attended access - Clinician (NHS ID) or Patient (NHS login), referred to as user restricted
- Application restricted - sometimes referred to as: System to system; unattended; server to server
- Further detailed guidance here: Securing your API (see page 774)

Identity

- System (application) - every single request will identify who the third party software writer is
- Attended or unattended access
- Delegation to another person
- Mechanism to address Spine unique system identities (ASIDs) mapping
- Transmission of that identity to your solution in a format you want

Authorisation

- It is critical to design your API to work with the many authentication / authorisation mechanisms supported by the platform. This is made simple to do, simply work with the APIM implementation patterns and the Apigee OAuth service
- On **production**, there is a very simple system authorisation. Each **App** has to be manually approved, as the result of a SCAL based on-boarding process. See https://digital.nhs.uk/developer/guides-and-documentation/onboarding-process
- (future) Coarse grained authorisation (access control at either a system or person level) to reduce the assurance burden on API Consumers
- (future) Linking into the National framework for authorisation (which will transition from the existing RBAC to a new system)

Logging

All proxies wired to deliver logs to the NHS Digital Cloud Splunk instance, see Logging to Splunk (see page 1082)

295 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125409294/APIs+proxies+repos+and+dashboards

Analytics

Splunk is the primary NHS Digital tool for this, and there is dedicated Splunk app for APIM where API Producers can build reports and dashboards.

You can also access the Production redacted (open access) indexes from any app in Splunk, so you can integrate APIM data with your own.

7.14.4.16 Privacy considerations in your design

7.14.4.16.1 Overview

APIM should be thought of as a "utility" that the API data passes through, so you don't need to understand the underlying workings - nor define them in your service DPIA. As part of gaining approval to go live, API Producers often ask about the APIM assurance. You can cross reference to the key APIM documents in this area:

- Information Asset (IA 0770[296])
- Data Protection Impact Assessment (DPIA 0143[297])
    - There is also a DPIA for the Digital Onboarding Service (DOS DPIA 0193[298])
- System Level Security Policy (SLSP 0289[299])

7.14.4.16.2 Considerations

Whilst APIM provides the underlying platform, you must still consider the following questions / aspects when designing **your** service.

> Remember to try and document the outcomes of these decisions in your Confluence space - later stages such as Service transition for APIs (see page 947) and API penetration testing process (see page 968) can go faster if you can link to existing pages

7.14.4.16.2.1 Does the service you are publishing via APIM already have a confirmed class of data?

- Cyber Security have a guidance spreadsheet that helps you to rank data from Class I (least sensitive) to Class V (most sensitive)
    - If you have completed a Technical Architecture Solution Design Overview (SDO), this has the classification spreadsheet as part of the process

296 https://unifiedregister.digital.nhs.uk/InformationAsset/Display?_recordGUID=b7ce497e-5fda-463b-8eac-1da9e53bec96
297 https://unifiedregister.digital.nhs.uk/DPIA/DisplayDPIA?_responseGUID=83954642-03b2-4dd0-a873-9166069d0825
298 https://unifiedregister.digital.nhs.uk/DPIA/DisplayDPIA?_responseGUID=743245af-ee92-41fb-a087-76e50bf964ac
299 https://unifiedregister.digital.nhs.uk/SLSP/Display?_recordGUID=cbe6b49e-04d7-4ee5-bc02-8e62af275887

    - You can read more here cloud risk framework[300] and here data risk model[301]
- The APIM Management platform has been designed and approved to contain Class V data

7.14.4.16.3 Data minimisation

A key principle in GDPR, and general best practice, is to keep the amount of information you pass out from your API to a minimum. This is mostly about protecting the confidentiality of the data - the less people that have access and/or a copy of your data the better. There are a number of other ways, that this design principle also helps with the wider design:

- The payloads are smaller, which helps with responsiveness
- Easier to understand
- Helps with designing your APIs to be atomic
- It can enable wider use of your API - a smaller dataset reduces the sensitivity of data allowing you to have different tiered access modes. The highest security is only required for the largest / most sensitive dataset

When you come to your pen testing, if you've documented how you've minimised data you can cross reference it from your Evidence template for Cyber to mitigate an external Pen Test (see page 978)

7.14.4.16.3.1 URL path and query parameters

Consider the values in the URL path or query parameters - do any of them contain Personally Identifiable Information / Data (PII or PID) or sensitive data?

- This is an **expected** (and **accepted**) impact of the RESTful model, there is no need to change your API design if (by doing so) it goes against REST
    - REST APIs are the NHS England preferred interaction pattern[302]
    - The FHIR REST standard[303], itself, follows the RESTful model and proposes unique identifiers in GET requests
    - Remember that, for many APIs, the percentage of traffic this applies to might be quite low:
        - Only some endpoints, that make up an API, will have such information
        - The most sensitive data will be inside the payload
    - You should (still) keep the number of these type of fields to a minimum
- With the TLS standard, these values **are** encrypted in transit and it is only at the point of decoding that these values could be logged

---

300 https://digital.nhs.uk/data-and-information/looking-after-information/data-security-and-information-governance/nhs-and-social-care-data-off-shoring-and-the-use-of-public-cloud-services/cloud-risk-framework

301 https://digital.nhs.uk/data-and-information/looking-after-information/data-security-and-information-governance/nhs-and-social-care-data-off-shoring-and-the-use-of-public-cloud-services/cloud-risk-framework/risk-framework

302 https://digital.nhs.uk/developer/architecture/integration-patterns-book/interaction-methods#synchronous-interaction

303 https://www.hl7.org/fhir/http.html

- As noted above all components of the API Management ecosystem are designed to securely deal with any such parameters:
    - The DPIA is fully approved
    - An example is that, in Production, the raw logs are only accessible to staff who have SC clearance. APIM then makes the logs available to the rest of NHS Digital by redacting the data into separate indexes
- You can see this in many existing APIM APIs, e.g. Personal Demographics Service (PDS) FHIR API[304]

7.14.4.16.3.2 Are existing transparency notices (how NHS Digital exposes privacy notices for a specific party) required to be updated?

- Generally the addition of the APIM platform to your solution does not require a change in any existing transparency notices:
    - APIM is part of NHS Digital, so that data does not change the Data Processor (or Controller depending on the API)
    - Notices should be in simple language, and this level of detail would not be useful for many readers
    - For instance, the Spine PDS API transparency notice (after review) was decided to not be changed
- Many solutions now use cloud "utility services" such as Content Delivery Networks (CDNs) to improve speed and protect against Denial of Service (DOS) attacks. These cloud providers decode every request, log key details such as URLs and have potential access to every part of every request
    - These platform are often not called out at high level privacy / transparency notices, and APIM continues with this pattern
- The final decision must come from your representative in Privacy, Transparency and Ethics (PTE)

7.14.4.17 Spine based APIs

7.14.4.17.1 Overview

As an API Producer you might be creating a proxy for, or building a new lightweight hosted container, which calls out to the Spine APIs. The APIM Platform is looking to make integration with Spine easier for both Consumers and Producers. As an API Producer the key elements you need to consider are:

- Apigee (your proxy config) will inject the relevant Spine details (i.e. an ASID) into the request to support existing audit and compliance requirements
- API Consumer will not need to use TLS MA Certificates, but the use of OAuth standard to secure connections to APIM
- Connecting to Spine from the Apigee ecosystem will be managed by the APIM Platform design patterns and the APIM CD process

304 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#get-/Patient/-id-

- API Producers will not have to retrieve or issue TLS MA Certificates for Spine connections, even in the APIM AWS Hosted Containers

If you do not have Spine experience, talk to the Spine team to help you get started. This page will help with some key terms: Spine and Accredited System Identifier (ASID)[305]. You can then read more details: Spine and IGWAY3 security[306] and Spine Core HL7 FHIR API Standards[307]

7.14.4.17.1.1 Passing information to Spine

Spine requires a number of fields, in addition to the payload. There is a work to do this in a standard way, but it is not complete yet. See Passing information to your API backend (see page 611) for general information, and the current state of Spine backend integration design.

7.14.4.17.1.2 Unique system identifiers - ASIDs

Spine relies on Accredited System Identifiers (ASIDs) for system level authorisation and audit requirements. APIM is looking to de-couple API Consumers from having to manage ASIDs, party keys and associated certificates. As this strategy is being worked through, an API Producer must still be aware of ASIDs. Key points:

- The ASID passed to Spine (in this context) is always the API **Consumer** ASID (Connecting Party), so as an API Consumer you do not have to manage the ASIDs
- For API Management API Consumers - the ASID per End User Organisation (EUO) is no longer required. During design of your API you should consider the following questions to determine what common APIM and Digital Onboarding features you will use:
  - Does the API need to keep a record of which individual accessed the data or service?
  - Does the API need to keep a record of which organisations accessed the data or service?
  - Does the API need to actively reject a request based on the organisation making the request?
  - The implementation of the features required to meet those aspects isn't finished yet - but a full reasoning and solution description is here: D060 - Authorisation for End User Organisations[308]
- Connections via APIM are not subject to the normal Spine Party Key checks - but the other existing system authorisation checks do occur (e.g. interaction IDs must be correct)
- **One to one** linkage between Apigee App and a Spine ASID

You **must** design your onboarding, and authorisation models to focus on a single Apigee App. If you do not do this it has a wide ranging effect on how an API Consumer connects to other APIs

305 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114037475/Spine+and+Accredited+System+Identifier+ASID
306 https://nhsd-confluence.digital.nhs.uk/display/APM/Spine+and+IGWAY3+security
307 https://nhsd-confluence.digital.nhs.uk/display/APM/Spine+Core+HL7+FHIR+API+Standards
308 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/540181015/D060+-+Authorisation+for+End+User+Organisations

7.14.4.17.2 API Producer responsibilities when interacting with Spine APIs

As an API Producer, if you own the Spine backend you **should** understand what ASIDs are used for in your own API. However, authorisation settings rarely change, and the team knowledge of this might reduce over time.

Key responsibilities:

- Hide the concept of a Spine ASID from your API Consumers, where possible
- If you have more complex use cases, then make sure your API adopts the Single ASID model (see page 444)
- In the INT environment, your proxy should make use of the generic ASID.
    - This means that a new API Consumer can use your API **without** contacting NHSE - reducing friction and saving all parties time
    - Do this by using the shared flow (see page 418) to automatically inject it
    - Proxygen does **not** support this currently
- Nominate a tech person in your squad to manage and document how ASIDs are permissioned and used in your API
- Support the Onboarding lead to make a simple process to provision ASIDs for your API Consumers

7.14.4.17.3 As an API Producer do I need my own ASID?

Most API Producers do not need their own ASID, because the ASID is there to help manage authorisation of the **API Consumer system** not the owner of the backend. There are two situations where you as an API Producer might need to request and manage their own ASID:

- Do you undertake any associated Spine processing?
    - Looking up against a Spine API that the API Consumer should have not access (i.e. it's ASID should never have message sets)
    - e.g. checking that an NHS number is valid *(to update: need a better example, as a PDS trace could be done under an API Consumer ASID)*
- Testing - your API should work first time for API Consumers, not fail because their Consumer ASID was setup incorrectly. To do that setup correctly your team needs to understand how an ASID for your API needs to be setup

7.14.4.17.3.1 ASID use in different environments

There are a few implications on how ASIDs are used in different environments:

- In Path To Live environments:
    - An API Producer should have **their own** ASID to allow them to ensure the Message Set / Interaction IDs are fully know and understood

    - For each environment, there is a generic APIM ASID. Once the API Producer team has their first of type consumers in INT, post on API Producer Support Channel to have the Message Set(s) added to the APIM ASID (they will use KOP-025[309] to do this).
- In Production:
    - An API Producer should have **their own** ASID to allow a **smoke test** of their API
    - Each API Consumer will need an ASID - the process is **not** finalised yet, but see Spine and Accredited System Identifier (ASID)[310]

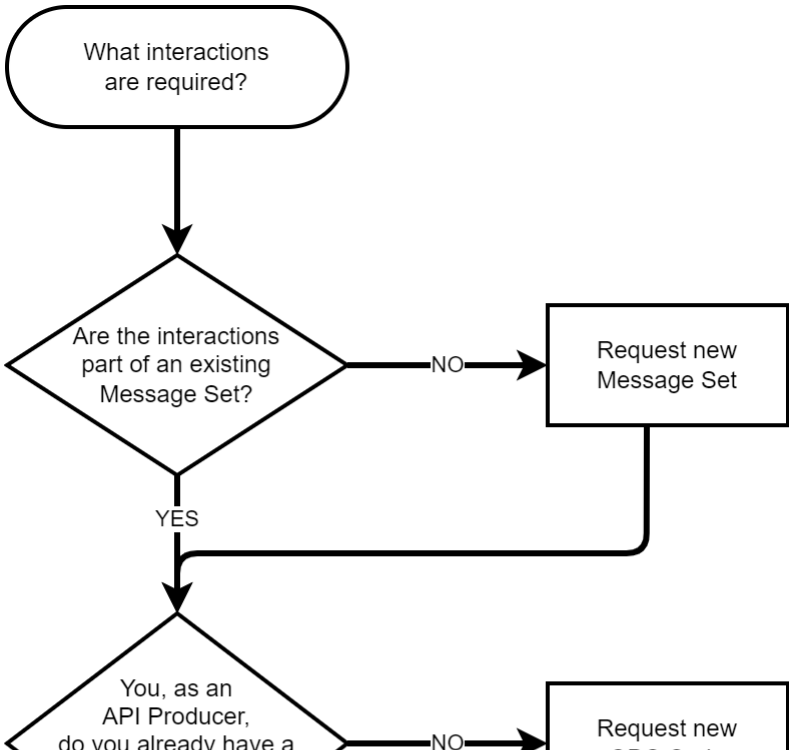
There some further details to be understood when you are considering configuration and testing - see Environments (see page 758) page for those details.

7.14.4.17.3.2 How to request your own API Producer ASID

The ASID requires a number of other items to be configured before it can work:

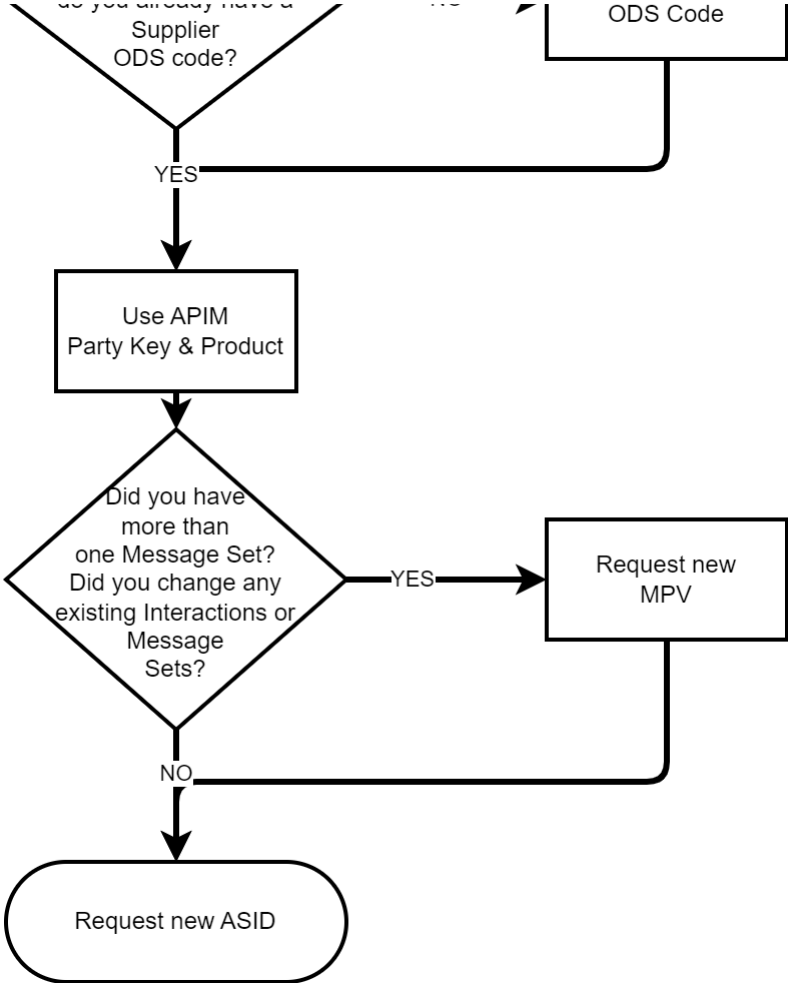
- Interactions - Spine authorisation tags
- Message Sets - API Producer grouping of Interactions
- MPV (Manufacturer Product Version) - a grouping of Message Sets, under the APIM Product
- Party Key - generally the APIM Party Key will be used

This is the basic flow:



309 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/203358467/KOP-025+Updating+generic+ASIDs+with+new+message+sets

310 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114037475/Spine+and+Accredited+System+Identifier+ASID#SpineandAccreditedSystemIdentifier(ASID)-new-consumer-asid



Requesting a new message set or MPV

The DIR team have to do this (simply email dir@nhs.net[311]), and the new Message Set will initially be released to all Path to Live (PTL) environments. Once testing is done, then a release to Production will have to be scheduled.

Path to live - requesting your own ASID

> This external page has key information out working in Spine PTL environments, with links to detailed environment steps:
> https://digital.nhs.uk/services/path-to-live-environments/connect-to-a-path-to-live-environment

311 mailto:dir@nhs.net

The ITOC team manages PTL, to start the process use this form: https://digital.nhs.uk/forms/combined-endpoint-and-service-registration-request

On the form fill in the follow details:

No.	Section	Description
1	Contact Details	No login details - simply fill in your contact details
2	What to do	Option 3 - "Other endpoint related activity"
8	Other information	For INT (or DEV) can an additional ASID be created: Recipient ODS Code = "T141D" Recipient Name = "NHS Digital" Manufacturer ODS = "X09001" Associated Party Key = "T141D-822150" **MPV** If you already have an MPV, then state it's name here. However, if it needs to be created request it to be setup in this request, using the format: • Product = "API Management Platform" • Version = "<API Producer Name> V1" (or similar version) • Message Set(s) = list the names of the message sets Category Bag = none

NOTE: As you are connecting via the APIM platform the FQDN is not relevant, and you don't need a certificate (and the associated CSR process).

Shared ASID

During development, API Management provide a "Generic ASID" that allows you to get started with testing while you're waiting for your ASID request. See: Injecting ASIDs for Spine APIs (see page 418)

7.14.4.17.3.3 Production - requesting ASIDs

First time into Production

If you are deploying your proxy for the first time, you need to make sure you do the following:

- Work with the DIR team to create and/ or make sure the correct versions are in Production:
    - Interactions
    - Message Sets
    - MPV
- Go back and ensure the APIM Generic ASID is updated

This will allow you to do smoke testing and to setup your API Consumers.

Requesting ASIDs from the DIR team

The DIR team manage Production configuration, and there are two types of ASID requests:

- For your API Consumers - this will be the vast majority of requests
- Smoke testing in Production

In both instances the request is an email to dir@nhs.net[312] (Attention of Martin Slater, and cc to aubyn.crawford1@nhs.net[313] so I can support if needed), with the following key details:

312 mailto:dir@nhs.net
313 mailto:aubyn.crawford1@nhs.net

	Value to send over to DIR

ODS Code (Recipient)	All external consumers will have (or will be asked for an ODS code in the SCAL process). If they don't have one, then they can use this form: https://digital.nhs.uk/services/organisation-data-service/update-your-data/forms#organisation-code-request
	Internal project teams:
	• This **includes** external consultancies that are embedded in NHS Digital (e.g. all staff have Azure AD short code @hscic.gov.uk[314] accounts)
	• APIM Management have to request a site code for the ODS org 'Internal NHSD Programme ODS Code Request' (special dispensation from the ODS teams for APIM Consumers - see Compliance touch points[315]):
	  • Use 'Site Code Request Form' here: https://digital.nhs.uk/services/organisation-data-service/update-your-data/forms#site-request
	  • In the "Organisation HQ Name" put 'Internal NHSD Programme ODS Code Request', which is the "X26" code
	  • Address should be "7-8 WELLINGTON PLACE, LEEDS, WEST YORKSHIRE, LS1 4AP", which is a duplicate but the ODS team expect that
	• The process is quick
	International (global) ODS codes are not formally supported yet, but investigations started Speak to Aubyn / Alex Lord - also see Compliance touch points[316]

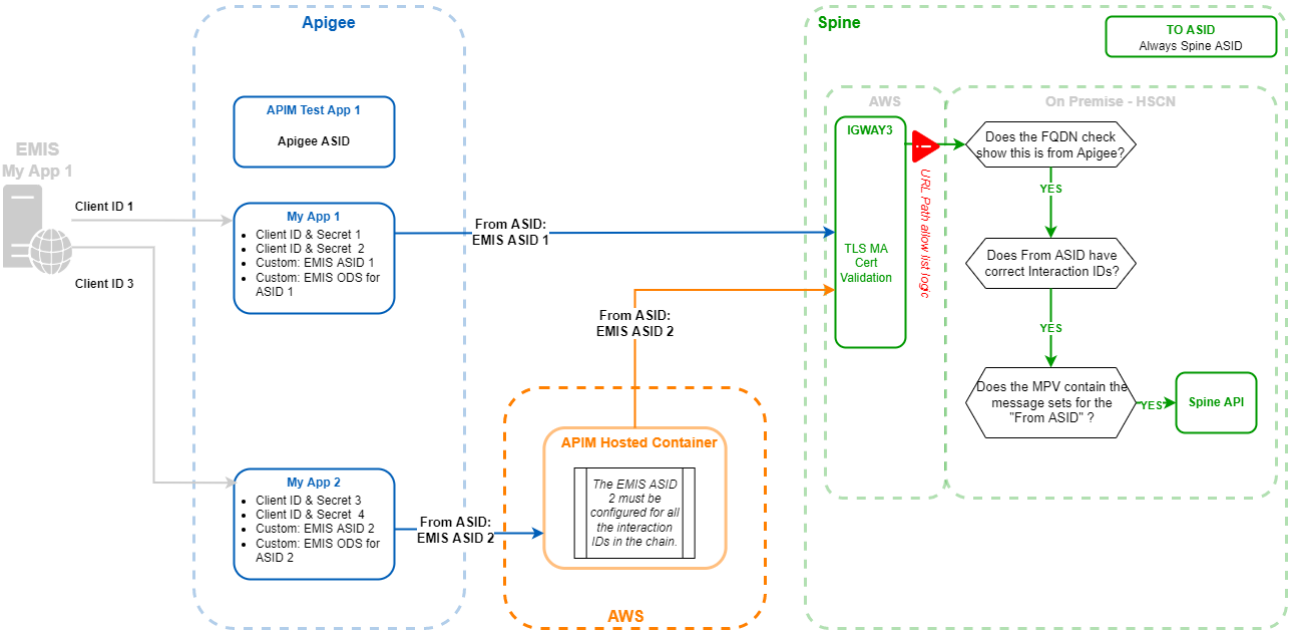
Name of Developer	If they already have an ODS code, the name should be the same - lookup using https://odsportal.digital.nhs.uk/Organisation/Search
Application Name	Name of the application that will hold the ASID
APIM Party Key to Use	X26-816405 *(MHS Party Key setup specifically for smaller APIM Production Consumers)* *** **WHEN** e-RS are ready for first Production ASID then need to confirm how this works against the way e-RS currently does in HSCN - Synapsis currently have 14 sites ARX02 ***
SCAL or LSB Approval	Attach to the request the approvals from either the SCAL process (via the Onboarding Lead) or from LSB. If the process is a Smoke Test include the IAO approval as well - for the exception process that needs to be followed for this see Smoke testing (see page 1145)

314 http://hscic.gov.uk
315 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182976643/Compliance+touch+points
316 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182976643/Compliance+touch+points

	Value to send over to DIR
MPV	The agreed process is to request an **MPV** at this point - often there are discussions about Interactions and Message Sets. However, working at the MPV level adds some clarity, and reduces the need for multiple emails. *** NOTE that over time an ASID will require more permissions to be added, so an API Consumer might start with (Personal Demographics Service) PDS API, but then add Electronic Prescription Service (EPS), or e-Referral Service (e-RS) or Summary Care Record (SCR). At this point the DIR team might choose to create a new MPV to incorporate multiple MPVs ***
Cat Bag	No additional values

7.14.4.17.4 Deep dive into ASID processing

For those interested, an example of how ASIDs are processed:



This diagram is from the Spine and IGWAY3 security[317] page.

317 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141201006/Spine+and+IGWAY3+security

7.14.4.17.5 What does this mean for your API Consumers?

As the number of API Consumers is very limited, all the processes are manual, with options to automate. The intent is that as the Digital Onboarding Service develops, the ability to manage Spine based permissions will be incorporated and automated.

Until that time there a scalable solution is in place for onboarding, these are some considerations:

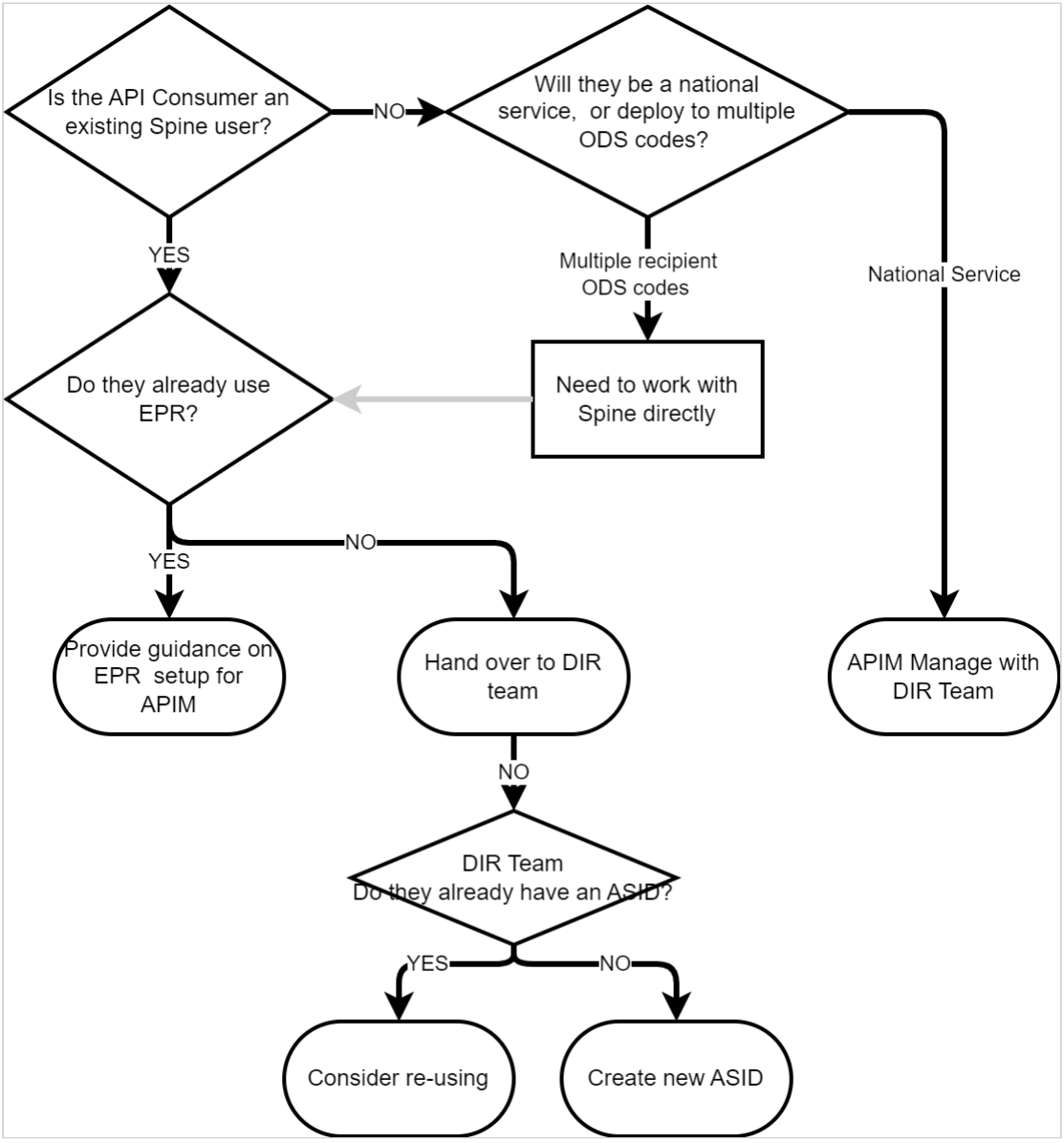
- As an API Producer you create a SCAL for on-boarding consumers - this process needs to include how ASIDs are managed for consumers. Key pages covering this:
  - Spine and Accredited System Identifier (ASID)[318]
  - API consumer onboarding (see page 1005)
  - Service transition for APIs (see page 947)
  - D014 - System Unique Identifiers, ASIDs and simplifying on-boarding[319]
  - D009 - How should End User Organisation be represented / passed by the API Consumer?[320]
- Attaching an ASID to an Apigee Application is currently a request on the Producer Support channel
- Where an API Consumer has **multiple ODS** codes of end organisations, each ODS code will require it's own App with ASID for that ODS code attached. This is **not** a scalable model - when the first use case that involves this arises we will work on the many options available to look at managing this.

The following flow chart is a guide for the Onboarding Lead to work with API Consumers and their Spine authorisation needs. Some API Consumers will already be familiar with managing ASIDs, and that could remain:

318 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114037475/Spine+and+Accredited+System+Identifier+ASID#SpineandAccreditedSystemIdentifier(ASID)-new-consumer-asid

319 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110869210/D014+-+System+Unique+Identifiers+ASIDs+and+simplifying+on-boarding

320 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110866420/D009+-+How+should+End+User+Organisation+be+represented+passed+by+the+API+Consumer



7.14.4.17.6 Injecting ASIDs for Spine APIs

Some back-end systems (notably, Core Spine systems) require an Accredited System Identified (ASID) passing with API requests. The ASID identifies the calling app, and is used in the production environment for audit purposes.

Spine requires your API Proxy to pass the ASID to the backend in the XAPI_ASID header; ordinarily this header is stored as a custom Apigee attribute on the calling app, however API Management also provide a generic ASID in PTL for you to use for self-service testing in PTL when they don't have their own. The shared flow below will pull the ASID from the calling app and add the required header, with an option to fall back to the generic ASID in PTL.

Some Spine APIs might not use this header.

7.14.4.17.6.1 Explanation

This flow attempts to resolve a value for the XAPI_ASID header required by Spine using the following order of precendence:

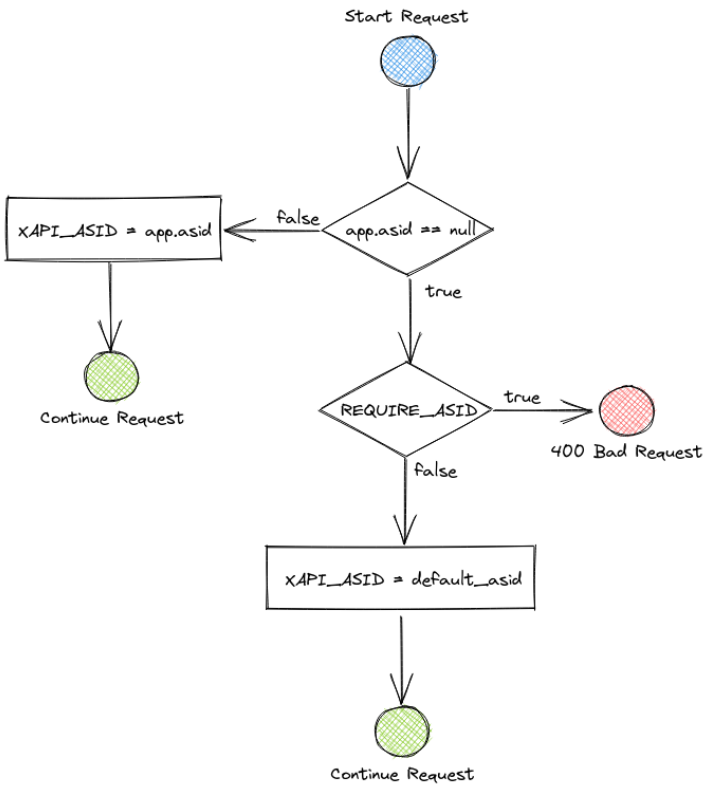
1. ASID configured by the calling app as an attribute in Apigee
2. API Management's Generic ASID

The diagram on the right illustrates the process; you can set the `REQUIRE_ASID` parameter at either the environment or proxy level in `azure/azure-release-pipeline.yml` - an example can be found here[321].

You can implement this through a shared flow; or manually if you need to change the default functionality.

The generic asid functionality **cannot** be used in Production, and as a safety measure we do not populate the `default_asid` value in Production. If you are using this flow, please ensure that you have the `REQUIRE_ASID` value set to `true` in Production, or set it at the Proxy-level and only override it when required explicitly.

321 https://github.com/NHSDigital/summary-care-record-api/blob/master/azure/azure-release-pipeline.yml



7.14.4.17.6.2 Shared-Flow Implementation (Recommended)

Add this policy to your proxy:

policies/FlowCallout.GenericAsid.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.G
enericAsid">
    <DisplayName>FlowCallout.GenericAsid</DisplayName>
    <FaultRules/>
    <Properties/>
    <Parameters>
      <Parameter name="RequireAsid">{% if REQUIRE_ASID == 'True' %}true{% else %}
false{% endif %}</Parameter>
    </Parameters>
    <SharedFlowBundle>GenericAsid</SharedFlowBundle>
</FlowCallout>
```

Add a step to your pre-target flow:

```
<Request>
```

```
  <Step>
    <Name>FlowCallout.GenericAsid</Name>
  </Step>
</Request>
```

Add the REQUIRE_ASID jinja_template var to your Azure pipeline steps.

```
apigee_deployments:
  - environment: internal-dev
    make_spec_visible: true
    post_deploy:
      - template: templates/tests.yml
        parameters:
          service_name: ${{ variables.service_name }}
    jinja_templates:
        REQUIRE_ASID: false
  - environment: internal-dev-sandbox
```

7.14.4.17.6.3 Manual Implementation

Add this policy to your proxy:

GetSharedAsid.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<KeyValueMapOperations async="false" continueOnError="false" enabled="true" name="Key
ValueMapOperations.GetDefaultAsid" mapIdentifier="secret-squirrel">
    <Get assignTo="private.apigee.NHSD-ASID" index="1">
        <Key>
            <Parameter>default_asid</Parameter>
        </Key>
    </Get>
</KeyValueMapOperations>
```

This policy will grab the `default_asid` value from the platforms KVM and store it as `private.apigee.NHSD-ASID`. If you have a generic ASID that you'd like to use in place of the API Management Generic ASID, replace this policy with your own `KeyValueMapOperation`.

Add this snippet into your target config:

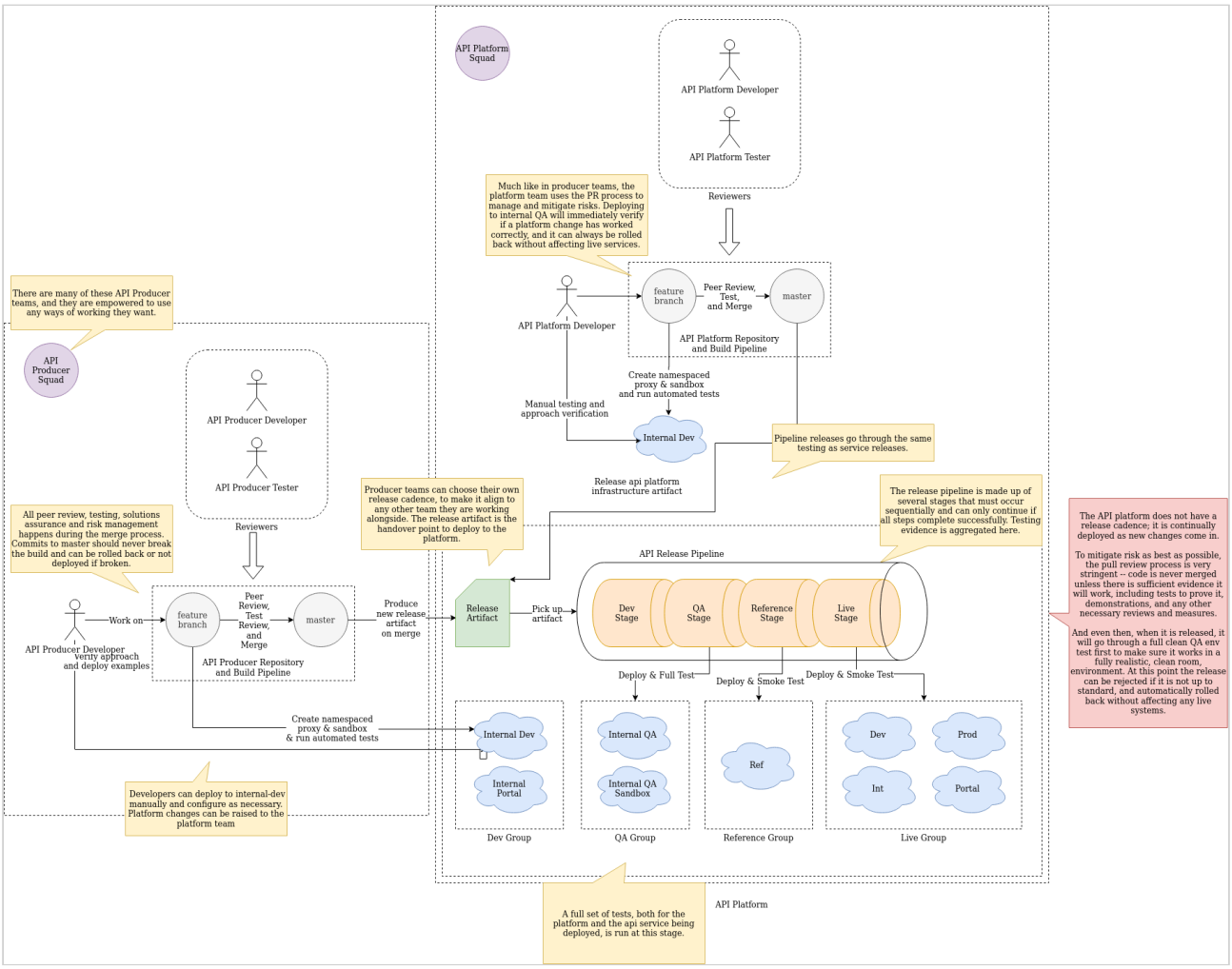
```
<!-- If we require an ASID, fail if app.asid is null -->
{% if REQUIRE_ASID == 'True' %}
```

```
<Step>
  <Name>RaiseFault.400BadRequest</Name>
  <Condition>(app.asid is null)</Condition>
</Step>
{% endif %}
<Step>
  <!-- Populate the ASID from the app -->
  <Name>AssignMessage.PopulateAsidFromApp</Name>
  <Condition>(app.asid isNot null)</Condition>
</Step>
<Step>
  <!-- Populate the ASID with the default ASID -->
  <Name>KeyValueMapOperations.GetDefaultAsid</Name>
  <Condition>(app.asid is null)</Condition>
</Step>
<Step>
  <!-- Add the ASID as a header -->
  <Name>AssignMessage.AddAsidHeader</Name>
</Step>
```

This snippet will check if your app requires a non-generic ASID with `REQUIRE_ASID` and fail if it hasn't been provided; it will then retrieve the ASID from the App if one is provided, otherwise it will fall back to the generic ASID.

7.14.4.18 Continuous deployment target model

7.14.4.18.1 Deployment Model



For details on the AWS Hosted Containers flow - see DRAFT - Deployment isolation (see page 432).

7.14.4.18.1.1 Design concepts

(Check against Laurence PPT for better ideas / wording here):

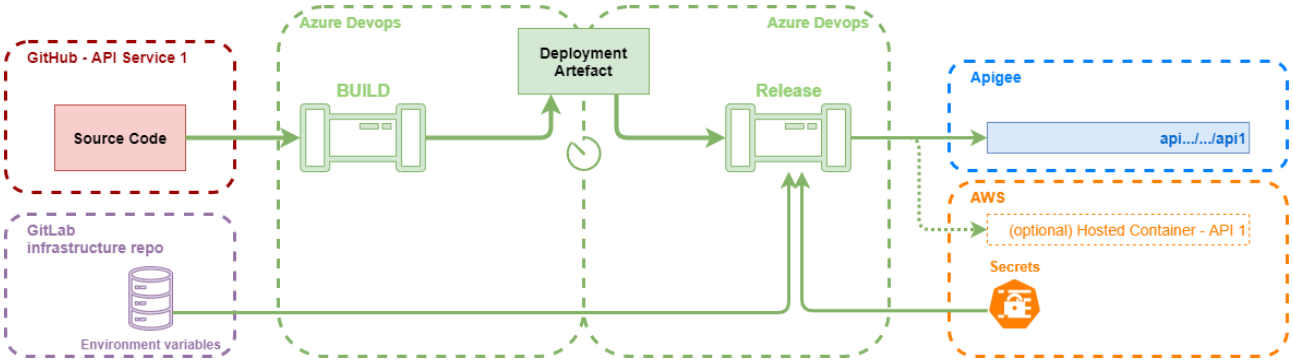
- Azure DevOps is the central coordinator of the CD process
- Common pipeline concept:
  - Stores common scripts in GitHub
  - These scripts carry out a specific task in the CD process
  - Azure DevOps uses a boostrap concept to use the correct script

7.14.4.18.1.2 Common pipelines

The common pipelines scripts covering the following areas:

- Build
- PR
- Deploy service task
- Release

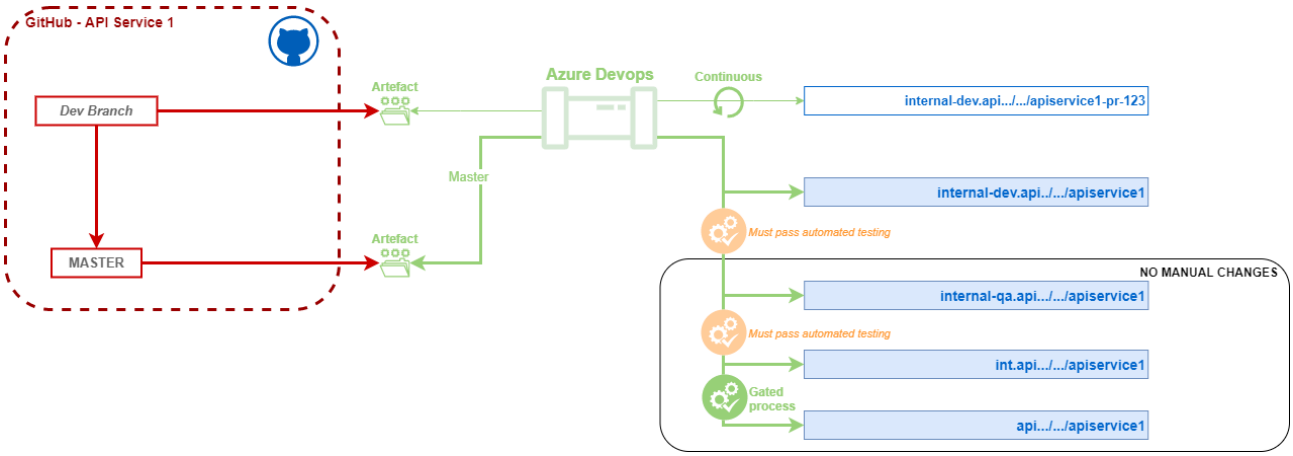
7.14.4.18.1.3 Simplification



Future:

- Bring utils code (which forms part of the release) into the deployment artefact (this might be done with versioning of the utils repo).

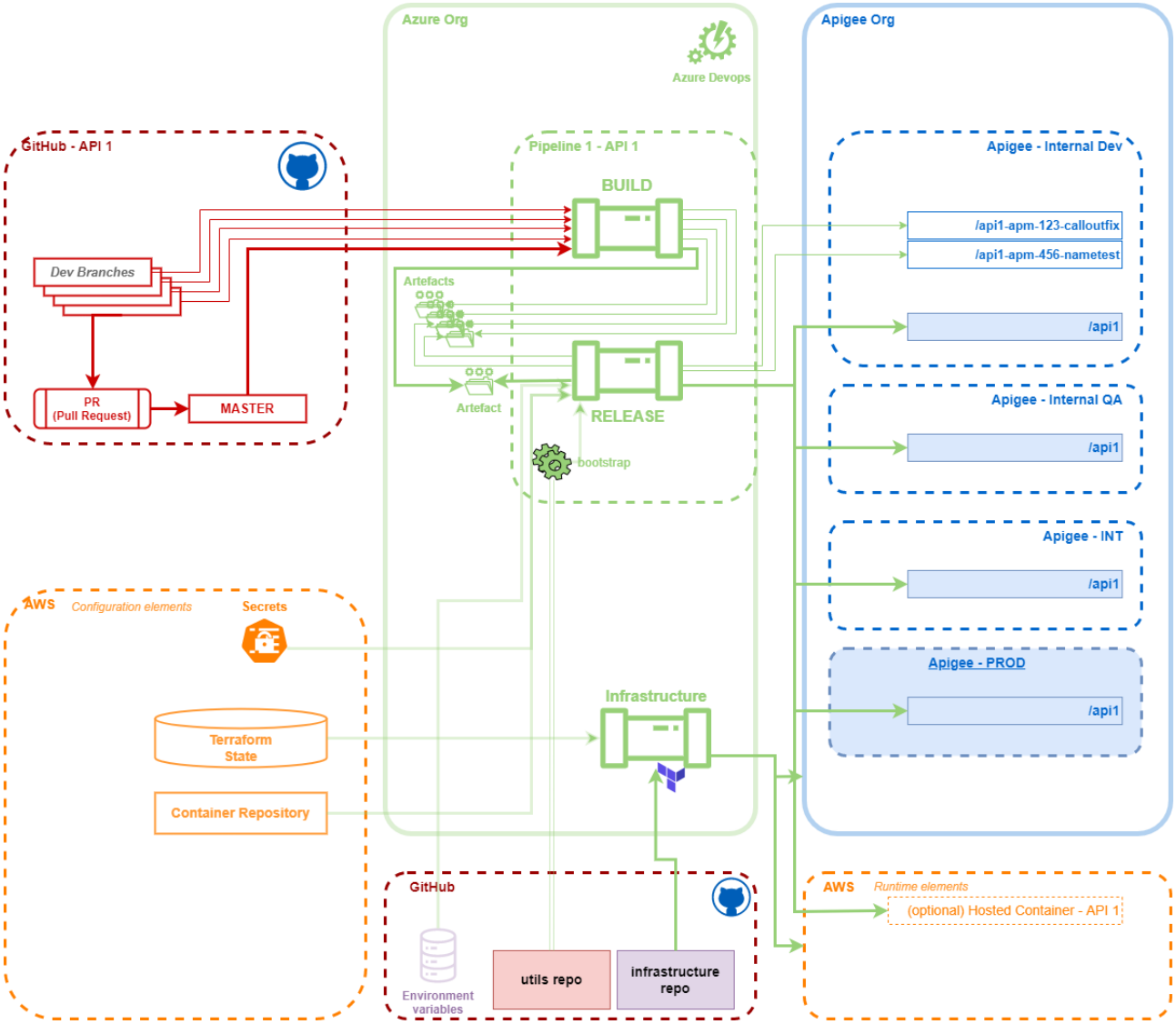
Focusing on the flow through multiple environments:



- Build process:
  - On any update to master a new tag (with a calculated version number) is automatically created
  - Azure pipelines picks up the tag and uses that tag to run the build process, ensuring consistency and tracability in build where multiple updates come along in quick succession
- Gated processes
  - In the lower environments, the deployment to the next environment is halted if all the testing does not pass

- Production releases are always gated, and actual releases occur Tuesday and Thursday nights
- API Producers can choose to implement manual gates on INT
- For details about how this operates see: Using manual approval on pipelines (see page 913)

Next level down of detail, although this still excludes the AWS Hosted Containers:



7.14.4.18.1.4 Goals

Goals

- To have all configuration applied in a set of automated processes
- Ensure that sensitive variables and secrets are kept secure
- Contain all non-environment related aspects are in source code
- To have tested the release process multiple times prior to first application on production
- To demonstrate (or not!) that we can achieve zero down time

Repos in use:

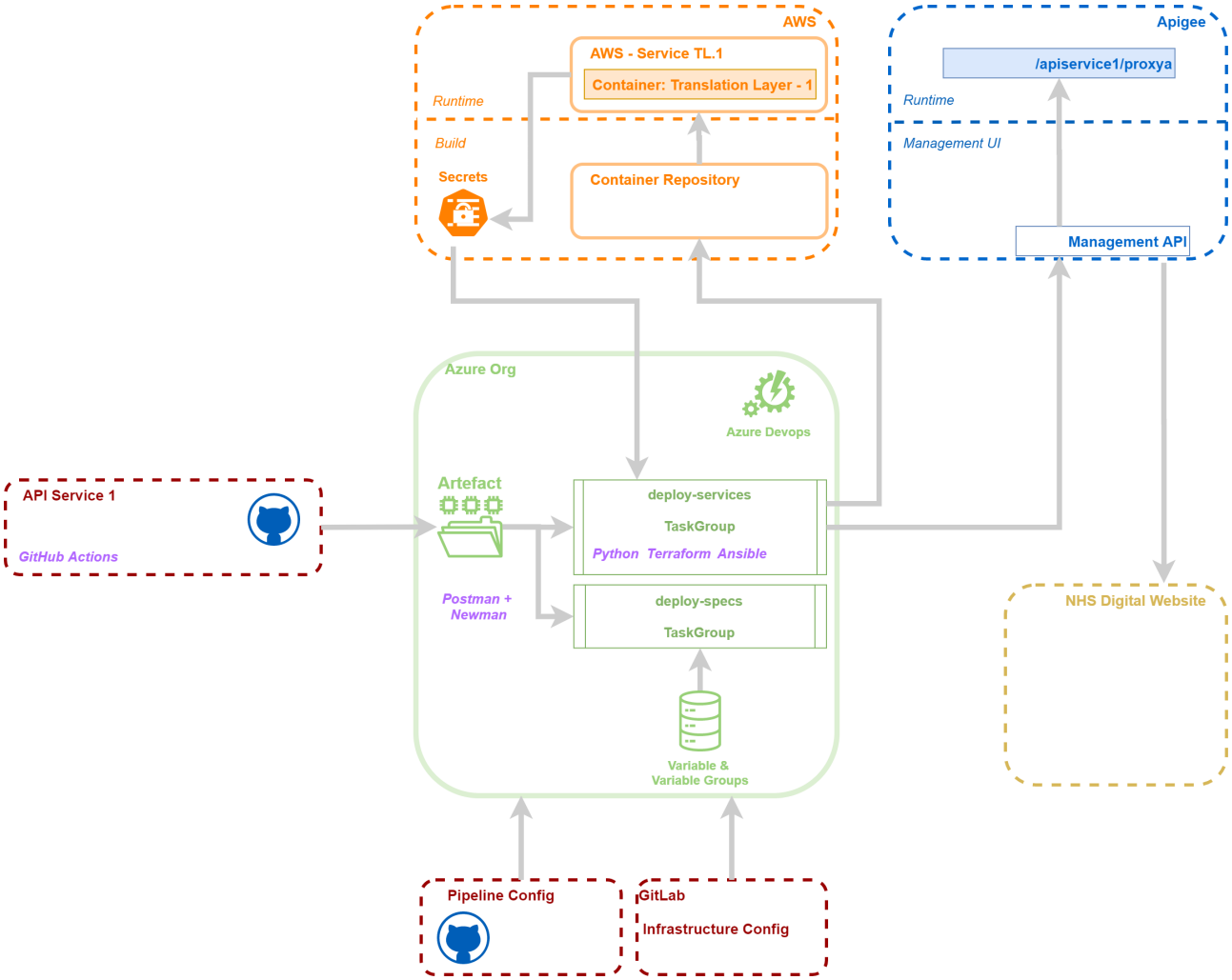
Name			To Do	Pipeline
utils	Git Hub	Generic reusable components for deployment to Apigee and AWS. Effectively this is where the common pipeline "lives"	No	Release
infrastructure	GitLab	Environment specific - anything is NOT a per project asset in Apigee • KVMs; target servers; shared flows AWS Configuration - Terraform, secrets management	• Minor amounts of Terrraform left in Azure DevOps • Some proxy related config	Infrastructure (excludes Terraform, which is currently manually run)
template	Git Hub	Contains templates for initial build of CI/CD repos and pipelines		Manual

Top level the platform must have (**tbc**!) to following to achieve the goals:

- Proxy and spec artefact
- YAML of the Build pipeline
- YAML of the Release pipeline
- Variables and secrets held securely

Aspects such as Terraform state, are part of the tooling, but are (tbc) driven and controlled by the above

7.14.4.18.1.5 Deployment pipeline tools



Secrets:

- Secrets manager stores the most sensitive values, and is only pulled out during release to AWS. Stores historic values.
- SSM - is used for environments variables, and some secrets, but low level ones and there is no historic version s

7.14.4.18.1.6 Configuration

With the common pipelines, the configuration values have moved away from Azure Devops, leaving:

- KVM config is in YAML
- Proxy config (XML) **should not** need to have any values in it that change per environment

- Non-sensitive environment variables are in GitLab
- SSM for other non-sensitive environment variables
- Secrets manager (can contain KVM sensitive values)

Previously Azure Devops, used the built-in, but the common pipelines removes their use - so these will no longer be required:

- Pipeline variables
- Environment
- Variable Groups
- Variables

UNLESS you are using / require custom values which you specify through PR (todo: Ben S to put page up)

Remaining items:

- How to provide environment isolation?
- Tidy up
- How to version control the settings?
- How to deploy the settings?
- Guidance on how to make sure that Proxy XML Config code doesn't contain env. values.

Breaking down into detail, the following table sets out the main points of **configuration** and what can be set there:

Location		Name	Description	Risks	To Do
GitHub-Producer	Pre-Build	Workflow / Action	Branch protection Naming validations Linting Release number	Medium - Producer could change PRs to allow unvalidated (malicious) code into Master	mm: Allow Limited config ? mm ::Also .. should build number / be consistently managed? mm: manually create secrets in secrets store ?

Location		Name	Description	Risks	To Do
GitHub-Producer	Build	azure-piplines.yaml	Build, which includes: • Sandbox • Spec - bundled inside the same artefact (optional) • Linting • Unit Tests (optional)	Medium - changes here could break the artefacts for use during release Artefact might not be in line with release config	• Source control the API Management Utils to template the proxies during the build (currently this is done at release) • Platform Service Module (Terraform module) • mm: snapshot configuration from source ssm / secrets to a versioned keys (e.g. clone secrets/my-ip/api-key → ssm/my-ip/v1.2.3/api-key
~~Azure~~	~~Release~~	~~Variables Groups~~			mm: update variable groups in line with build snapshot and test
AWS	Release	Environment	Holds information about an environment, covering: • Public DNS • SSL Certificate • Spine Internet Gateway Endpoint • ASID • NHS ID Connection		mm: update versioned secrets as per tested 'release' version ? (with structure supporting versioning)
Infrastructure	Release	-	• Key Value Maps • Cache • Target Servers • Sharedflows		move to automated management
Manual	Release	-	• TLS Keystores • References • Virtual Hosts		

Location		Name	Description	Risks	To Do
AWS	Release	Terraform state			

For details see: Build and release pipelines (see page 422)

7.14.4.18.1.7 Technology stack

⚠ Missing elements of tech stack:
- AWS Cloud Development Kit - might be used to simplify Terraform use
- Error tracking during build & release - investigate Splunk
- Security scanning tools - linting and active scanning

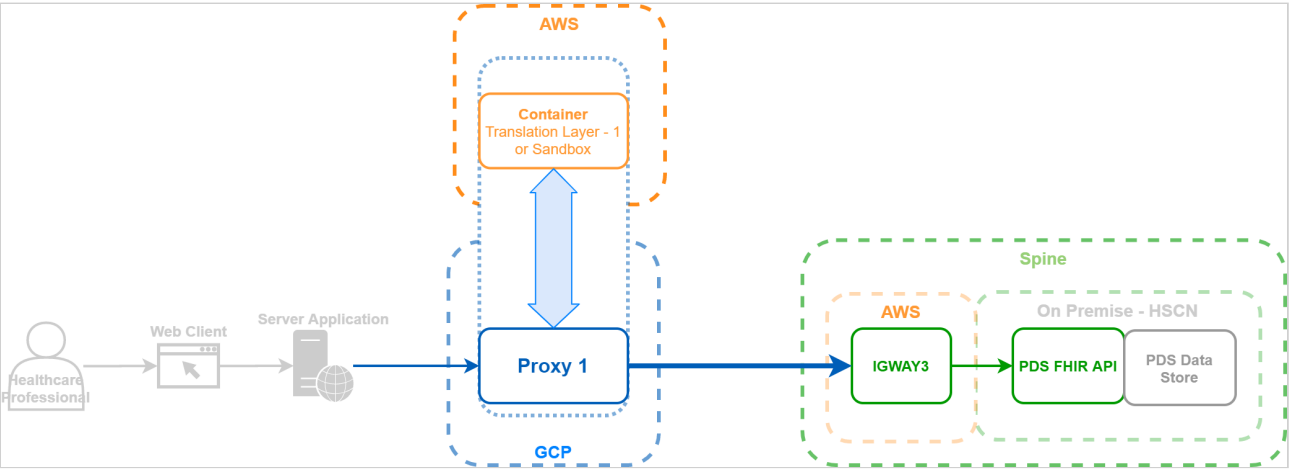
AWS Hosted Targets Replacement

Key points from initial investigation / spike work (Aubyn rough notes):

- Managed to add processing to Terraform, rather than dropping over to Ansible
- Scoped to "Service" = "repo" = which might contain 1 or more proxies
- Uses existing Proxy naming / Name Space for the AWS workspace being built
- Don't need to create a new Docker Repository for each build, use tags to help with separation
- Protections:
  - Each service has own deploy user
  - Service names (namespace) + tagging
- Deploy user is created based on namespace as part of Terraform:
  - Permissions created
  - Access keys put into secret store *todo make them short lived*
  - Release service can then pick up keys
  - (to record details) Creates a firebreak between workspace and the Squad / Deployment Pipeline
- Terraform folder focused on just creating the workspace
- Ansible:

    - build-ecs-proxies
    - deploy-ecs-proxies:
        - Includes monitoring
- AWS clusters:
    - Using existing environment names to make it easier
- Secrets Manager... 40cents per secret per month
    - ssm - free.
    - So keys are set during build... allowing both "sides" of the security are deployed: Splunk HEC tokens per environment
    - Zero down-time deploy: Thinking about how the translation layer might have config that relates to the Proxy Service config

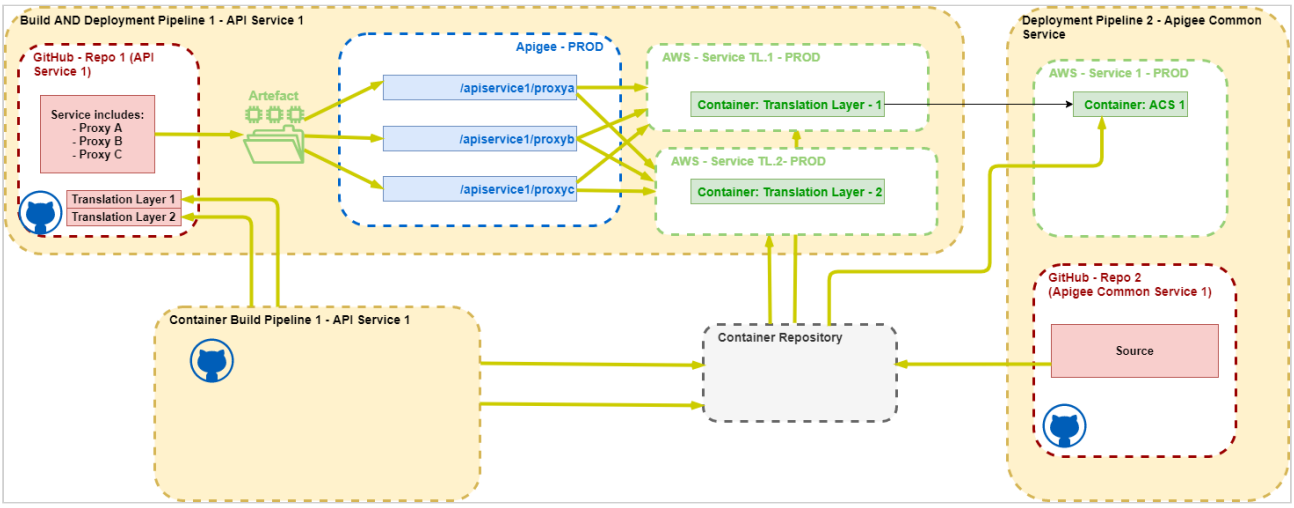
Concept view:



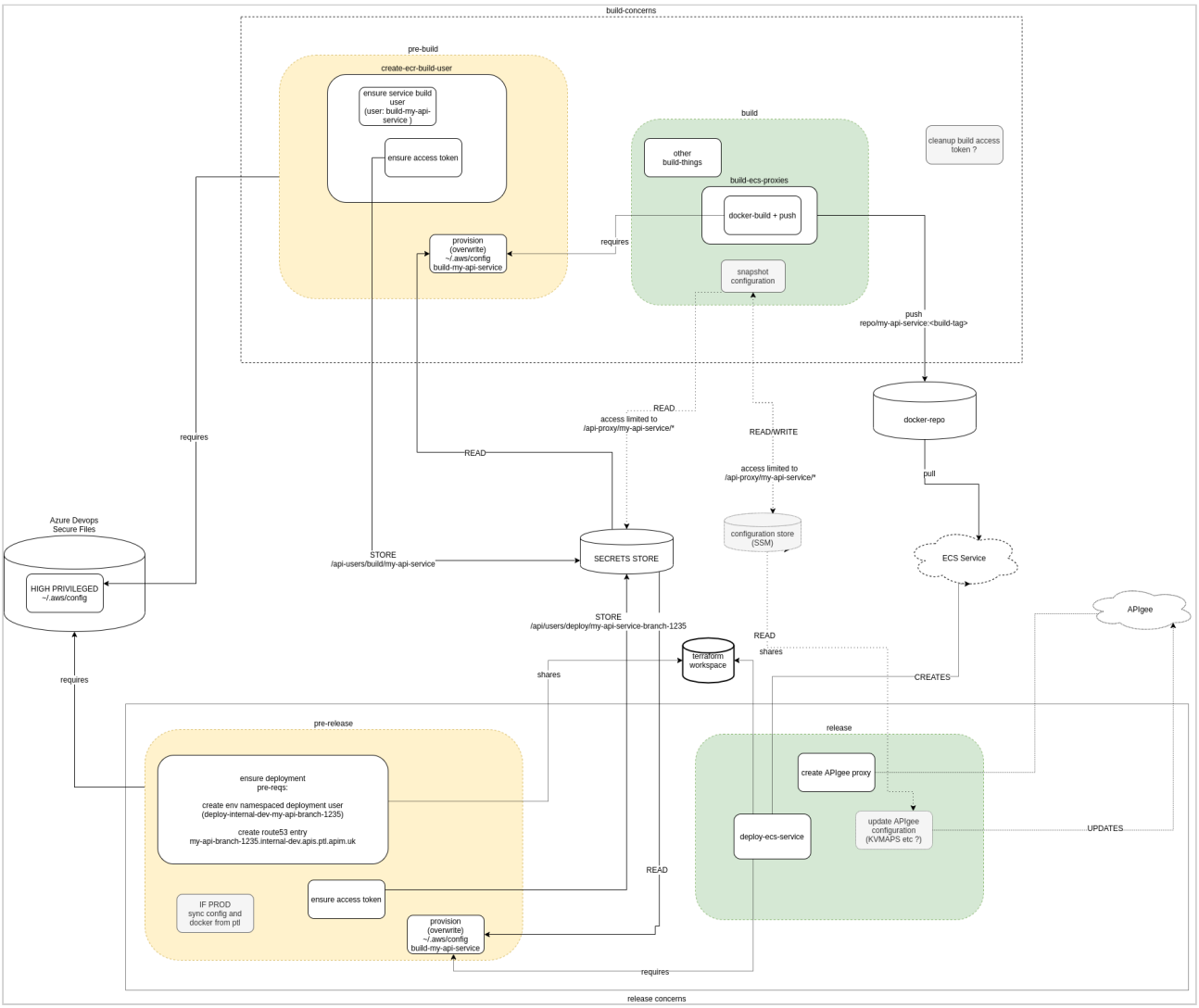
To discuss further

- Azure Devops can provide:
    - some of the account isolation.
    - Approvals workflow
- Boostraping Devops from GitHub
- Diagram created to show interactions: DRAFT - Deployment isolation (see page 432)

Deployment Relationships



7.14.4.18.2 DRAFT - Deployment isolation



7.14.4.19 Proxy complexity and sharing policies

This page looks at two aspects of proxy design:

- How do we share common proxy configuration between API Producer squads?
- When is a proxy too complex for Apigee?

7.14.4.19.1 Sharing mechanisms

The following are ordered in preference, with the best option higher up in the list. This excludes the concept of simply calling out to another service, which does offer significant encapsulation.

	Pros	Cons
Shared Flow + Flow Hooks	• Adds to every proxy in an environment • Encapsulates policy configuration	• Errors can't be handled as well • Only one shared flow per hook
Shared Flow	• Encapsulates policy configuration	• Might not be able to have high flexibility - resulting in copying policy config into a proxy rather than using shared flow • No versioning • Referencing shared values must be entirely shared not context based (Alex H - is this still true?) • No dependency checks on Shared Flow during deployment, so runtime could trigger a 500 error
Template applied during Repo Creation	• Easy to use • 100% customisable	• Once a repository is built the config has to be applied manually retrospectively • Proxies will drift over time as the template improves
Design Patterns (this type of page)	• Easy to create • 100% customisable • Could include XML snippets to make easier	• Could make mistake whilst configuring • Proxies will drift over time as the template improves

Proxy chaining

Investigate suitability of proxy chaining - which is a design pattern suggested by Apigee. However, it has limitations (e.g. must be bundled in the same API Product as the main proxy).
https://docs.apigee.com/api-platform/fundamentals/connecting-proxies-other-proxies

7.14.4.19.2 Forward proxy pattern and sharing logic

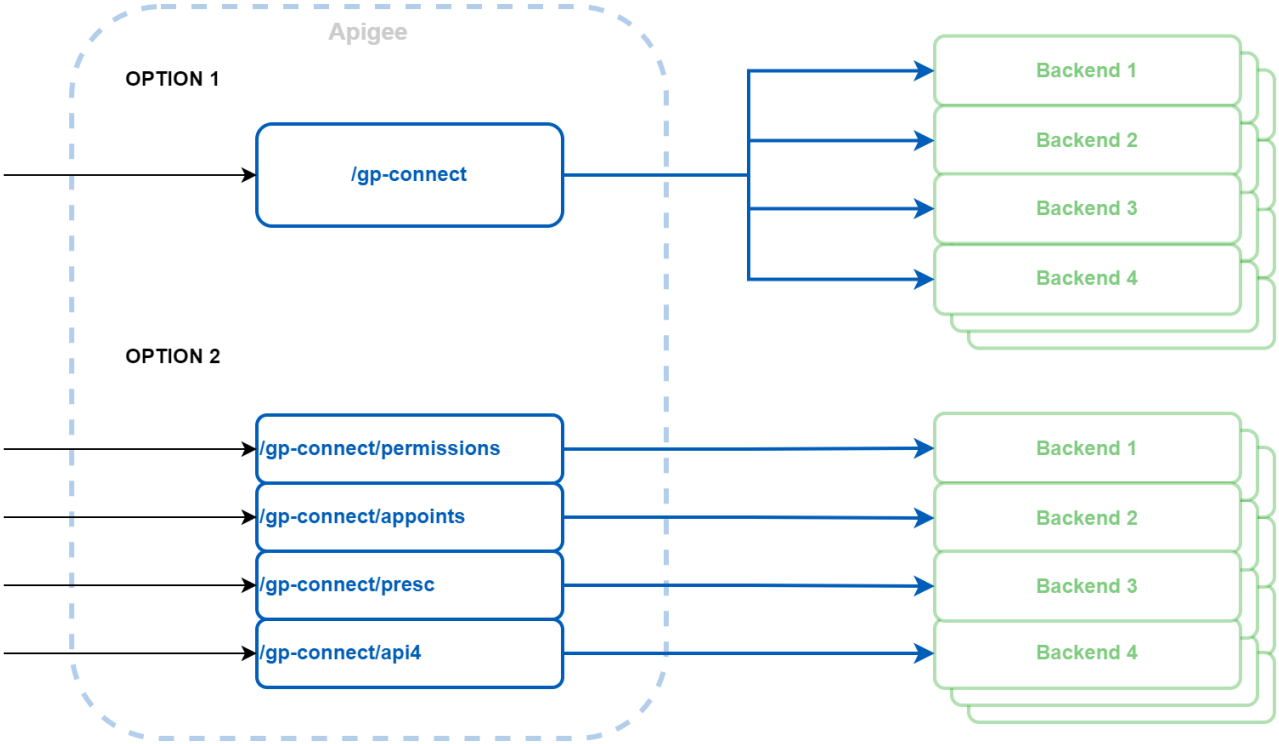
Most APIs will use the simplified proxy concept being encapsulated in the Proxy as a Service (PaaS). For more complex APIs, the standard GitHub repo, CI/CD pipelines will continue to be used. The above notes about sharing logic really applies to platform wide logic that speeds up delivery when re-used.

There are a number of forward proxy based APIs (e.g. Bookings and Referral - BaRS or the new GP Connect Patient Facing Services). With the careful design work going on, these proxies have been slimed down to routing logic that is (relatively) easy to implement in Apigee. The design principle (see Routing requests to multiple backends (see page 280)) is that the Spine Security Proxy (SSP) will not be replicated on APIM as a global solution, but each API (that needs the forward proxy) will follow a standard design pattern.

One team following this design pattern for one proxy is straight forward. However, the GP Connect Patient Facing Service (PFS) requires up to **nine** APIs all doing the same style of routing. Managing the same logic across all nine, would indicate the solution should consider being:

- One proxy for all APIs
- This results in one Repo and one Pipeline for all APIs

There are two main options can be visualised as:



In the case of GP Connect PFS, the decision was taken to follow option 2 - using Shared Flows to ensure the same logic is applied across all APIs, this has the following advantages:

- Functional scope for each API is very separate
- Development cadence - separate teams are working on each API, and might want to release at different times
- Keeps the one API = one Repo = one Pipeline model used in APIM
- Multiple versions of one API (i.e. multiple OAS specs in the one repo) is the existing model. However, if there are different APIs in one Repo then that changes the original approach and boundaries
- Operational considerations:
    - Blast radius - proxies (and their pipelines) are isolated from each other, so a failure or change in one doesn't impact another
    - Logging, redaction and reporting will be a mixture of central and per API
    - Base-path: having "/gp-connect/permissions" is valid, and means that any invalid requests to the permissions API go to the correct team

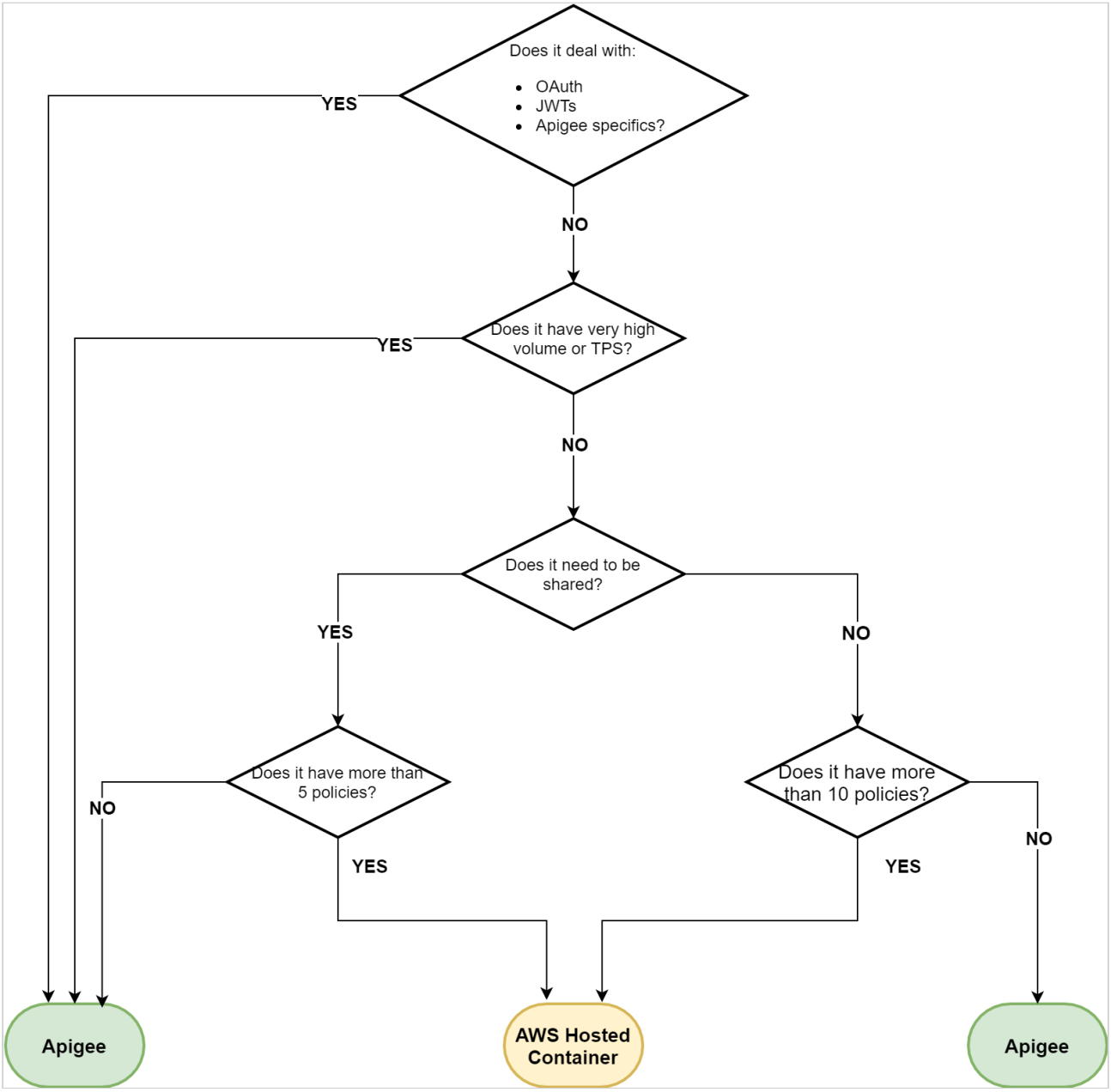
To follow the agreed model, the following considerations need to be taken into account:

- Breaking changes (major versions) of shared flows **must** be name spaced (e.g. shared-flow-v2), which then mitigates most of the following issues
- Deployments with shared flows:
    - Tied to APIM infrastructure releases, in PTL this should have minimal impact. In Production weekly infra releases should mitigate this, especially longer term when the proxy changes should be minimal
    - Shared Flows work with a revision selector per environment, changing the revision impacts all APIs in that environment immediately

        - Small outages can occur with shared flow updates. This means deployments must be out of hours, but this still an APIM standard change situation
- Shared flow testing:
    - Can't be done locally
    - This is not as mature, on the platform, as other tooling and should be worked on
    - Likely to follow the PR type testing, using name spaced proxies temporarily deployed
    - Either Deathstar or Bebop should do work in this area - Navin Bose maybe some tickets here?

7.14.4.19.3 When is a proxy too complex for holding logic in the Apigee layer?

Proposed decision tree:



7.14.4.20 Guidance for making API security decisions

This guidance is draft and is being prepared for submission to the relevant governance body for approval.
Since it was first drafted no other guidance has really come out as an alternative.

7.14.4.20.1 Introduction

7.14.4.20.1.1 Why is guidance needed?

There are different factors which contribute to decisions around the level of security which should be applied to APIs and different aspects of API security which they influence. Clear guidance is required so that decisions about the security aspects of individual APIs can be made objectively and consistently, and so that they can be governed according to clear and unambiguous criteria.

7.14.4.20.1.2 In scope

This guidance covers the following aspects of API security decisions:

- The way that the connecting system must identify and authenticate itself to the API.
- The NIST Identity Assurance Level (IAL) that must have been met for user-restricted access.
- The NIST Authenticator Assurance Level (AAL) that must have been met for user-restricted access.

7.14.4.20.1.3 Out of scope

This guidance does not cover:

- Making decisions about whether application-restricted access should be allowed or not for an API
- How user-restricted and application-restricted access should be implemented if both are allowed for a single API, e.g. whether there should be different versions of the API for user-restricted vs. application-restricted access
- Downstream access control requirements for data fetched from APIs. It is acknowledged that it is current practice for a national API which allows a local system to obtain data requires strict access control (IAL3 + AAL3) but that a clinician may subsequently access the obtained data via their local system access using only user name and password. This is a complex issue and addressing it is beyond the scope of this guidance
- The specific checks or processes that a user must go through to achieve any given Identity Assurance Level
- The specific methods, authentication devices or protocols that should be used for a user to achieve any given Authenticator Assurance Level
- Authorisation model, role based access control or the use of scopes to control access

7.14.4.20.1.4 Assumptions

This guidance assumes that:

- Security features of an API are *not* driven by aspects of the systems which connect to them, i.e. the security features are common for all connecting clients and do not differ, for example, between a limited-user mobile app and a major national system.

- The stringency of different security aspects of APIs increase in lockstep, i.e. we would not expect to see an API which required an IAL 3 identity, but only an AAL1 authentication and no authentication requirement for the connecting system.

7.14.4.20.2 How to use this guidance

The guidance consists of a series of decision trees and a matrices of security requirements. To use this guidance:

1. Agree the data classification (I to V) of your backend store and the class of the data exposed via your API
2. Work through each decision tree to assess the security-related features of the API.
3. Use the requirements matrices to determine the security features to implement based on the assessed features.

If a single API has endpoints with different security-related features, e.g. publicly-available reads and restricted writes, then group the similar endpoints and assess each group separately.

7.14.4.20.3 Assessment 1: data classification

NHS England publishes guidance for public cloud hosting[322], and in particular there is a cloud risk framework[323] and an associated data risk model[324] to help classify the data risk. Data risk classifications go from 1 (least risky) to 5 (most risky).

If this is early stage, just complete the spreadsheet classification quickly - later you can refine / change it. This data classification is used in:

- Architecture submissions - the Solution Design Overview (SDO)
- System Level Security Policy (SLSP)
- Service transition

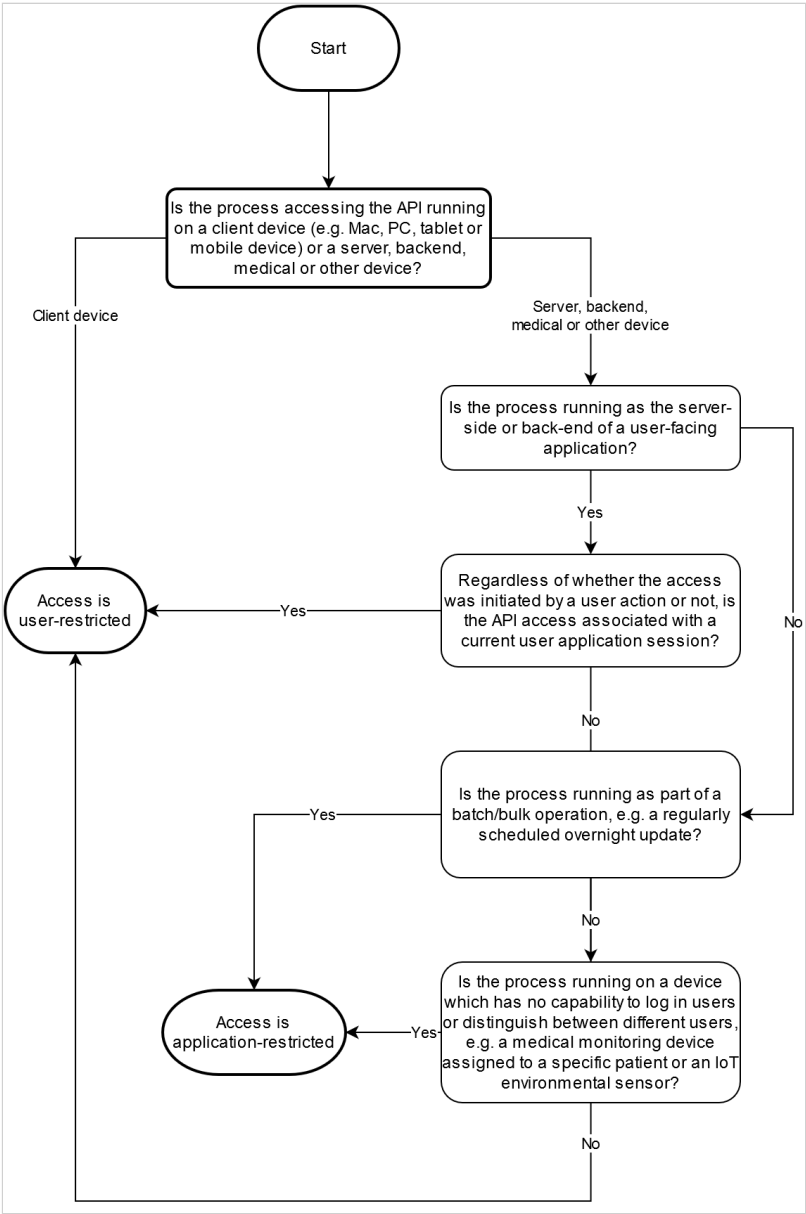
7.14.4.20.4 Assessment 2: user-restricted vs application-restricted access

Deciding whether API access is user-restricted or application-restricted is critical to understand risks around the data or services being accessed. The meaning of user and application restricted is defined on the public-facing security and authorisation page[325].

This assessment aims to avoid the circumstance where the user is present and *should* authenticate to the API, but the API consumer attempts to argue that the security should be application-restricted because it is e.g. performing a background process, such as pre-fetching data, which the user did not explicitly initiate.

The following flow diagram is there to help disambiguate and guide decisions:

322 https://digital.nhs.uk/data-and-information/looking-after-information/data-security-and-information-governance/nhs-and-social-care-data-off-shoring-and-the-use-of-public-cloud-services
323 https://digital.nhs.uk/data-and-information/looking-after-information/data-security-and-information-governance/nhs-and-social-care-data-off-shoring-and-the-use-of-public-cloud-services/cloud-risk-framework
324 https://digital.nhs.uk/data-and-information/looking-after-information/data-security-and-information-governance/nhs-and-social-care-data-off-shoring-and-the-use-of-public-cloud-services/cloud-risk-framework/risk-framework
325 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#application-restricted-apis



Examples:

Scenario	Restricted	Rationale
An overnight process retrieves an event message from the National Event Management System (NEMS), automatically determines that data linked from the event is needed to process the event and calls the National Record Locator (NRL) API to retrieve the linked data.	Application	The access is clearly part of an overnight/automated process and is not on behalf of a specific user.

Scenario	Restricted	Rationale
The user of a clinical system selects a different patient to be the focus of the application. The application automatically decides to prefetch data for the patient from an API in case the clinician wishes to view it.	User	Although the decision to prefetch the data is automated and not in response to an explicit user action, the access is user-restricted as it is associated with a specific user's application session.
An application running natively on a tablet device runs a timed overnight process to prefetch data relating to the user's planned appointments for the next day from an API.	User	Although the access is part of an automated process which can execute when the user is not currently logged into the device, all access from the device is on behalf of the known user of that device and is user-restricted by default.
The backend of a client-server web application, which can be accessed via any browser including from a tablet device, runs a timed overnight process to prefetch data relating to the user's planned appointments for the next day from an API.	Application	Unlike the previous native application example this access is not as closely coupled to a specific user on a specific device or user session and is closer to a true overnight batch process.

OAuth has a provision for delegation, where:

- The "subject" (sub) claim always relates to the person / identity that the action is about
- The "actor" (act) claim then contains a chain of identities that are acting on behalf of the subject

Some of the scenarios above, conceptually, delegate the user actions to a machine identity.

7.14.4.20.5 Assessment 3: Establish baseline identity and authenticator assurance levels (xAL levels)

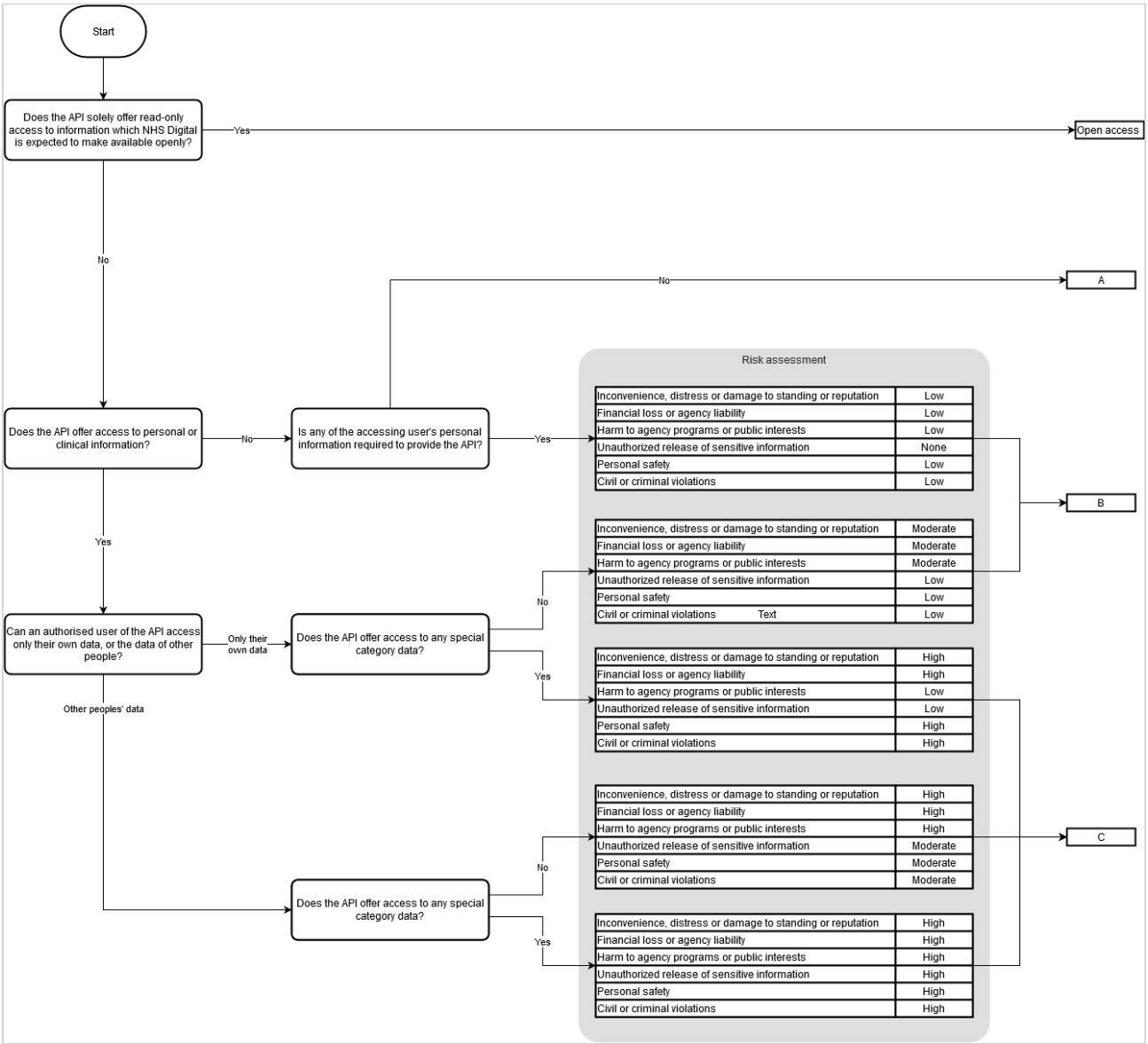
Use the flowchart below to determine the *minimum* identity and authenticator assurance levels which should apply to user-restricted access of the API. This flowchart applies the NIST guidance[326] on xAL level selection, (see sections 6.1 and 6.2 of the linked document for details).

Special category data is defined by the GDPR as:

- personal data revealing **racial or ethnic origin**;
- personal data revealing **political opinions**;

326 https://doi.org/10.6028/NIST.SP.800-63-3

- personal data revealing **religious or philosophical beliefs**;
- personal data revealing **trade union membership**;
- **genetic data**;
- **biometric data** (where used for identification purposes);
- data concerning **health**;
- data concerning a person's **sex life**; and
- data concerning a person's **sexual orientation**



7.14.4.20.6 Assessment 4: Client system authentication

Use the matrix below to determine the minimum method that the client system calling the API should use to authenticate itself. This is the minimum level because there may be circumstances in which there are reasons to use a more rigorous level of client system authentication, e.g. one API in a group of APIs could be offered with client ID & secret authentication, but private key & JWT is used to maintain consistency across the set.

The table gives different outcomes for staff vs. patient access as NHS login complies to the DCB0351 standard[327].

		Assessment 1 Outcome		
		Application-Restricted	User-Restricted (Clinician)	User-Restricted (Patient)
Assessment 2 Outcome	Open access	Client ID	Client ID	Client ID ID Verification: none User Authentication: none
	A	Client ID & Secret	Client ID & Secret IAL 1 & AAL 1	Client ID & Secret ID Verification: none Authentication: low (single factor)
	B	Private Key JWT - Web Server	Client ID & Secret IAL 2 & AAL 2 2nd factor authentication	Client ID & Secret ID Verification: medium Authentication: high (two factor)
	C	Private Key JWT - Web Server Use of HSM for Private Key	Private Key JWT - Web Server IAL3 & AAL3	Private Key & JWT ID Verification: high Authentication: high (two factor)

***NOTES**:*

- *In the table above "Private Key JWT" refers to the method to prove system identity, used between a third party web server (API consuming service) and Apigee. This does **not** relate to the end client, and any private / public key infrastructure used to verify the identity of a person.*
- *Levels A to C sound like a new standard, which is not the intent here. During the initially work will use A to C as the holding pattern*

A key goal is to make integration easier with NHS Digital, without compromising security. However, OAuth usage is still maturing, and the following are potentially key blockers for new entrants:

- Private Key JWT for system authentication
- Use of an HSM (or equivalent secure enclave)

Private beta is the ideal time to get feedback from third party suppliers on the real difficulties. NHS Login has a number of larger suppliers who are fully comfortable with Private Key JWTs. An number of OIDC certified

327 https://digital.nhs.uk/data-and-information/information-standards/information-standards-and-data-collections-including-extractions/publications-and-notifications/standards-and-collections/dcb3051-identity-verification-and-authentication-standard-for-digital-health-and-care-services

libraries support this, but for application-restricted access the "client_credentials" flow is less well supported with libraries.

For the duration of Private beta propose that:

- We gain feedback from third party suppliers on:
    - How easy they find it to implement Private Key JWT
    - What the adoption of HSMs is like and how hard / costly is it to do this
- For Level 3 (C) APIs, application-restricted access, the standard will be to require an HSM (or equivalent secure enclave)
    - Exceptions would be agreed on a case by case.
    - During private beta, we are expecting many suppliers to ask for (and be granted) exceptions
- For Level 3 (C) APIs, user-restricted access. The standard will be Private Key JWT. It should be noted that having a person involved makes it significantly harder to break into the security chain and the additional security gained with Private Key JWT isn't as powerful.
    - Exceptions would be agreed on a case by case
    - During private beta, we are expecting many suppliers to ask for (and be granted) exceptions

7.14.4.21 Spine based APIs and single ASID model

The API Management Platform publishes a range of APIs, some of which are part of the Spine family of APIs[328]. A number of decisions were made to simplify management of this situation for all groups.

7.14.4.21.1 Hidden or single ASID Model

When connecting to services such as PDS FHIR API, the Spine requirements for an Accredited System Identifier (ASID) are hidden:

- NHS E teams request and setup your ASID for you
- It is attached to your APIM Application ID
- You connect and Spine knows who you are, and what permissions you have

If you decided to use a second internet facing Spine API (e.g. EPS), the permission are added to the existing ASID behind the scenes.

Prior to API Management publishing the internet facing Spine APIs, to use a Spine service you would require/ understand:

- Party Key
- Certificates
- Accredited System Identifier (ASID)
- Were you a CMS or MHS?

The FHIR APIs being published had a much narrower focus and contextually did not need that explicitly setup.

328 https://digital.nhs.uk/services/spine

7.14.4.21.2 What about one ASID for each End User Organisation?

Many third-parties provide their service to other clinical groups or Trusts, these are known as End User Organisations (EUOs). In Spine the traceability and control was achieved by creating one ASID for each EUO - which involved technical setup and a number of parties.

To reduce backend setup, but keep traceability, the API Platform requests that the EUO is sent in a header with each request. This supports a single ASID model.

7.14.4.21.3 How are Spine permissions managed?

In the normal process the DIR team uses the End Point Registration (EPR) to update the Spine Directory Service (SDS).

- Default settings are applied for API Management APIs that don't need all the options
- Each API Producer teams has different rules on how the message sets and interactions are applied based on the access mode

7.14.4.21.4 Do all internet facing Spine APIs use this new model?

Due to prioritisation and backlog of onboarding automation, not all APIs do this. If you want to use the e-Referral Service - FHIR API[329] in Healthcare worker user restricted mode (⚠️ there are a few more), you will need to:

- Have at least one separate API Platform Application for the e-Referral Service:
    - If you have EUOs, you then need to have one per EUO
- You will need to work with the Spine DIR team to:
    - Create an ASID for each EUO
- Inform .... onboarding team of which ASID needs to be connected to which application
- What else...?

7.14.4.21.5 Documenting your API

7.14.4.21.5.1 Overview

This page explains how to write and publish API documentation for RESTful APIs on the NHS Digital API Platform.

329 https://digital.nhs.uk/developer/api-catalogue/e-referral-service-fhir

7.14.4.21.5.2 Why documentation matters

Our user research tells us that API consumers really value good API documentation. And also that they really struggle if the documentation is poor. Over the years we have learned what good documentation looks like.

Your API documentation is an important part of your product, and you should treat it as a first class citizen, alongside the API code itself. It's really easy to write poor documentation. It is much harder to write good documentation. But write it you must.

If you don't believe us, just read this blog article on why we need to improve the developer experience with NHS Digital, "Why the TRUD must die[330]".

7.14.4.21.5.3 What documentation you need

API documentation comprises an *API specification* and *supplementary API documents*.

API specification

An *API specification* is the main document you produce. This is a single page on the NHS Digital website that describes pretty much everything an API consumer needs to know about using your API. It is auto-generated from an OAS file that you write (see below).

Your API specification gets listed in our API catalogue (https://digital.nhs.uk/developer/api-catalogue).

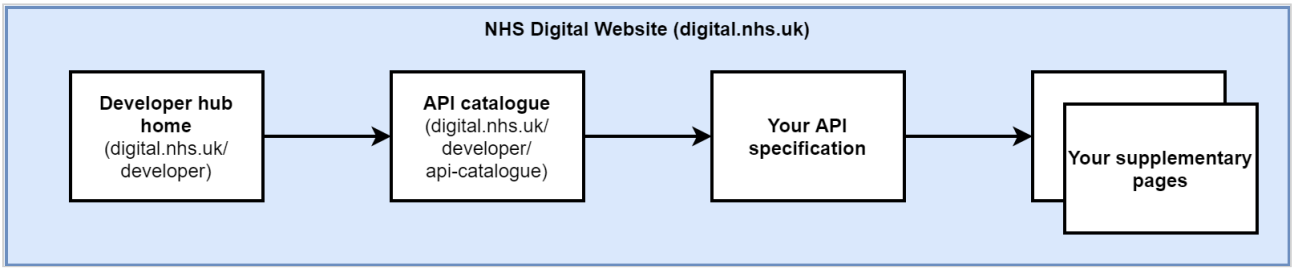
If you want to deploy multiple specs for a single API, see Multi spec deployments (see page 907) instructions (there isn't an established pattern to deploy multiple specs with a Proxygen managed API, please reach out to us on the #platforms-apim-producer-support if this is required).

Supplementary API documents

You might also need some *supplementary API documents* to support your API specification, for example:

- A test data page, explaining what test data is available for testing the API in path-to-live environments
- A "developer guide" - a page or pages explaining how your API fits into a bigger picture
- A form where developers can request access to your API

Here's an illustration:



330 https://web.archive.org/web/20220929143019/https://marcus-baw.medium.com/why-the-trud-must-die-ac27377ee189

7.14.4.21.5.4 How API specifications are published

API specifications on the NHS Digital API Platform are written in a machine-readable format called Open API Specification 3.0[331] (OAS 3.0). OAS files include general information about the API as well as detailed information on the API's endpoints - request and response parameters, headers and bodies.

Legacy Method

Under the legacy APIM deployment method, each API has a GitHub repo which includes all the API assets, including the OAS file for the API. When you first create a GitHub repo from our template, you'll get a dummy "starter" OAS file for your API which you can then edit to include your actual API documentation. When you make changes to the OAS file in your API repo, the build and deployment pipelines will build and deploy those changes as per our SDLC. Behind the scenes, the pipeline pushes your API specification to the API Platform (Apigee Edge). **UPDATE**: there is currently a bug preventing the automated update of specifications to Apigee Edge, if you need to deploy changes through this method, you will need to contact us via the platforms-apim-producer-support[332] Slack channel to manually upload your spec to Apigee. This does not impact Proxygen managed specs, which use a different method (see below (see page 448)).

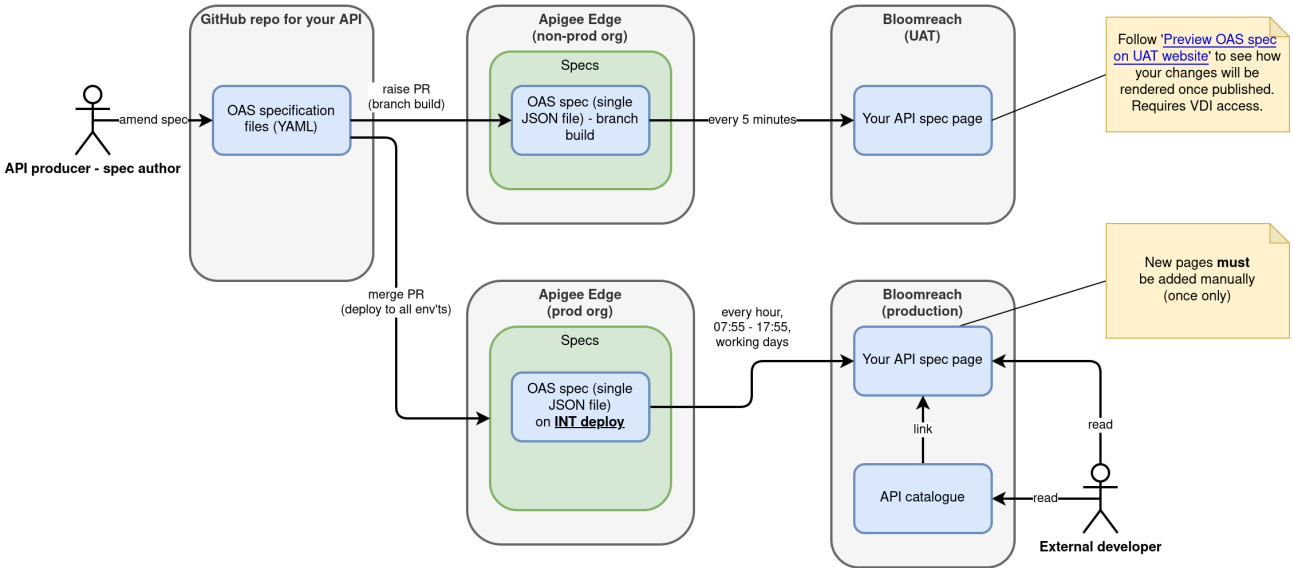
The public (developer-facing) version of your spec is deployed to the NHS Digital website - Bloomreach, in the Developer Hub API Catalogue[333], for example, https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir. This is updated / deployed automatically after:

- PR must be merged onto the main branch
- Deployment successfully reaches INT
    - This updates the spec in Apigee Edge for those environments (See **UPDATE** above)
- Wait for a Bloomreach poll which (currently) is hourly during working hours 07:55 to 17:55 for Production

You can Preview API specifications on the Bloomreach training environment (see page 515) before publishing your changes to the production API and integration catalogue site, allowing you to discover and address any potential issues with rendering or the OAS code.

Here's a diagram:

331 https://swagger.io/specification/#specification
332 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY
333 https://digital.nhs.uk/developer/api-catalogue



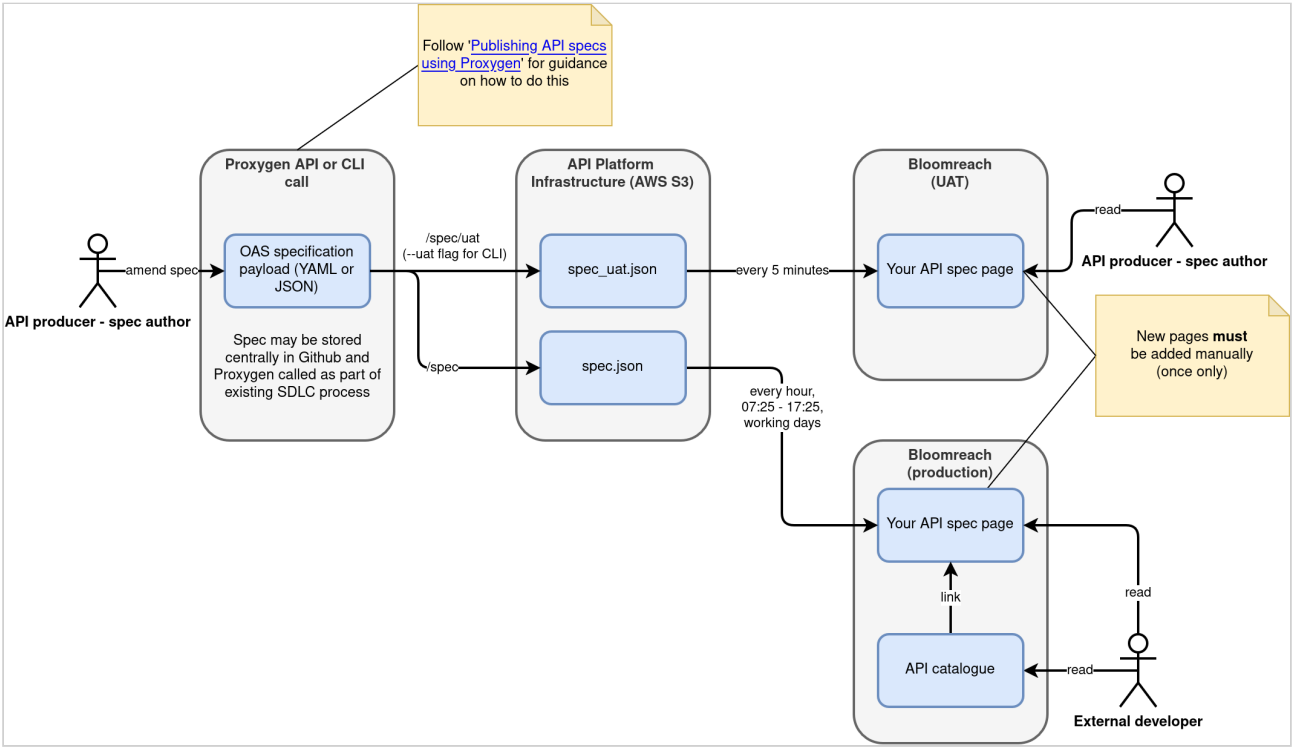
Proxygen Method

If you're using Proxygen to manage your API proxy deployments, you should use the Proxygen functionality to publish your API documentation instead of the legacy method mentioned above. For details on how to do this, see Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571).

Unlike the legacy method, Proxygen stores the spec in our APIM Infrastructure (AWS). The automated Bloomreach job then pulls any changes and updates the content on the website. Similar to the legacy method, it is still possible to preview specifications on the Bloomreach UAT environment before publishing to Production, details can be found on the Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571) page.

For an example, the Proxygen API specification itself uses GitHub Actions workflows to set up and authenticate with the Proxygen CLI and publish any specification updates: https://github.com/NHSDigital/proxygen-api-specification, https://digital.nhs.uk/developer/api-catalogue/proxy-generator. Unlike the legacy method, it's down to the API Producer to decide what tools they want to integrate Proxygen with as part of their SDLC.

Here's a digram:



7.14.4.21.5.5 How supplementary API documents are published

Supplementary API documents are created manually, in Bloomreach CMS. They can be previewed there to see how they will look on the website once they are published. Once ready, the reviewer then publishes the documents which makes them available to the wide audience.

7.14.4.21.5.6 How to deliver your API documentation

Pre-requisites

Before you can write your API documentation, you'll need:

- a draft design for your API - see API design (see page 322)
- the right skills in your team - see Who should write your API documentation (see page 452)
- A GitHub repo set up for your API proxy - see Creating a GitHub repository and deployment pipelines for your API (see page 580). If using Proxygen, you must be set up and authenticated with either the Proxygen API or CLI - see Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571)
- Bloomreach CMS access, but only if you need to write supplementary API documents; contact us to arrange it. This is not required if the only thing to publish is the API specification.

The process

1. Learn how the process works (by reading this page) and what we'll be looking for by reading the API Specification checklist (see page 530).
2. Set up your OAS file and configure your pipeline (see page 554) to make your specification available to the Apigee platform if using the legacy method, otherwise set up your API with Proxygen - see Getting set up with proxy generator (see page 681).
3. Write your API documentation[334], including the specification and any supplementary API documents.
4. Draw any diagrams (see page 503) you want to include in your API specification, or the supplementary documents.
5. Review your API specification within your team on Bloomreach User Acceptance Test (UAT) environment - see Previewing API specification on Bloomreach UAT website[335], as well as Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 577) for Proxygen.
6. Create any supplementary API documents in Bloomreach production - see Using Bloomreach CMS[336] - but do not publish them yet.
7. Review your API documentation with us[337].
8.

   If you have published you API in the INT environment, the spec **should** be publicly visible. Conversely, you should not delay a technical push to INT just because the spec isn't quite ready
9. Publish your documentation in Bloomreach production - the first time you do this ask us to create your API spec page in Bloomreach - see Creating an API specification page in production Bloomreach CMS (see page 526). Or do it yourself if you have a Bloomreach account.
10. Get your API specification ready for the API catalogue[338] by adding an SEO summary (see page 548) for search engines (Google, etc) and adding taxonomy keys [339]to make your API appear on the catalogue under the correct filters (see page 537) and with the right tags.
11. Ask us to add your API to the API catalogue - see Adding an API specification to the API catalogue in Bloomreach CMS[340].

---

334 https://nhsd-confluence.digital.nhs.uk/display/APM/Write+your+API+documentation
335 https://nhsd-confluence.digital.nhs.uk/display/APM/Previewing+API+specification+on+Bloomreach+UAT+website
336 https://nhsd-confluence.digital.nhs.uk/display/APM/Using+Bloomreach+CMS?src=contextnavpagetreemode
337 https://nhsd-confluence.digital.nhs.uk/display/APM/Review+your+API+documentation+with+us
338 https://digital.nhs.uk/developer/api-catalogue
339 https://nhsd-confluence.digital.nhs.uk/display/APM/Search+engine+optimisation+%28SEO%29+for+API+documentation
340 https://nhsd-confluence.digital.nhs.uk/display/APM/Adding+an+API+specification+to+the+API+catalogue+in+Bloomreach+CMS

7.14.4.21.5.7 Writing your API documentation

Ensuring effective API documentation is available for your API is crucial for the success of your API on the NHS API Platform. Clear, comprehensive, and easy-to-understand documentation empowers API Consumers to quickly understand, integrate with, and utilise your API, ultimately driving adoption and reducing support overhead.

Overview

Your API specification is the foundational element of your technical documentation. It should be written as an OpenAPI Specification (OAS) file in YAML format and stored within your API proxy repository.

Process

Step 1: Make sure you have the right skills

See Who should write your API documentation[341].

Step 2: Understand what goes in your API specification

See What goes in your API specification (see page 453).

Step 3: Write your API specification

You have a couple of options for adding your spec content:

1. Edit the raw YAML files via the GitHub UI - recommended for occasional use - see Editing OAS files in GitHub (see page 473).
2. Use BeautifyTools - recommended for heavier use - see Editing OAS files in Beautify Tools and Preview in Bloomreach UAT (see page 499) - Provides a user-friendly interface for editing YAML and potentially previewing the generated documentation.
3. Use Stoplight - recommended for heavier use if for some reason BeautifyTools isn't working for you - see Editing OAS files in Stoplight[342] - Offers a more feature-rich environment for API design and documentation, including visual editing and validation.

Step 4: Write additional pages

Beyond the core API reference generated from your OAS file, you will need to create additional pages to provide a comprehensive documentation experience. These should cover the conceptual information, getting started guides, authentication details, error handling, and other relevant topics. See What goes in your API documentation.

341 https://nhsd-confluence.digital.nhs.uk/display/APM/Who+should+write+your+API+documentation?src=contextnavpagetreemode

342 https://nhsd-confluence.digital.nhs.uk/display/APM/Editing+OAS+files+in+Stoplight?src=contextnavpagetreemode

Step 5: Make sure you have followed documentation standards

Before publishing your documentation, it's essential to review and validate it to ensure accuracy, completeness, and adherence to NHS API Platform standards.

See:

- API specification checklist (see page 530)
- API content style guide (see page 488)
- Formatting OAS API specifications [343] Linking API specifications to FHIR standards [344]

Exceptions: REST APIs

Some APIs on the NHS England API and Integration catalogue [345] do not have a formal, machine-readable specification like OpenAPI Specification (OAS). Whilst the specification pages may not list endpoints in a typical /resource/{id} format with explicit HTTP methods (GET, POST, etc.) like an OAS spec, it may describe the inputs required to perform actions (e.g. searching by NHS number, retrieving by prescription ID etc). This implicitly defines the ways you interact with the API, which is a core part of an API specification.
Under the "Technology" section, the pages would clearly labels these APIs as a "REST API". This indicates it follows RESTful principles, even if the documentation isn't in a standard REST specification format like OAS. REST APIs are designed around resources, and these resources are accessed and manipulated via unique identifiers, which are typically URIs (Uniform Resource Identifiers). These URIs, particularly the paths to specific resources or collections of resources, are commonly referred to as endpoints in the context of APIs. When you make an HTTP request (like GET, POST, PUT, DELETE) to a REST API, you are directing that request to a specific endpoint to interact with the resource located there.

Who should write your API documentation

Overview

This page explains what skills you need in your team to write good API documentation.

Details

Your audience is a mixture of:

- technical people
- non-technical people e.g. product owners, entrepreneurs, CTOs

So, your API documentation needs to be a combination of:

- technical detail
- product information - in plain English

With that in mind, you might need up to three separate people to write your docs:

- **a technical person** who can describe the endpoints, request and response parameters, data fields and example request and response bodies

---

343 https://nhsd-confluence.digital.nhs.uk/display/APM/Formatting+OAS+API+specifications?src=contextnavpagetreemode

344 https://nhsd-confluence.digital.nhs.uk/display/APM/Linking+API+specifications+to+FHIR+standards?src=contextnavpagetreemode

345 https://digital.nhs.uk/developer/api-catalogue

    - for example, a technical architect, developer or technical BA
- **a product person** who can describe your API as a product - the overview of what it does, its status, how to onboard for it and so on
    - for example, a product owner or a BA
- **a wordsmith** who can write in plain English and follow the fairly rigorous GDS style guidelines
    - for example a technical author, content designer, or a product owner or BA with really good English language skills

If you're really lucky you'll have one person on your team who can do all three.

If not, we are here to help.

We have a pool of technical authors/content designers who can help you with the wordsmithing. And we have product and technical people who can advise on your content.

For each API, we assign a single point of contact to help you with your spec - see the "Technical author point of contact" column in the table at API register#APIcatalogue-newAPIs.

What goes in your API documentation

Overview

This page explains what should go in your API documentation, including

- your API specification
- any supplementary documents

Details

Introductory information

Your API specification should include introductory information under the following standard headings which explains, in plain English:

- Overview - What the API does
    - Write your Overview description for an API or an endpoint from the user's perspective, not from the API's perspective. In other words, explain what the user can do with this API, as opposed to what the API can do for the developer.
        - When describing an API, start the sentence with "Use this API to", for example:
        - "Use this API to find patients and to retrieve and update their personal details held nationally by NHS Digital." (as opposed to "This API finds patients...")
    - Include a summary diagram of how your API works in your Overview - directly under the section heading is the best location to support the description that follows. See Drawing diagrams in (see page 503)draw.io[346] for how to draw diagrams.
    - Provide links to and from the service page for your service.

346 http://draw.io

        - In the first few sentences of your Overview, make sure you link to information about the service that your API connects to -
          this is mostly likely somewhere in the https://digital.nhs.uk/services area of the NHSD website.
        - Example: PDS FHIR API links to Personal Demographics Service (PDS)[347]
        - You also need to add a link from the service page to your API once your API spec is live. Contact us for how to do this.
        - Example: Access data on the Personal Demographics Service[348] links to all of the PDS API specs including the PDS FHIR API.
    - If your API gets data from a central source, include a short statement in your Overview (and possibly a link to a companion page) about the scope of coverage of your data - such as, England, Wales, Isle of Man, Channel Islands, completeness of data coverage or specific data excluded.
        - Example: The ODS APIs have a companion page (applicable to ODS in general, not just their APIs) in the Services area to explain this, linked from their API Overviews: https://digital.nhs.uk/services/organisation-data-service/about-the-organisation-data-service#ods-data-coverage
- Who can use this API - What it can be used for, legally, and how to confirm your use case - this is **really** important to state up front.
    - It's important to link to a list of approved use cases for your API and how to request access approval for your API. This might be a form for developers to fill in, or just publishing your group's email contact address.

      Example: https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-description__who-can-use-this-api
- Related APIs - Any related APIs that access the same services or are used with this one
- API status and roadmap - The API status and its roadmap (as a link to the interactive product backlog[349], filtered to show just the features for your API). For details of how to set this up, see Interactive product backlog[350].
- Service level (once your API is in beta) - see Service levels for APIs (see page 288).
- Technology - The technology the API uses - REST and maybe FHIR
    - For a FHIR API, if it uses local FHIR extensions such as UK Core, which version/package has the API been built against (see Designing your FHIR API)
- Network access - Network access options for the API - internet and/or HSCN
- Security and authorisation - How authentication and authorisation works
- Errors - A standard section to explain basic HTTP error messages 200 to 299, 400 to 499, 500 to 599 with a link to the refence guide for more details - your individual endpoint-specific errors should still be included under each endpoint.

---

347 https://digital.nhs.uk/services/demographics
348 https://digital.nhs.uk/services/demographics/access-data-on-the-personal-demographics-service
349 https://nhs-digital-api-management.featureupvote.com/
350 https://nhsd-confluence.digital.nhs.uk/display/APM/Interactive+product+backlog?src=contextnavpagetreemode

- Open Source - You should also provide details of your open source resources - see Open source (see page 457)
- Environments and testing - What testing facilities are available
- Onboarding - How onboarding works. Optionally, you can include a contact email address for your API, at the end of this section. Tell us what it is so we can forward any queries from api.management@nhs.net[351].

For an example, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir.

Endpoint information

For each endpoint in your API, you should include:

- An overview of what the endpoint does
    - Write your Overview description for an API or an endpoint from the user's perspective, not from the API's perspective. In other words, explain what the user can do with this API, as opposed to what the API can do for the developer.
        - When describing an endpoint, you can keep things shorter by omitting "Use this API to", for example:
        - "Get current information for the patient with the given NHS Number." (as opposed to "Returns current information...")
- Any important business rules or other information
- Details of sandbox test scenarios (if you have a sandbox)
- Request details
    - Parameters (path / query)
    - Headers
    - Body - examples and schema
- Response details
    - Headers
    - Body
        - happy path
        - alternative flows / error conditions
        - in particular, if your API includes rate limiting (see page 1037), make sure you have included HTTP status 429 as a possible error condition - see the PDS API spec[352] for an example of this

To make your endpoint information clear:

- write field descriptions in clear and plain English
- make sure any coded values also include the possible values

See also Standard data for specs and sandboxes[353].

---

351 mailto:api.management@nhs.net
352 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
353 https://nhsd-confluence.digital.nhs.uk/display/APM/Standard+data+for+specs+and+sandboxes

Do not include in your response schemas

By default, the OAS API specification contains a boolean field that you can use on schemas called ***readOnly***. This field must be avoided in response schemas.

Attention to shared schemas between different components that can represent requests schemas and the readOnly field might have an important meaning.

Supplementary documents

You might need to create some supplementary pages for your API to prevent overloading your API specification. For example, you might want:

- a page with specific details around test data[354]
- a form where developers can request access to your API[355]
- a developer guide[356] to your specific area of software development

The disadvantage is that this information is not searchable as part of your main API spec page using CTRL/F.

You create supplementary documents directly on the NHS Digital website, using Bloomreach - the NHS Digital website CMS (see page 508). For more details, contact us.

Try this API

By default, your API spec will include a "Try this API" link for each endpoint, which will allow external developers to send a request directly to your sandbox from their browser.

Note that:

- If you don't have a sandbox, you'll want to turn this feature off - see Formatting OAS API specifications - Toggle support for 'Try this API' (see page 458)'
- Sometimes it doesn't work because of issues with CORS - for details see Cross-origin resource sharing (CORS) (see page 786)

Content style

Your API spec is a mixture of technical details and descriptive information, including general overviews, field descriptions and so on.

When writing the descriptive information, bear in mind you are writing for an audience of external software developers, who don't necessarily know NHS Digital terminology.

You MUST follow the API content style guide (see page 488) when writing descriptive information.

You may find other exemplar API's useful for comparison, see our research on our References[357] page.

---

354 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir/pds-fhir-api-test-data
355 https://digital.nhs.uk/services/demographics/personal-demographics-service-access-request
356 https://digital.nhs.uk/developer/guides-and-documentation/developer-guides
357 https://nhsd-confluence.digital.nhs.uk/display/APM/References

Formatting

See Formatting OAS API specifications (see page 458) for details of some quirkier aspects of formatting that need to be observed in order to ensure correct rendering of your specification on NSHD website.

Summaries and search engine optimisation (SEO)

Your API documents need to be discoverable - on the API catalogue, using NHS Digital's website search engine and by Google and the other search engines. To do this, add a Summary, Short summary and SEO summary to your API in Bloomreach.

- Make the Summary (which appears in the page header) and Short summary (which appears on the API catalogue) the same as each other and phrase them consistently with other APIs. Start with a verb such as "Access", "Request", "Share", "Receive" and so on, depending on its function.
    - For example, the Immunisation History - FHIR API has these both set to: "Access a patient's coronavirus (COVID-19) immunisation history."
- Make the SEO summary (which is used by external search engines eg Google) use additional keywords and phrases, for details see External SEO.[358]
    - For example, the PDS FHIR API SEO summary says: "You can read and update the following data: NHS number (read only), name, gender, addresses and contact details, birth information, death information, registered GP, nominated pharmacy, dispensing doctor pharmacy, medical appliance supplier pharmacy, related people such as next of kin (read only)."

To find out more, read Using Bloomreach CMS for Search Engine Optimisation (SEO) of API documentation[359].

Open source

As per the NHS service standard[360], you should code your API and underlying service in the open.

Also, to make integration easier, you should provide open source client libraries and/or sample client apps that API consumers can use and a starting point.

You should document what open source resources are available for your API in the "Open source" section of your API specification.

This might include:

- the GitHub repo for your API (the one that has your API spec in it, as well as the API proxy and API sandbox code)
- the GitHub repo for your back-end service, if it's coded in the open (and if not, why not?)a
- any open source client libraries or example clients you have built

For examples, see:

358 https://nhsd-confluence.digital.nhs.uk/display/APM/Search+engine+optimisation+%28SEO%29+for+API+documentation#Searchengineoptimisation(SEO)forAPIdocumentation-ExternalSEO
359 https://nhsd-confluence.digital.nhs.uk/display/APM/Using+Bloomreach+CMS+for+Search+Engine+Optimisation+%28SEO%29+of+API+documentation
360 https://service-manual.nhs.uk/standards-and-technology/service-standard-points/12-make-new-source-code-open

- https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#overview--open-source
- https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#overview--open-source

Formatting OAS API specifications

This page documents selected, non-obvious formatting aspects that need to be followed in OAS specifications' source in order to ensure correct rendering on NSHE website[361].

OAS version

OpenAPI Specification version supported is of range `3.0.x` (required to be specified in the top-level `openapi` field of the specification as actual value, e.g. 3.0.0).

In theory, 3.x.x is supported; however, this hasn't been verified by our team.

RapiDoc

A new spec renderer called RapiDoc (see page 557) was deployed in 2023 as the previous custom-built renderer did not completely support the OAS v3.0 specification. Although most formatting remains the same, certain aspects have changed. For a full breakdown of the differences, see the New RapiDoc renderer page.

Rich text formatting

As defined by OpenAPI Specification[362] rich text formatting is supported in '`description`' fields (and only in those fields) which can be populated using CommonMark[363] flavour of Markdown or plain text.

See this cheatsheet[364] for basic syntax.

> **Injected HTML**
>
> Avoid injecting HTML into the Markdown of the '`description`' fields if possible - this can cause issues when the HTML is rendered from the Markdown. For example, using `<a>` tags instead of using CommonMark syntax `[link](https://example.com)`. There may be instances where this cannot be avoided, for example, using the `<br>` element is the only way to produce a line break within table cells (see Tables (see page 459)). In these cases, make sure the HTML is valid to prevent rendering issues.

361 https://digital.nhs.uk/developer/api-catalogue/
362 https://spec.openapis.org/oas/v3.0.3#rich-text-formatting
363 https://commonmark.org/
364 https://commonmark.org/help/

Tables

Tables can be embedded in a Markdown text using format compatible with GitHub Flavoured Markdown[365]'s 'pipe' syntax.

Preceding blank lines

Tables need to be preceded and followed by an empty (blank) line in order to ensure correct separation from the surrounding text or other elements (e.g. '`##`'-annotated header lines).

In YAML, a typical multi-line string notation[366] makes it easy to express a blank line, as shown in the first example.

However, when a multi-line string value is expressed in a single line (be it in YAML or JSON), make sure to use *two* new line characters: '`\n\n`' as illustrated in the latter two examples. Also, when this notation is used, make sure that the there are no white space characters between '`\n\n`' and the first '`|`' character of the table.

Examples:

YAML - multiline string defined across multiple lines

```
...
info:
  description: |
    Preceding text.

    | Column 1 Header | Column 2 Header |
    | --------------- | --------------- |
    | Cell 1.1        | Cell 2.1        |
    | Cell 1.2        | Cell 2.2        |

    Following text.
...
```

YAML - multiline string defined in a single line

```
...
info:
  description: "Preceding text.\n\n| Column 1 Header | Column 2 Header |\n|
--------------- | --------------- |\n| Cell 1.1 | Cell 2.1 |\n| Cell 1.2 | Cell 2.2 |
\n\nFollowing text."
...
```

365 https://docs.github.com/en/free-pro-team@latest/github/writing-on-github/organizing-information-with-tables
366 https://yaml-multiline.info/

JSON

```
{
  ...
  "info": {
    "description": "Preceding text.\n\n| Column 1 Header | Column 2 Header |\n|
--------------- | --------------- |\n| Cell 1.1 | Cell 2.1 |\n| Cell 1.2 | Cell 2.2 |
\n\nFollowing text."
  }
  ...
}
```

Alignment

Make sure that left edge of the table is correctly aligned with the preceding text - in the example below, the table shares the same indentation level as the '*Where status...*' and '*Check diagnostics...*' lines:

```
description: |
  Where status code 422 (Unprocessable Entity) is returned then an...
  Check diagnostics property for specific information regarding the...

  | Error code                        | Description                     |
  | --------------------------------- | ------------------------------- |
  | REFERENCE_NOT_FOUND               | A supplied reference could...   |
  | INAPPROPRIATE_VALUE               | A value, which is acceptabl...  |
  | TOO_MANY_ITEMS                    | In a list where a maximum n...  |
```

Email addresses

If email addresses embedded in Markdown text are meant to be be rendered as hyperlinks, make sure to apply the '`mailto:`' prefix as in the following example:

> *you can email your request to the [API management team](**mailto:**api.management@nhs.net[367]).*

Headings

Headings in Markdown should be defined using ATX notation[368] with leading '###' characters.

In terms of the hierarchy, the highest level of heading can be any 'standard' 1 to 6 level (inclusive). Note that the actual levels used in the `h1-h6` HTML rendered from Markdown may not directly correspond to the ones defined in Markdown. The reason is that the levels are automatically adjusted so that the highest level (hierarchy-wise, not order-wise) rendered from Markdown comes right after the level of the preceding/

367 http://nhs.net
368 https://spec.commonmark.org/0.29/#atx-heading

containing heading in the page. For example, if endpoint's description contains heading of level 1 (single '#' character), it will actually be rendered as `h3`, to ensure that it is one level down from `h2` of the 'containing' *Endpoints* heading imposed by the page's structure.

The top-level headings (again, hierarchy-wise, not order-wise) in the API description will have hyperlinks rendered in the navigation panel (this will change to top-level once ADZ-343[369] is released); note that this only pertains to the `info.description` field; headings from other description fields do *not* get their hyperlinks rendered in the side nav menu):

369 https://nhsd-jira.digital.nhs.uk/browse/ADZ-343

Rendering	YAML	JSON

	``yaml
	--- 	
	openapi: 3.0.0
	info:
	  ...
	  description: |-
	    ## Overview
	    API overview text.
	
	    ## Another level 2 heading
	    Some more text.
	
	    ### Level 3 Heading
	    Yet more text.
	...
	paths:
	  "/Patient/{id}":
	    ...
	    get:
	      description: |-
	        ## Overview
	        Endpoint description.
	``
		``
		{
		
		"
		o
		p
		e
		n
		a
		p
		i
		"
		:
		"
		3
		.
		0
		.
		0
		"
		,
		
		"
		i
		n
		f
		o
		"
		:
		{
		
		.
		.
		.
		
		"
		d
		e
		s
		c
		r
		i
		p
		``

Rendering	YAML	JSON
		tion":"##Overview\nAPIoverviewtext.\n\n##Anotherlev

Rendering	YAML	JSON
		el2heading\nSome m ore text.\n\n###Level3Heading\nYetmo

Rendering	YAML	JSON

		`retext.",`
		
		`},`
		
		`...`
		`"paths":{`

Rendering	YAML	JSON
		"/Patient/{id}": { "get": {

Rendering	YAML	JSON
		"description": "##Overview\nEndpointdescription."

Comments in embedded snippets of code in API Description

Make sure that any comment lines that use leading '#' characters present in code snippets (common in bash, python, etc.) in the API Description Markdown (field `info.description` in the spec's body), are *not* followed by a space character.

For example, instead of having the comment as:

```
# Current time formatted as yyyyMMddHHmm
```

type it as:

```
#Current time formatted as yyyyMMddHHmm
```

This needs to be observed until bug ADZ-916 is fixed.

Grouping and ordering endpoints in the navigation menu

For details on grouping and ordering endpoints, refer to the Grouping and ordering operations in the navigation menu (see page 562) section.

Rendering endpoints

RapiDoc uses the standard way of grouping known as tags[370]. The tag order in the global tags section controls the order of those groups. The order of the operations are controlled by the order they appear in the OAS file itself. However, if the same endpoint has multiple operations, then they are sorted by method (e.g. GET before POST).

'example' and 'examples' fields

OAS[371] allows defining examples for various parts of the API specification, such as examples of header/parameter values (Header Objects[372], Parameter Objects[373]) or request and response body payloads (Media Type Objects[374]). Objects that define the headers, the bodies, etc., can also reference schemas which describe them further. Example values can be specified directly against the objects themselves but they can also be specified within their Schema Objects[375].

On NHSD website these example values are rendered in dedicated fields or sections titled *Example* or *Examples*.

Be mindful that the general rule imposed by the OAS standard is that when *both* the object and its schema define examples, the examples defined by the object win. The rendering on NHSD website follows this rule and in such a case ignores examples defined in the schema. Conversely, where the parent object does not define an example, but its schema does, the example from schema *will* be rendered.

---

370 https://swagger.io/docs/specification/grouping-operations-with-tags/
371 https://spec.openapis.org/oas/v3.0.3
372 https://spec.openapis.org/oas/v3.0.3#header-object
373 https://spec.openapis.org/oas/v3.0.3#parameter-object
374 https://spec.openapis.org/oas/v3.0.3#media-type-object
375 https://spec.openapis.org/oas/v3.0.3#schema-object

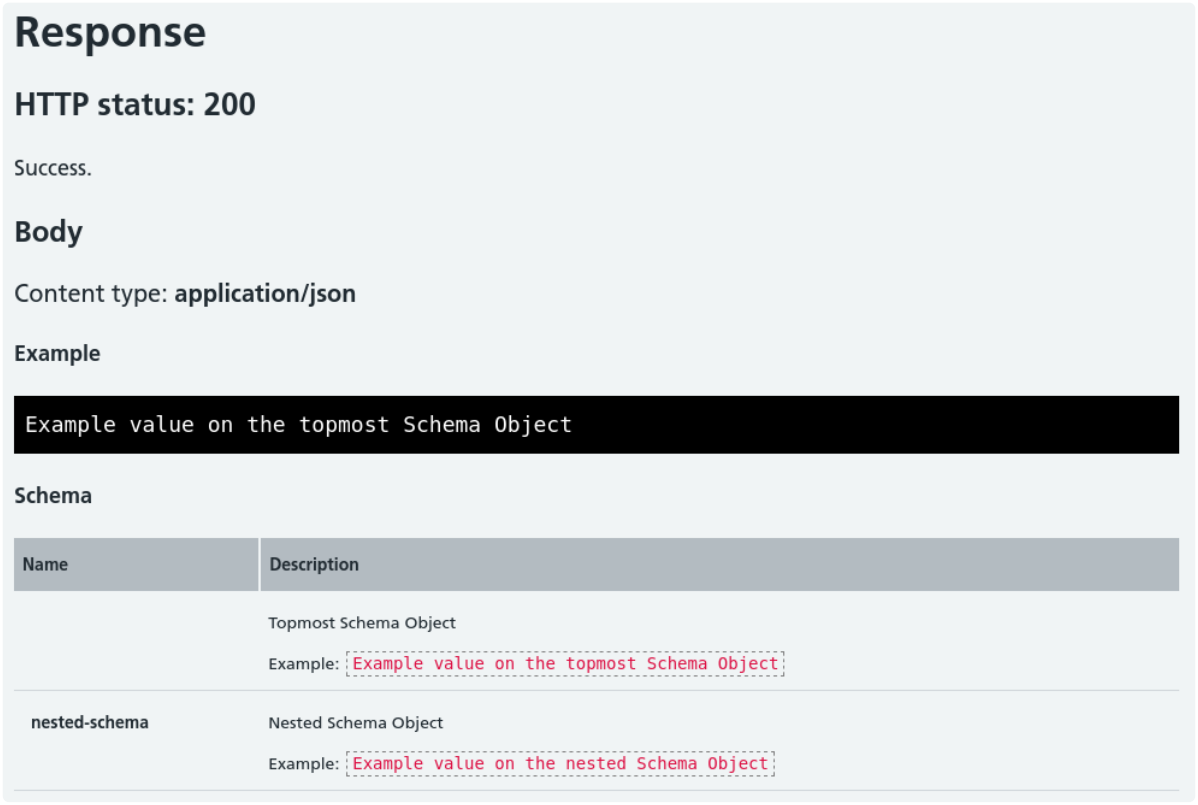
Note, however, that Schema Objects typically embed other Schema Objects. Where the header or body object does not define an example (the latter of the two cases from the previous paragraph), only examples from the *topmost* Schema Object will be rendered for the parent object and examples defined by schema objects lower in the hierarchy will be ignored.

The above were referring to situations where examples are rendered in sections describing details ***of the parent objects***; e.g. we have a '*Body > Content type: application/json*' section and we render '*Example*' or '*Examples*' of the body payload. However, where a body object has its own schema, such schema will be rendered in its own section titled 'Schema'. There examples defined at any and all levels of the hierarchy will be displayed.

To illustrate, for a body defined in this way:

```
...
      responses:
        '200':
          description: 'Success.'
          content:
            application/json:
              schema:
                description: 'Topmost Schema Object'
                example: 'Example value on the topmost Schema Object'
                properties:
                  nested-schema:
                    description: 'Nested Schema Object'
                    example: 'Example value on the nested Schema Object'
...
```

...the following will be rendered:



Note that if the '`application/json`' object had an '`example`' (or '`examples`') property of its own, value of that property would be displayed in the 'Example' section under the 'Content type', instead.

'example' vs 'enum'

Note that:

- If an object (Parameter Object[376], Header Object[377], Schema Object[378]) defines both '`example`' *and* '`enum`' fields, only the '`enum`' field will be rendered on the API specification page (with label '*Allowed Values*').
- Despite the above, in the case of request parameters it's still useful to define '`example`' field if the *Try this API* feature is to be used, as such field is used to populate request fields in the *Try this API* with default values.

Source's language

NHSD website reads specifications from Apigee and requires that specifications available there they are defined in JSON.

376 https://spec.openapis.org/oas/v3.0.3#parameter-object
377 https://spec.openapis.org/oas/v3.0.3#header-object
378 https://spec.openapis.org/oas/v3.0.3#schema-object

Specification source files are typically defined as YAML files which, as the specification is published to Apigee, are compiled in to a single JSON file (see Referenced content[379] below for more details).

Referenced content ($ref fields)

Following JSON Schema[380], OAS standard permits[381] the use of '`$ref`' fields[382] to support reusable components[383].

The use of `$ref` fields is fine within the source YAML files, but is not supported in the final JSON file which makes the input to rendering of the specification on NHSD website. In order for the referenced components to be correctly rendered on the website, in the final JSON form of the specification the references need to be resolved and the referenced content *inlined* in their place.

The standard scripts provided by the Platforms team make it easier to compile multiple YAML source files into a single JSON specification file, handling the resolution and inlining for you.

Unique identifier for your Spec

It is important to track the specification, and to link the specification into traffic usage (held in Splunk) and other NHS Digital systems - for that wider vision see: APIs, proxies, repos and dashboards[384].

Once your Repo has moved to using a manifest file (see Manifest.yml reference (see page 896)), this will include a Spec GUID and should be inserted as an extension property x-nhs-api-spec-guid:

YAML

```yaml
openapi: '3.0.0'
x-nhs-api-spec-guid: a062e39c-b843-4833-8d24-8fc1434900a0
info:
  ...


...
```

This will allow (all future):

- Visibility of the Spec GUID in Bloomreach and web stats
- Links this particular OAS file to Manifest related processing (CI / CD)
    - Extra metadata is likely to be added, and these values should be included in another extension property e.g. "x-manifest"

Try this API

379 http://nhsd-confluence.digital.nhs.uk#APIspecificationformattingguide-Referencedcontent($reffields)
380 https://json-schema.org/
381 https://swagger.io/docs/specification/using-ref/
382 https://spec.openapis.org/oas/v3.0.3#document-structure
383 https://spec.openapis.org/oas/v3.0.3#components-object
384 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125409294/APIs+proxies+repos+and+dashboards

Toggle support for 'Try this API'

The 'Try this API' feature enables external developers to interact with APIs directly from the specification's web page, eliminating the effort of having to set up any client software. Having said that, whilst most APIs do support 'sandbox' instances that are a typical target of this feature, some do not.

By default this feature is enabled for all APIs and for those APIs that do not support it, it has to be explicitly disabled.

To disable 'Try this API' add `x-spec-publication.try-this-api.disabled` as a property of the top level object of your specification and set its value to true.

Examples:

YAML

```yaml
openapi: '3.0.0'
info:
  ...
x-spec-publication:
  try-this-api:
    disabled: true



...
```

JSON

```json
{
  "openapi": "3.0.0",
  "info": { ... },
  "x-spec-publication": {
    "try-this-api": {
      "disabled": true
    }
  },
  ...
}
```

Response headers

If you successfully make a request to your sandbox API from *Try this API* but the responses displayed do not contain all expected headers returned from the server, make sure to configure your API proxy to enumerate them via header `Access-Control-Expose-Headers` as described on page Cross-origin resource sharing (CORS) (see page 786).

You may want to first verify whether the server actually returns the expected headers by inspecting raw requests and responses in you web browser's Dev Tools.

Requests not working

The most common cause for requests to sandbox API from *Try this API* to fail is misconfiguration of CORS parameters. Refer to instructions on page Cross-origin resource sharing (CORS) (see page 786) for how to configure your API proxy correctly.

Inspecting actual, raw requests and responses in you web browser's Dev Tools often helps in diagnosing the issue.

Editing OAS files in GitHub

Overview

This is a 'How to' guide for making edits to API specs generated as OAS files from Apigee.

YAML files hold all the content that will be published to the API spec and can be edited using GitHub.

GitHub[385] is essentially a versioning manager tool used to make changes to the YAML file which can be reviewed, approved, and merged into the master branch as a new version.

Accessing GitHub

If you don't have a GitHub account, you will need to:

- Open a GitHub account https://github.com/ and sign up
- Be added to the team including two-factor authentication for signing in
- Be added as a contributor to the master file

There are a number of online tutorials for GitHub beginners. They do not cover these project specifics, but they are a good starting point and can be found here[386].

Locate the Master file

Note: Some repositories have their primary branch called **main** instead of **master**.

To locate the master file, perform the following steps:

1.Navigate to the following URL in GitHub:

https://github.com/NHSDigital/.

The NHS Digital GitHub screen is displayed.

385 https://github.com/NHSDigital
386 https://docs.github.com/en/repositories/working-with-files/managing-files/editing-files

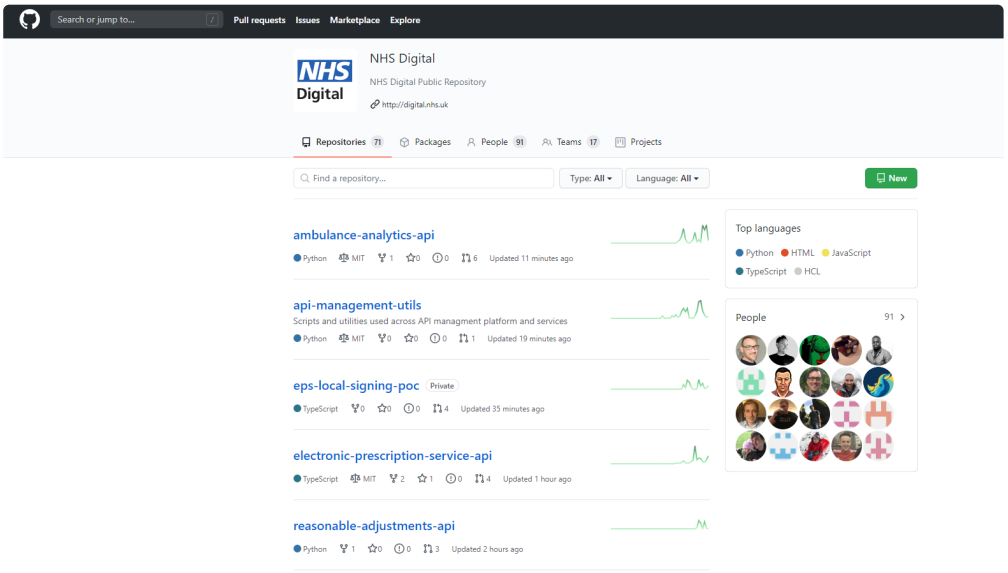
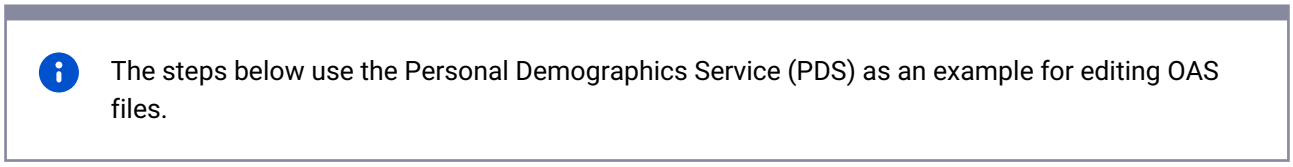
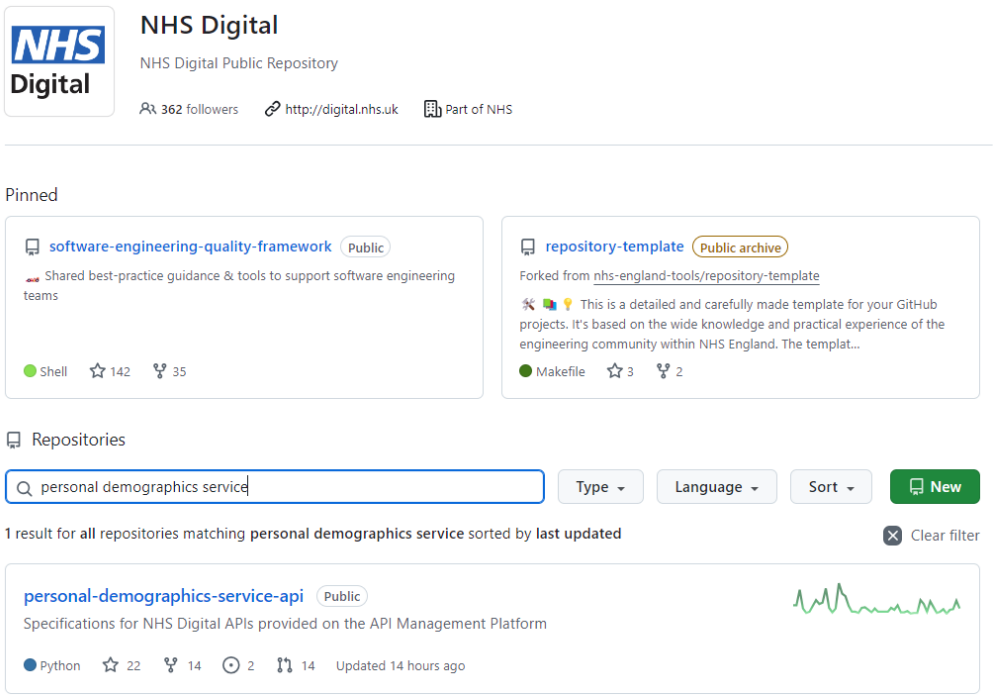


fig. 1 NHS Digital GitHub screen



The steps below use the Personal Demographics Service (PDS) as an example for editing OAS files.

2.Enter the name of the repository via the **Find a Repository** search field, for example, PDS[387].



387 https://github.com/NHSDigital/personal-demographics-service-api

fig. 2 Find a repository screen

3.Navigate to the **specification** folder.

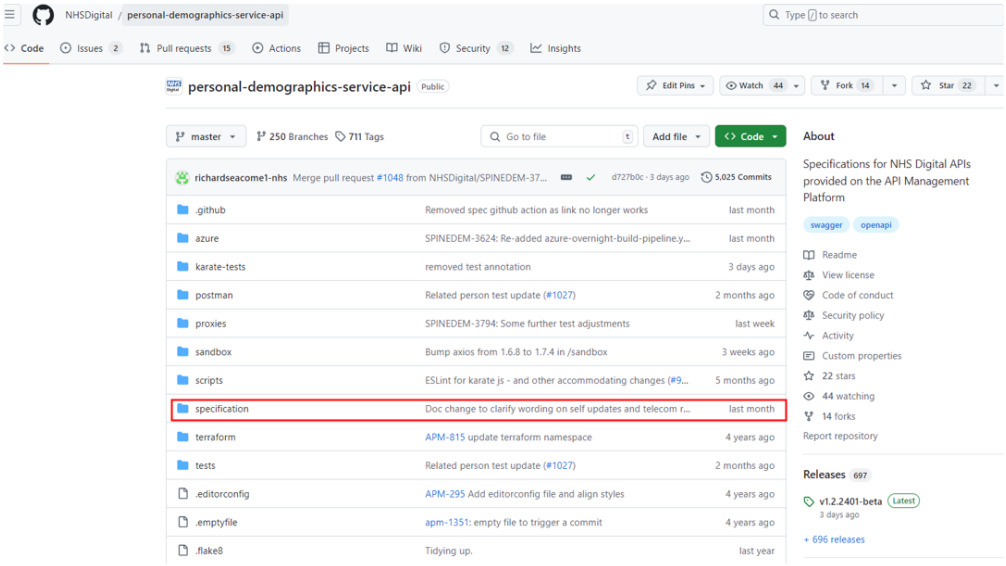


fig. 3 The Specifications folder

4. Click the **.yaml** file to open the editor.

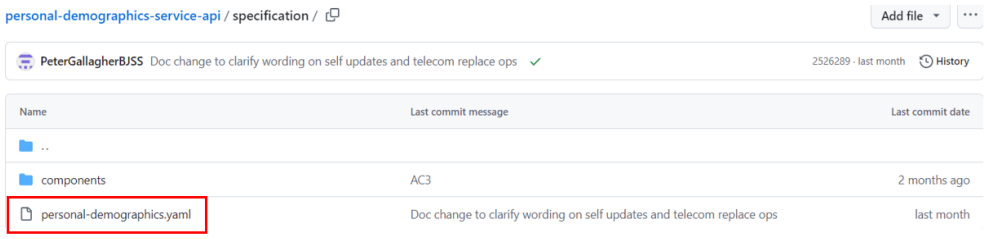


fig. 4 The PDS YAML file

Select a branch

To select a branch, perform the following steps:

1.If you already have some changes to a branch that have **not** been merged into the **master branch**, then select that branch from the drop-down box in the top left.

2.If you **do not** have an existing branch then skip this step as GitHub will prompt you for a branch after you have made your changes.

> The branch protection rules won't let you edit the master branch directly.

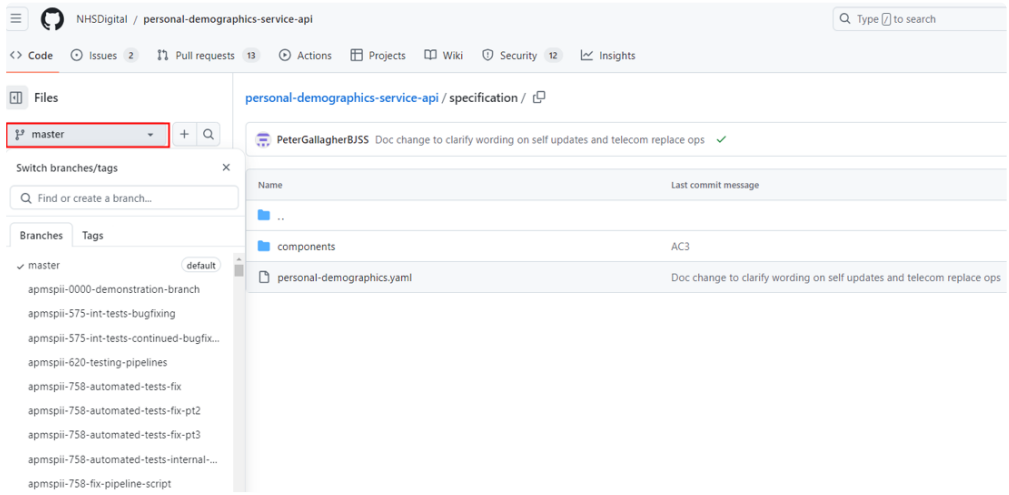


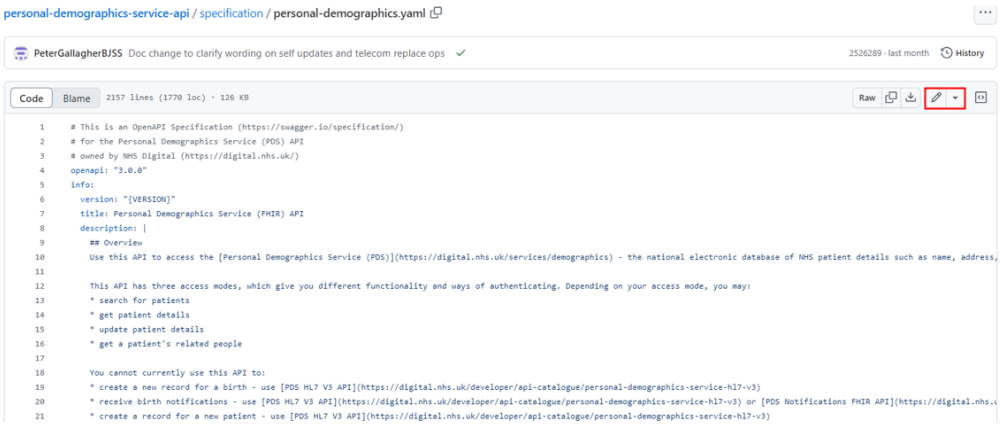
fig. 4 Select a branch for changes not merged to the master branch

Edit the YAML file

> We use the CommonMark format.
> CommonMark does not have tables, so we use an official extension to allow tables with the GitHub markdown syntax.
> If you need to add a table in YAML, ensure you add a line after the table introduction paragraph and the table itself to avoid rendering issues. For further details, see ADZ-2091[388].
> There are a number of styling and writing rules to consider when writing for NHS pages, so you may want to consult our style guide[389].

To edit the YAML file, perform the following steps:

1.Click the ***Edit this file*** *button located a*t the top right corner of the display.



388 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2091
389 https://confluence.digital.nhs.uk/display/APM/Documentation+checklist

fig. 5 Edit button in YAML file

At this stage it is useful to know what you need to change, and where the change is located on the page.

- The Master file contains all the sections listed in the published specification.

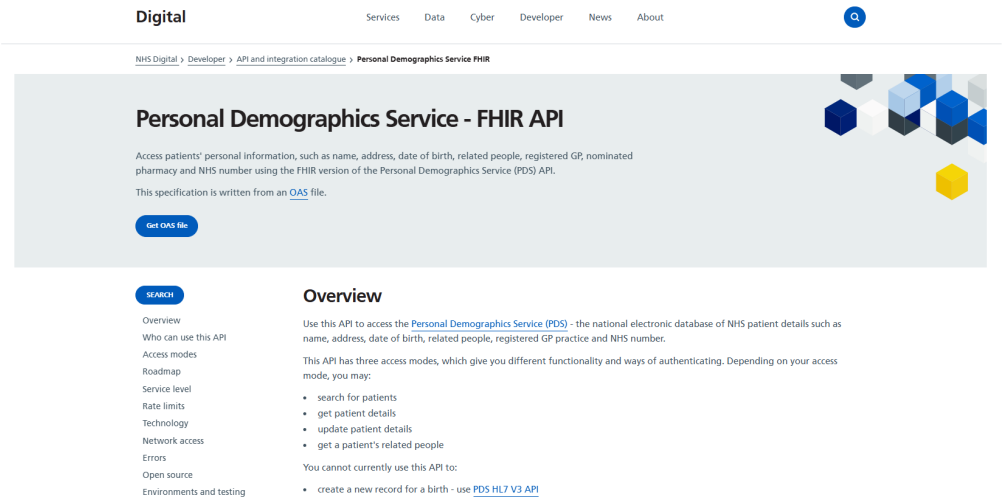


fig. 6 The PDS API (OAS) Overview page

- You can search the .yaml file for the section you need to modify, for example, "Who can use this API".

2.Make the necessary edits to the YAML file, ensuring the correct indentation rules are followed.

Commit your changes

If you have multiple email addresses associated with your GitHub account you can select which one to associate with your commit (your NHS one is a sensible choice for work in NHS Digital repositories, though not required).

> For more information on commits, branches, and merging click here (see page 928).

Once you have made changes to the YAML file, you can commit your changes.

To commit your changes, perform the following steps:

1.Select the **Commit changes** button to open the **Propose changes** form.

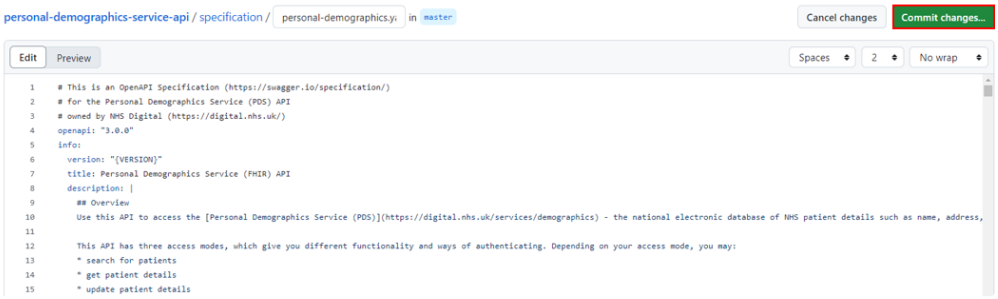


fig. 7 Commit change button

2.Enter a description of the change including the Jira ticket number, for example "**APM-000-AMP-1312-link change**".

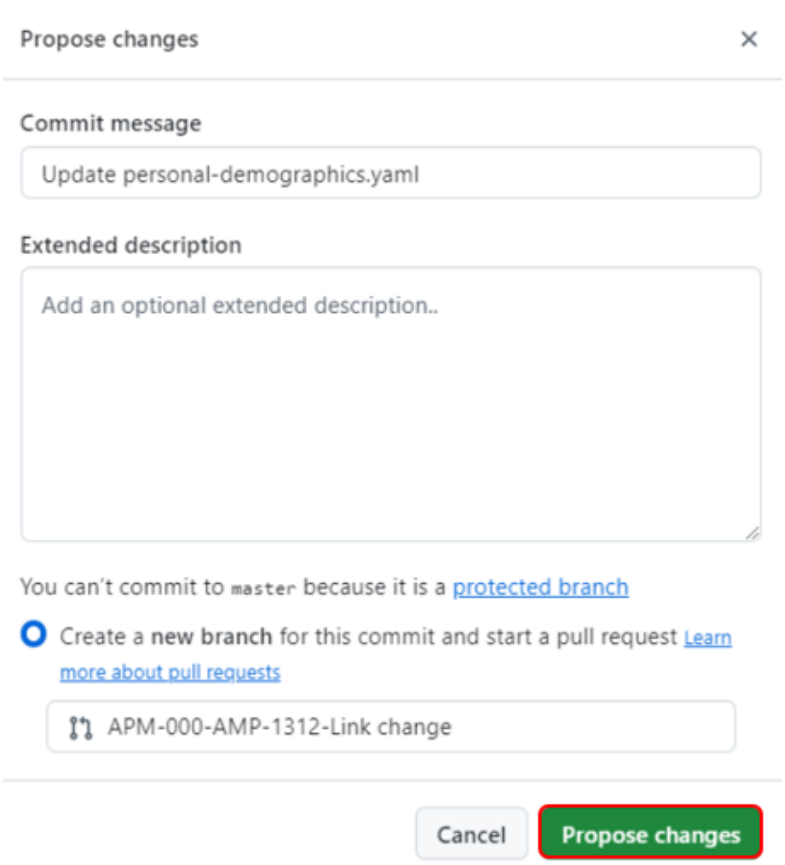


fig. 8 Propose changes button

> You can type a more detailed description in the **Extended description** box, although this step is not mandatory.

3.Click the **Propose changes** button to create the pull request.

Commit changes to a non-master branch

If you selected an existing (non-master) branch in the optional step earlier then you should continue to commit to that branch.

Create a new branch

If you had the `master` branch selected, you should see a greyed out radio button saying "*You can't commit to* `master` *because it is a protected branch*".

If you have the `master` branch selected and do **not** see the message "*You can't commit to* `master` *because it is a protected branch*, your repository is **mis-configured**".
Please contact the Deathstar triage engineer via the **#platforms-apim-producer-support** channel to fix this for you.

Create a pull request

Once you have proposed your change, a PR (Pull Request) is generated which must be reviewed and approved before the change can be merged with the master branch.

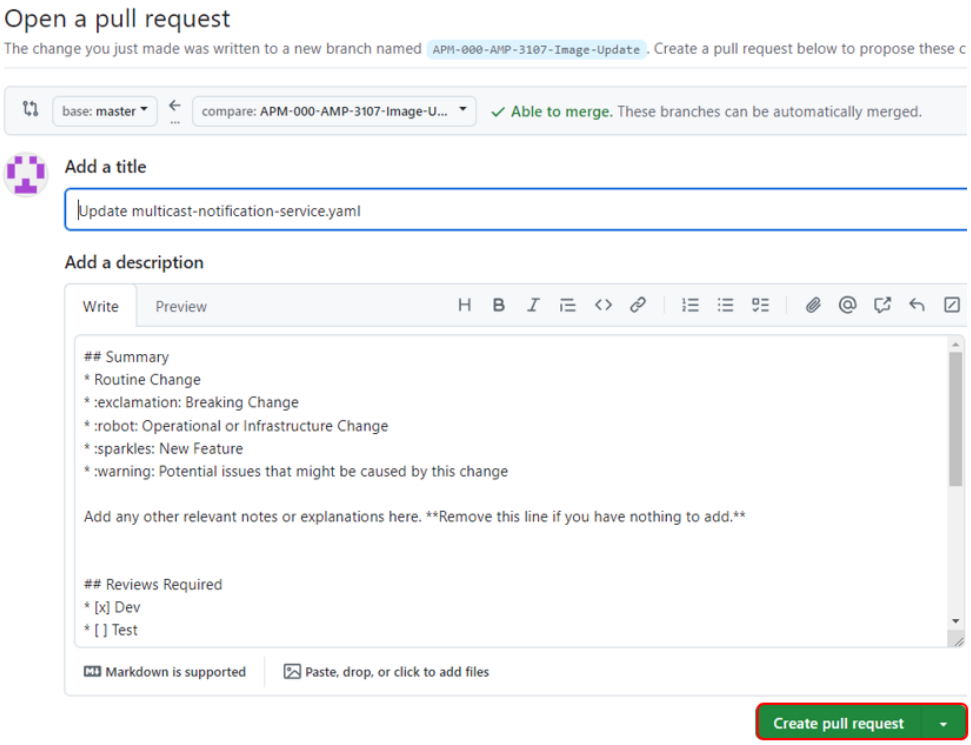


fig. 9 Pull Request #297 submitted for inclusion

1.Send the PR to the appropriate authority, by:

- Select a reviewer within the repo from the top right corner, they will get a notification by email
- Copy the PR URL and send it to them manually

The reviewer will then review your changes and approve the PR.

Merge changes

When your pull request has been approved, you can merge the changes with the Master branch.

To merge your changes, perform the following steps:

1.Click the **Merge pull request** button.

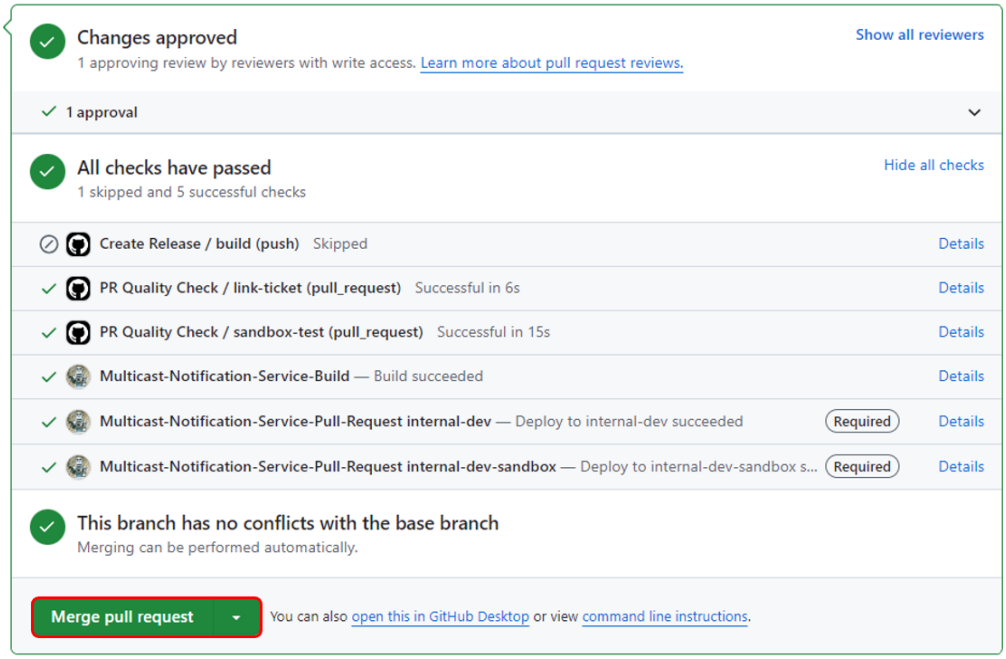


fig. 10 Merge pull request button

2.Confirm the merge by clicking the **Confirm merge** button.

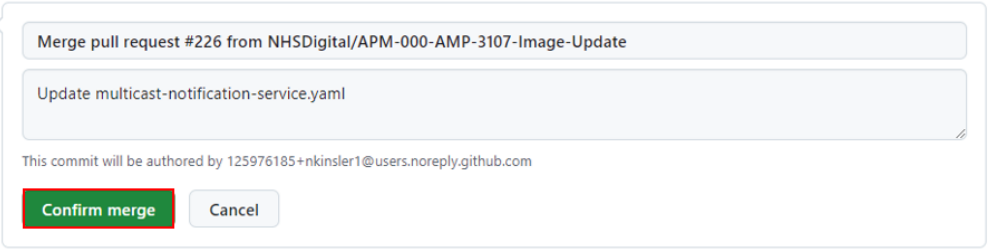


fig. 11 Confirm merge button

Delete branch

When you have confirmed the merge, you can delete the branch.

To delete the branch, perform the following steps:

1.Click the **Delete branch** button to safely delete the branch.

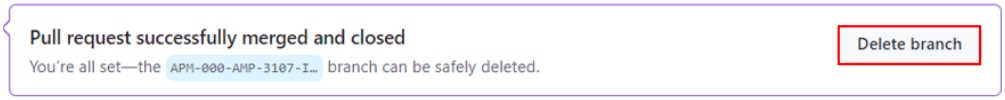


fig. 12 Delete branch button

View "live" changes

When the pull request has been approved and the changes merged to the master branch, you can view your updates on the live web site[390]. In this example the updated page would be viewed via the PDS page[391].

> Updates to the live web site may take a while to publish. Changes are usually visible at five to the hour after the PR has been merged.

Fix YAML errors

Depending on the type of error, you can validate your YAML code using online checkers, such as Beautify Tools' YAML validator[392].

However, sometimes you may get sandbox or pipeline failures (see screen shot below), and you will need to "rerun" the job or submit a message to the Producer Support channel[393] in order to resolve the problem.

Before contacting Producer Support, try and troubleshoot any errors by clicking the **Details** button next to the error, as this will give you an indication of what's wrong, and you may find that you can fix the issue yourself.

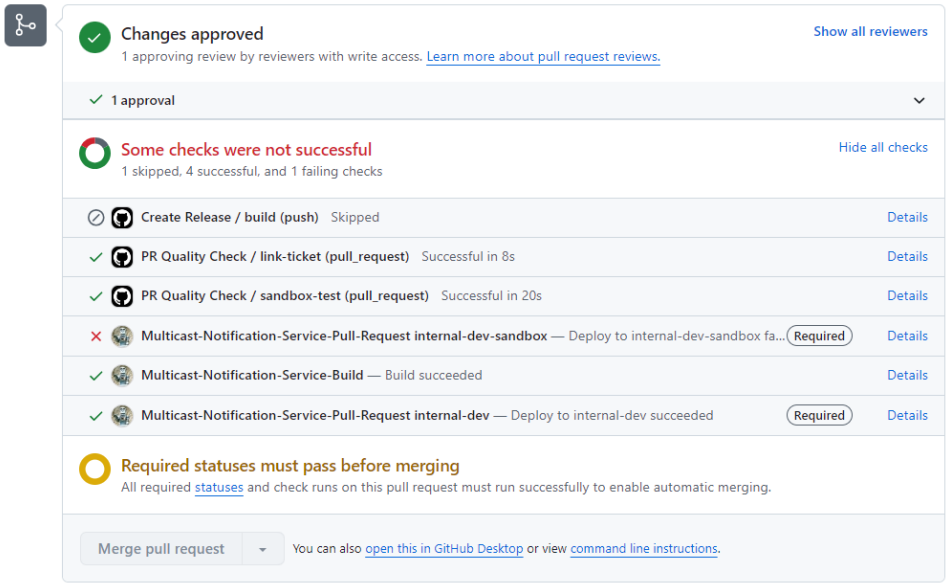


fig. 13 Check changes screen

Publishing fix - Broken Apigee flow pipelines

390 https://digital.nhs.uk/developer/api-catalogue
391 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
392 https://beautifytools.com/yaml-validator.php
393 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY

If the Apigee flow pipelines are broken, changes won't filter to the "live[394]" environment. As a workaround, you will need to send the JSON file to the Producer Support Slack channel. Once they receive the JSON file, it will manually be uploaded to Bloomreach and the changes will be published to the "live" environment.

To publish your content (when the Apigee flow pipelines are broken), perform the following steps:

1. Open the 'YAML to JSON' online converter (https://jsonformatter.org/yaml-to-json) and convert YAML to JSON format.
2. Open the JSON minify online converter (https://jsonformatter.org/json-minify) and minify your JSON (derived from Step **1**).
    a. Click the **Download** button and rename the file to the name of the API specification.
3. Open the online Swagger editor (https://editor.swagger.io/) and paste your minified JSON (derived from Step **2**) and verify if the UI is correct.
    a. Press '**OK**' to convert from 'JSON' to 'YAML'.
4. Once verified, share the minified JSON (derived from Step **2**) to '*platforms-apim-producer-support*' Slack channel and ask them to update the JSON in Bloomreach.

The changes will appear in the live environment once the JSON has been updated to Bloomreach.

Review your API specification in Bloomreach training environment

See Previewing API specifications on the Bloomreach training environment (see page 515).

Editing OAS files in Stoplight and other open source editors

Overview

Stoplight is a tool for editing and previewing OAS files. It's easier to use than raw YAML editing. Stoplight also has a build in .git source control which makes everything nice and safe. This guide won't go into the proper usage of .git, please see Editing OAS files in GItHub. (see page 473)

You can download the stoplight studio editor[395] free, it has a premium upgrade option available but you don't need that to edit any .YAML / API OAS files.

Swagger Editor[396] also enables you to render your API specification visually and interact with your API while still defining it. However, whatever editor you use, ensure you copy the code back to Github in YAML format. For example, to convert files edited in Swagger Editor, select **Edit>Convert to YAML**.

Note: If you just want to validate some YAML outside of GitHub then Stoplight might be overkill. There are simpler online checkers available, such as Beautify Tools' YAML validator[397] which we've used successfully.

Prerequisites

- Download the stoplight studio editor[398].

394 https://digital.nhs.uk/developer/api-catalogue
395 https://stoplight.io/studio/
396 https://swagger.io/tools/swagger-editor/
397 https://beautifytools.com/yaml-validator.php
398 https://stoplight.io/studio/

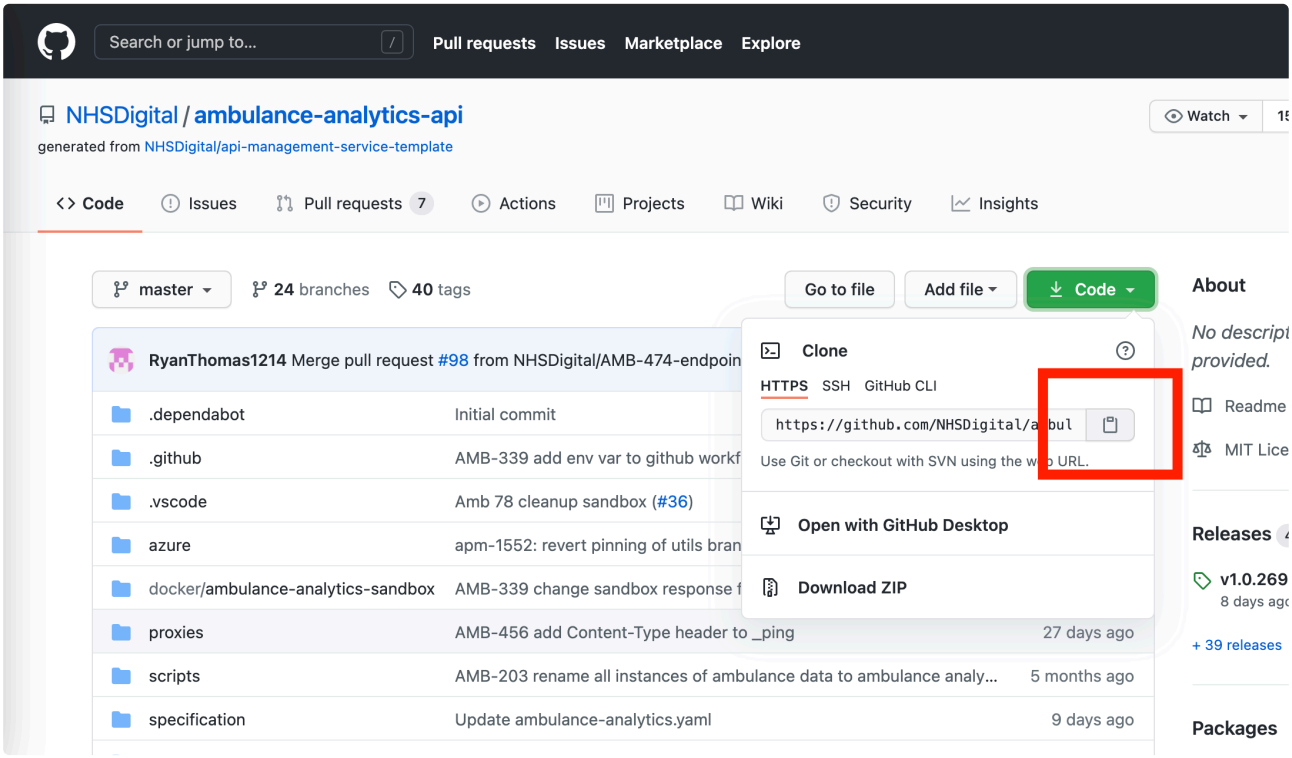
- Github developer account using your NHS Email.
- Account needs adding to `NHSDigital/api-gateway` permission group. (Only allowed by a github admin)

1 - Get the source code

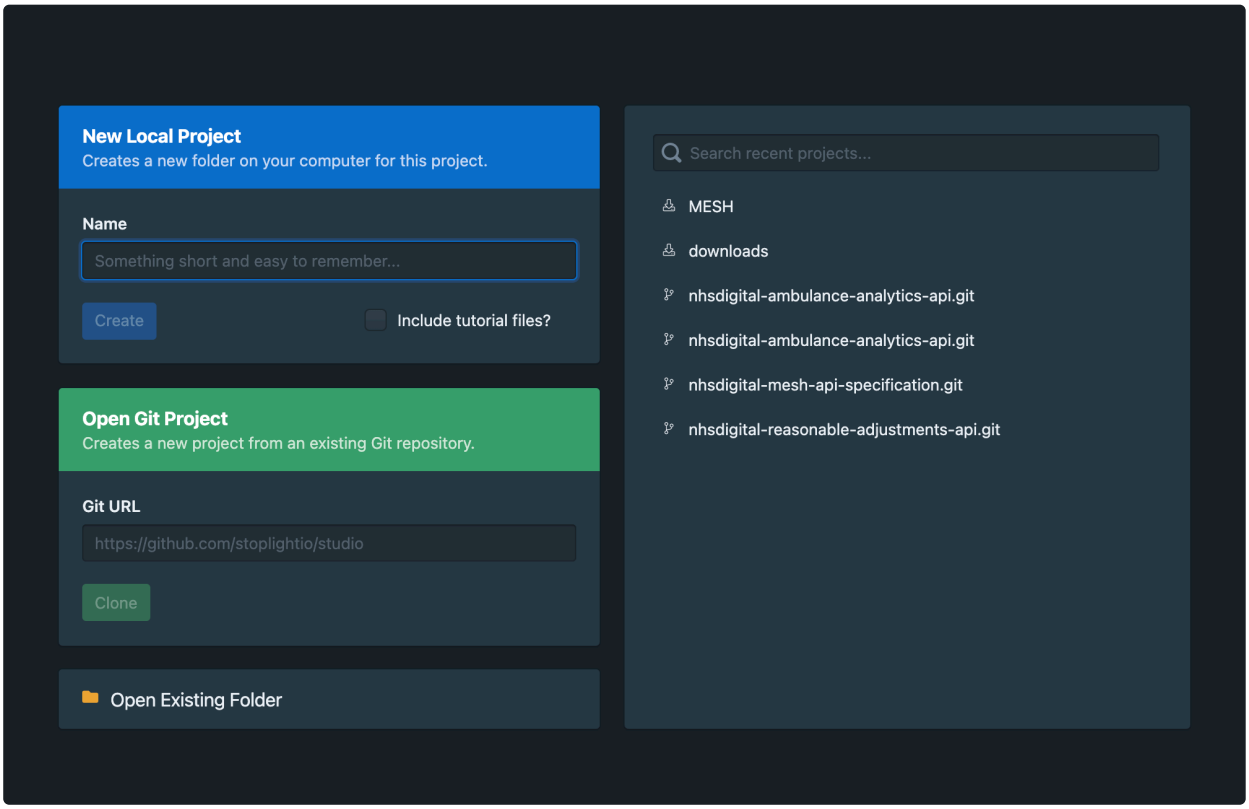
If the project and the spec already exist, you can obtain the source code from github.

e.g. https://github.com/NHSDigital/ambulance-analytics-api

Copy the .git URL for cloning.

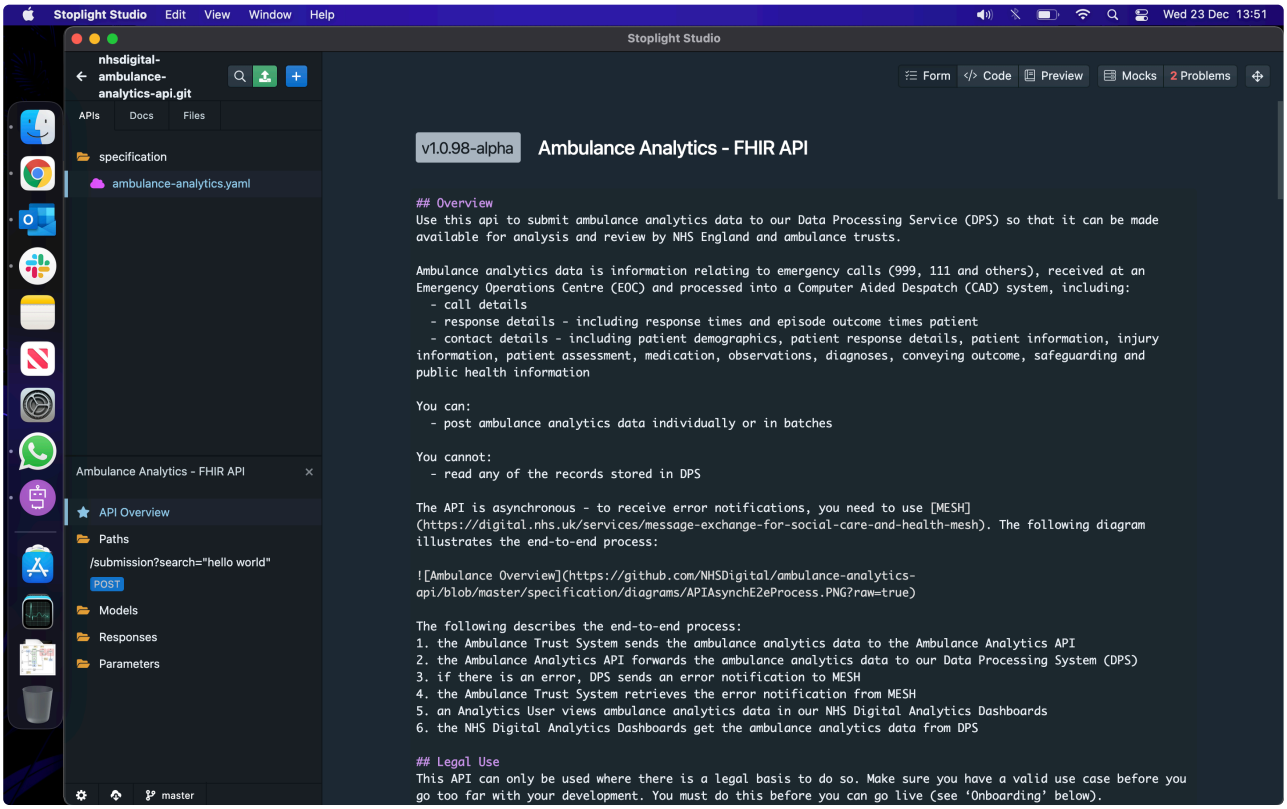


Next copy it into Stoplight.



Once copied, click 'Clone', and next time you launch it will appear on the right as can be seen here. Next time you launch you can double click that link to begin editing.

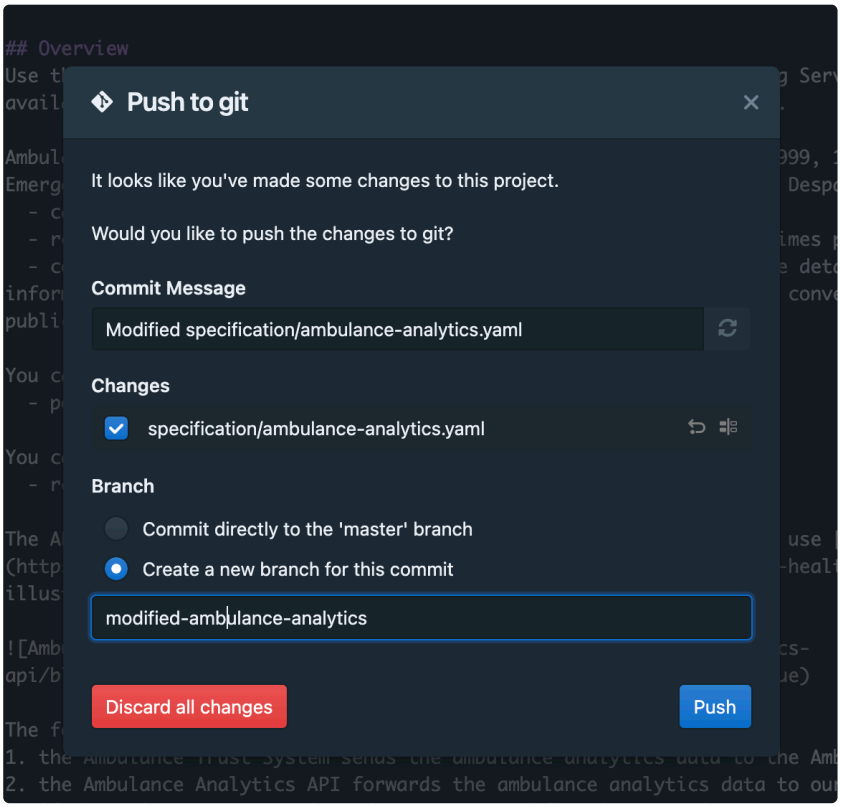
2 - Create a branch



Branches

The very bottom left you can see it says 'Master', This shows us what branch we are on. Clicking on this shows us the branch selector.

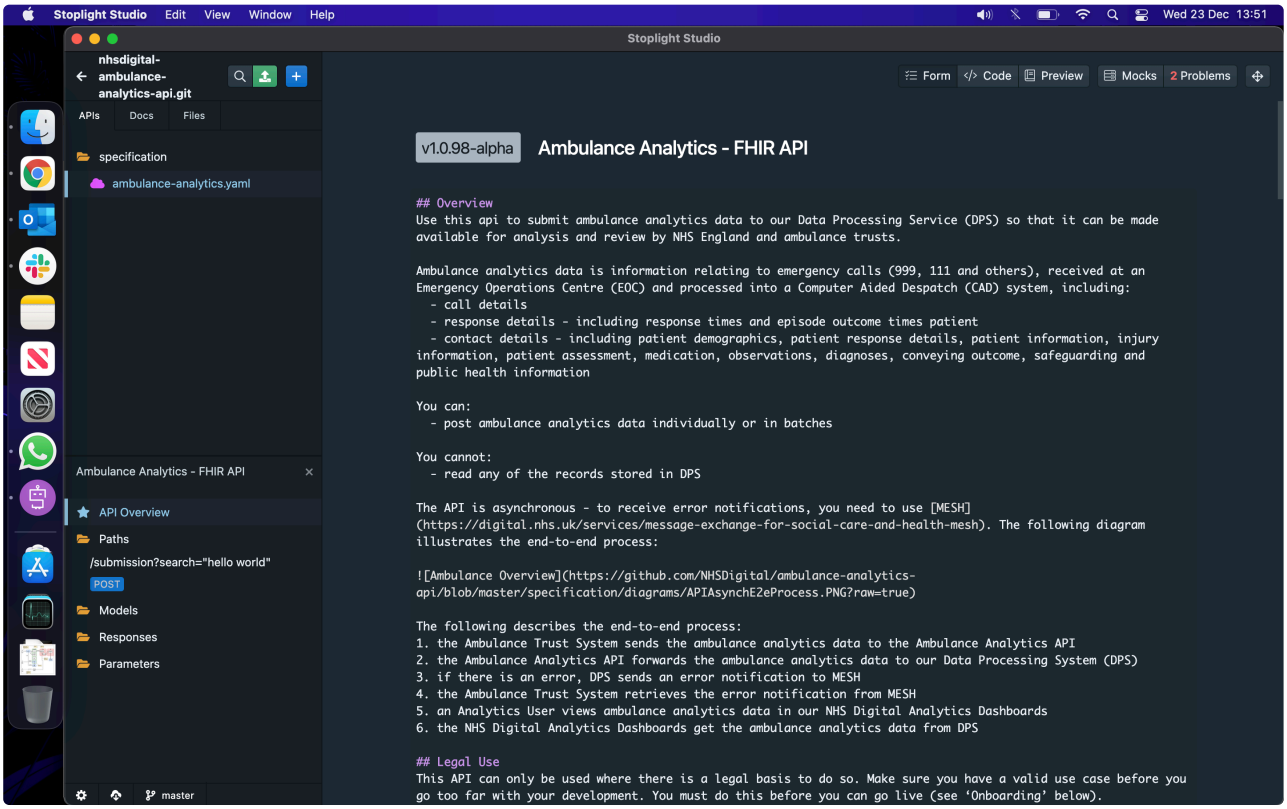
To create your own branch thought we need to make a few changes. Once you have the green arrow button will become enabled.



Here i have selected the option to create a new branch, It is important to give the branch a meaningful name, and commit message. Your developers will explain in detail the reason for this.

By clicking Push your changes will forever be saved.

3 - Making changes



First thing with looking at the above screen shot is getting familiar with the interface and what the different areas mean.

In the main window, we can see the spec in the 'form' editor, this follows basic MarkDown, which we know the spec is written in.

The other two views for the spec are, 'code' and 'preview'. You can switch between them by using the selector in the top right.

Bottom left quadrant is the spec overview, here you can select to view the 'Overview' or switch between endpoints. Editing the spec or updating the technical information.

API content style guide

Overview

This page gives guidance on writing documentation for NHS Digital APIs.

Principles

- **Get someone with content skills to help write your API documentation.** It should ideally be a combination of a developer / architect and a content designer / tech author. The former will focus on technical accuracy and the latter on readability. If you do not have a tech author, as the API management team, we may be able to help, please ask.

- Your audience includes not just programmers and architects, but also product owners, business analysts, testers, entrepreneurs and others. Whilst you cannot explain every technical detail, you can do your best to make the content accessible to all these roles.
- Assume the reader does not know NHS terminology, especially acronyms. Either explain it or link to an explanation. For example:
    - not everyone knows what N3 or HSCN is
    - not everyone knows what Spine is
- **Test your content with real users.** During alpha and beta, find software developers who will be using your API and make sure they can understand the content. If they cannot, iterate it until they can. Record your findings as part of your user research[399].
- If you want to know why these principles are needed, read this blog article on why we need to improve the developer experience with NHS Digital, "Why the TRUD must die[400]".

Style guides

The two main general writing style guides that you need to use are:

- NHS Digital A to Z of house style[401]
- GDS Style guide A to Z[402]

In the spirit of not duplicating the same information here, you can look up specific advice on, for example:

- abbreviations and acronyms
- active voice
- ampersands
- apostrophes - for example, the plural for API is APIs, not API's
- bullet points
- capitals
- commas
- contractions
- dashes and hyphens
- e.g., i.e., etc.
- email
- exclamation marks
- headlines, titles and headings
- numbered lists
- semicolons
- specific word usage - for example, internet, web, webpage, website, NHS Digital, NHSmail, Spine
- use of present tense
- and many more

Other style guides you might find useful are the:

---

399 https://nhsd-confluence.digital.nhs.uk/display/APM/User+research

400 https://medium.com/@marcus_baw/why-the-trud-must-die-ac27377ee189

401 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style

402 https://www.gov.uk/guidance/style-guide/a-to-z-of-gov-uk-style

- NHS Digital content style guide[403] for general guidance on writing NHS Digital content
- NHS content style guide[404] for general guidance on writing content for digital NHS services, but bear in mind its target audience is citizens, not software developers
- GOV.UK content design guide[405] for very general guidance on writing for public services, but again bear in mind our specialist audience

A to Z of API content

This is an A to Z of content style guidance that is specific to writing API documentation. It does not repeat things mentioned in any of the above links.

A - API and endpoint descriptions

When writing a description for an API or an endpoint, write it from the developer's perspective, not the API's perspective.
In other words, explain what the developer can do with this API, as opposed to what the API can do for the developer.

When describing an API, start the sentence with "Use this API to", for example:

- "Use this API to find patients and to retrieve and update their personal details held nationally by NHS Digital."
- (as opposed to "This API finds patients...")

When describing an endpoint, keep things shorter by omitting "Use this API to", for example:

- "Get current information for the patient with a given NHS Number."
- (as opposed to "Returns current information...")

Use simple language to describe the operation of the API and its endpoints in context, and start your decription with a verb, as follows:

Preferred verb	Suitable context-dependent alternatives	Avoidable synonyms
get	read, search, check, download, retrieve, track, validate, lookup	
update	update, upload, acknowledge	patch, put
add	create, send, submit, convert, request, upload	

403 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk
404 https://beta.nhs.uk/service-manual/content
405 https://www.gov.uk/guidance/content-design/writing-for-gov-uk

Preferred verb	Suitable context-dependent alternatives	Avoidable synonyms
remove		delete

Examples, from different APIs:

- **Search** for a patient[406]
- **Update** patient details[407]
- **Lookup** MESH address[408]
- **Track** outbox message[409]
- **Get** reasonable adjustments[410]
- **Get** impairments[411]
- **Get** patient's Summary Care Record[412]
- **Upload** patient's Summary Care Record[413]

In your Overview, this looks something like:

"You can:

- search for a patient
- get patient details
- update patient details

"You cannot:

- create a new record for a birth
- receive birth notifications
- create a record for a new patient"

Also bear in mind that you can reasonably "create a patient record", but you literally cannot "create a patient".

B - bulleted lists

Do not use more than one level in a bulleted list, in other words don't indent lists:

- because this is right
- as this is correct
    - because this is not right

406 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-Default-search-patient
407 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#api-Default-update-patient-partial
408 https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#api-Default-mesh-address-lookup
409 https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#api-Default-outbox-tracking
410 https://digital.nhs.uk/developer/api-catalogue/reasonable-adjustment-flag-fhir#api-Default-readFlag
411 https://digital.nhs.uk/developer/api-catalogue/reasonable-adjustment-flag-fhir#api-Default-readList
412 https://digital.nhs.uk/developer/api-catalogue/summary-care-record-fhir#api-Default-get-scr
413 https://digital.nhs.uk/developer/api-catalogue/summary-care-record-fhir#api-Default-upload-scr

- as this is not correct

This is bad content design and not supported by NHS or GDS guidelines, allegedly.

D - diagrams

Diagrams can be useful in API documentation to explain business processes or technical flows. However, they do bring accessibility challenges which you must take into consideration. If you want to add diagrams to your OAS page, contact us first.

For each diagram, consider:

- NHS Digital guidance on image sizing, aspect ratio, file formats - see Images on the NHS Digital website[414]
- NHS Digital guidance on making content accessible[415] in the NHS Digital service manual especially regarding use of alternative text in informative diagrams or images
- GDS guidance on how to write alt text, produce accessible diagrams, choose images and copyright standards - see Content design: planning, writing and managing content - Images[416]

D - dyslexia

User research was carried out in January 2022, interviewing people who use the Developer hub who also have accessibility needs. Two volunteers had dyslexia, and some findings specifically around their needs were:

- Be concise
- Break dense content up in to shorter paragraphs
- Use left justification, opposed to full justification
- Use pictures or "how to" videos in place of text when possible

Further writing advice can be found on the British Dyslexia Association website[417].

E - endpoints

Use this term consistently in FHIR API specs as a subheading to describe how your developer can interact with your API's endpoints. See the PDS FHIR API specification[418] which we developed as an example for API specifications.

Note that:

- technically speaking "endpoint" is just the "URL" part of the verb-URL pair eg GET URL, POST URL, and so on, but we have deliberately chosen to ignore this for ease of use purposes

414 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-look/images-on-the-nhs-digital-website

415 https://service-manual.nhs.uk/accessibility/content

416 https://www.gov.uk/guidance/content-design/images

417 https://www.bdadyslexia.org.uk/advice/employers/creating-a-dyslexia-friendly-workplace/dyslexia-friendly-style-guide

418 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir

- in user research at HMRC and NHSD developers and product owners both recognise this language and prefer it to alternative (and sometimes more technically accurate) terms such as "requests", "interactions" or "operations"
- under Endpoints we distinguish further subheadings of "requests" to the endpoint and "responses" from the endpoint

E - English language variant

Use British English, not American English.

There may be exceptions, for example, some API field labels might be based on international standards which use American English. You should still describe them in British English despite some inherent awkwardness, for example:

- The `Authorize` header submits the required authorisation.
- `Organization` is the owning organisation.

G - glossary of developer terms

We maintain a glossary of NHS developer terms and abbreviations[419], along with their definitions.

If your API adds terms not covered in it, ask us to add the relevant new entries. The same applies to our pages on API technologies at NHS Digital[420] and the A to Z of developer resources[421].

H - healthcare worker

Use "healthcare worker" when you mean "people who work in health and social care" rather than one of the many alternatives:

- healthcare professional - excludes admin staff
- health and care professionals - excludes admin staff
- health and care worker - acceptable alternative if you have a specific reason to emphasise that social care is included, but that is often taken as read
- health and care staff - staff excludes contractors
- users - too general
- end users - too general
- mobile workers - too specific unless that's exactly what you mean
- smartcards users - too specific unless that's exactly what you mean

N - NHS number not NHS Number

Although not API specific, the NHS Digital A to Z of house style neglects to mention that "NHS number" has a lowercase "n" in number - we have been incorrectly saying "NHS Number".ADZ-906 will fix this.

419 https://digital.nhs.uk/developer/guides-and-documentation/glossary-of-developer-terms
420 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital
421 https://digital.nhs.uk/developer/guides-and-documentation/a-to-z-of-developer-resources

P - patient

Use "patient" when you mean "people who receive health and social care or make use of services" rather than one of the alternatives:

- citizen - too much baggage to do with being a legally recognized resident or national of a country, either native or naturalized.
- service user - acceptable alternative "patients and service users" if you have a specific reason to emphasise that social care is included, but "patient" is sometimes used as an equivalent anyway eg when talking about "NHS numbers... as the only national unique patient identifier...across health and social care"
- people - too broad a term which can covers patients, carers, their family and friends.
- public - too general
- patients and the public - unclear whether this includes "healthcare workers" or not. Normally we want to differentiate between software for "healthcare workers" and "patients" so this pairing is of limited use with APIs
- public and people - too general

Typical usage in an API context is:

- "patient-facing applications" - versus ones used by healthcare workers
- "patient access mode for the PDS FHIR API" - versus healthcare worker access mode, or application-restricted access mode
- API catalogue filter: Care setting -> Patient - versus other Care settings like Hospital, Dentistry, Pharmacy or Social care.

V - videos

Videos are easier to understand than text for some people. They do bring accessibility challenges, so see the advice on making content accessible in the NHS Digital service manual[422] especially regarding making video and multimedia accessible. If you want to add videos, please contact us first.

Exemplar API pages

Here are some examples of pages to show how you should organise and format API specification pages, in line with the style guides:

- Personal Demographics Service - HL7 V3 API[423]. A manually written description for a legacy HL7 V3 API, re-formatted using BloomReach (see page 508)into the current NHS style. Note: we migrated this API to the API platform at bronze level[424] and applied our developer hub minimum bronze standard[425] to its API cover page.

422 https://service-manual.nhs.uk/accessibility/content
423 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-hl7-v3
424 https://nhsd-confluence.digital.nhs.uk/display/APM/Migrating+APIs+onto+the+API+Platform
425 https://nhsd-confluence.digital.nhs.uk/display/APM/Writing+a+bronze+API+service+cover+page

- Personal Demographics Service - FHIR API[426]. A description which is generated as part of defining the API (see page 473). The editable text is embedded in one or more nested yaml files. Note: there are some limitations on what formatting we can apply using the yaml file as a source for documentation.

Linking API specifications to FHIR standards

Overview

This page explains how your API specification and FHIR resources, profiles and Implementation Guides should link together.

Details

If your API conforms to the FHIR standard, things are a bit more complicated.

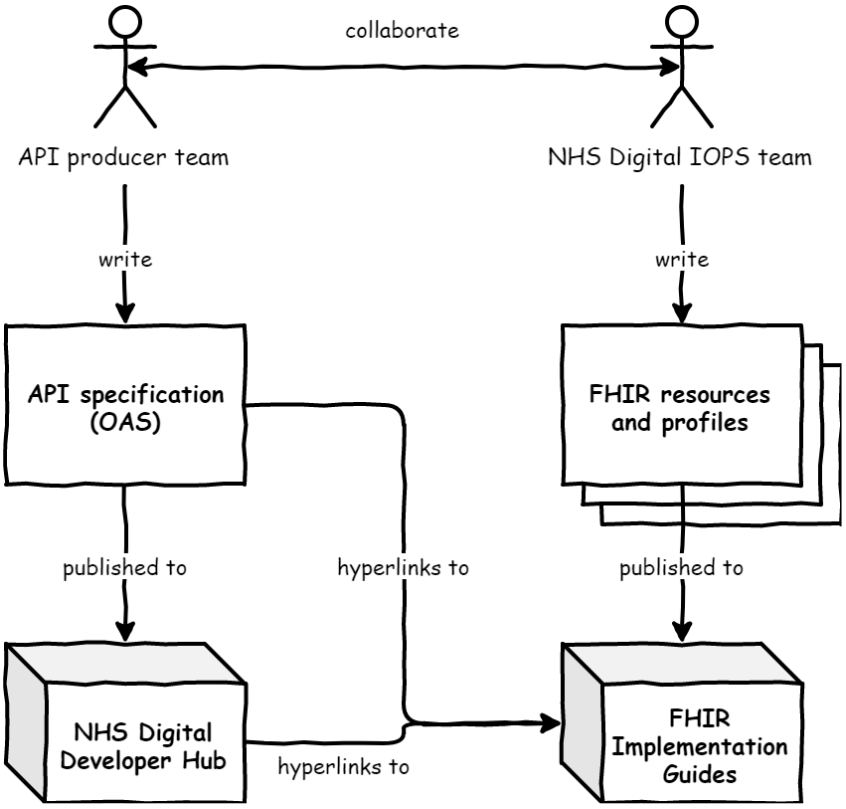
There will likely be some FHIR resources and profile standards that your API follows.

Your API specification should be able to stand alone - it should be possible for developers to understand how your API works without having to refer to FHIR standards documentation.

That said, your API documentation should probably refer out to the relevant FHIR standards via hyperlinks in the OAS specification.

The following diagram illustrates the relationship.

426 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir



The differences between the Developer Hub and the FHIR Implementation Guides are explained below.

Developer Hub

Overview -

The Developer Hub can help to accelerate the onboarding process by providing developers a centralised place for discovery of NHSD APIs and tools that can be used to successfully use these APIs, including:

- API documentation
- Tutorials
- Code snippets
- Examples

The Developer Hub can also allow developers to:

- Register apps in order to access APIs
- Test API (mocking service)
- Provide feedback and make requests

NHSD Web standards –

The Developer Hub can be used to specify NHSD usage of common web standards such as XML, JSON, HTTP, OAuth, etc.

API Definition Language –

- The Developer Hub can be used to publish API documentation generated from OpenAPI Specification[427] (OAS)/ RESTful API Modelling Language[428] (RAML) files.
- The API Definition Language can be used to auto-generate documentation, mocked endpoints and interfaces for API implementations and encourages reuse, enables discovery and pattern sharing and aims to support best practice.
- OAS/ RAML API Definition Languages are industry agnostic – they can be used to define any API whether healthcare specific or not.

API Console –

A Developer Hub API Console could support API calls using a mocking service:

- Returns early sample API responses defined in the API definition
- Useful for early developer feedback before API is implemented

What is the difference between API Specs and FHIR Implementation Guides?

The API specs (OAS/ RAML) and FHIR Implementation Guides are specifications for describing APIs. The main difference between them (at a high level) is that the OAS/ RAML standards describe non-industry specific RESTful APIs whereas FHIR is a dedicated healthcare API specification standard which is used to describe not only RESTful APIs but other exchange paradigms such as documents and messaging.

API Specs:

It is assumed that the alpha DDC API gateway/s will initially support OAS generated interfaces to API implementations. The assumption is based on Swagger (OAS 2) - as MuleSoft 4 (Anypoint API Designer) is NOT compatible with our OAS 3.0 specifications.

OAS API specifications define a standard and programming language-agnostic interface description for REST APIs, allowing provider backend service capabilities to be discovered, and understood, without access to source code - in essence it enables consumer systems to interact with provider services with minimal implementation logic.

The OAS API specification at its simplest level defines what you can call, what you send and what you get back from an API implementation.

OAS allows an API to define -

- Resources/nested resources
- Methods and HTTP responses
- Query parameters
- Modularise/compose API definitions using reusable fragments e.g. data types, traits, resource types, extensions, security schemes, libraries etc

The alpha DDC API gateway/s (Apigee and MuleSoft) phase will initially support OAS generated interfaces to API implementations e.g. PDS that are conformant to FHIR (*this needs confirmation*), with a view to supporting dual OAS and FHIR RESTful interfaces in a future phase.

To support the alpha phase, the intention is to link from the OAS specs to standalone FHIR specs (published on GitHub?) which will provide greater detail on the NHSD (England/ UK) FHIR profiles (resources) which can be returned from an API call. The NHSD (England/ UK) FHIR profiles can include extensions, bindings to terminologies and profile slicing (see below).

427 https://swagger.io/specification/

428 https://raml.org/

FHIR Implementation Guides

FHIR Implementation Guides are NHSD published FHIR Specifications based on the STU3[429] and R4[430] FHIR International standards. They specify NHSD (England/UK) FHIR customisation of the platform FHIR International standard and include England/UK FHIR assets such as profiles, extensions, terminologies, slicing and custom data types which are constrained to support specific England/UK healthcare data exchange use cases.

In addition, NHSD RESTful FHIR API Implementation Guides can also define RESTful API constructs similar to the OAS API definition language. For example, the FHIR resources identified by the API endpoint (URL), FHIR methods used to interact with the resources and FHIR query parameters used for search filtering. As stated above, it is anticipated that during the Alpha phase that OAS API specs will link to the FHIR Implementation Guides in order to provide the FHIR API definitions.

NHSD FHIR Implementation Guides align to many FHIR principles. The following principles are highlighted as they support the case for the publication of standalone NHSD FHIR Implementation Guides using the available bespoke HL7 International community IG Publisher tool.

FHIR usability –

- FHIR resources can be understood by technical and non-technical people alike. Even if the syntax (XML/JSON) is hard to understand, non-technical people can view a rendered set of FHIR profiles, value sets etc in any browser or text reader and understand the contents within them.
- Narrative content supports different views – e.g. text, snapshot, differential..
- Detailed descriptions – definitions of elements including Control, Type, Bindings, Invariants, Is Modifier etc

FHIR Decomposition –

Aside from principal FHIR components (resources and RESTful APIs) the NHSD FHIR Implementation Guides may include some of the following components:

- Information Model – the components of FHIR related to the creation of FHIR resources e.g. Base Classes, Data Types
- Constraints – the components of FHIR addressing constraints, validity and conformance (e.g. conformance layer and rules)
- Terminology – the components of FHIR related to clinical terminologies and ontologies
- Usage – the component of FHIR addressing the use of FHIR in a run-time capacity e.g. FHIR messaging

Composability –

- FHIR profiles are highly composable and NHSD FHIR specifications allow for complex structures to be built from atomic resources e.g. The GPConnect AccessRecordStructured or the EOL GetRecord FHIR structures.

FHIR Design Patterns -

- NHSD FHIR Implementation Guides can assist implementers to better understand the relationships between FHIR resources and to support the use of abstractions of the resources when implementing common tasks.

---

429 http://hl7.org/fhir/STU3/
430 http://build.fhir.org/index.html

FHIR Implementation Guides can also be used to define -

- Different FHIR Paradigms e.g. Documents, messaging and services
- FHIR examples using healthcare standards
- FHIR Mappings, conversions etc
- Implementation support: testing, validation, reference implementations, code snippets, UK Core, data standards
- Business requirements and use cases

HL7 IG Publisher Swagger API definitions -

A future release of the HL7 International community IG Publisher tool is anticipated to include the functionality to generate Swagger (OAS 2) API definitions for RESTful FHIR APIs defined by an HL7 FHIR implementation guide.

It is expected that the tool will be able to generate a single swagger file for all resources (single), or a file for each resource (split).

See: https://confluence.hl7.org/display/FHIR/IG+Publisher+Documentation#IGPublisherDocumentation-Swagger

Editing OAS files in Beautify Tools and Preview in Bloomreach UAT

Overview

Beautify Tools[431] Use this application if you just want to validate a YAML code outside of GitHub. It is a simple online validator to Validate YAML formatting.

WARNING - AVG is currently blocking this site as being infected with **URL:Phishing**

Advantages

This method of publication is preferred due to the following advantages:

1. validate YAML using Beautify Tools YAML validator
2. copy and paste YAML in GitHub Repo
3. use the named spec page in UAT to view publication without approval from owners (see section 2 below)

Prerequisites

- Project repo is created in GitHub

431 https://beautifytools.com/

- Account needs adding to `NHSDigital/api-gateway` permission group. (Only allowed by a GitHub admin)

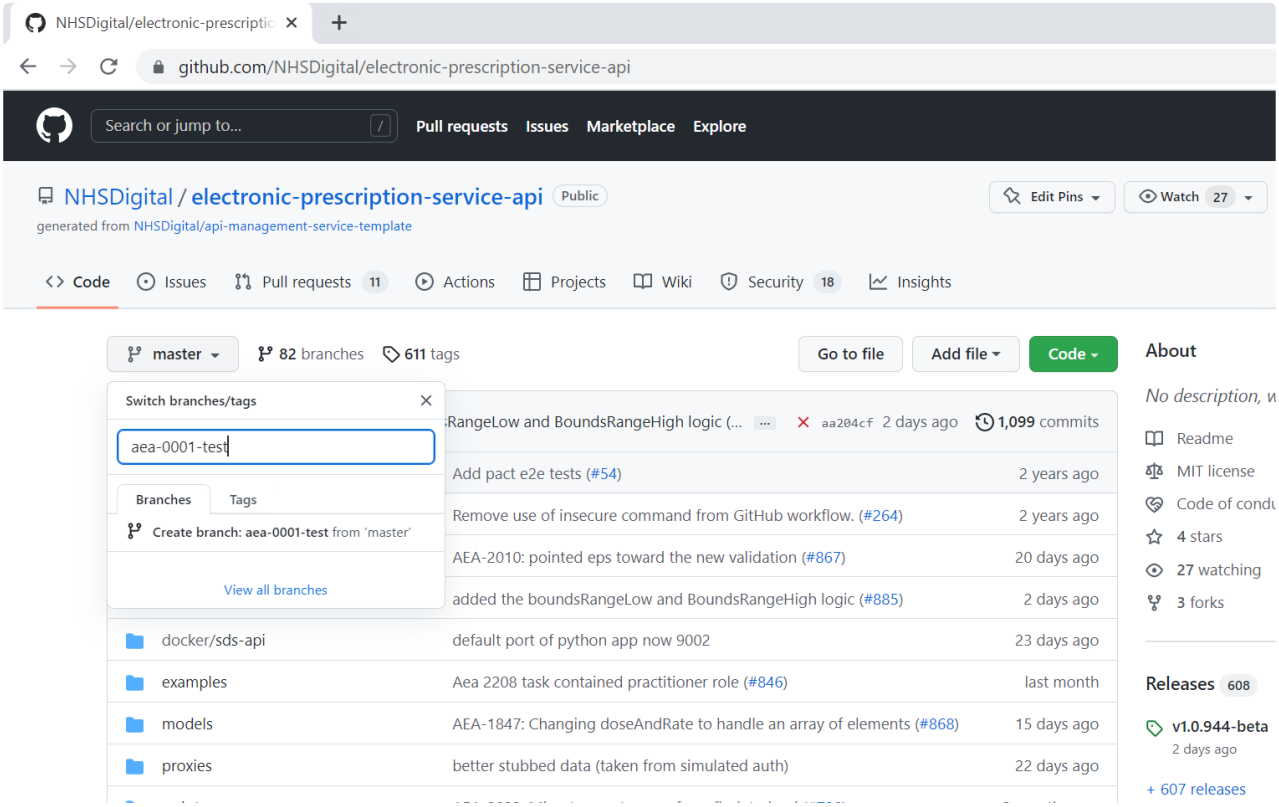
1 Create a branch in the GitHub repo

If the project and the spec already exist, you can create a branch from the master as shown below.

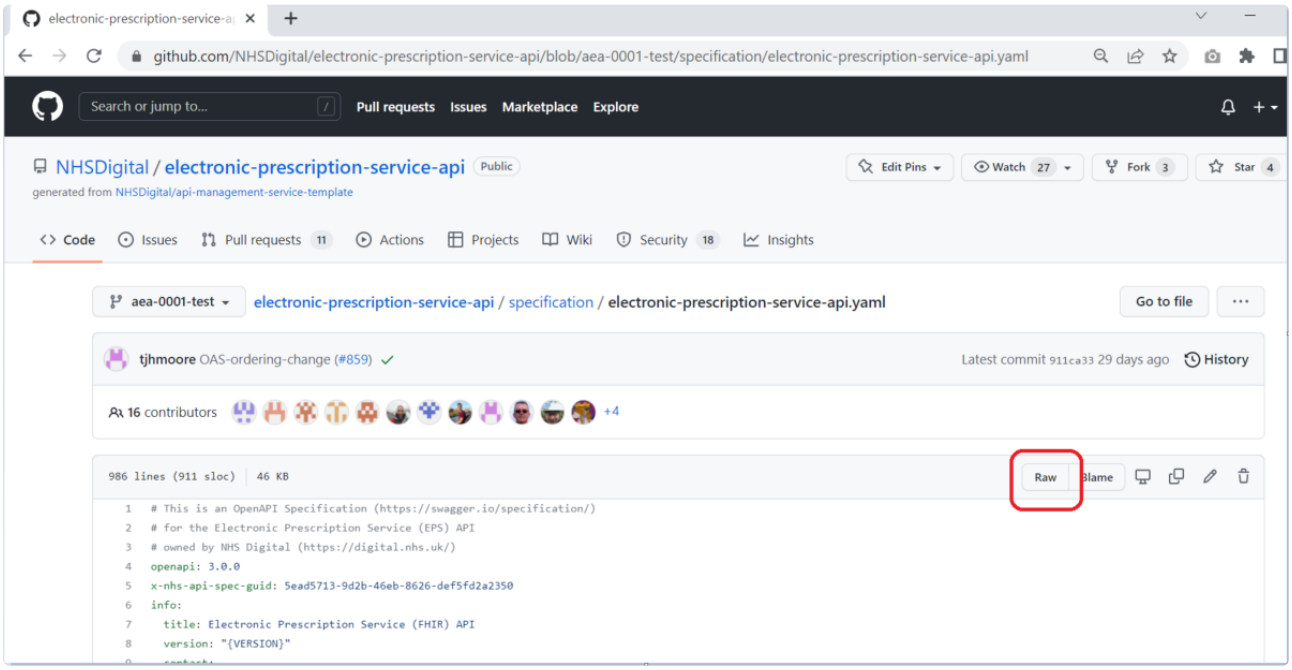
1.1 Search for branch name

- search for a branch name - its good practice to start with a JIRA reference
- click on create branch, to create a new branch

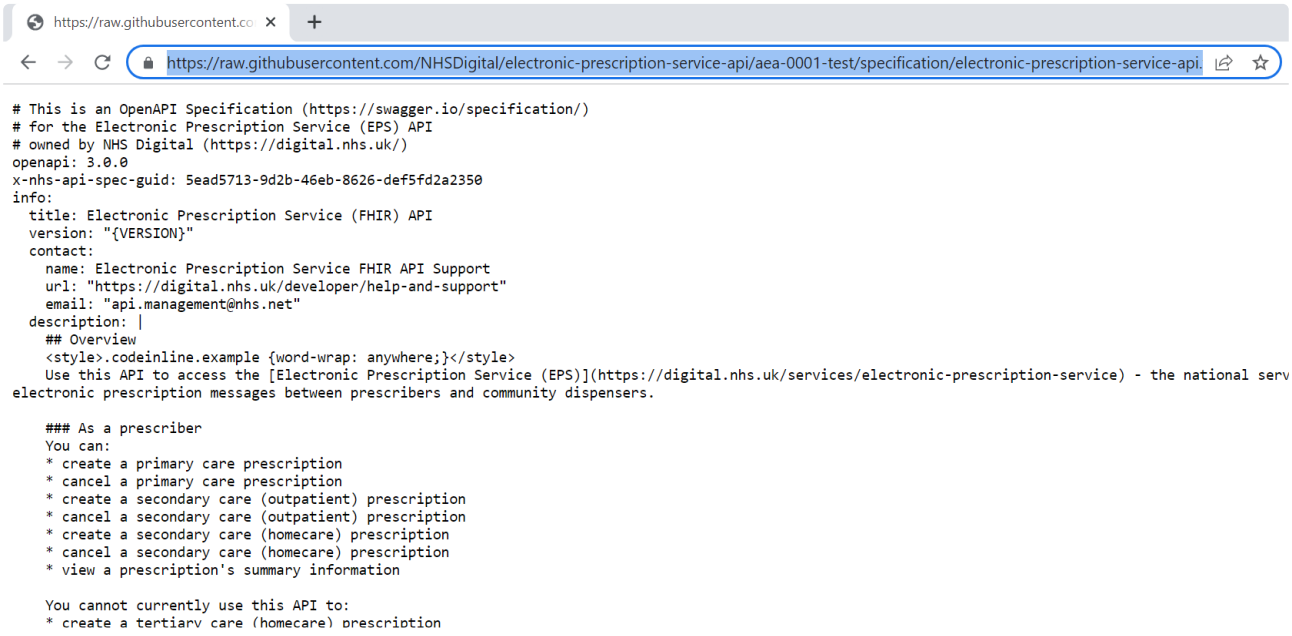
e.g. https://github.com/NHSDigital/electronic-prescription-service-api



1.2 Locate the YAML file you want to edit and click on Raw

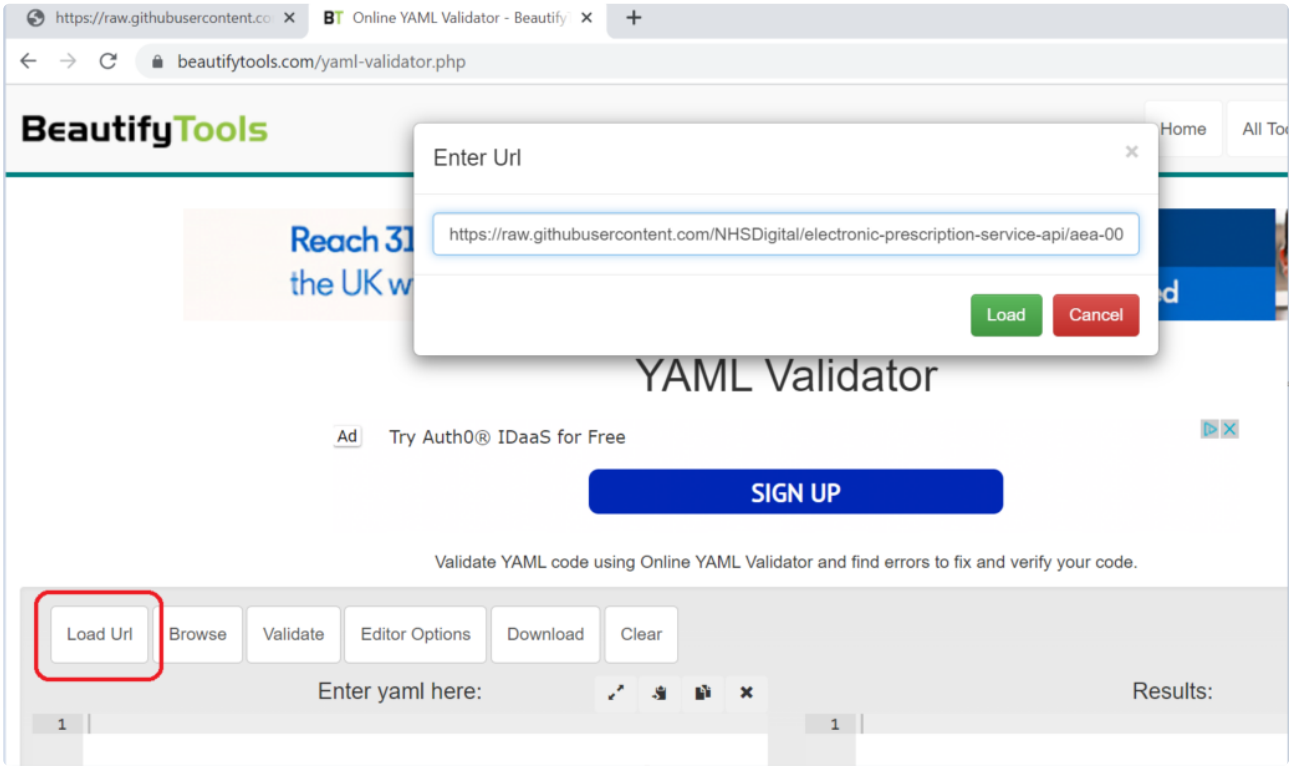


1.3 Copy the raw.git URL for cloning.

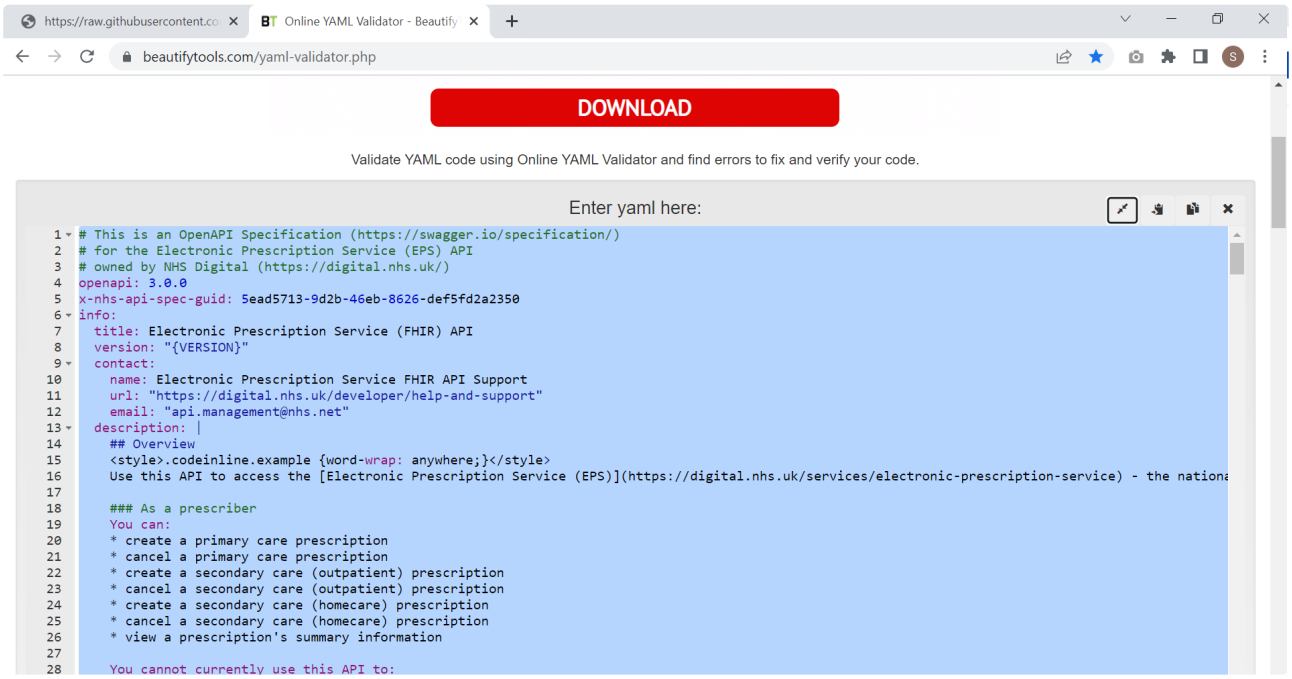


1.4 Load the raw file in Beautifytools YAML Validator

- open https://beautifytools.com/yaml-validator.php in browser
- Click on Load URL
- paste the raw.git URL and click on Load



1.5 Edit the YAML and Validate



1.6 Copy and Paste the edited file into the GitHub branch

edit the source code file

select all and delete

paste the validated YAML file in your branch file and commit changes.

2. Previewing API specification on Bloomreach UAT website[432]

Drawing diagrams in draw.io

Overview

Most API specifications and many other of our introductory content pages benefit from a diagram.

Some people learn better visually rather than by reading text.

However if you have accessibility needs, you must be able to understand the content without using the diagrams. This means making sure that the words on the page convey the same amount of information as shown in the diagrams.

Also for accessibility reasons, we always use Scalable vector Graphics (SVG) format diagrams and descriptive alternative (ALT) text for diagrams.

---

432 https://nhsd-confluence.digital.nhs.uk/display/APM/Previewing+API+specification+on+Bloomreach+UAT+website

Drawing tools

We use draw.io as our drawing tool, which allows you to embed and store the master copies of your diagram in Confluence.

We've built a wide range of "source drawings" for our diagrams, which we store under Developer hub assets (see page 503) - contact us if you need access.

The instructions for creating and editing diagrams are as follows.

Creating a new asset

1. If this is the first image on a page, create a new Confluence page as a child of this page (see page 503) - by copying one of the other existing child pages.
2. Create a new section on the page by typing a suitable heading, such as the title of the new diagram eg NRL (Producer) (Consumer) FHIR - How NRL works with SDS and SSP
3. Create a new draw.io[433] diagram by clicking on the "+" icon and selecting "draw.io[434] Diagram".
4. Choose an existing diagram on one of the other pages that's similar to your diagram.
5. Click the pencil icon to edit it, select all objects with your mouse, then copy (CTRL/C) and paste (CTRL/V) the diagram content into your new diagram.
6. Click "save".

See "Editing an existing asset" below for how to edit and export the diagram

Top tip: to copy an existing diagram, you can also edit then export it to SVG with a new name, then import it into a blank diagram for more editing.

Editing an existing asset

1. Float over the asset and click the pencil icon to edit it (in draw.io[435]).
2. Click "File"→"Export" to export the drawing as an SVG file, and select the transparent background checkbox.
3. Click on the "download" option on the popup to download the image at 100% size to your local PC.
4. Save the drawing.
5. If using a Virtual Private Network (VPN) or a local NHS office connection, go to step 7.
6. If using Virtual Dektop Interface (VDI), upload the drawing from your local PC to your VDI virtual desktop file system using the Open File Transfer Panel (VDI HTML version).
7. Sign in to Bloomreach.
8. Select Images.

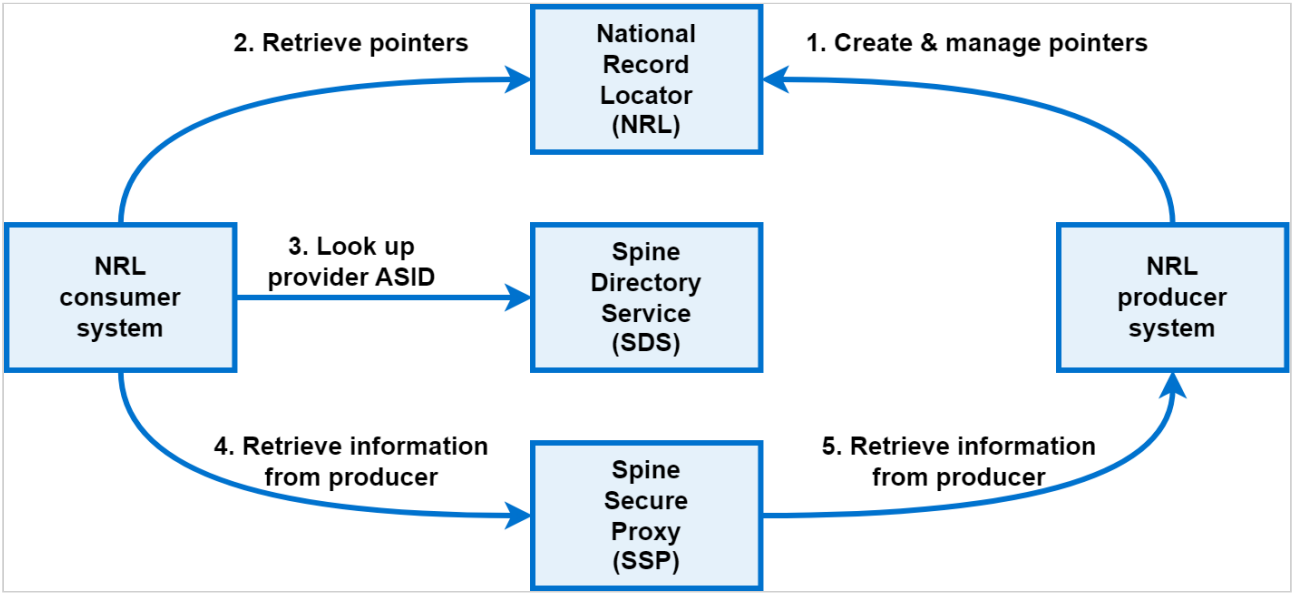
---

433 http://draw.io
434 http://draw.io
435 http://draw.io/

9. Navigate to website/developer/<page name folder> - you need to create a new folder if this is the first image on a page.
10. Upload the image to the folder
11. Edit your document to insert the image. You need to add an "+SVG" section to the document and move it to the right location then add the link to the image and some appropriate ALT text.

Warning - Bloomreach preview might not show any sizing or transparency problems - you have to publish the document to see the finished result.

Example asset



Example ALT text

For the above diagram, the alternative text is:
"How NRL works with Spine Secure Proxy and the Spine Directory Service to retrieve information pointers from producers and give them to consumers."

Documentation for APIs with multiple versions

Overview

If you make a breaking change to your API, and it's out of beta, you will probably want to create a new version of the API.

The new version will need to sit alongside the existing version because you'll have API consumers still using it.

Therefore, you need an API specification for both versions.

Here's how to do it.

Read our guide on API version control

See Version control (see page 380) for a detailed discussion on API version control.

That page talks about whether or not to have a new OAS file for your new version.

At the moment we haven't really explored using the same OAS file for multiple versions. If you want to do that, well, maybe chat to us first.

Otherwise, read on.

Create a new spec

You'll need to add a new OAS file to your API repo, sitting alongside your existing OAS file. The easiest way to do this is to copy your existing one, and then edit it with whatever changes you make.

Bear in mind you might also need to make copies of any supporting files such as schemas or examples.

You'll also need to alter your build pipeline to deploy both versions of the spec.

We can help you do this if you ping us a message on the API producer support Slack channel.

Here's an example of a repo with multiple specs: https://github.com/NHSDigital/patient-care-aggregator-api/tree/master/specification(note that the specs in this repo are for different APIs, not different versions of the same API, but the principle is the same).

Publish the new spec

We recommend having a "version chooser" page as your main page in the API catalogue. Then you have a separate page for each version, linked from the chooser page. Each version can have its own status, so you can show the new version as "in development" until it's ready to use.

For an example of this, see https://digital.nhs.uk/developer/api-catalogue/national-record-locator-fhir.

If you want to publish the page for your new spec but not have it "visible" until you want to make it properly public that's fine, you can publish it but just not create the version chooser page until you're ready. But bear in mind that it's a good idea to make things pugblic sooner so people have more advance warning of what's coming.

You'll need Bloomreach access to set this up. You probably don't have that, so get in touch via producer support and we'll do it for you.

Write a migration guide

If you want API consumers to migrate to the new version of your spec, it's an extremely good idea to write a migration guide and link it from your old spec.

For an example, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-smsp#api-status-and-roadmap, which links to https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-smsp/migrating-from-the-pds-smsp-api-to-the-pds-fhir-api.

7.14.4.21.5.8 Publishing your API documentation in Bloomreach CMS

> **Publishing your API alpha with proxy generator**
>
> If using proxy generator see Building and deploying your API alpha with proxy generator (see page 571)

Overview

The steps required to publish your *API documentation* differ depending on whether you need to publish *API specification* or *supplementary API documents*.

If you *only* have API specification to publish, we can set it up within Bloomreach CMS for you.

If you have any supplementary API documents to publish as well, you create them in Bloomreach CMS yourself, *as well* as set up your API specification there.

See Documenting your API (see page 445) for distinction between the two parts of API documentation.

Prerequisites

Only if you have *supplementary API documents* to publish:

- Account in Bloomreach CMS (production and UAT) with editor privileges.
- Access to NHS network via VDI or HSCN VPN or from an NHS office.

See Get access to tools, systems and environments (see page 144) for how to obtain these.

Deploying documentation to Apigee: Process: Publishing in Bloomreach

Step 1: Write your API specification

Read and follow Writing your API documentation (see page 451).

Step 2: Test / review your API specification within your team on Bloomreach User Acceptance Test (UAT) environment

Once your API specification has published from GitHub to non-prod Apigee, have it set up for preview in Bloomreach CMS UAT as per Previewing API specifications on the Bloomreach training environment (see page 515).

Once it's set up there, any further updates you make to your OAS specification on GitHub automatically updates the corresponding page in UAT.

Test and review it there with your team.

Step 3: Create your supplementary API documents in Bloomreach production

If you have any supplementary API documents to publish:

- Read the documentation available and the hints and tips we put together, see Using Bloomreach CMS[436].
- Create them in production Bloomreach (see page 508), but do not publish them, yet.
- Test and review them there with your team.

Step 4: Engage with us to review your API documentation

When you're happy with your internal review engage with us (see page 2056) to review your documentation - for what we look for see API specification checklist (see page 530).

Step 5: Publish your documentation in Bloomreach production version

Once we've approved your API documentation:

- If you *only* have API specification to publish, ask us (see page 2056) to set it up in production Bloomreach.
- If you have any supplementary API documents to publish as well, however:
    - Create API specification page by following Creating an API specification page in production Bloomreach CMS (see page 526).
    - Publish your supplementary API documents in Bloomreach CMS.

Step 6: Add your API documentation to the API catalogue

If you *only* have API specification to publish, ask us (see page 2056) to add it to the API catalogue for you.

If you have any supplementary API documents to publish as well, however, add your entire API documentation to the API catalogue yourself by following Adding your API specification to the API catalogue in Bloomreach CMS (see page 535).

Using Bloomreach CMS

Overview

Bloomreach is the content management system for the NHS Digital website, which hosts the developer hub pages.

You can log in at https://cms.digital.nhs.uk (not internet-facing - see below) using your Virtual Destop Infrastructure (VDI) or Virtual Private Network (VPN) if working remotely.

Note: API specification pages are produced in OAS as part of defining the API[437] itself and then content is managed in GitHub repositories[438].

---

436 https://nhsd-confluence.digital.nhs.uk/display/APM/Using+Bloomreach+CMS?src=contextnavpagetreemode
437 https://nhsd-confluence.digital.nhs.uk/display/APM/Publishing+API+documentation
438 https://nhsd-confluence.digital.nhs.uk/display/APM/Guide+to+OAS+page+publishing+in+GitHub

(OAS pages are now automatically published in Bloomreach instead of Apigee. Any changes to content still need to be made to the YAML files in the GitHub repos rather than in Bloomreach. The only exceptions are page title, API taglines and taxonomy tags which must be amended in Bloomreach.)

External NHS Digital site:

- Features of the NHS Digital website CMS[439] (Beta)

Internal NHS Digital intranet (Sharepoint library):

- Bloomreach User Guide[440]
- Bloomreach Publications Admin User Guide[441]
- How to write and add SEO summaries[442]
- Bloomreach Key Fact & Infographics User Guide[443]
- How to Link a Supplementary Page to a Publication[444]
- How to Add Visualisation, Change Notices and Survey links to Publication doc type[445]

Internal API Management Confluence page:

- Search engine optimisation (SEO) for API documentation[446] - adding SEO summaries and SEO keys in Bloomreach

External Bloomreach Experience Manager 14 documentation - recently discovered, believed to be the right version:

- https://documentation.bloomreach.com/14/library/architecture/brxm-architecture.html

Accessing Bloomreach

439 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/features-of-the-nhs-digital-website-cms

440 https://hscic365.sharepoint.com/:w:/r/InformationandAnalytics/Documents/Statistical%20Services/Library/Hippo%20User%20Guide/RPS01%20-%20Bloomreach%20Hippo%20CMS%20User%20Guide%20v0.5.docx?d=w21303dea36fa453bbef2aa86b031a308&csf=1&e=ix55jO

441 https://hscic365.sharepoint.com/:w:/r/DIS/Documents/Statistical%20Services/Library/Bloomreach%20CMS%20User%20Guide/RPS02%20-%20Bloomreach%20Administration%20User%20Guide%20v0.1.docx?d=w87459b621e4d4325a797aa1a28fd457f&csf=1&e=dtKAPU

442 https://hscic365.sharepoint.com/:w:/r/InformationandAnalytics/Documents/Statistical%20Services/Library/Hippo%20User%20Guide/RPS03%20-%20How%20to%20write%20and%20add%20SEO%20summaries%20v0.1.docx?d=w5c9a4d38e234480191c56440a2407541&csf=1&e=np9Vdg

443 https://hscic365.sharepoint.com/:w:/r/DIS/Documents/Statistical%20Services/Library/Bloomreach%20CMS%20User%20Guide/RPS04%20-%20Bloomreach%20Key%20Facts%20%26%20Infographics%20User%20Guide%20v0.1.docx?d=wfea8e1a722684911862c313b8956ad4e&csf=1&e=Dazm7G

444 https://hscic365.sharepoint.com/DIS/Pages/Statistical%20Services/Linking-a-Supplementary-Page-to-a-Publication-(June-2020).aspx

445 https://hscic365.sharepoint.com/DIS/Pages/Statistical%20Services/How-to-Add-Visualisation,-Change-Notices-and-Survey-links-to-Publication-doc-type.aspx

446 https://nhsd-confluence.digital.nhs.uk/display/APM/Search+engine+optimisation+%28SEO%29+for+API+documentation

Before using Bloomreach you will need to request access via the web team who will grant the necessary permissions. Send an email to websitedevelopment@nhs.net[447] requesting Bloomreach access, and they will send your login details along with your editing permissions via email.

Note: If you can't publish a page, it means you do not have the necessary publishing permissions - you will need to ask a colleague to publish the changes or request additional publishing rights from the web team by emailing websitedevelopment@nhs.net[448].

Once the web team have created your Bloomreach account, you can use this URL (https://cms.digital.nhs.uk) to access the Bloomreach production environment or this URL (http://cms-training2.nhsd.io[449]) to access the training environment.

This URL[450] is not internet-facing, you need to get network access - there are a few options:

- be in an NHSD office and be on the wired network
- use your Virtual Desktop Infrastructure (VDI)
- use AWS HSCN VPN (see page 168)- this is better than VDI because you can use your browser natively, not via a virtual desktop

Bloomreach page checklist

The following checklist of hints and tips should help you create and edit web pages in Bloomreach.

Initial sections

Title	Not too long (ideally <65 characters) Unique in the table of contents/source page
Summary	Not too long (ideally <3 sentences and <200 characters)
Short summary	Same as summary in most cases
SEO Summary	A longer version of the summary - colour codes tell you when you hit the right length

Sections

Title	Must have a title for each section No questions in headings

447 mailto:websitedevelopment@nhs.net
448 mailto:websitedevelopment@nhs.net
449 http://cms-training2.nhsd.io/
450 https://cms.digital.nhs.uk

Main Heading (Heading 1)	These provide the page Table of Contents on the left-hand side
Sub Heading (Heading 2)	These do not appear in the page Table of Contents
Bullets	When order does not matter Lead in line, with initial lower case, no full stops, no trailing "and" or "or"
Numbered list	When order does matter No lead in, full sentences with initial caps and end with full stops

Tables

Headers	Make sure that the table uses header rows properly. We generally have non-sorting headers, set by Table properties: Advanced tab > Id field, type cannotsort
Spacing	Don't set width or height or column control restrictions on tables - we want them full width (such as it is)

Diagrams

Accessibility	Make sure you include words that convey the same information as your diagram
ALT text	Include ALT text for each diagram or else the NHSD web team will flag it with their accessibility checker

Links

Links	Internal links are links to pages elsewhere on NHS Digital (https://digital.nhs.uk/). These will remain up to date if pages move around. External links are links outside NHS Digital, or links to a part of an NHS Digital page using hash (#). These will break if destination pages move around. Email address links are external links that you need to specify with a "Link Type" of "E-mail" in the pulldown box. (The default link type is URL which wrongly creates an http: link instead of a mailto: link.)
Link Text	It should show either: • text description of the internal destination page • full web address (e.g. www.google.com[451]) Avoid redundant text such as “click here” or “see more”
Link Testing	Clicking it should take you to the correct place in these environments: • Bloomreach editing environment • The “published” NHS environment Issues such as opening within a frame or a second window should work correctly The NHSD web team run a link checker to flag broken links

What if I've edited the wrong page?

If you've already published your wrongly edited page, to put things right, click on Document → Show revision history and you should see the previous good version of the page in the list. Click on the good version and restore it.

If you finshed editing and saved your changes (with Save or Done), you must first take the document offline using Publication → Take offline, before you can see the full page revision history. Click on the good version and restore it.

Bloomreach known issues and fixes

This section includes known issues and fixes for Bloomreach.

API status code - reference guide[452]

As part of ADZ-2181 - API status in reference guide - "lozenges" we updated the reference guide to reflect the same colours as the API catalogue status filters.

451 http://www.google.com/
452 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Status+code+-+reference+guide

If edit this normally in Bloomreach, the colours disappear when you save the page.

The workaround is:

- edit the API status section in html
- click on Done directly, without returning to the rendered preview (non-html) view
- use the code below to cut and paste the HTML
- update/replace the code below with your new code to keep it up-to-date

---

<h3>APIs, message integrations and publish-subscribe events</h3>

We tag our APIs (see page 508)>, message (see page 508) integrations and publish-subscribe (see page 508) events with a status as follows:

In development - early prototyping - the API is available, and might be available for testing via a sandbox service or an integration environment - but we expect to make breaking changes based on developer feedback
Beta - the API is available in production - it might still be subject to breaking changes - and its service (see page 508) level can be anything from bronze to platinum
In production - once out of beta, if we need to make breaking changes, we'll normally publish a new version of the API - we'll only make breaking changes in exceptional circumstances
Internal - the API is not currently available for integration by external third parties - if you still want to use it, <a data-fragment-id="" data-uuid="0d7053ae-f6f9-46e2-ba24-dada6e4e568c" href="http://">contact us
Under review for deprecation - the API is under review and we're considering deprecating it - if you have any concerns, <a data-fragment-id="" data-uuid="0d7053ae-f6f9-46e2-ba24-dada6e4e568c" href="http://">contact us
Deprecated - the API is still available and service levels still apply, but we plan to retire it at some point - we are unlikely to make any updates and new integrations are not permitted - once deprecated, we will consult with developers before deciding on a retirement date
Retired - the API is no longer available for use

<h3>API standards</h3>

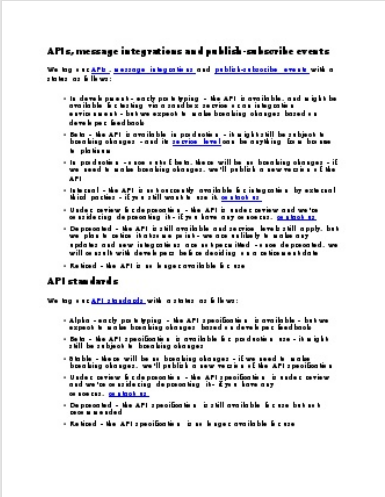
We tag our API (see page 508) standards with a status as follows:

Draft - is still being developed or waiting for assurance or endorsement by qualified bodies
Active - is stable, maintained and has been assured or endorsed for use by qualified bodies
Under review for deprecation - is under review and we're considering deprecating it - if you have any concerns, <a data-fragment-id="" data-uuid="0d7053ae-f6f9-46e2-ba24-dada6e4e568c" href="http://">contact us
Deprecated - is an older version of a standard which is being phased out
Retired - is not being maintained and should not be used

For more details on deprecation and retirement, see our <a data-fragment-id="deprecation-and-retirement-policy" data-uuid="1d843c15-c5b7-4eca-a2b1-765d8c5fa37d" href="http://">deprecation and retirement policy.

We align these API standard statuses with those found in the (see page 508)NHS Data Standards Directory.

Archive - API status code - reference guide[453]



APIs, message integrations and publish-subscribe events

We tag our APIs, message integrations and publish-subscribe events with a status as follows:

- In development - early prototyping - the API is available, and might be available for testing via a sandbox service, but we expect to make breaking changes based on developer feedback
- Beta - the API is available in production - it might still be subject to breaking changes - and its service level can be anything from best effort to platinum
- In production - open or by invitation - there will be no breaking changes - if we need to make breaking changes, we'll publish a new version of the API
- Internal - the API is not currently available for integrations by external third parties - if you still want to use it, contact us
- Under review & deprecating - the API is under review and we're considering deprecating it - if you have any concerns, contact us
- Deprecated - the API is still available and service levels still apply, but we plan to make changes at some unlikely to make any updates and new integrations are not expected - soon deprecated. we will consult with developer feedback deciding on a retirement date
- Retired - the API is no longer available for use

API standards

We tag our API standards with a status as follows:

- Alpha - early prototyping - the API specification is available - but we expect to make breaking changes based on developer feedback
- Beta - the API specification is available for production use - it might still be subject to breaking changes
- Stable - there will be no breaking changes - if we need to make breaking changes, we'll publish a new version of the API specification
- Under review & deprecating - the API specification is under review and we're considering deprecating it - if you have any concerns, contact us
- Deprecated - the API specification is still available for use but not recommended
- Retired - the API specification is no longer available for use

---

Interrupted numbered lists

Sometimes you need to use a numbered procedure. To do this use the "Numbered List" icon in the toolbar.

453 https://nhsd-confluence.digital.nhs.uk/display/APM/API+Status+code+-+reference+guide

However, you sometimes need to add another component (such as a code sample) within the middle of the numbered list, so this breaks the list in two and messes up the numbering.

The fix is to go to the second part of the list, and view the HTML code. This starts with:

```
<ol>
<li> text</li>
<li> text</li>
</ol>
```

This would display as:

1. List
2. List

To make the numbering restart at 4, change `<ol>` to `<ol start="4">`

Broken spec formatting

If the spec formatting is broken (e.g. large gaps, broken tables) and you can't identify a reason within the spec or haven't changed the spec, then republishing may fix it.

Bloomreach does not have a republish button but you can instead edit a page and save it without making any changes. It will then let you publish.

Technical details

When a spec is published and the page loaded it is rendered and then the result is cached. There is an intermittent issue with the spec renderer and each time a page is rendered there is a chance of this happening. So it could occur on publish or it could occur when the cache expires and the page is next loaded.

Previewing API specifications on the Bloomreach training environment

New URL for environment used for previewing API specifications

API specifications have been temporarily moved to an environment called "Training", located here: training2.nhsd.io[454]

Quarterly clear out of Training environment!

On the **first day of every quarter**, starting from 1st June 2023, the non-production Bloomreach environments (UAT, Training, Dev, etc) are being refreshed with production data.
This means your training preview API specification gets deleted and **must** be recreated, by completing the steps below!

454 http://training2.nhsd.io/

A reminder will be sent out in the #platforms-apim-producer-announcements channel on 25th of the preceding month, via Slackbot.
Further information can be found here[455].

Overview

API specifications are published on the NHS Digital website[456], which is powered by Bloomreach Content Management System (CMS). As the specifications are being worked on, they need to be checked for correctness of rendering and content before they can be published.

Bloomreach User Acceptance Test (training) environment is used to preview API specifications. This page describes how to publish API specifications to Bloomreach training and view the rendered page, as it will be seen by your users.

Prerequisites

- Apigee account in organisation *nhsd-nonprod*
- Access to NHS network via VDI or HSCN VPN or NHS office
- Account in Bloomreach training - only needed if you have *supplementary API documents* to publish as well as your *API specification*; see Documenting your API (see page 445) for the distinction

See Get access to tools, systems and environments (see page 144) for how to obtain these.

Convert the YAML specification into a JSON file

1.Ensure the npm dependencies are installed in your project by using the terminal to run "**npm install**". Once the dependencies are installed, refer to step 2.

2.Use the terminal to run "**npm run publish**" to convert the YAML specification into a JSON file. A new "build" folder is created containing the single JSON file.

455 https://nhsd-confluence.digital.nhs.uk/display/CW/Prod+db+restore+to+other+environments
456 https://digital.nhs.uk/developer/api-catalogue

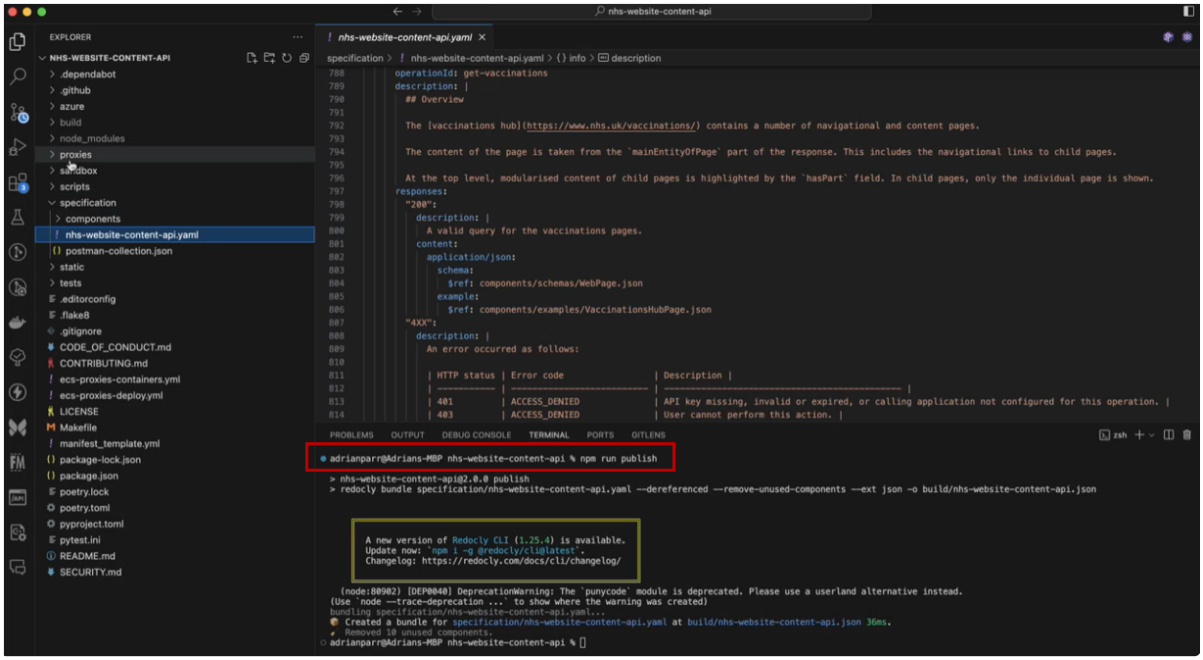


fig 1: Run "npm run publish"

Login to non-prod Apigee

Note: If you don't have access to the non-prod Apigee environment, open the **platforms>APIM>producer>support>** channel >click the **Workflows** button >select **Apigee non prod account**, and complete the form to request access.

If you do have access to non-prod Apigee, perform the following steps:

1.Login to non-prod Apigee[457] and enter your email and password credentials.

457 https://login.apigee.com/login

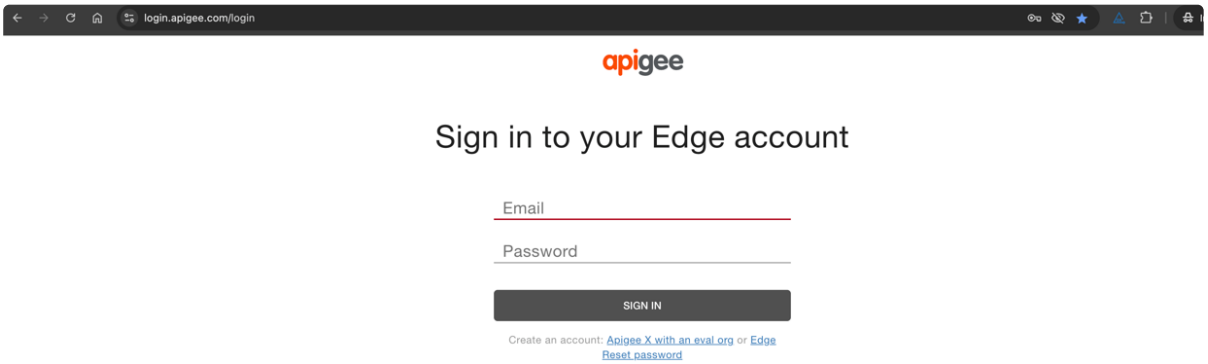


fig 2: Sign in to your Edge account

2.Complete the two factor authentication step. If two factor authentication isn't setup refer to section 4.1.

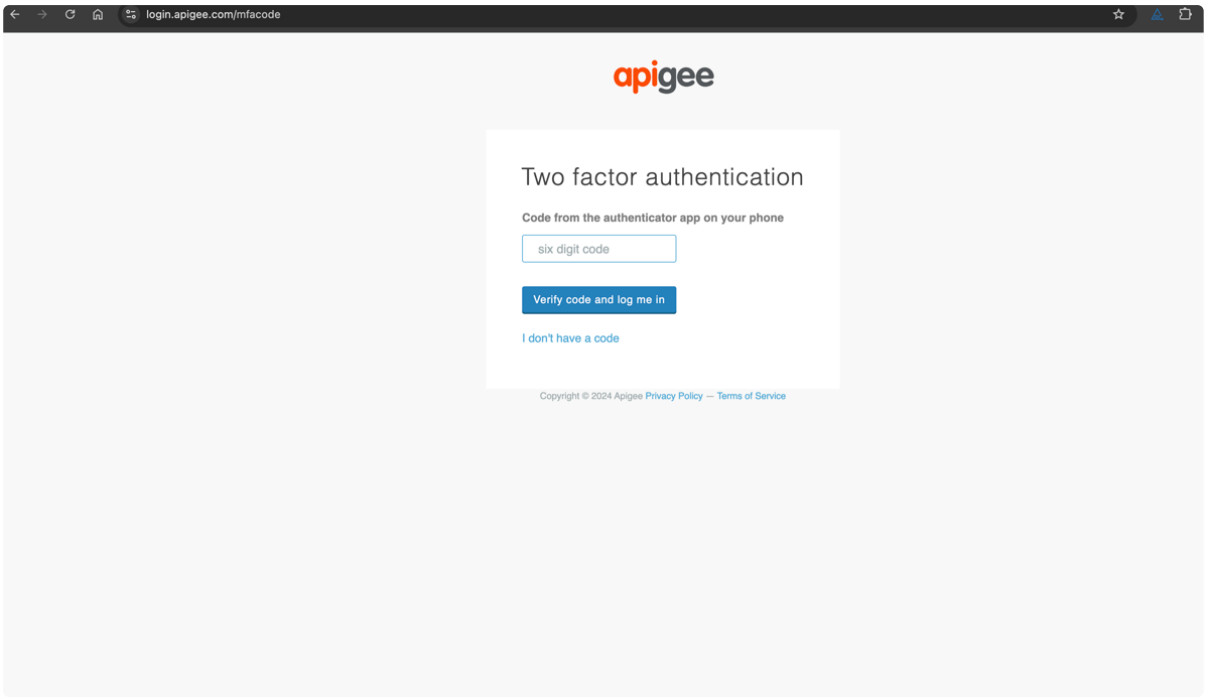


fig 3: Two factor authentication

Setup two factor authentication

If two factor authentication isn't setup, perform the following steps:

1.Click the drop-down menu in the top left-hand corner> select **Edit Account.** Follow the instructions to setup two factor authentication.

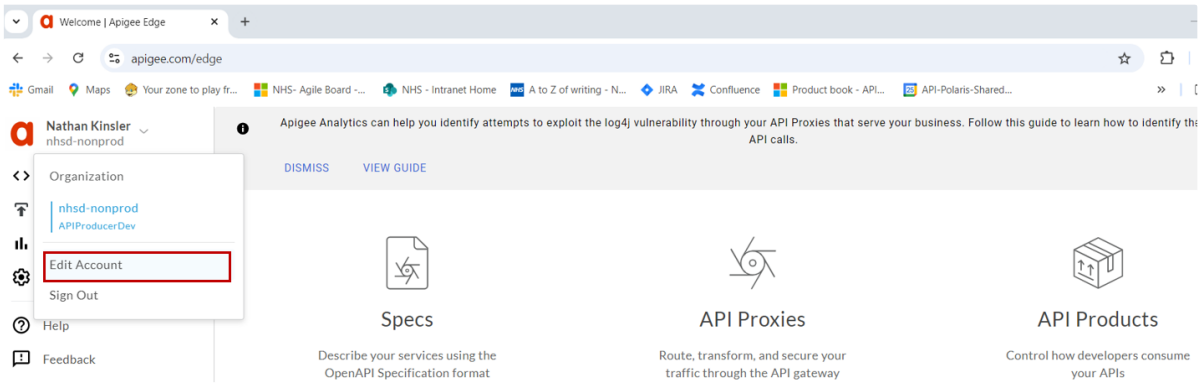


fig 4: Edit Account to setup two factor authentication

2. Once you're logged in, you will see "**nhsd-nonprod**" in the top left hand corner, indicating your are in the correct environment.

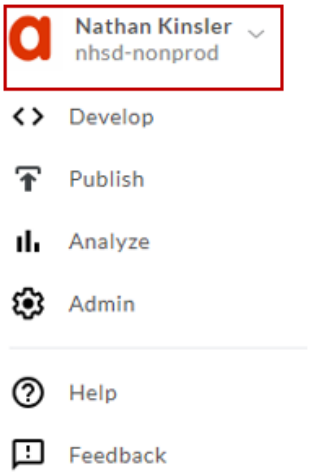


fig 5: NHSD-nonprod environment

Create a named specification in non-prod Apigee

Creating a named spec

1.Log in to non-prod Apigee[458].

2.Select **Develop**>**Specs>+Spec** button>**New Spec**.

458 https://login.apigee.com/login

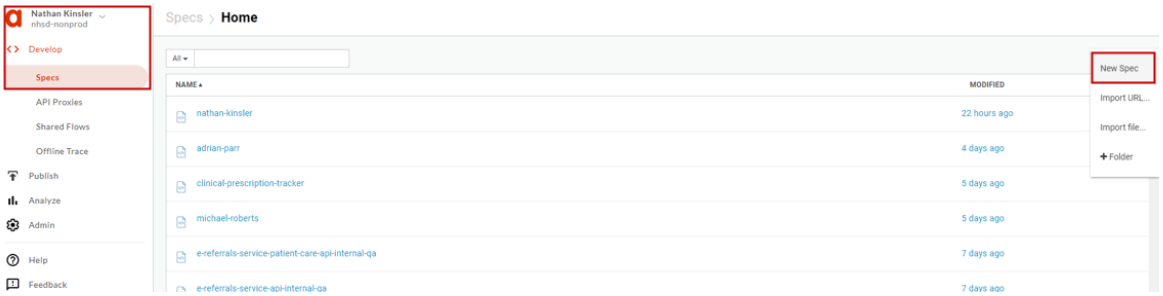


fig 6: Create new spec

3.Delete the YAML data in the newly created spec.

Paste JSON data

1.Paste the JSON data from the file generated in section 3 into the editor. Make sure to keep it as JSON; if a dialogue is displayed asking if you "would like to convert your JSON into YAML" - click cancel.

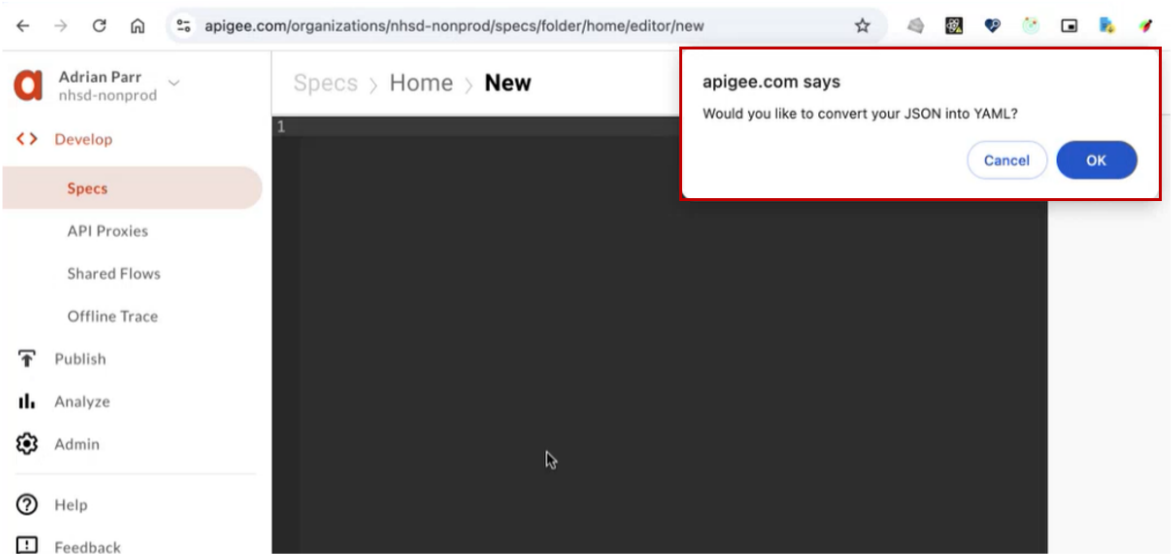


fig 7: JSON to YAML dialogue

2.Click the **Save** button. The "**Name your spec**" form is displayed.

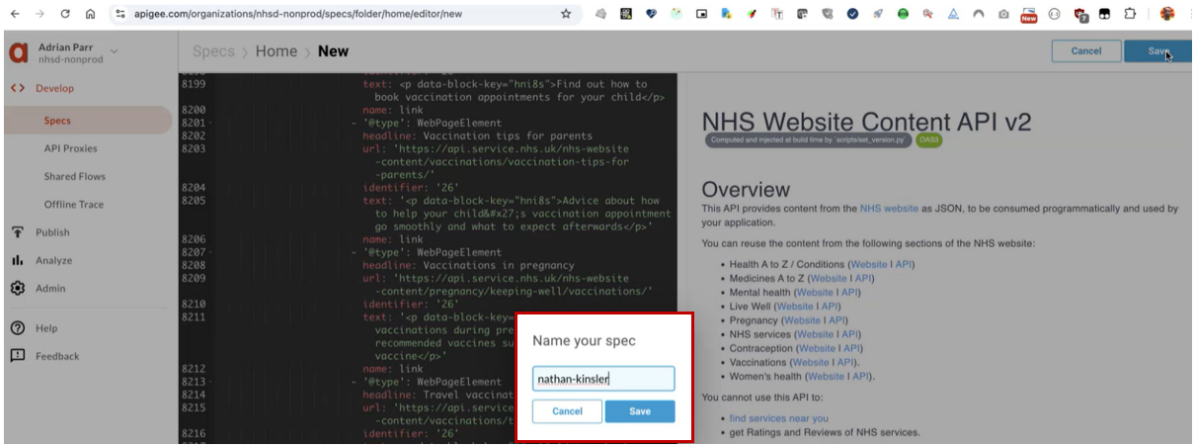


fig 8: Name your spec form

3.Enter the name of your spec and click **Save**.

You **MUST** name your spec using the format '*yourfirstname-yourlastname*', e.g. "joe-bloggs".

A green confirmation banner is displayed confirming the new spec has been created.

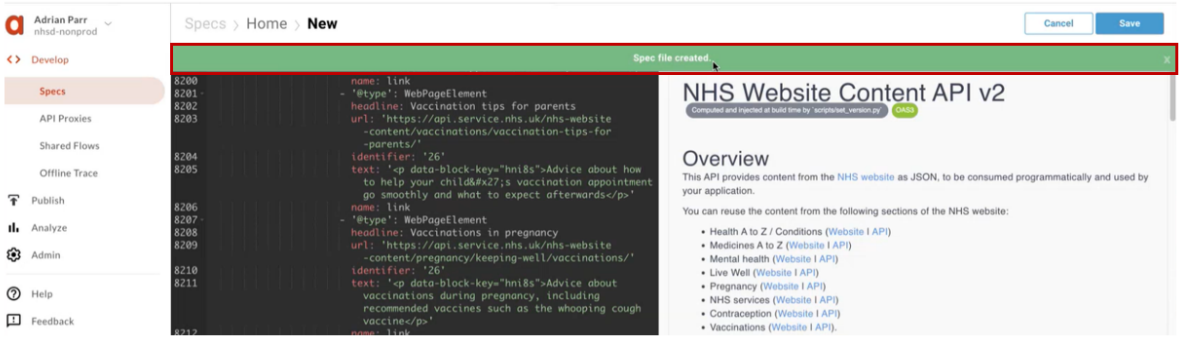


fig 9: "Spec file created" confirmation banner

Get "spec id"

Before you can preview the published page in the test environment, you will need the "spec id".

The Specification id is what links the API specification document in Bloomreach with the corresponding specification in Apigee.

To obtain the "spec id", perform the following steps:

1.Login to non-prod Apigee[459].

2.Select **Develop**>**Specs** from the left-hand navigation.

459 https://login.apigee.com/login

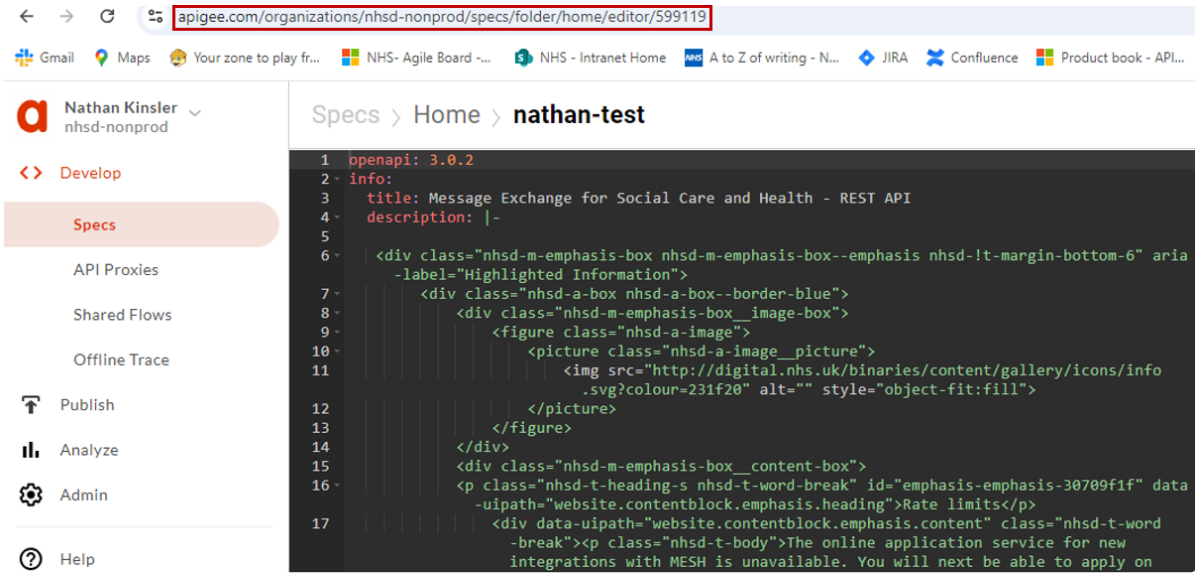


fig 10: Spec ID example - 599119

3.Search for the named spec, for example "joe-bloggs".

4.Click the spec. The "spec id" can be found at the end of the URL, for example, 599119.

5.Copy the "spec id" and follow the steps in section 7.

Create API specification document in Bloomreach Training

Contact platforms-apim-producer-support

If you don't have access to Bloomreach contact the **platforms-apim-producer-support** channel requesting a named spec be created.

Provide the following information in the platforms-apim-producer-support Slack channel[460]:

- the specification id, (as documented in section 6) and your name (matching the name of the specification)

Once this is done, you can view your changes as documented in section 8.

If you already have a Bloomreach training account, create a named API specification document yourself following the steps in section 7.2.

Create new API specification document

1. Log into CMS at: cms-training2.nhsd.io[461].

2.Under *Content > Corporate Website > Developer > API and integration catalogue > Named.*

460 https://nhsdigital-platforms.slack.com/archives/G016JRWN6AY
461 http://cms-training2.nhsd.io/

3. Hover over the *Named* catalogue, click the 'three dots' menu, choose *Add new folder*....in the **new folder** dialogue, enter the following information:

- set *Name* to the first and last name of the requester using format '*firstname-lastname*',
- click the ***OK*** *button*.

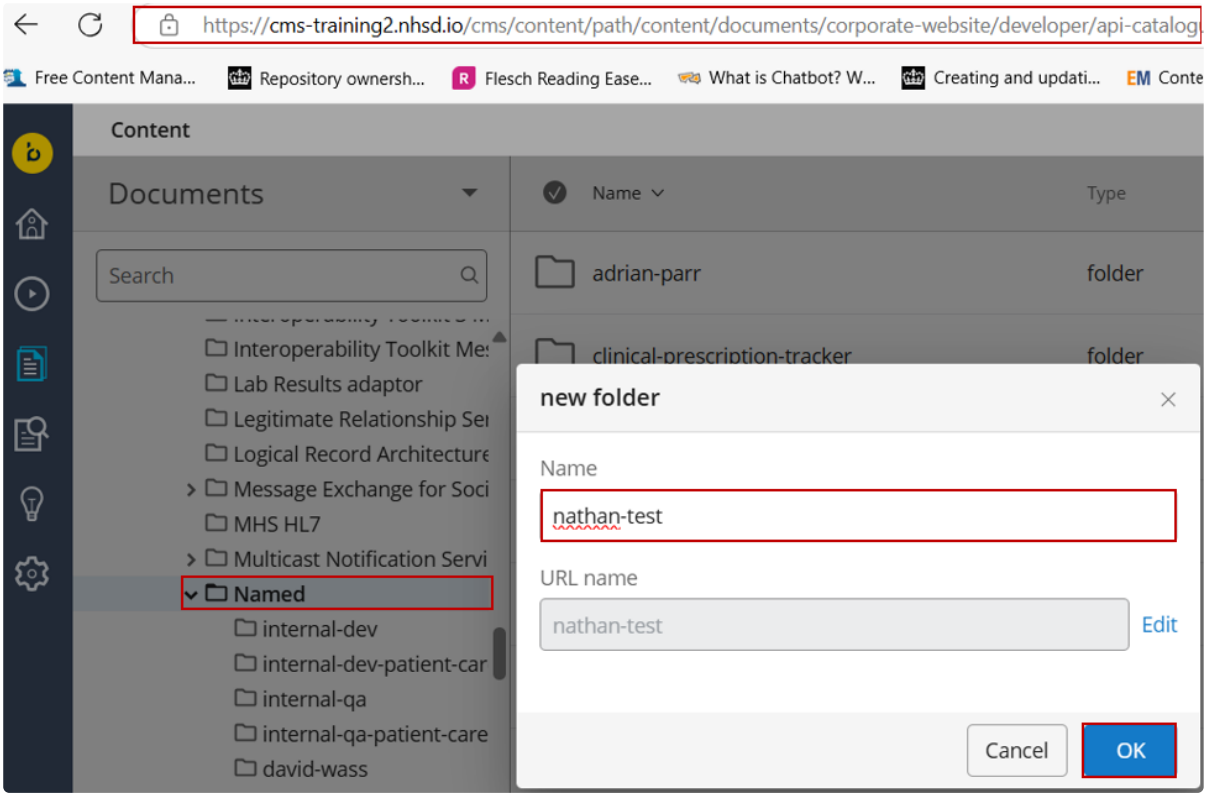


fig 11: Bloomreach training environment - Add new folder form

4.Click the newly created folder, hover over it, click the 'three dots' menu, choose ***Add new document***..., in the **new document** dialogue, enter the following information:

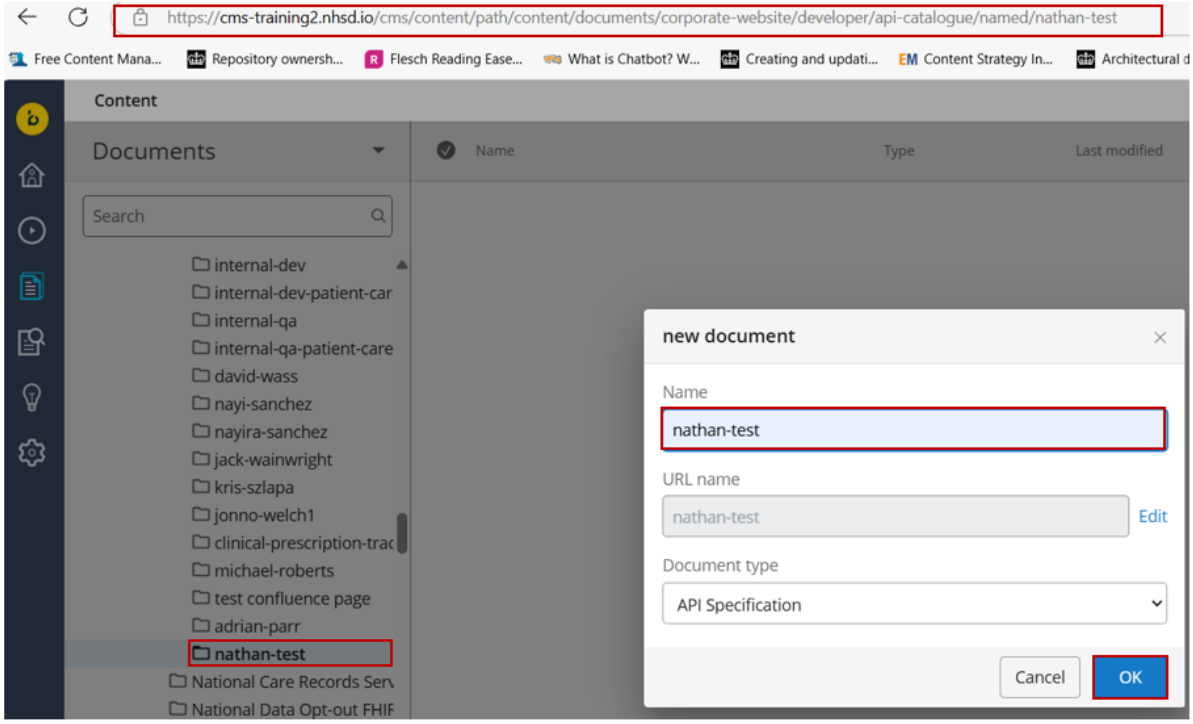


fig 12: Bloomreach training environment - Add new document form

- set *Name* to the first and last name of the requester
- populate field *URL name* with word *content*
- from the drop-down menu choose **Document type *API Specification***
- click the ***OK*** *button*

The API specification is now visible in the Bloomreach interface. Open the API specification and complete step 5, as follows:

5. Populate the required fields (marked by '*') in the spec with:

- The name of the API: the full name of the requester, with first characters capitalised.
- Specification Id: the specification id number received from the requester (or found by you in Apigee by requester's name).
- Summary: text 'API specification summary'.
- Short Summary: text 'API specification short summary'.
- SEO Summary: text 'API specification'.

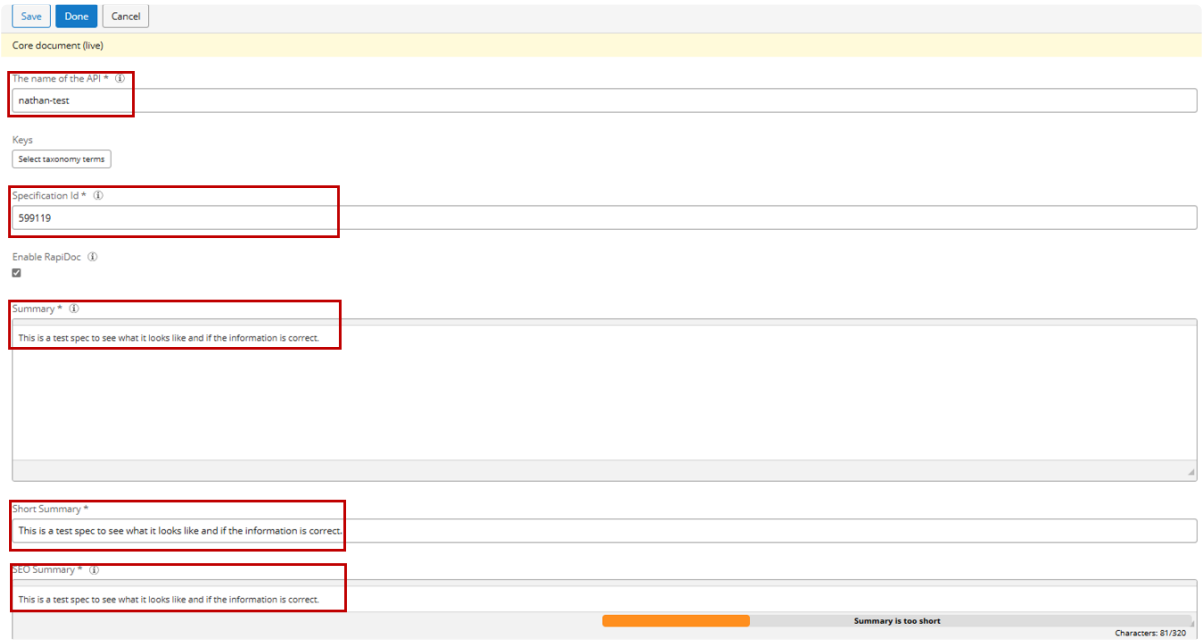


fig 13: Bloomreach mandatory fields

6. Click the ***Done*** *button.*

> **Do not** publish the spec as our automated batch process will handle the publishing. The spec page will change from red to green once it has published.

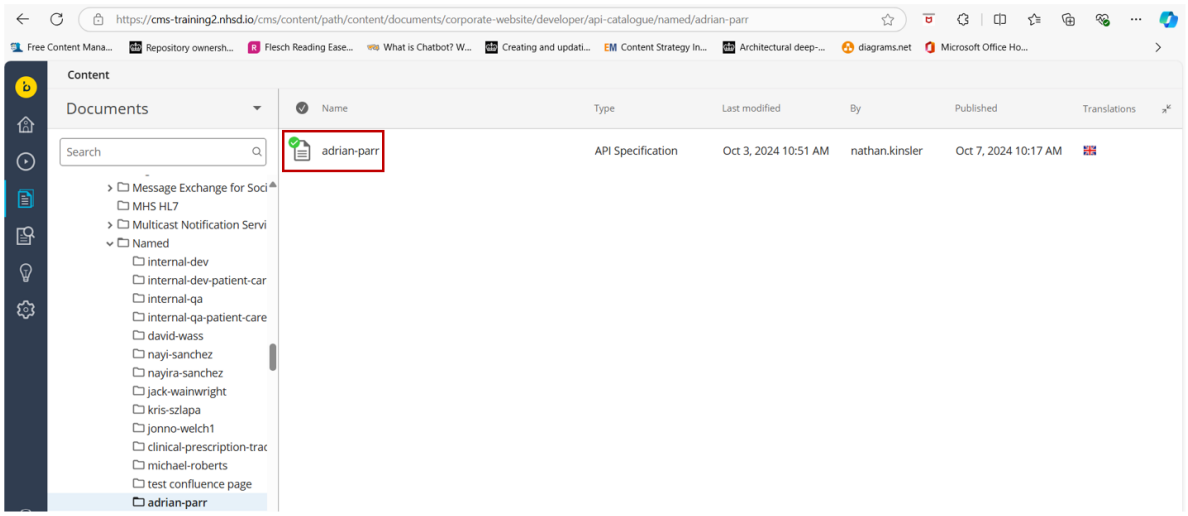


fig 14: Sample published spec in Bloomreach training environment

View the changes in Bloomreach training environment

To view the changes:

1. Log into VPN (see page 168) or VDI[462] if you're not connecting from one of NHS England offices.

2. In your web browser navigate to the URL of your specification. Use the following URL, substituting *yourfirstname-yourlastname* with your real names: http://training2.nhsd.io/developer/api-catalogue/named/yourfirstname-yourlastname

Note: After you've created your specification it can take up to an hour for the changes to be published from Apigee to Bloomreach. If your changes haven't been published to Bloomreach, contact us (see page 2056).

Preview additional changes

Complete section 3 (Convert the specification YAML file into a JSON file) and section 5.2 (Paste JSON data) to continue previewing your updates in the Bloomreach test environment.

Things to look out for

- If your specification content fails to publish to training, check there are no error messages displayed when viewing the spec in Apigee.
- When editing documents in Bloomreach CMS, you may see the UI becoming unresponsive or encounter an error message similar to '*Something went wrong content not found while creating a new document.*'. In these cases try reloading the screen, logging out of the CMS and logging back in, and waiting a few minutes.

Creating an API specification page in production Bloomreach CMS

Overview

When your API specification is to be published to external developers for the first time, an API specification page first has to be manually created for it in Bloomreach CMS. This page explains how.

If you *only* have API specification to publish (no supplementary pages), ask us to set it up within Bloomreach CMS for you and ignore the rest of this page.

If you have any supplementary pages to publish as well, however, you create them in Bloomreach CMS yourself, *as well* as set up your API specification there. In this case, follow the steps below.

See Documenting your API (see page 445) for distinction between the two parts of API documentation.

Prerequisites

- Account in production Bloomreach with editor privileges.
- Access to NHS network via VDI or HSCN VPN or NHS office.

See Get access to tools, systems and environments (see page 144) for how to obtain these.

---

462 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=11845016

Steps

Step 1: Deploying the specification to Apigee or Proxy Generator

If your API is "old style" (not using Proxy Generator), you deploy your API spec to Apigee. If it is "new style" (using Proxy Generator), you deploy your API spec to Proxy Generator

Deploying to Apigee

Once you have written your spec (see Writing your API documentation (see page 451)) and have merged the changes into the main branch in GitHub the spec is deployed to Apigee via the deployment pipelines.

The production version of your spec is deployed to the `int` environment, therefore the spec must be successfully released to `int` for your spec to appear in Apigee.

If this is your first time deploying the spec or your changes aren't appearing in Bloomreach the first place to check is your release pipeline.

> NAVIGATE TO AZURE[463] and check your release pipeline has successfully deployed to `int`.

If the int release was successful but your changes still haven't appeared in Bloomreach then reach out to the Platform team in #platforms-apim-producer-support[464].

Deploying to Proxy Generator

See Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571).

Step 2: Get specification ID

If using Apigee

Ask on Slack channel #platforms-apim-producer-support[465] that someone with access to production Apigee finds it for you.

Point them to the instruction How to obtain specification id from production Apigee (see page 529) on this page.

If using Proxy Generator

See Publishing API specs using the Proxygen Generator API or Proxygen-cli (see page 571).

---

463 https://dev.azure.com/NHSD-APIM/API%20Platform/_build?view=folders
464 https://nhsdigital-platforms.slack.com/archives/G016JRWN6AY
465 https://nhsdigital-platforms.slack.com/archives/G016JRWN6AY

Step 3: Get title and summary

Title is typically the title embedded in the specification's source file and follows standard pattern e.g. "`Personal Demographics Specification - FHIR API`".

Short description is a single phrase summarising the key purpose of the API.

Step 4: Log in to production Bloomreach

Connect to NHS network. If you're in one of NHSD offices you're already connected. Otherwise connect through VPN or VDI[466].

Navigate to https://cms.digital.nhs.uk and log in.

Step 5: Create API specification document

Create a new folder for the API spec:

- navigate to *Corporate website > Developer > API and integration catalogue*,
- in the folder tree on the left, hover mouse cursor over *API catalogue* and click the 'three dots menu', choose *Add new folder...*,
- give the new folder a name that follows the pattern set by other existing APIs, e.g. "`Personal Demographics Specification FHIR`", and confirm.

Create a new document in the new folder:

- open your newly created folder,
- in the folder tree on the left, hover mouse cursor over your newly created folder and click the 'three dots menu', choose *Add new document...*,
- Set *Document name* as per standard pattern e.g. "`Personal Demographics Specification - FHIR API`".
- Set *URL name* to "`content`" which causes it to appear on the URL for the folder as opposed to being a sub-page in the folder.
- Set *Document type* to "`API Specification`".

Step 6: Populate API specification with required details

In the specification's edit screen:

- Set *The name of the API* as per standard pattern e.g. "`Personal Demographics Specification - FHIR API`".
- Set *Specification Id* to the specification ID from step 2.
- Set *Summary* to the value obtained earlier. Its text is then displayed as the sub-title in the title bar on the page. Follow the pattern e.g. "`Access the Personal Demographics Service (PDS) -`

466 https://www.vdi.hscic.gov.uk/

`the national electronic database of NHS patient details such as name, address, date of birth and NHS Number.`"

- Set *Short Summary* to the same as *Summary*. Its text is then displayed in API catalogue[467] right under the title of your specification.
- Set *SEO Summary* to the same as *Summary* then add additional text chosen from your Overview until the acceptable length indicator turns green.
- Set the *Keys* to include all the filter tags that are relevant for your API, as explained on Adding your API under the correct API catalogue filters (see page 537)

Step 7: Save API specification document

Save and close the document - **but do not publish it**.

Now...wait!

Our spec publishing/updating batch process runs once an hour at 5 minutes to the hour on working days between 07:55 and 17:55. It will spot the new page and generate the HTML spec content. It will then automatically publishes the page.

If you've already published the page, our batch process won't do anything because it thinks the page has already been updated - because the page last published date is later than the spec last modification date. If this happens you need to ask someone with prod Apigee access to update your API specification in Apigee to force the update to occur - any small change to the content will do, for example adding extra space in the description text.

How to

Obtain specification id from production Apigee

Sign into to Prod Apigee Edge, navigating to the list of specs[468] (organisation "`nhsd-prod`"; displayed under your name in the top-left corner).

This should take you to the list of specifications but if it didn't, navigate to the same URL again or simply click option *Develop* > *Specs* in the top-left corner.

Find the integration test instance of the spec (suffix "`-int`") and click it to open it for editing.

Specification id is the number at the very end of the URL e.g. for https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/specs/folder/265511/editor/320397 it would be "`320397`".

7.14.4.21.5.9 Reviewing your API documentation with us

Overview

---

467 https://digital.nhs.uk/developer/api-catalogue
468 https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/specs/folder/home

Before you publish your API documentation, you need to review it with us first.

This is to ensure quality and consistency.

Process

Step 1: Complete the API specification checklist

This is effectively a self-service review and covers the most common points.

To do this, follow the instructions on API specification checklist (see page 530).

Step 2: Engage with us for a review

You should already have a member of our tech author team allocated to your API as a key point of contact - see API register (see page 1204).

Get in touch with your tech author contact, send them a link to your checklist and ask for a review.

Depending on how they are feeling, this might be done off-line, or they might ask for a "little chat" 🙂.

API specification checklist

> **Instructions for using this template**
>
> Make a copy of this page, change the name to "API specification checklist - <your API name>, and save it in your API's confluence space.

Overview

This page provides an audit trail / checklist for how the API specification for the above API has been developed in line with the NHS Digital API delivery process[469], including links to evidence where appropriate.

The checklist helps us ensure consistency and quality across all our APIs.

Status codes

TO DO

IN PROGRESS

DONE

N/A

469 https://nhsd-confluence.digital.nhs.uk/display/APM/API+delivery+process

Checklist

Activity	Status	Comments
Have a read of the PDS FHIR API specification[470] to see an example of an "exemplar" API specification. It's not perfect, but it should give you a good idea of what we're looking for, in terms of content and writing style.	TO DO	TBC
Write your API specification (see page 451).	TO DO	TBC
Name your API to match our naming pattern, including the technology it uses. See New API onboarding questionnaire (see page 530). Example: "Personal Demographics Service - FHIR API"	TO DO	TBC
Use the standard section headings for FHIR APIs - see What goes in your API specification (see page 453). If you need to add additional subheadings, ideally make them subsidiaries of the standard ones: Overview Who can use this API Related APIs API status and roadmap Service level Technology Network access Security and authorisation Errors Open source Environments and testing Onboarding Optionally, you can include a contact email address for your API, at the end of your Onboarding section. Tell us what it is so we can forward any queries from api.management@nhs.net[471]. If you have headings that don't fit, talk to us about it, we're happy to make exceptions.	TO DO	TBC

470 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
471 mailto:api.management@nhs.net

Activity	Status	Comments
Use the same standard links as used in other API specs. These are links to other documentation on the developer hub such as API status[472], API technologies[473], Network access for APIs[474], security and authorisation[475]patterns, testing environments[476], onboarding processes[477] and so on. Examples: "This access mode is in [478]beta[479]", "This API is a FHIR[480] API.", "To onboard for this API, follow the Supplier Conformance Assessment List (SCAL) process.[481]"	TO DO	TBC
If you need to explain some aspect of your API in more detail, you can choose to add additional pages in Bloomreach CMS to support your API spec - see Additional pages[482] for details. Example: PDS FHIR API has an additional page for PDS FHIR API test data[483] which would otherwise make its "Environments and testing" section very long.	TO DO	TBC
Follow the NHS and GDS writing style standards. These are summarised in the API content style guide[484]. The biggest problems we see in new API specs are with over-complicated writing style, especially where there is no technical author on the API producer team or the spec writer does not have time to dig in to the GDS and NHS writing style guidelines. If you are short of time and need a simplified writing style guide, see "Writing style - how to avoid the most common problems", below.	TO DO	TBC

472 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#api-status
473 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital
474 https://digital.nhs.uk/developer/guides-and-documentation/network-access-for-apis
475 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation
476 https://digital.nhs.uk/developer/guides-and-documentation/testing
477 https://digital.nhs.uk/developer/guides-and-documentation/onboarding-process
478 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital#fhir
479 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#api-status
480 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital#fhir
481 https://digital.nhs.uk/developer/guides-and-documentation/onboarding-process#onboard-using-the-supplier-conformance-assessment-list-scal-process
482 https://nhsd-confluence.digital.nhs.uk/display/APM/What+goes+in+your+API+specification#WhatgoesinyourAPIspecification-Additionalpages
483 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir/pds-fhir-api-test-data
484 https://nhsd-confluence.digital.nhs.uk/display/APM/API+content+style+guide?src=contextnavpagetreemode

Activity	Status	Comments
Ask us to add your API spec to our API catalogue (see page 535) after you publish the alpha version and we review it. You need to request which of our API catalogue filters and tags apply to your API. See Making your API appear under the correct API catalogue filters[485].	TO DO	TBC

Writing style - how to avoid the most common problems

Do this	Why it's a problem	Status	Comments
Use active verbs[486] to clarify who does something, and what they do it to.	Passive verbs[487] are hard to understand as they can conceal who does an action. Good example - active - "**you can search** for patients" Bad example - passive - "patients **can be searched for**"	TO DO	TBC
Use commands or imperative verbs to tell the reader what to do. (This is a technical writing guideline, not an NHS or GDS one.)	Using commands gets rid of wordiness and confusion about who or what initiates an action, especially in procedural steps or instructions. Examples: "**Use** this API to..." "In step 11: **submit** your completed SCAL to..." "For more details, **see** FHIR[488]."	TO DO	TBC

485 https://nhsd-confluence.digital.nhs.uk/display/APM/Making+your+API+appear+under+the+correct+API+catalogue+filters?src=contextnavpagetreemode
486 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style#active,-passive
487 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style#active,-passive
488 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital#fhir

Do this	Why it's a problem	Status	Comments
Use present tense to explain what happens. (This is a technical writing guideline, not an NHS or GDS one.)	Users read content to help them perform tasks or to gather information. These activities occur in the user's present time, so the present tense is appropriate in most content. Additionally, sentences that use the present tense are easier to read than sentences that use past or future tense. Examples: Good example - present - "This only **gets** populated on a retrieval and not a search." Bad example - future/passive - "This **will** only **be populated** on a retrieval and not a search." There are exceptions when things must be in the past or future, such as explaining, "Once our roadmap is complete, the above APIs will become redundant."	TO DO	TBC
Use "you" (meaning the reader, ie the developer) or "we/us" for NHS Digital. GDS guidelines say, "Address the user as 'you' where possible. " and also that, "Using 'we' is fine, as long as you're making it clear as much as possible who the 'we' is."	Users are more engaged with content which talks to them directly. You talk to users directly by using second person, addressing the user as you. Second person also promotes a friendly tone. Examples: "For details of sandbox test scenarios, or to try out the sandbox using **our** 'Try this API' feature, **see** the documentation for each endpoint." "Alternatively, **you can** try out the sandbox using **our** Postman collection." If you need to clarify exactly who you mean, then the first time say, for example, "**As a developer, you** might find the following resource useful" or "**at NHS Digital, we** produce this data."	TO DO	TBC

Do this	Why it's a problem	Status	Comments
Use positive contractions. Avoid negative or conditional contractions.	Positive contractions are currently viewed as a good thing which help make written language simpler and more natural, such as "you're", "we're", "it's", "there's", "here's" and so on. GDS user research shows that readers with lower English literacy levels (eg people with English as a second language) can mis-read negative contractions[489] such as "can't" or "don't" or "shouldn't" as the opposite, meaning "can" or "do" or "should" so you must avoid using these. The same also applies to avoiding conditional contractions[490] such as "should've", "would've", or "could've" according to GDS readability guidelines. Good example - "**Here's** an example:" Bad example - "The asynchronous pattern is used for interactions which **don't** require an immediate response" - should be "**do not**"	TO DO	TBC

7.14.4.21.5.10 Adding your API specification to the API catalogue in Bloomreach CMS

Overview

Once you've published your API specification to the NHS Digital with Bloomreach CMS, there are a few things you need to do before you can add it to the API catalogue

Process

Step 1: Add your API under the correct API catalogue filters

Add the right keys to your API specification in Bloomreach, see Adding your API under the correct API catalogue filters[491]

489 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style#contractions
490 https://digital.nhs.uk/about-nhs-digital/corporate-information-and-documents/nhs-digital-style-guidelines/how-we-talk/a-to-z-of-house-style#contractions
491 https://nhsd-confluence.digital.nhs.uk/display/APM/Adding+your+API+under+the+correct+API+catalogue+filters?src=contextnavpagetreemode

Step 2: Add Search Engine Optimisation (SEO) to your API specification

Add an SEO summary to your API specification in Bloomreach, see Using Bloomreach CMS for Search Engine Optimisation (SEO) of API specification (see page 548)

Step 3: Add your API specification to the API catalogue

Edit the API catalogue in Bloomreach to add your API specification, see Adding an API specification to the API catalogue in Bloomreach CMS[492]

Adding an API specification to the API catalogue in Bloomreach

APIs should be listed in the API and integration catalogue[493] which contains a list of all the APIs currently available, or in development, from NHS England. You might have to update the API catalogue if you introduce a new API.

Prerequisites

To add or remove API specs to the API and integration catalogue[494], you will need:

- Bloomreach access[495] (contact the Web team[496] if you do not have access to Bloomreach).
- the correct editing permissions - the Web team[497] can help with the necessary editing permissions.

Note: The procedures below are only applicable for the following document types in Bloomreach.

- API Specification
- General
- Service

Add an API spec to the API and integration catalogue

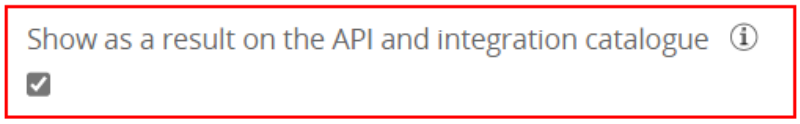
To add an API spec to the API and integration catalogue[498] perform the following steps:

1. Open Bloomreach[499].
2. Navigate to the page you want to add to the catalogue.
   a. Double click the page.
   b. Click the **Edit** button.

---

492 https://nhsd-confluence.digital.nhs.uk/display/APM/Adding+an+API+specification+to+the+API+catalogue+in+Bloomreach+CMS?src=contextnavpagetreemode
493 https://digital.nhs.uk/developer/api-catalogue
494 https://digital.nhs.uk/developer/api-catalogue
495 https://cms.digital.nhs.uk/cms/?0&initialPath=/content/path/content/documents/corporate-website/developer/api-catalogue/multicast-notification-service
496 mailto:england.websitedevelopment@nhs.net
497 mailto:england.websitedevelopment@nhs.net
498 https://digital.nhs.uk/developer/api-catalogue
499 https://cms.digital.nhs.uk/cms/content/path/content/documents/corporate-website/developer/api-catalogue/multicast-notification-service

c. Scroll to the bottom of the page and select the "**Show as a result on the API and integration catalogue**" checkbox.



d. Click the **Done** button.

e. Select **Publication**>**Publish.**

Your API spec will now appear in the API and integration catalogue[500].

Remove an API spec from the API and integration catalogue

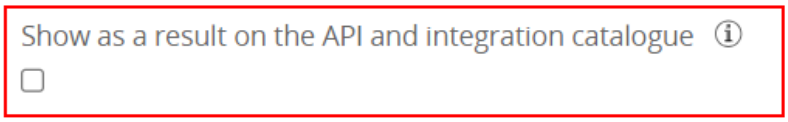
To **remove** an API spec from the API and integration catalogue[501], perform the following steps:

1. Open Bloomreach[502].
2. Navigate to the page you want to remove from the catalogue.

a. Double click the page.

b. Click the **Edit** button.

c. Scroll to the bottom of the page and deselect the "**Show as a result on the API and integration catalogue**" checkbox.



d. Click the **Done** button.

e. Select **Publication**>**Publish.**

Your API spec will no longer be displayed in the API and integration catalogue[503].

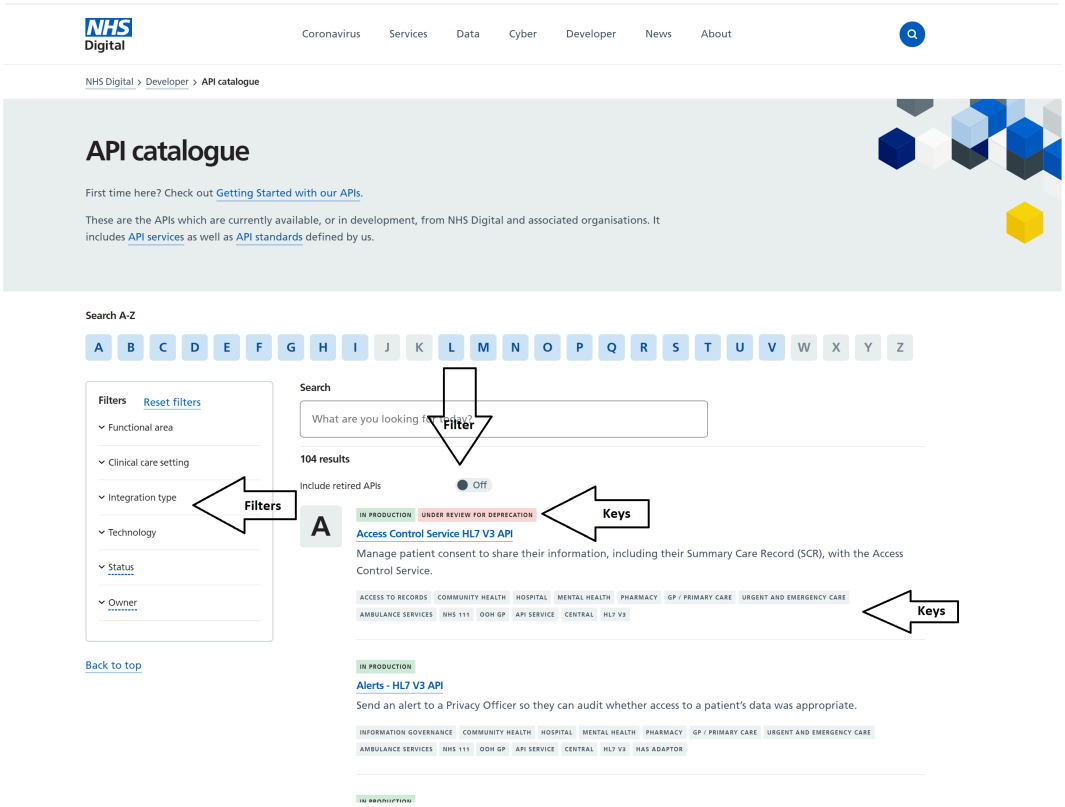
Adding your API under the correct API catalogue filters

Overview

This page explains how to make your API specification appear on the API catalogue page under the correct filters with the right keys.

To do this, add the right Bloomreach CMS taxonomy keys to your API specification.

500 https://digital.nhs.uk/developer/api-catalogue
501 https://digital.nhs.uk/developer/api-catalogue
502 https://cms.digital.nhs.uk/cms/content/path/content/documents/corporate-website/developer/api-catalogue/multicast-notification-service
503 https://digital.nhs.uk/developer/api-catalogue



Context

The API catalogue uses 6 groups of filters to help software development team members narrow down and select the APIs that they're interested in.

We apply taxonomy keys to API specifications to create these filters under 6 major categories:

- Functional area (formerly Business use)
- Clinical care setting (formerly Care setting, formerly Services)
- Integration type (formerly API service or standard)
- Technology
- Status (formerly API status)
- Owner (new)

Our overall approach is to use a subset of the existing Bloomreach CMS taxonomy, add some missing terms and rename others, to produce a suitable keyword mapping.

We chose this terminology with the help of a panel of stakeholders drawn from across NHS Digital, who reviewed, iterated and finally agreed on it.

As an API producer, you need to select which taxonomy keys apply to your API specification, and possibly request the addition of any new ones that are missing. If you need help deciding on your taxonomy, contact us via API producer slack support (see page 2056).

To apply taxonomy keys to a document in Bloomreach, see "Internal SEO" under Using Bloomreach CMS for Search Engine Optimisation (SEO) of API specification (see page 548).

The filters

Functional area

This is the broad clinical or business purpose which your API supports, such as prescribing, demographics or security.

The full list of keys, together with its location in the Bloomreach CMS taxonomy hierarchy, is as follows:

Key	Location
Access to records	Technology and Architecture / Capability / Care Capability / Access to Records
Clinical decision support	API Catalogue / Clinical Decision Support
Continuity of care	Technology and Architecture / Capability / Care Capability / Continuity Of Care
Demographics	Data and Information/People and Places / Demographic
Key care information	Technology and Architecture / Capability / Care Capability / Access to records / Key Care Information
(Medication management)	Technology and Architecture / Capability / Care Capability / Medication Management
Prescribing	Data and Information / Health and Social Care / Pharmacy / Prescribing
Dispensing	Data and Information / Health and Social Care / Pharmacy / Dispensing
Vaccination	API Catalogue / Vaccination
Messaging	Technology and Architecture / Standards / Messaging
Patient communication	API Catalogue / Patient Communication
Referrals and bookings	API Catalogue / Referrals and Bookings
Reference data	Content Type / Document / Data / Reference Data

Key	Location
Information governance	Service Types / Infrastructure / Information Governance
Security	API Catalogue / Security
Tests and diagnostics	API Catalogue / Tests and Diagnostics

Care setting

This is the type of setting or care provider organisation the API supports, such as GP/primary care, NHS111 or hospital.

The full list of keys, together with its location in the Bloomreach CMS taxonomy hierarchy, is as follows:

Key	Location
Community health	Data and Information/Health and Social Care/National Health Service/Community Health
Dentistry	Data and Information/Health and Social Care/National Health Service/Primary Care/Dental Practice
Eye care	Data and Information/Health and Social Care/National Health Service/Primary Care/Eye Care
HOSPITAL	Data and Information/Health and Social Care/National Health Service/Secondary Care/Hospital
A&E / emergency department	Data and Information/Health and Social Care/National Health Service/Secondary Care/Hospital/Accident and Emergency
Inpatient	Data and Information/Health and Social Care/National Health Service/Secondary Care/Hospital/Inpatient
Outpatient	Data and Information/Health and Social Care/National Health Service/Secondary Care/Hospital/Outpatient
Maternity	API Catalogue/Maternity

Key	Location
Mental health	Data and Information/Health and Social Care/National Health Service/Mental Health
Patient-facing	API Catalogue / Patient
Pharmacy	Data and Information / Health and Social Care / Pharmacy
GP / primary care	Data and Information/Health and Social Care/National Health Service/Primary Care/General Practice
Transport / infrastructure	API Catalogue / Transport/Infrastructure
Social care	Data and Information/Health and Social Care/Care Service/ Social care
Urgent and emeregency care	API Catalogue/Urgent and Emergency Care
Ambulance	Data and Information/Health and Social Care/National Health Service/Ambulance
NHS 111	Data and Information/Health and Social Care/National Health Service/Primary Care/111
Urgent treatment centres	API Catalogue / Urgent Treatment Centre
A&E / emergency department	Data and Information/Health and Social Care/National Health Service/Secondary Care/Hospital/Accident and Emergency
OOH GP	Data and Information/Health and Social Care/National Health Service/Primary Care/Out-of-hours

Integration type

This shows the type of integration we consider the specification to describe: an API central or intermediary service, a message integration, a publish-subscribe event, or an an API standard, as defined in the Introduction to healthcare services[504].

504 https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/integration-and-apis#overview

The full list of keys, together with its location in the Bloomreach CMS taxonomy hierarchy, is as follows:

Key	Meaning	Location
API service	One or more endpoints are available for developers to integrate with. There is typically an onboarding process (even if very lightweight) and service wrapper	API Catalogue / API Service
central	Integrate to access NHS Digital services – typically data services that support healthcare processes, such as PDS, ODS, SDS	API Catalogue / API Service / central
intermediary	Integrate to enable information to flow across organisational or system boundaries, with NHS Digital acting as an intermediary or broker between healthcare organisations, such as GP Connect Access Document - FHIR API[505]	API Catalogue / API Service / intermediary
Message integration	Where one system sends a message to another system but does not expect an immediate response, for example, Transfer of Care Emergency Care Discharge - FHIR[506]	API Catalogue / API Service / Message integration
Publish-subscribe event	Where a system wants to broadcast the fact that something has happened to all interested parties, for example, Personal Demographics Service Notifications - FHIR	API Catalogue / API Service / Publish-subscibe event
API standard	A specification only, published to support interoperability, allowing interactions such as information flow across organisational or system boundaries	API Catalogue / API Standard

Technology

This is the technology used by the API - broadly corresponding to the list in API technologies at NHS Digital[507].

The full list of keys, together with its location in the Bloomreach CMS taxonomy hierarchy, is as follows:

Key	Meaning	Location
Internet facing	The API is availble on the internet	API Catalogue / API Technologies / Internet facing

505 https://digital.nhs.uk/developer/api-catalogue/gp-connect-access-document-fhir
506 https://digital.nhs.uk/developer/api-catalogue/transfer-of-care-emergency-care-discharge-fhir
507 https://digital.nhs.uk/developer/guides-and-documentation/api-technologies-at-nhs-digital

Key	Meaning	Location
FHIR	This is a FHIR API	Technology and Architecture / Standards / Messaging / HL7 / FHIR
DSTU1	This is FHIR version First Draft Standard for Trial Use	Technology and Architecture / Standards / Messaging / HL7 / FHIR / DSTU1
DSTU2	This is FHIR version Second Draft Standard for Trial Use	Technology and Architecture / Standards / Messaging / HL7 / FHIR / DSTU2
STU3	This is FHIR version Standard for Trial Use Release 3	Technology and Architecture / Standards / Messaging / HL7 / FHIR / STU3
R4	This is FHIR version Release 4	Technology and Architecture / Standards / Messaging / HL7 / FHIR / R4
REST	This is a REST API	API Catalogue / API Technologies / REST
SOAP	This is a SOAP API	Technology and Architecture / Capability / Technical Capability / Protocols / SOAP
HL7 V3	This is an HL7 V3 API	Technology and Architecture / Standards / Messaging / HL7 / HL7 V3
MESH	This is an messaging API that runs over MESH (or it is the MESH API itself!) (or it is NEMS, which relies on MESH)	API Catalogue / API Technologies / MESH
NEMS	This is an event-based API that runs over NEMS (or it is the NEMS API itself)	API Catalogue / API Technologies / NEMS
Has adaptor	This API has an "adaptor" - an open source component you can download and use to make integration easier	API Catalogue / API Technologies / Adaptors

Key	Meaning	Location
Has sandbox	This API has a self-service sandbox for having a play (generally this is all APIs on Apigee)	API Catalogue / API Technologies / Sandbox
Uses API platform	The API is hosted on the API platform - ie on Apigee.	API Catalogue / API Technologies / Uses API platform
Uses OAS	The API specification is generated from an underlying OAS file (which is downloadable via the API spec page)	API Catalogue / API Technologies / Uses OAS
Uses digital onboarding	The API uses digital onboarding	API Catlogue /API Technologies / Uses digital onboarding

Status

For the latest definitions, always check the https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#statuses.

APIs, message integrations and publish-subscribe events

By default, retired APIs (or API standards) are not listed on the page. The "Include retired APIs" toggle controls them.

When the toggle is set "on", the Status filter includes an option to select retired APIs and API standards.

The current list of keys for APIs, message integrations and publish-subscribe events but NOT API standards, were heavily revised in November 2022. Together with their locations in the Bloomreach CMS taxonomy hierarchy, they are as follows:

Key	Meaning	Location
In development	Early prototyping - the API is available, and might be available for testing via a sandbox service or an integration environment - but we expect to make breaking changes based on developer feedback	API Catalogue / Status / APIs / In development (API)
Beta	The API is available in production - it might still be subject to breaking changes	API Catalogue / Status / APIs / Beta (API)

Key	Meaning	Location
	- and its service level[508] can be anything from bronze to platinum	
In production	Once out of beta, there will be no breaking changes - if we need to make breaking changes, we'll publish a new version of the API	API Catalogue / Status / APIs / In production (API)
Internal	The API is not currently available for integration by external third parties - if you still want to use it, contact us[509]	API Catalogue / Status / APIs / Internal (API)
Under review for deprecation	The API is under review and we're considering deprecating it - if you have any concerns, contact us[510]	API Catalogue / Status / APIs / Under review for deprecation (API)
Deprecated	The API is still available and service levels still apply, but we plan to retire it at some point - we are unlikely to make any updates and new integrations are not permitted - once deprecated, we will consult with developers before deciding on a retirement date	API Catalogue / Status / APIs / Deprecated (API)
Retired	The API is no longer available for use	API Catalogue / Status / APIs / Retired (API) (makes the "Retired" filter work) **AND** API Catalogue / API / Retired (API or API standard) (makes the "include retired toggle" work)

API standards

API standards use a different set of statuses which we plan to revisit following the heavy revisions to the API statuses above. They are currently:

508 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#service-levels
509 https://digital.nhs.uk/developer/help-and-support
510 https://digital.nhs.uk/developer/help-and-support

Key	Meaning	Location
Alpha	This is an early prototype - the specification is available, - but we expect to make breaking changes based on developer feedback	API Catalogue / Status / API standards / Draft (API standard)
Beta	This API standard is available - it might still be subject to breaking changes	API Catalogue / Status / API standards / Active (API standard)
Under review for deprecation	The API standard is under review and we're considering deprecating it - if you have any concerns, contact us[511]	API Catalogue / Status / API standards / Under review for deprecation (API standard)
Deprecated	This API standard is deprecated with plans to retire it - do not begin a new integration	API Catalogue / Status / API standards / Deprecated (API standard)
Retired	The API standard is no longer available for use	API Catalogue / Status / API standards / Retired (API standard) (makes the "Retired" filter work) **AND** API Catalogue / Status / Retired (API or API standard) (makes the "include retired toggle" work)

Owner

As we begin to add non-NHS Digital APIs to the catalogue, we need tags to differentiate the owner of the API.

Key	Meaning	Location
NHS England	API is owned by NHS England	API Catalogue / API Owner / NHS England
NHS BSA	API is owned by NHS Business Services Authority (coming soon)	API Catalogue / API Owner / NHS BSA
NHS Wales	API is owned by NHS Wales (coming soon)	API Catalogue / API Owner / NHS Wales

511 https://digital.nhs.uk/developer/help-and-support

Key	Meaning	Location
Ordnance Survey	API is owned by Ordnance Survey	API Catalogue / API Owner / Ordnance Survey

Which key values to assign to a particular API

This is a decision for the subject matter experts (such as the product owner) in your API producer team.

For more context on the meaning of individual keys, see the taxonomy hierarchies above.

What to do if the right value doesn't exist

Occasionally a new API might need a key that we haven't provided.

The original taxonomy terms went through a lengthy discussion and agreement process with stakeholders from across NHS Digital.

Agreement process

If the right key doesn't exist, contact us via API producer support slack (see page 2056) to discuss your needs.

Web team rules about new taxonomy terms

Use existing terms where possible and only create new ones where necessary using the following principles:

- the topic should be broad and ideally 1 or 2 words describing a single concept e.g. "Social Care"
- if they describe 2 concepts, they should be related to each other and have Title Case e.g. "Strategies and Policies"
- every topic could be considered to have a home page of it's own
- topics can be combined with others to filter results down
- topics by default are designed for search and faceted search filtering only.
- a topic should not repeat itself in the entire taxonomy unless it has a different definition
- topics should represent what content could be filed in, not how our organisation is managed
- topics should not be in plural unless necessary. Plural terms will be entered as a synonym.
- if and is necessary use "and" and not "&"
- avoid the use of "/" to express 2 forms of the same word. Only the primary most popular term from Analytics should be used, and the alternative term present as a synonym
- if the primary term has an abbreviation, it is recommended to place it in a separate field within the data. This should have no use of full stops or bracket . e.g. "QOF"

- camelCase should not be used unless the term is a verb or is describing 2 similar concepts for example "Mortality and Death"
- abbreviations should always be separated out, as they can then be concatenated together by the system or using a function like Concatenate within Excel for example, visually in the format suggested Full Form "Quality Outcomes Framework (QOF)"
- do not move any term from it's current structure as this will impact other areas of the business

The ability to add taxonomy terms to the doc in the CMS will shortly be locked down so all new terms must be approved by the web team.

Implementation process

Once you've agreed the need for a new taxonomy term, the implementation is fairly straightforward:

1. Contact us via API producer support slack (see page 2056) and remind us of the instructions in KOP-029 API catalogue page management[512].
2. Update the taxonomy key structures shown on this page with the changes you've made.

Using Bloomreach CMS for Search Engine Optimisation (SEO) of API specification

Your API documentation pages should be discoverable by searching. This includes both internal search (on the NHS Digital website) and external search (e.g. Google)

External SEO

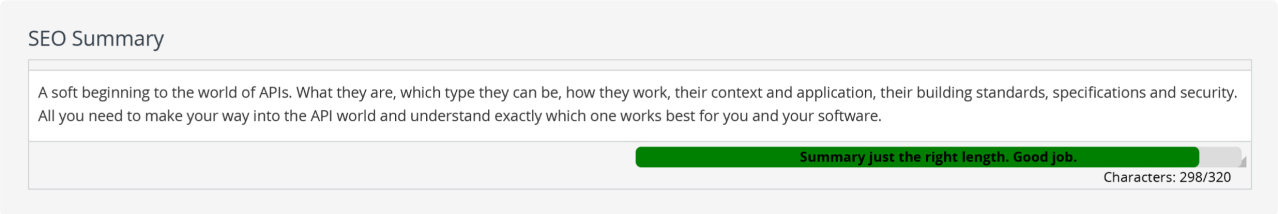
SEO Summary: what it does and how to write it

For each page, make sure you put appropriate words in the page's SEO Summary field in Bloomreach (see page 508).

This field acts as an HTML meta-description field so, even being a sentence or more, it should also contain the meta-keywords we want the external search engines to scan and use in search queries.

There's an indicator based on the input data that only points out the right length of the summary but tell us nothing of the quality of it.

Keep in mind to include keywords in the SEO Summary sentence to be sure the page will get indexed the best it can.



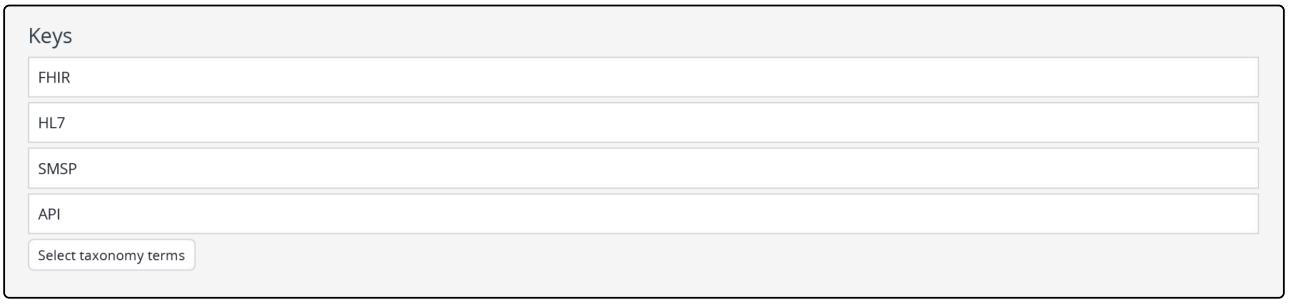
512 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/203363261/KOP-029+API+catalogue+page+management

Internal SEO

Note - freestyle use of taxonomy keys is currently on hold for API specs while we use this feature to create tags and filters

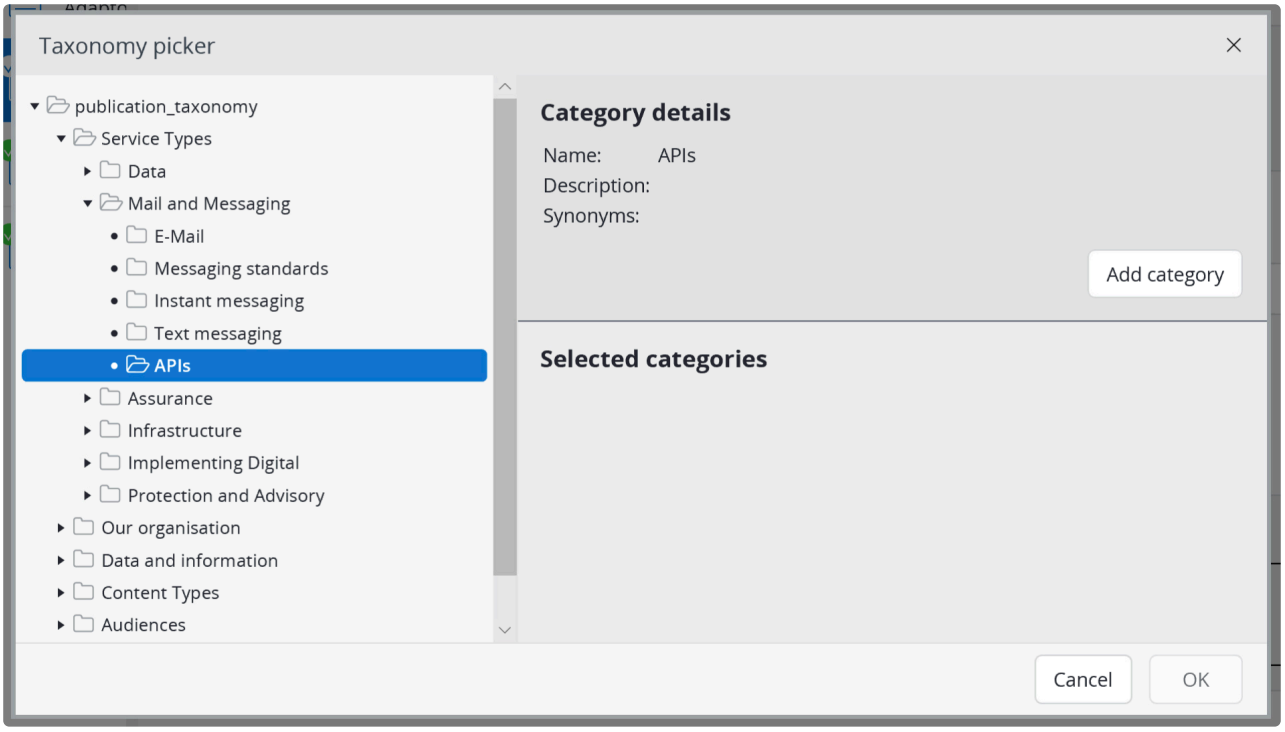
Taxonomy Keys

For each page, make sure you include appropriate entries in the page's Keys field in Bloomreach. The field is usually located towards the bottom of the page:

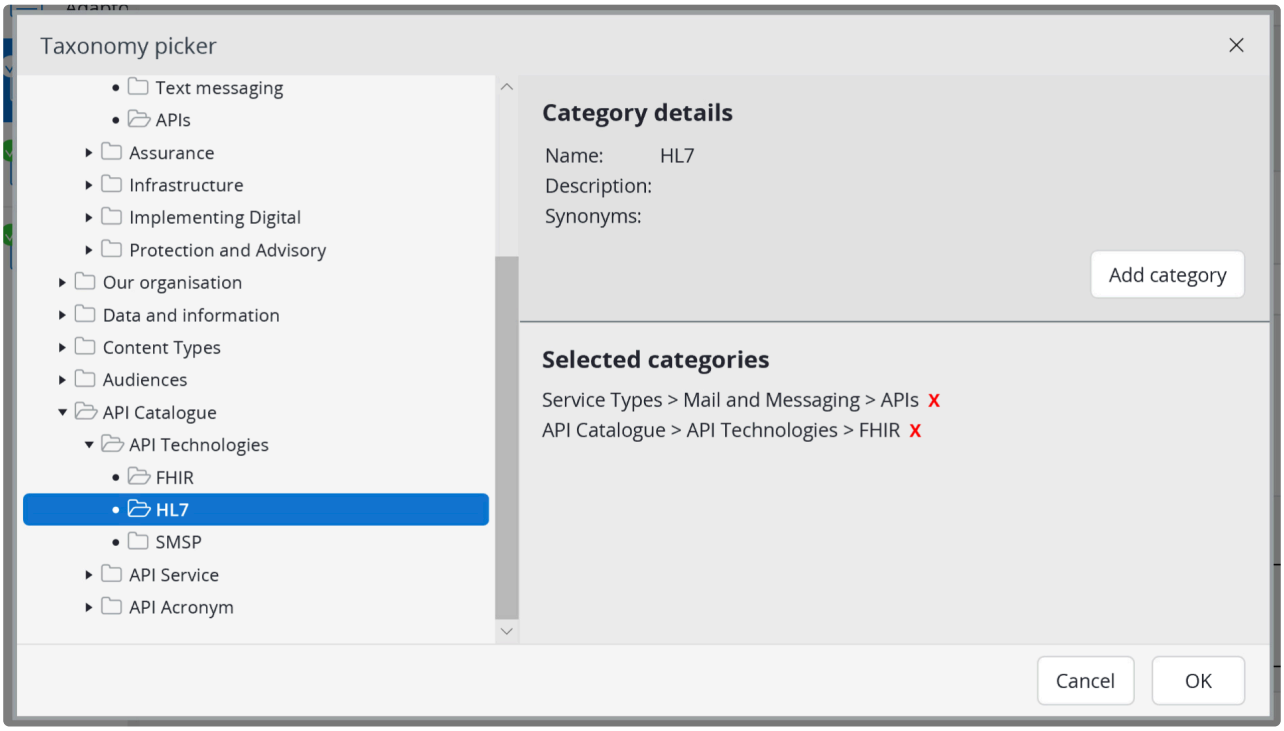


To include a new one click on **Select Taxonomy Terms** and the **Taxonomy Picker** will popup.

Select the term you want to add and click on **Add category** and it will be added to the Selected categories list.



To include multiple terms you need to select one at a time, click **Add category** each time unless you're done with all of them.

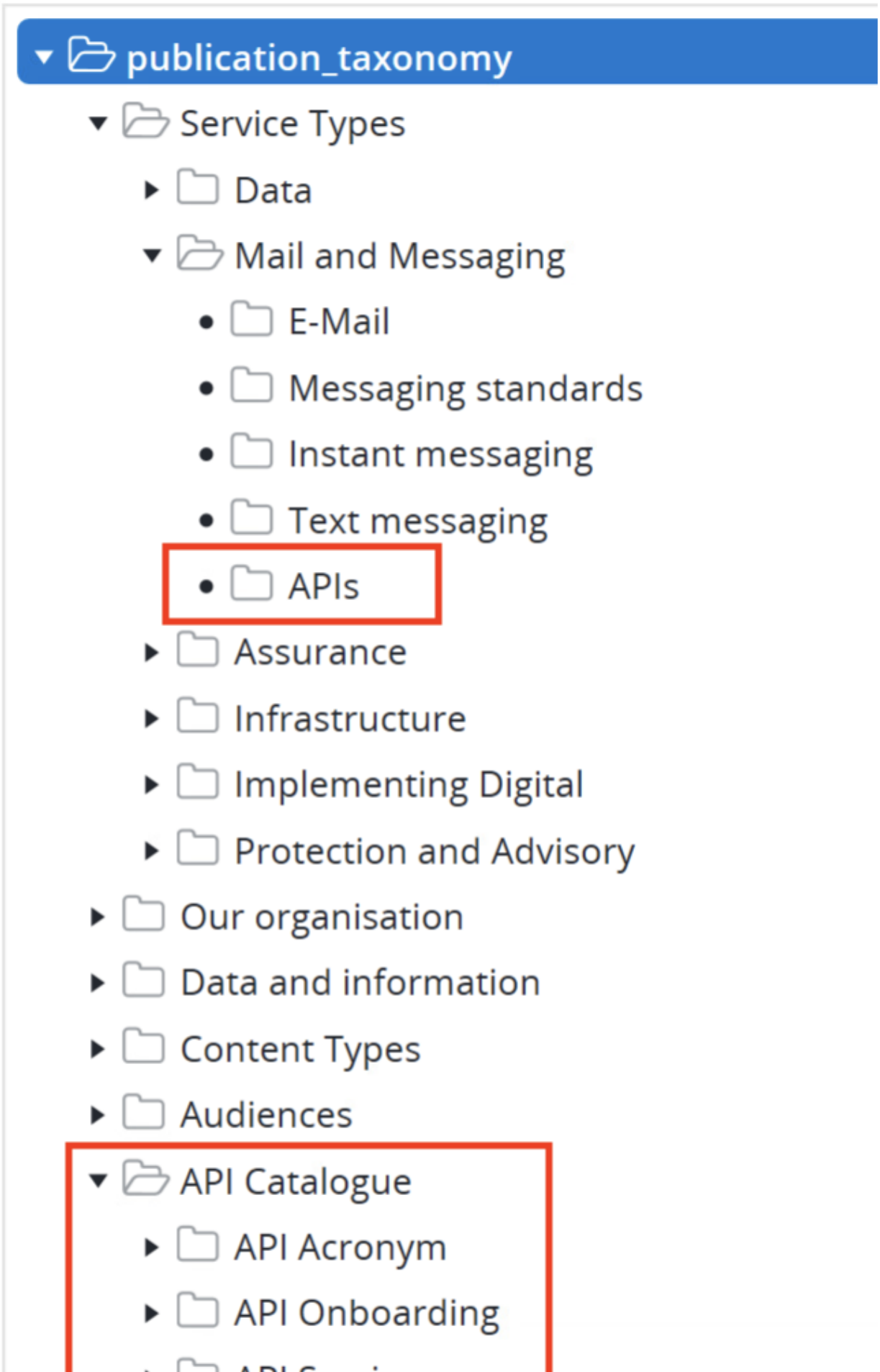


Best practice for adding keys to a page

A) The recommendation **is to utilise at least one key from every taxonomy category** (enclosed image), this means :

1. **Service > Mail and Messaging > APIs**
2. One from **API Catalogue > API Acronym**
3. One from **API Catalogue > API Service**
4. One from **API Catalogue > API Technologies**

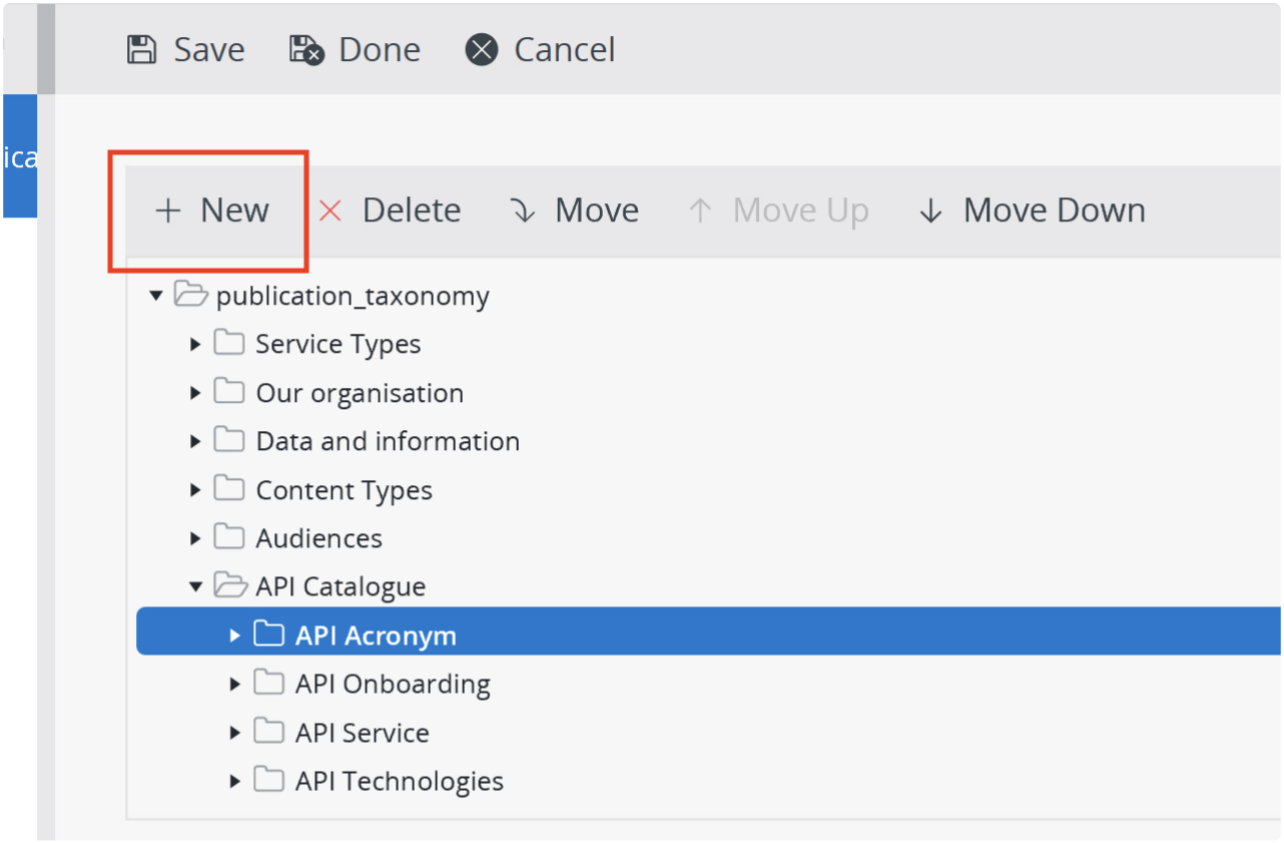
B) The keys from **API Catalogue > API Onboarding** are not mandatory for the specs page but they can add value for users searching for info about the Onboarding process for a particular API.


publication_taxonomy
Service Types
Data
Mail and Messaging
E-Mail
Messaging standards
Instant messaging
Text messaging
APIs
Assurance
Infrastructure
Implementing Digital
Protection and Advisory
Our organisation
Data and information
Content Types
Audiences
API Catalogue
API Acronym
API Onboarding

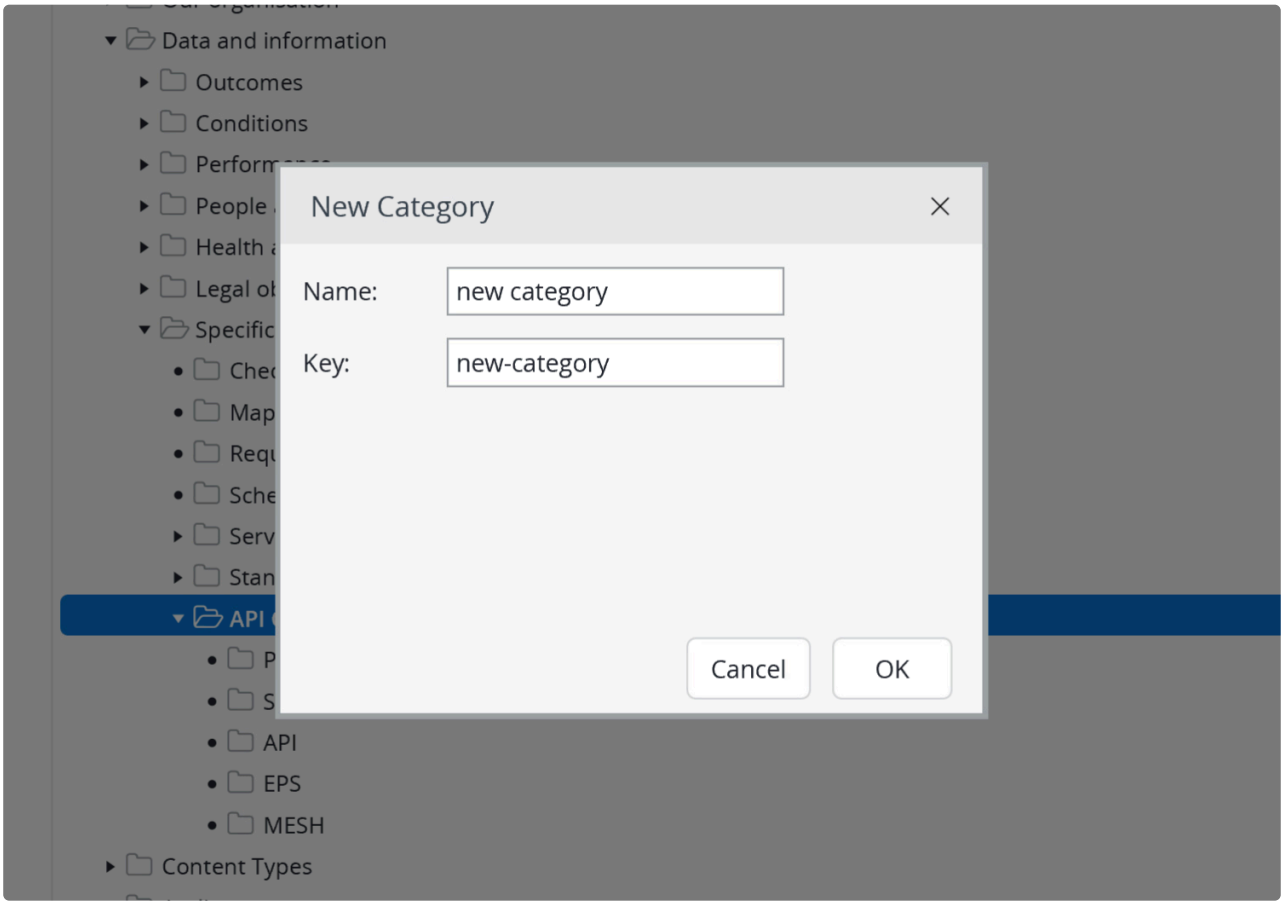
Adding new keys/taxonomies

If you want to add a key that doesn't already exist, you can introduce it into **publication_taxonomy** from the **Taxonomies** section of Bloomreach content.

1) Identify the parent category (API Catalogue) and click on **NEW**



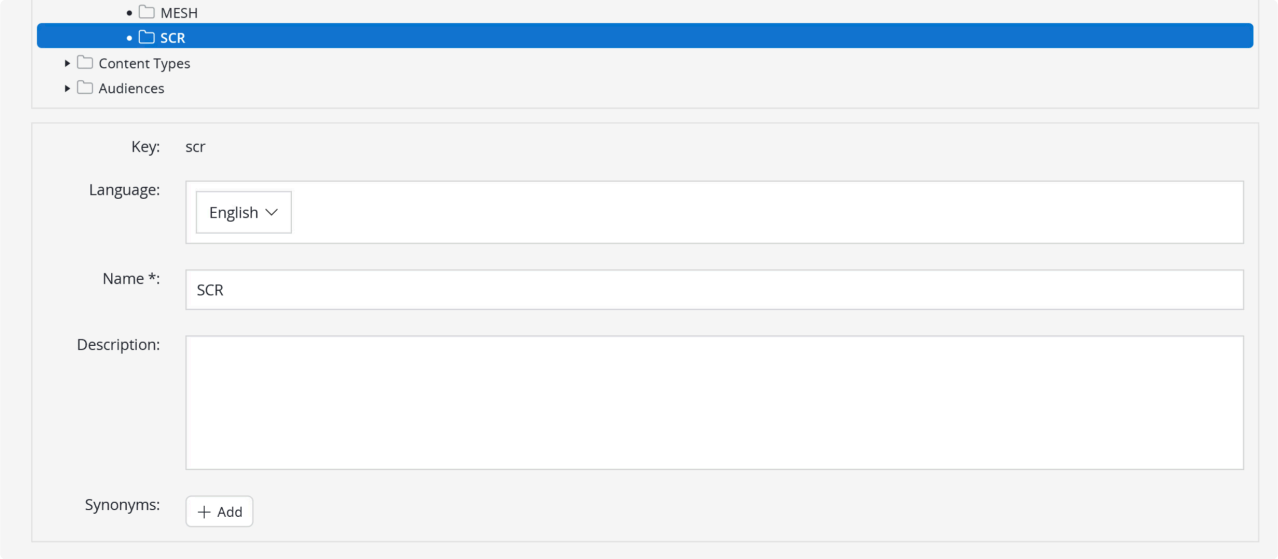
2) The **NEW CATEGORY** creation modal comes out. Insert **NAME** and **KEY.** The NAME is just to identify the category while the KEY is the term that identifies it during the search.



3)After creating it, you can further modify parameters like:

- Language
- Name
- Description
- Synonyms

Synonyms, in particular, is very useful because you can put here all the other terms that can identify that particular key, without having the hassle of creating new keys.



7.14.4.21.5.11 Configuring and setting up the OAS file

Overview

Some specific configuration is required to get your deployed spec to be visible on Bloomreach.

Bloomreach polls the Apigee Developer Portal for the OpenAPI spec to publish on Bloomreach, specifically the one associated with the `int` environment.

Therefore to publish your spec on Bloomreach you must first publish it on the Apigee's API catalogue in the Integration environment.

This requires a combination of `specs`, `products` and `api_catalog` objects to be deployed to Apigee.

These deployments are controlled by your `manifest.yml` file (or `manifest_template.yml` file).

Example

Here is a minimal exemplar `manifest.yml` file which publishes the contents of the `specification` directory:

Minimal manifest configuration for publishing a spec on Bloomreach

```
meta:
  api:
    name: prescription-transmission
    guid: f5f26970-ce52-4eef-95c5-a169810ae6e3
    spec_guids:
      - fac3b3c9-638c-4a00-aa4b-3c7172abe80b
```

```
  schema_version: 1.1
apigee:
  environments:
  - name: int
    api_catalog:
      - edgeAPIProductName: prescription-transmission-internal-dev
        visibility: true
        specId: prescription-transmission-internal-dev
        title: Electronic Transmission of Prescriptions
    products:
      - name: prescription-transmission-internal-dev
    specs:
      - name: prescription-transmission-internal-dev
        path: prescription-transmission.json
```

Apigee's Spec Publication Model

Apigee expects `spec` objects to be associated with API products, which control subscription access to APIs.

- Developers log into a portal and "subscribe" to API Products, which grants them access to use the API.
- If you are using the API management platform to publish a proxy, the `products` section of your manifest.yml file will be filled with detailed information controlling authentication mechanisms and rate limiting.

For the purposes of publishing a spec to Apigee's API catalog, an "empty" product will do.

Note: what ends up on Apigee is not quite an empty product, we inject some minimal content as a part of deployment as a truly empty product grants access to everything on the platform ⚠️
So for security reasons, we don't do that.

So, to expose your spec externally in Apigee's API catalog, you must map an `int` environment API Product to your spec.

This mapping is done via the `api_catalog` section in the manifest file.

- The `edgeAPIProductName` field should be the `name` of a product in the `products` section.
- The `specId` field should be the name of one of the specs in the `specs` section.

The above example is the minimal case where this is only one of each thing.

For more complicated API's, there are multiple products per proxy.

Note: The `.json` suffix on the spec path object comes from the fact that the deployment pipeline compiles the contents of the spec directory into a single large spec file.
For example `specification/prescription-transmission.yaml` will be assembled into `presciption-transmission.json`

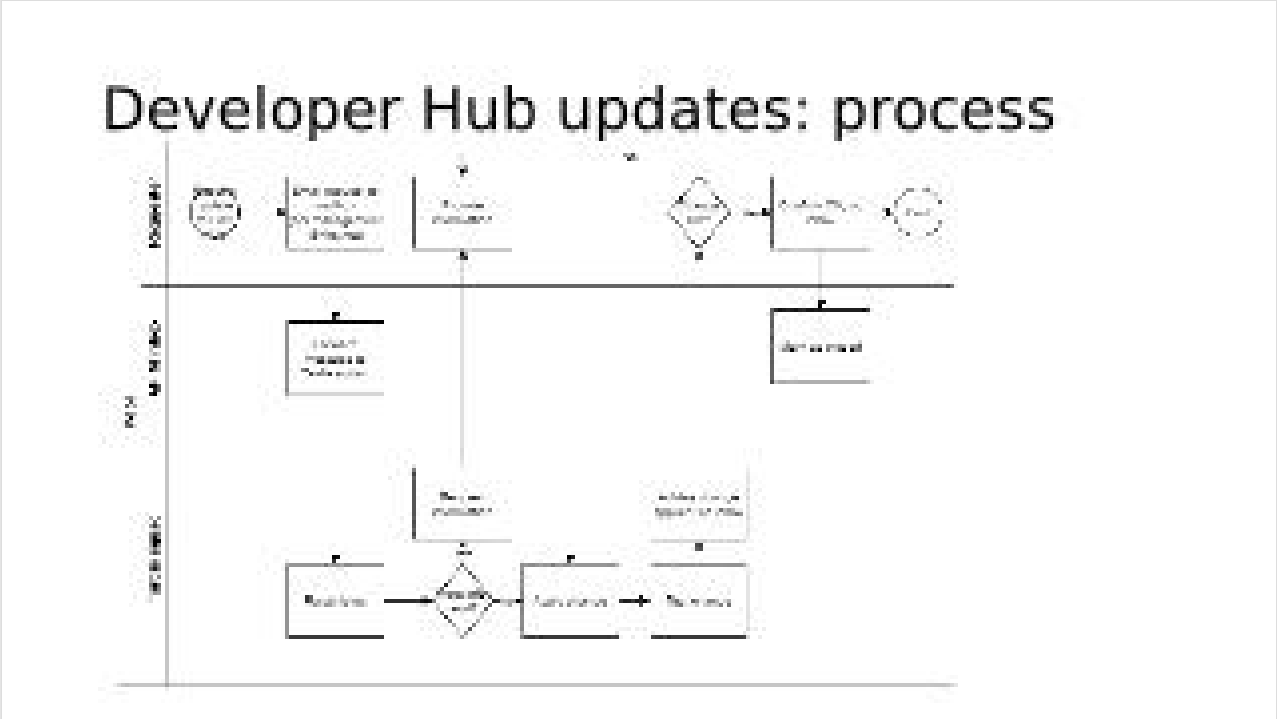
To actually publish the spec in Apigee's API catalog, set the `visibility` flag in the `api_catalog` section to `true` .

Then it is visible externally and will be pulled from Apigee `int` into Bloomreach `prod` .

7.14.4.21.5.12 Requesting changes to the Developer Hub

The API Catalogue is becoming the one-stop shop for Developers to learn about NHS Digital APIs and is replacing other Hubs or Zones - in the first instance, the Health Developer Network API Hub[513].

Updates in relation to particular APIs e.g. new or updated versions will typically come from the IOPs team - the Powerpoint below documents the process and also provides a template email to ensure you supply the correct information.



513 https://developer.nhs.uk/apis/

7.14.4.21.5.13 API specification content review meeting

The reasons for and process how to conduct the API specification content review meeting are covered below:

Purpose:

- Ensure your API specification is clear, concise, and easy to understand for external developers.
- Verify compliance with our API documentation guidelines (see page 445).
- Maintain consistency with other API specifications.

Process:

- Presentation: You will present your API specification to the review team.
- Review and Feedback: The team will provide feedback on the clarity, completeness, and consistency of the specification.
- Revision: You will incorporate the feedback and make necessary revisions.
- Additional Review: We may conduct an additional offline review for a more detailed analysis.

By following these steps, we aim to ensure the highest quality API specifications.

Recommended Attendees:

- API producer team
    - specification writer
    - product owner
- API platform team
    - tech author - Nathan Kinsler
    - producer liaison - Jonathan Eagle

To request a review, contact Nathan Kinsler

7.14.4.21.5.14 New spec renderer (RapiDoc) spring 2023

What is this?

The API Management team are implementing a new renderer for API specs, which will be integral to how the API specs are displayed in the API Catalogue (https://digital.nhs.uk/developer/api-catalogue)

We have chosen RapiDoc (https://rapidocweb.com/) as the new renderer.

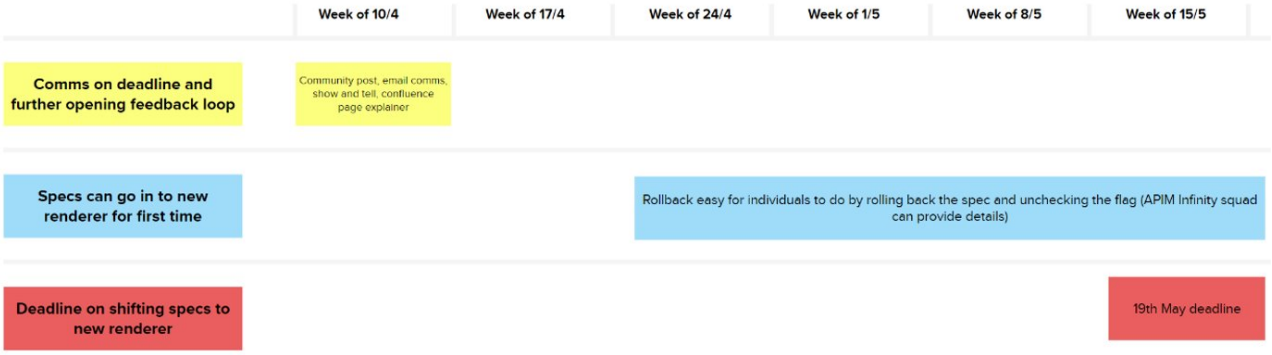
Why is it being done?

The previous custom-built renderer did not quite completely support the OAS v3.0 spec, and we encountered some issues with larger specs which use the more advanced aspects of that standard. Therefore, the decision was taken to find a new renderer tool which was compliant with all of OAS v3.0, and hopefully would also speed up spec deployment.

When is it being done?

April to May 2023. This is the current planned timeline:

(see page 557)



What do API Producers have to do?

1. From 24th April 2023, check their specs in the non-production environment[514] of RapiDoc, to verify that they still render correctly. NB: This step is necessary as previously specs may have used non-standard components to work with the old renderer. Also check you are using the VPN (see page 168) to ensure you can access this.
2. Make any required changes to the specs so that they render correctly in non-production RapiDoc.
3. Confirm to the API Management team (api.management@nhs.net[515]) that their spec is OK in non-production RapiDoc.
4. The API Management team will then switch over that spec to RapiDoc, away from the old renderer.

Please note : any remaining specs will be switched over to RapiDoc at the 19th May regardless, so API Producer teams have a four-week window to test and confirm their specs in non-production RapiDoc.

What's the rollback plan?

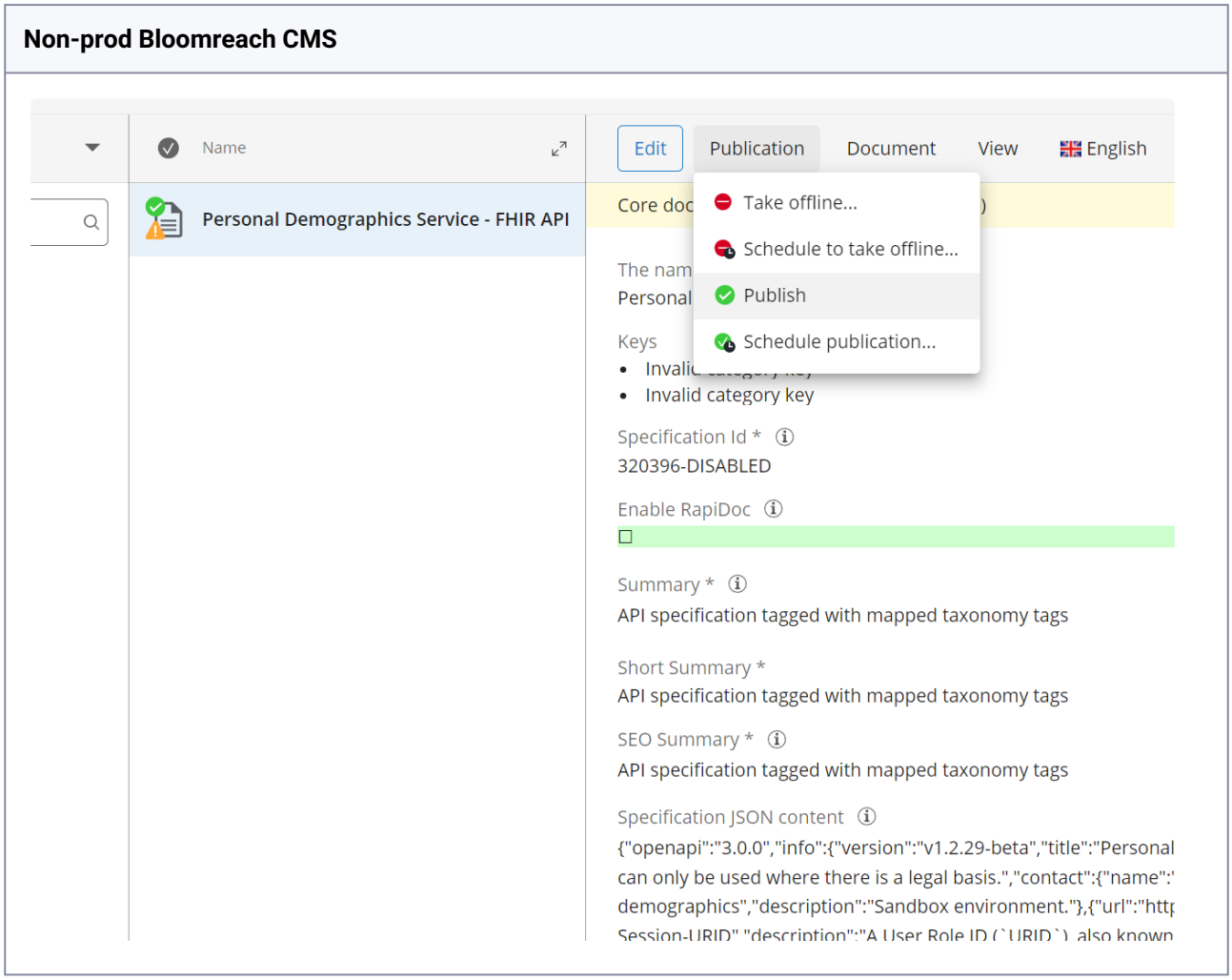
Should a spec go live in RapiDoc and not work correctly, it can be temporarily switched back to the old renderer by the API Management team while the reason for it not rendering correctly in RapiDoc is resolved.

514 https://training2.nhsd.io/developer/api-catalogue
515 mailto:api.management@nhs.net

How to toggle RapiDoc rendering

RapiDoc rendering for a single specification can be easily toggled on or off in any Bloomreach environment using the CMS. To find the toggle, navigate to the spec document in the CMS if you have access or contact us if not. To use Rapidoc rendering, click Edit and make sure the 'Enable RapiDoc' checkbox is checked. Click Done and you should now see the appearance of the checked box. After publishing the spec, it should now be using the new RapiDoc renderer after a quick refresh.

If still required to preview specs using the old renderer, instead make sure the 'Enable RapiDoc' checkbox is unchecked, click Done and Publish. After a refresh your spec should now be using the old renderer.



Testing spec changes in non-production Bloomreach using RapiDoc

To verify that any changes to specs render correctly using RapiDoc, you can preview specifications in a non-production Bloomreach environment before switching over your spec to RapiDoc in production.

We now provide several options to do this:

- Using the old method as documented here: Previewing API specification on Bloomreach UAT website[516]. The training environment mentioned (and any other non-prod Bloomreach environment) will use RapiDoc to render specs by default. It is still possible to preview specs using the old renderer, for instructions see How to toggle RapiDoc rendering (see page 559). This method requires publishing specs in non-prod Apigee and creating named specifications in both Apigee and Bloomreach CMS. The specification in Apigee is automatically imported into Bloomreach every 10 minutes if there is a change. This way takes time to initially set up and does not provide an instant feedback loop. Although once set up any spec changes pushed to your branch in github will be automatically published to Bloomreach once the import has run.
- Using the new upload method (see page 560). This method does not require logging into Apigee or the Bloomreach CMS and circumvents the 10 minute wait for the import job, so may be quicker when testing spec changes. The upload only happens in the browser though, so navigating away from the page or refreshing will reset the page to the content stored in the CMS.

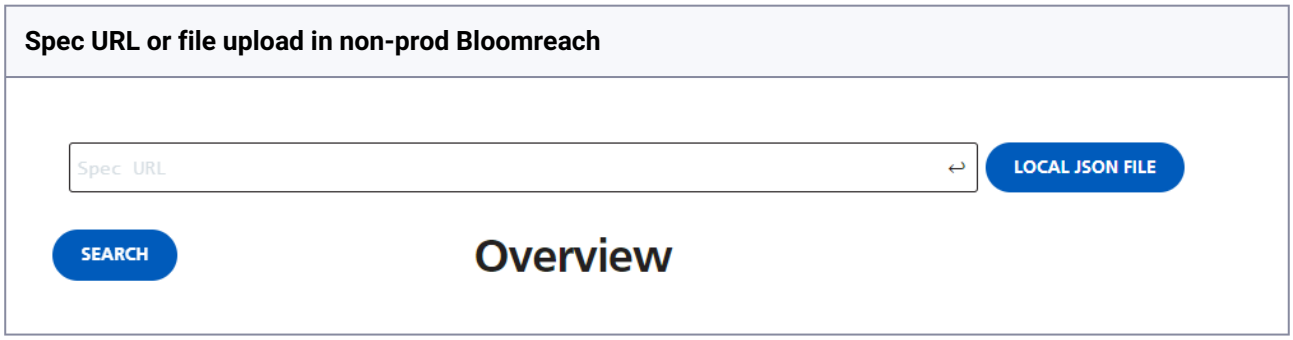
Uploading a spec via URL or file in non-prod Bloomreach

In non-prod Bloomreach environments only (such as training), RapiDoc provides the ability to load any spec URL or file. The spec URL or file provided should contain the whole spec in one single location/file. Below are the steps you may take using the file upload method:

1. Have the repository where you are making changes to your spec cloned locally (e.g. https://github.com/NHSDigital/personal-demographics-service-api/tree/master)
2. Follow the `README` for that repository to set up for development
3. Make changes to the OAS files under the `specification/` directory
4. Run `make publish` to output the specification as a single file into the `dist/` directory (as detailed in the repositories README)
5. Navigate to any spec page in a Bloomreach non-prod environment (e.g. https://training2.nhsd.io/developer/api-catalogue/personal-demographics-service-fhir)
6. Click "LOCAL JSON FILE" and open the specification file created earlier in the `dist/` directory - your specification should now render. If not, check the dev console for any errors, as the OAS file may be invalid.

As mentioned previously, this upload is client-side only so this won't overwrite the specification stored server-side in the CMS. Navigating away from the page or refreshing the page will reset the spec to the original one stored in the CMS.

516 https://nhsd-confluence.digital.nhs.uk/display/APM/Previewing+API+specification+on+Bloomreach+UAT+website



RapiDoc troubleshooting

'TRY' feature not working

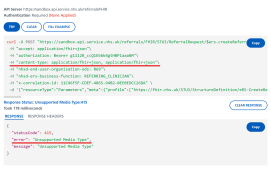
RapiDoc still allows users to try out APIs using sandbox environments. As per old functionality, this feature can also be disabled, see the Formatting OAS API specifications (see page 458) page. Instead of this feature being available via a pop up window, the 'TRY' feature is now built into the spec itself within the operations pages.

Although in most cases this feature works as expected with no changes required, we've noticed some instances where changes to the specification are required for RapiDoc's 'TRY' feature to work. In these instances, check the cURL request being executed and the returned response for clues.

In the below example, the content-type value has been repeated twice and so the response is returning invalid media type. Values for the "Content-Type" header should be defined in the OAS spec by the `requestBody.content.<media-type>` keyword, see the swagger documentation[517] for examples. This duplication happened as RapiDoc does not ignore header parameters named "Content-Type". Removing this header will resolve the issue.

517 https://swagger.io/docs/specification/media-types/

RapiDoc	Specification

	**YAML**
	`...`
	`parameters:`
	`# This header should be removed as it's already defined`
	`- in: header`
	`name: Content-Type`
	`description: |`
	`Indicates the media type of the associated resource.`
	`required: true`
	`schema:`
	`type: string`
	`enum:`
	`- application/fhir+json`
	`example: application/fhir+json`
	`...`
	`requestBody:`
	`required: true`
	`content:`
	`# Standard way the "Content-Type" header is defined`
	`application/fhir+json:`
	`schema:`
	`...`

Another difference to look out for between RapiDoc's 'TRY' feature and old renderer functionality is the application of default values when describing request parameters. The `default` keyword in the parameter schema is the one the server uses if the client does not supply the parameter value in the request. A common mistake when using the `default` keyword is using `default` to specify a sample value. This is not the intended use of `default` and can lead to unexpected behaviour when using RapiDoc's 'TRY' feature, as it does not automatically apply default values to the request. Use the `example` or `examples` keyword instead. See the swagger documentation[518] for more details.

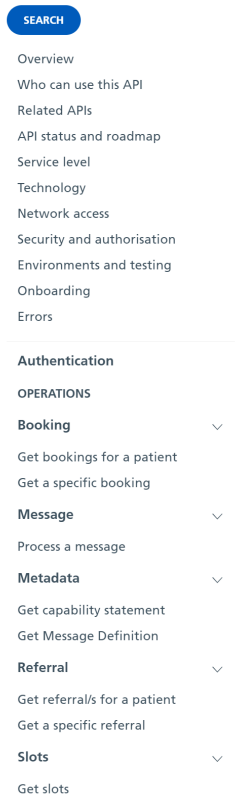
Grouping and ordering operations in the navigation menu

Grouping and ordering of operations is still possible with the new RapiDoc renderer. However, how this is achieved has changed so you may notice that the order of endpoints in the side navbar has changed. With the old renderer, a custom metadata object known as 'operation-order (see page 468)' was added to the specification. If this custom metadata was not supplied, the default ordering of operations was alphabetical.

518 https://swagger.io/docs/specification/describing-parameters/

RapiDoc uses the standard way of grouping known as tags[519]. The tag order in the global tags section controls the order of those groups. The order of the operations are controlled by the order they appear in the OAS file itself. However, if the same endpoint has multiple operations, then they are sorted by method (e.g. GET before POST)

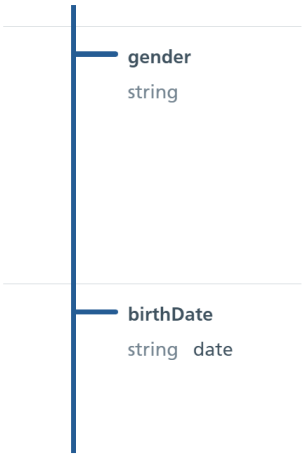
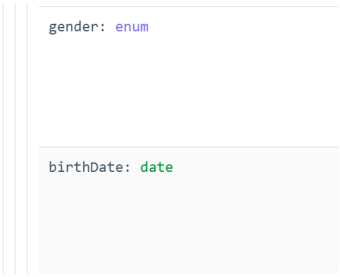
RapiDoc	Specification

	**YAML**
	`...`
	`tags:`
	`  # Defines groups and order of groups`
	`  - name: Booking`
	`  - name: Message`
	`  - name: Metadata`
	`  - name: Referral`
	`  - name: Slots`
	`...`
	`paths:`
	`  # Order of paths within the spec itself controls the order in which they appear in RapiDoc`
	`  /Appointment:`
	`  # Operations with the same path are ordered by method e.g. GET before POST`
	`  get:`
	`  tags:`
	`  - Booking`
	`  summary: Get bookings for a patient`
	`...`
	`  /Appointment/{id}:`
	`  get:`
	`  tags:`
	`  - Booking`
	`  summary: Get a specific booking`
	`...`

Schema data type differs between RapiDoc and old renderer

Although the data type of a schema is defined by the `type` keyword, it is also possible to define a more specific type for the same schema through additional keywords such as `enum` , `format` , `nullable` etc. In these cases, RapiDoc shows the more specific type. For example, when a schema is defined with a `type` of `string` and is followed by a list of `enum` , the type will be rendered as `enum.`

519 https://swagger.io/docs/specification/grouping-operations-with-tags/

Old renderer	RapiDoc	Specification
		**YAML** ... gender: type: string description: | ... enum: - male - female - other - unknown example: female birthDate: description: | ... example: '2010-10-22' type: string format: date ...

Auto filled examples

A difference between the old renderer and the new RapiDoc renderer is that if no examples are added to the request or response body objects (see the swagger documentation[520] for how to add examples), then the

520 https://swagger.io/docs/specification/adding-examples/

example tab is still rendered using values auto generated from the schema data types. This is intended functionality and works similar to Swagger UI[521].

RapiDoc	Specification
	YAML ... responses: '200': description: OK content: application/json: schema: type: object properties: messages: type: array items: type: string description: Array of strings of messageId in inbox ...

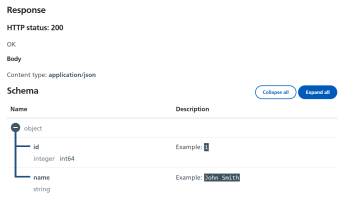
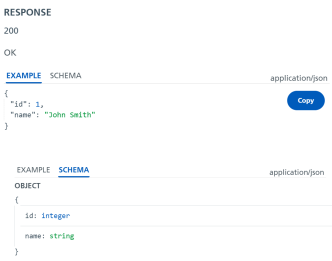
Examples not shown

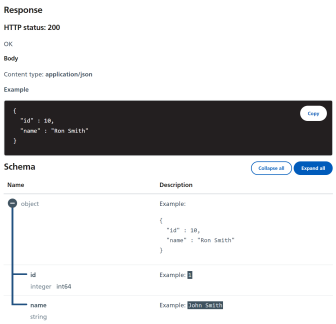
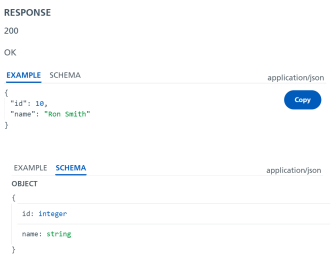
Examples can be defined at different levels, including the request body and response level, the object level and property level (see the swagger documentation[522] for more information).

RapiDoc does not present examples within the schema itself (as the old renderer did). Instead, examples defined at any level are combined into the example tab, with priority given to the example defined at the higher level:

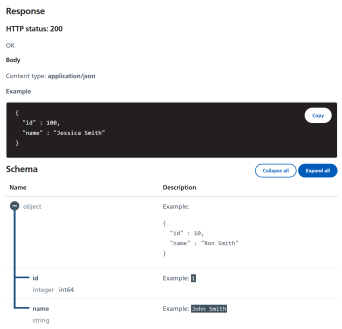
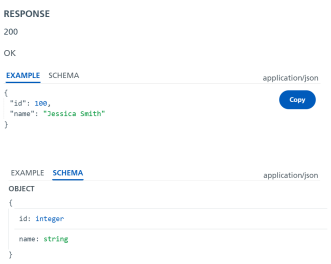
521 https://editor.swagger.io/
522 https://swagger.io/docs/specification/adding-examples/

Scenario	Old renderer	RapiDoc	Specification

Example defined at the property level; property example shown in the RapiDoc example tab.			**YAML**
			`...`
			`responses:`
			`'200':`
			`description: OK`
			`content:`
			`application/json:`
			`schema:`
			`type: object`
			`properties:`
			`id:`
			`type: integer`
			`format: int64`
			`example: 1 # Property-level example`
			`name:`
			`type: string`
			`example: John Smith # Property-level example`
			`...`

Scenario	Old renderer	RapiDoc	Specification
Examples defined at the object and property level; object example shown in the RapiDoc example tab			**YAML** ... responses: '200': description: OK content: application/json: schema: type: object properties: id: type: integer format: int64 example: 1 # Property-level example name: type: string example: John Smith # Property-level example example: # Object-level example id: 10 name: Ron Smith ...

Scenario	Old renderer	RapiDoc	Specification

Examples defined at the requestBody, object and property level; requestBody example shown in the RapiDoc example tab.			**YAML**
			`...`
			`responses:`
			`'200':`
			`description: OK`
			`content:`
			`application/json:`
			`schema:`
			`type: object`
			`properties:`
			`id:`
			`type: integer`
			`format: int64`
			`example: 1 # Property-level example`
			`name:`
			`type: string`
			`example: John Smith # Property-level example`
			`example:`
			`# Object-level example`
			`id: 10`
			`name: Ron Smith`
			`example:`
			`# requestBody-level example`
			`id: 100`
			`name: Jessica Smith`
			`...`

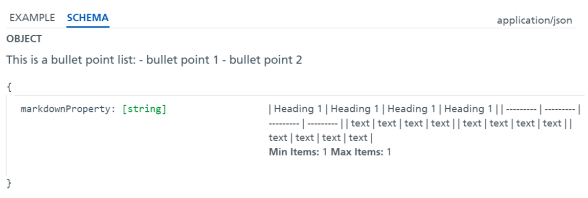
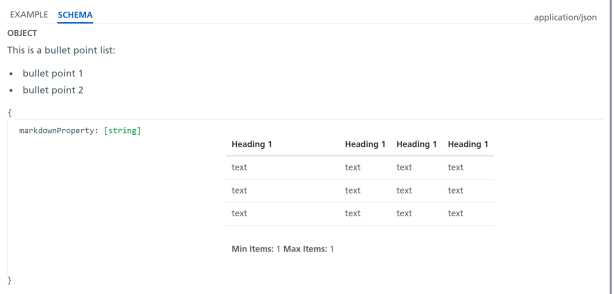
Broken internal spec links

Like the old renderer, markdown syntax can be used to create internal links to various sections of the document. However, how these links are generated has changed. Therefore, any internal links will be broken using the new RapiDoc render. Such links should be updated (or removed) before switching over to the new renderer. See the RapiDoc documentation[523] for details around link formatting. For example, the old link format `[A004](#api-Default-a004-retrieve-reference-data)` should be updated to `[A004](#get-/STU3/CodeSystem/-codeSystemType-)`

In addition, any links to the old 'endpoints' heading (i.e. `[Endpoints](#api-endpoints)`) should be removed, as this heading no longer exists as endpoints are separated into pages.

Markdown not rendering correctly

Like the old renderer, RapiDoc supports CommonMark markdown formatting. However, in a few certain scenarios markdown lists and tables have not rendered as expected. Adding a newline before the list or table should fix the issue:

Issue	Fix
	

523 https://rapidocweb.com/examples/links.html#overview--internal-links

Issue	Fix

YAML

```
...
              schema:
                type: object
                description: |
                  This is a bullet
point list:
                  - bullet point 1
                  - bullet point 2
                properties:
                  markdownProperty:
                    type: array
                    minItems: 1
                    maxItems: 1
                    description: |
                      | Heading 1 |
Heading 1 | Heading 1 | Heading 1 |
                      | --------- |
--------- | --------- | --------- |
                      | text | text |
text | text |
                      | text | text |
text | text |
                      | text | text |
text | text |
                  items:
                    type: string
...
```

YAML

```
...
              schema:
                type: object
                description: |
                  This is a bullet
point list:

                  - bullet point 1
                  - bullet point 2
                properties:
                  markdownProperty:
                    type: array
                    minItems: 1
                    maxItems: 1
                    description: |

                      | Heading 1 |
Heading 1 | Heading 1 | Heading 1 |
                      | --------- |
--------- | --------- | --------- |
                      | text | text |
text | text |
                      | text | text |
text | text |
                      | text | text |
text | text |
                  items:
                    type: string
...
```

RapiDoc code

For day to day operations the code is not relevant, however it forms a useful reference point:

- RapiDoc fork tailored to Hippo CMS: https://github.com/NHS-digital-website/hippo-RapiDoc
- Hippo CMS source for the main digital.nhs.uk site: https://github.com/NHS-digital-website/hippo

7.14.4.21.5.15 Publishing API specs using the Proxygen Generator API or Proxygen-cli

Overview

This page documents how to publish your *API specification* using the Proxy Generator API or the Proxygen-cli.

If you are looking to publish *supplementary API documents* refer to the Using Bloomreach CMS (see page 508) page.

The Proxy Generator API takes a spec-first approach by using the OpenAPI specification as a payload for deploying all platform resources, including the API specification itself.

You should use Proxy Generator to deploy your API documentation if:

- you're using Proxy Generator to manage your API proxy deployments; *OR*
- you're only deploying API documentation (read more about spec-only APIs (see page 696))

Prerequisites

Proxy Generator API prerequisites

Before publishing API documentation with the Proxy Generator API you must:

- have the documentation written, reviewed and ready to deploy (see page 451)
- be set up to use Proxy Generator (see page 681)
- authenticate with Proxy Generator (see page 682)

Proxygen-cli prerequisites

Before publishing API documentation with the Proxygen-cli you must:

- have the documentation written, reviewed and ready to deploy (see page 451)
- Install the Proxygen-cli (see page 934)
- Setup your settings and credentials as captured here

Deploying your UAT and production specification

The following sections detail how to publish your API specification. Before getting the API live on the API and integration catalogue[524], it is a good idea to publish to the UAT version of the catalogue, allowing you to preview how the API documentation will look. It can also be helpful for discovering and addressing potential errors within the OAS code.

Deploying your UAT or production specification using Proxy Generator API or the Proxygen-cli stores it on the API Platform infrastructure. If this is the first time deploying a specification, there are extra steps to publish it in both the UAT or production API catalogues, which are detailed here (see page 522).

524 https://digital.nhs.uk/developer/api-catalogue

To see how proxygen-cli commands can be integrated into CI/CD pipelines for publishing specifications, view the examples below:

- UAT: https://github.com/NHSDigital/proxygen-api-specification/blob/main/.github/workflows/publish-uat.yml
- Prod: https://github.com/NHSDigital/proxygen-api-specification/blob/main/.github/workflows/release.yml

Publishing specifications using the Proxy Generator API

To create or update your specification using the Proxy Generator API, perform the following steps:

Note: The steps below use Postman to update the spec, but other tools such as CURL can be used.

1. Open Postman (or your desired application) to make your API call.
2. Obtain a Proxy Generator access token (see page 682) to authenticate the API call.
    a. Click the **Authorization** button and select **Bearer Token** from the drop-down menu.

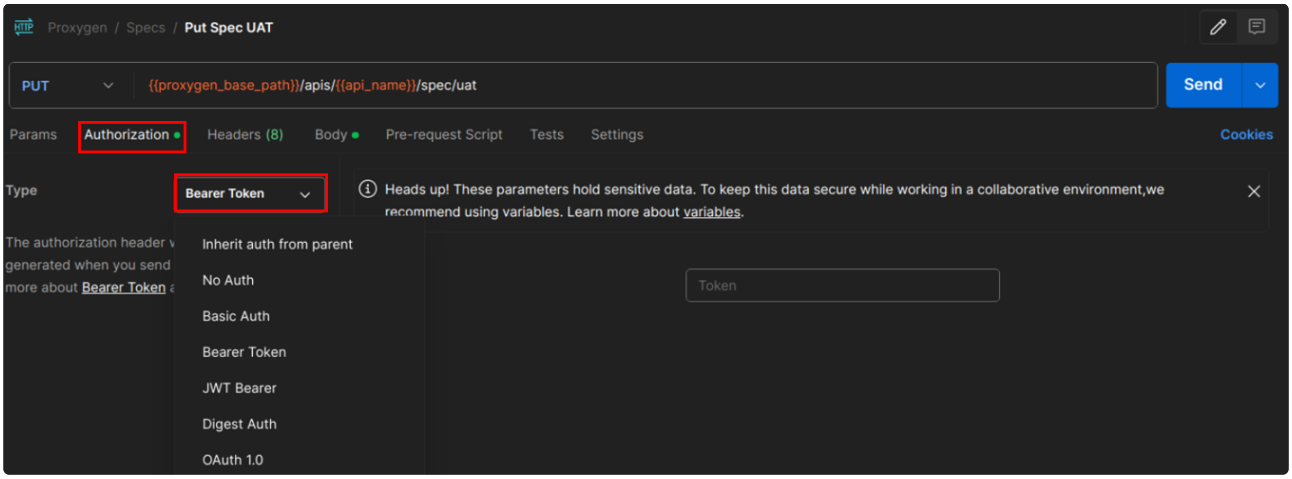


Fig 1: Put Spec - Authorisation and Bearer Token

3. Select the "PUT" command and enter the corresponding endpoint in the adjacent box, see fig 2.

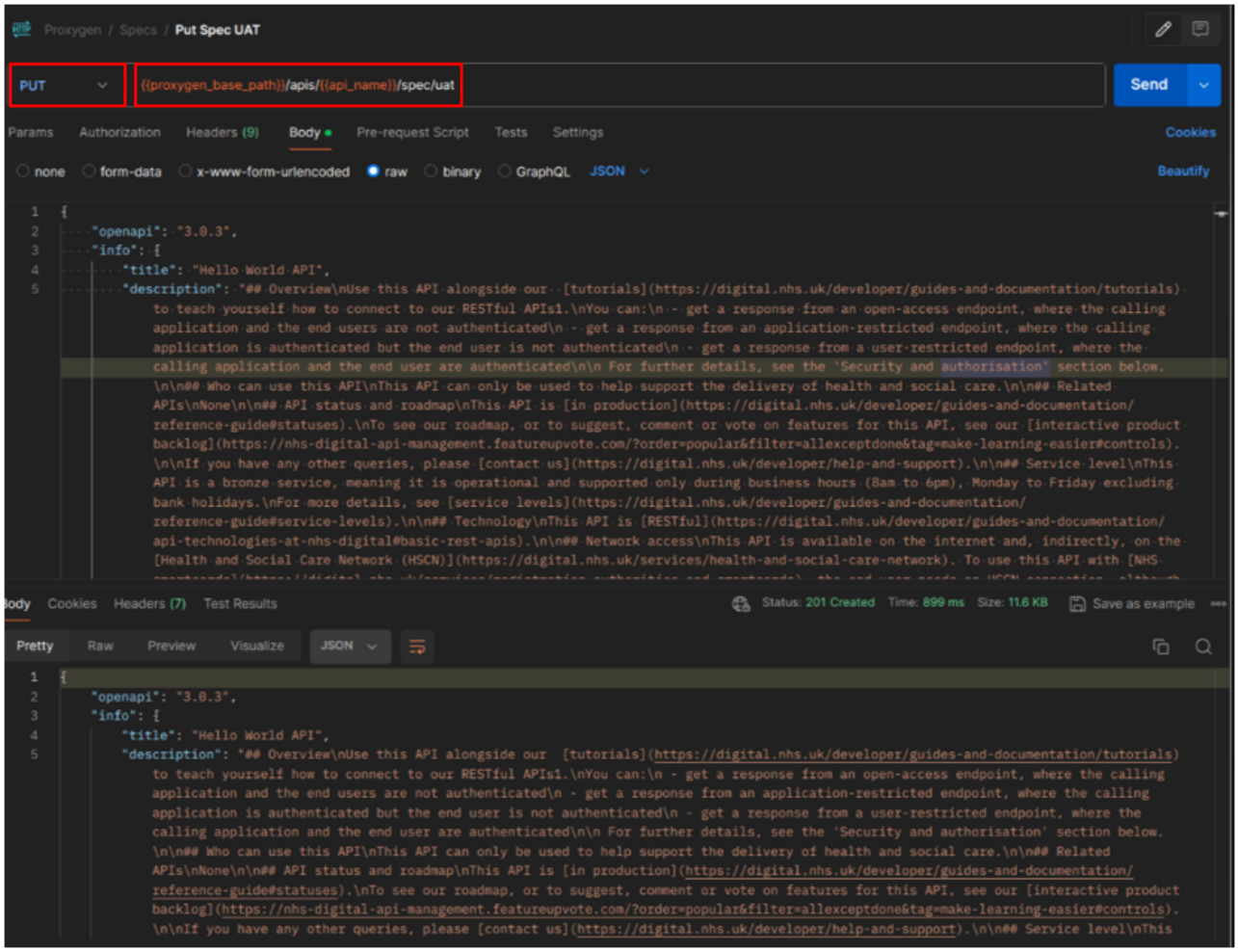


Fig 2: PUT command for UAT

4. Click the **Send** button.

The "200 OK" status message is displayed, and your spec is returned in the response body indicating the spec has successfully published.

Note: To view the spec in Bloomreach refer to the guidelines here (see page 522).

Endpoints for creating or updating specifications in production or UAT

Once the specification has been verified in UAT, use the production spec endpoint to publish the specification, as follows:

Create or update specification using the API
`# Production` `PUT /apis/{api_name}/spec`

```
# UAT
PUT /apis/{api_name}/spec/uat
```

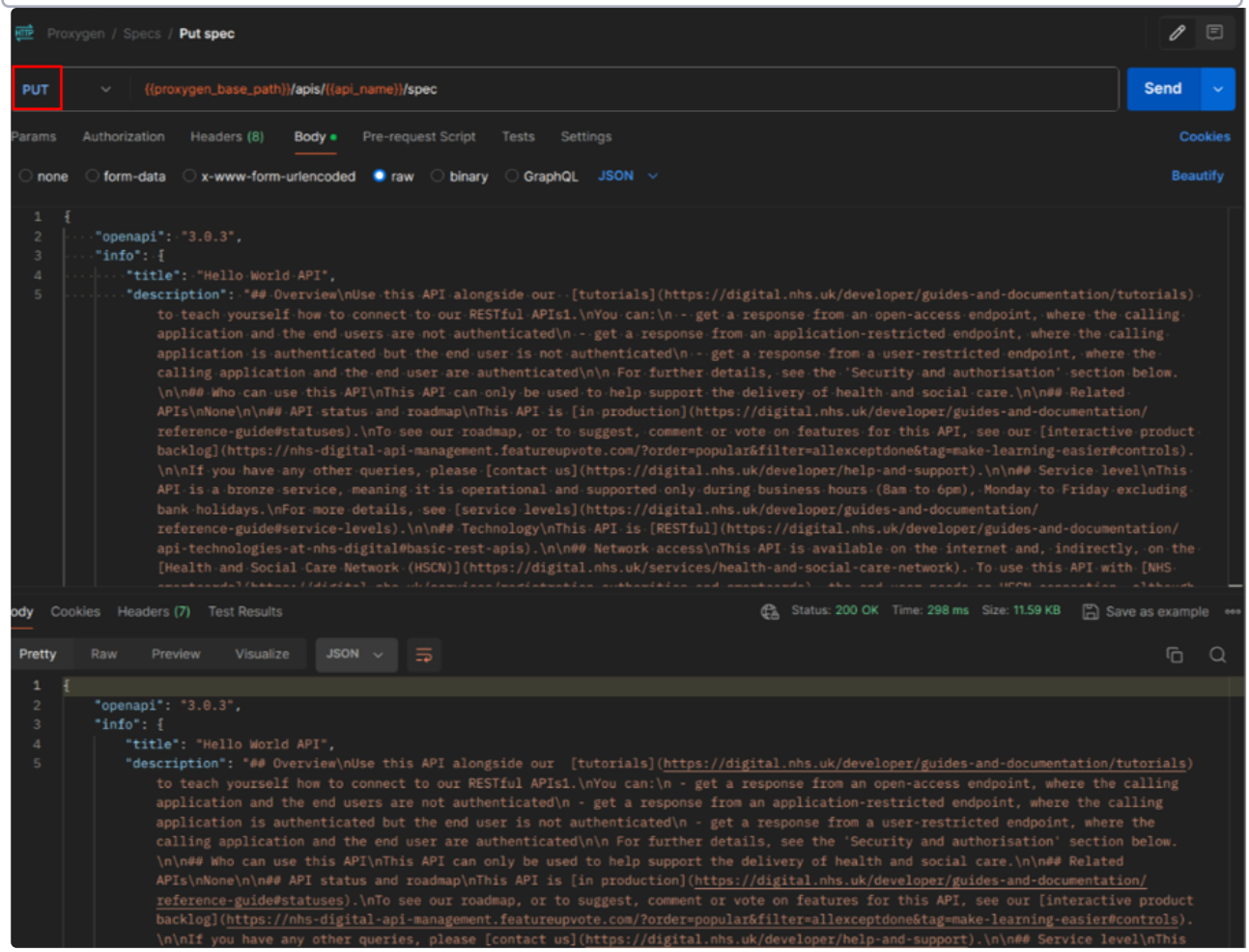


Fig 3: PUT specification endpoints - Proxygen Generator API

See the Proxy Generator OpenAPI specification for more information on schemas:

- Create/update production spec[525]
- Create/update UAT spec[526]

Get specification

Once your specification has been created, GET your OpenAPI Specification using the Proxy Generator API as follows:

525 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/spec
526 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/spec/uat

Get specification using the API

```
# Production
GET /apis/{api_name}/spec

# UAT
GET /apis/{api_name}/spec/uat
```

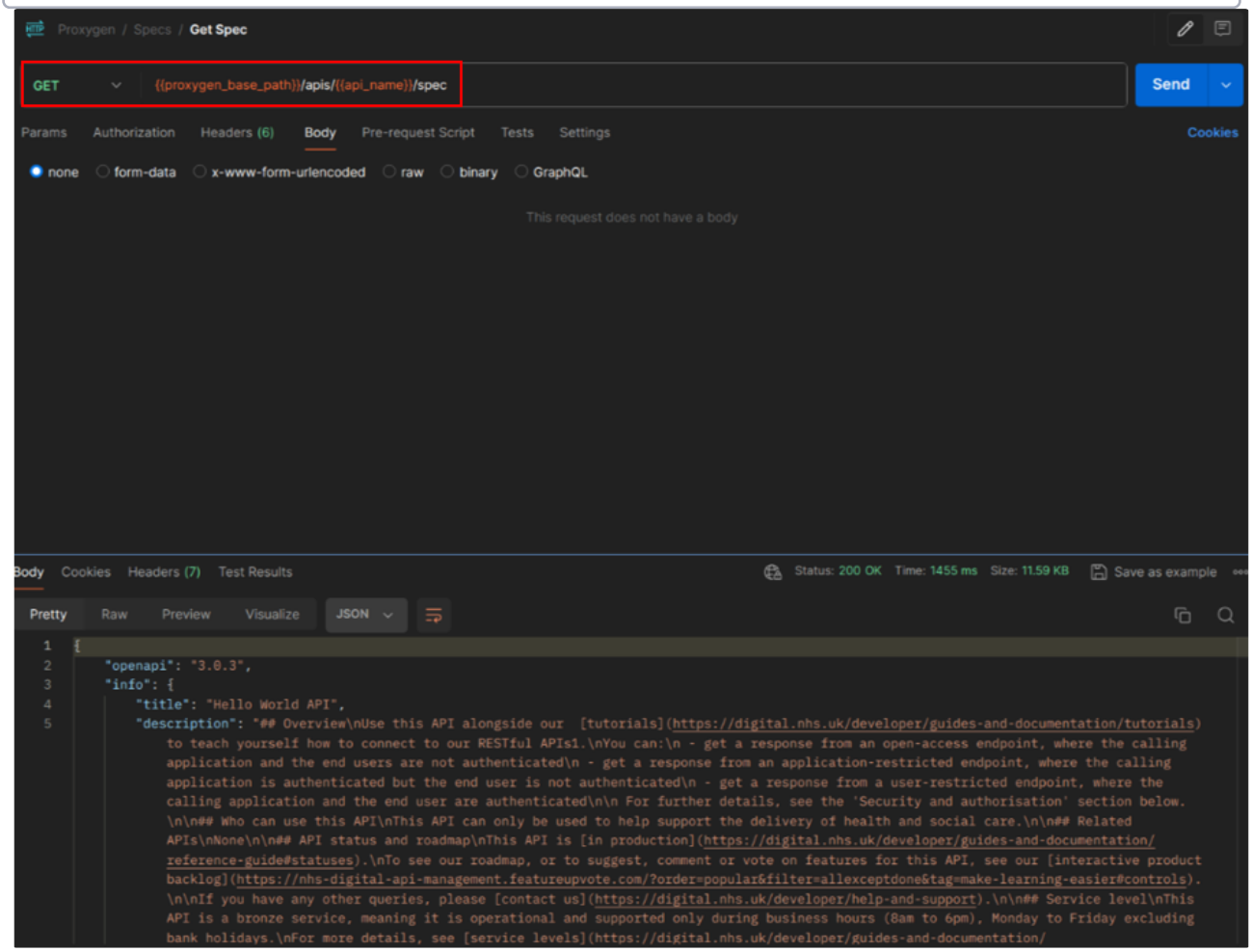


Fig 4: Get specification endpoints - Proxygen Generator API

See the Proxy Generator OpenAPI specification for more information on schemas:

- Retrieve production spec[527]
- Retrieve UAT spec[528]

Delete specification

527 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/spec
528 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/spec/uat

DELETE your OpenAPI Specification using the Proxy Generator API, as follows:

Delete specification using the API

```
# Production
DELETE /apis/{api_name}/spec

# UAT
DELETE /apis/{api_name}/spec/uat
```

Fig 5: Delete API specification endpoints - Proxygen Generator API

See the Proxy Generator OpenAPI specification for more information on schemas:

- Delete production spec[529]
- Delete UAT spec[530]

Publishing the spec using Proxygen-cli

Perform the following steps to publish the specification using Proxygen-cli:

1. Open the command line and run the following commands to publish the spec to production or UAT as follows:

Note: YAML or JSON can be used to publish specifications via the Proxygen-cli.

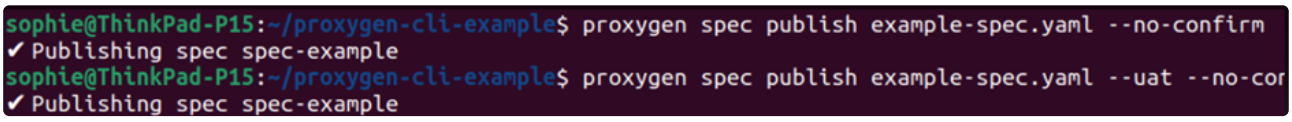


Fig 6: Proxyen-cli - publish specification commands

Note: The tick confirms the spec has successfully published.

Get specification

Once your specification has been created, GET your OpenAPI specification, as follows:

Get specification using Proxygen-cli

```
# Production
proxygen spec get

# UAT
proxygen spec get -- uat
```

529 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#delete-/apis/-api_name-/spec
530 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#delete-/apis/-api_name-/spec/uat

```
sophie@ThinkPad-P15:~/proxygen-cli-example$ proxygen spec get
components:
  securitySchemes:
    app-level0:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/app-level0
    app-level3:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/app-level3
    nhs-cis2-aal1:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/nhs-cis2-aal1
    nhs-cis2-aal3:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/nhs-cis2-aal3
    nhs-login-p0:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/nhs-login-p0
    nhs-login-p5:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/nhs-login-p5
    nhs-login-p9:
      $ref: https://proxygen.ptl.api.platform.nhs.uk/components/securitySchemes/nhs-login-p9
info:
  contact:
    email: api.management@nhs.net
    name: API Management Support
  description: |
    ## Overview
    Use this API alongside our  [tutorials](https://digital.nhs.uk/developer/guides-and-documentat
s) to teach yourself how to connect to our RESTful APIs1.
    You can:
     - get a response from an open-access endpoint, where the calling application and the end user
thenticated
     - get a response from an application-restricted endpoint, where the calling application is au
```

Fig 7: Proxygen-cli - ~Get specification commands

Delete specification

DELETE your OpenAPI Specification using the Proxygen-cli as follows:

Delete specification using proxygen-cli

```
# Production
proxygen spec delete --no-confirm

# UAT
proxygen spec delete --uat --no-confirm
```

Fig 8: Proxygen-cli - delete API specification commands

Publishing and viewing your specification on the UAT and production API catalogues

If you are publishing your specification for the first time then there are a few extra steps required, captured here (see page 522).

Once you have deployed your specification, Bloomreach is scheduled to render the page and publish the spec on the API and integration catalogue.

Scheduled jobs are deployed as follows:

- **Proxygen training & UAT:** Triggers every 10 minutes, starting on the hour, between midnight (12 AM) and 10 PM (22:00) every day
- **Proxygen production:** Triggers at 7:25 AM, 8:25 AM, and every hour on the 25th minute, up to 5:25 PM (17:25) on weekdays (Monday through Friday).

Before viewing your spec you must get the API platform team to configure Bloomreach. Post a query in the #platforms-apim-producer-support Slack channel (see page 2056) letting them know you're using Proxy Generator and provide the following details:

- your api_name (as used in the path when calling Proxy Generator) as your specification ID
- which Bloomreach environment you need the specification setting up in (UAT or production)

Once your specification has been set up by the API platform team you will be ready to view the specification in the API and integration catalogue.

Guidance for API platform team

To publish an API specification that uses Proxy Generator, see Creating an API specification page in production Bloomreach CMS. (see page 526)

Bear in mind, if deploying to UAT, the URL to access the Bloomreach CMS in UAT is https://cms-uat2.nhsd.io/cms/?0.

View your specification

UAT specifications must be viewed via the VPN, VDI or on the network in the NHS England office. Get set up with the VDI or VPN (see page 168) if you are not already set up.

Once your specification is published you can view it in the API catalogue:

- Production - https://digital.nhs.uk/developer/api-catalogue
- UAT - https://uat2.nhsd.io/developer/api-catalogue

7.14.5 Building your API alpha

Building your API alpha with proxy generator

If using proxy generator see Building and deploying your API alpha with Proxy Generator (see page 680)

7.14.5.1 Overview

This page describes the engineering tasks of building your API as part of the Alpha phase (see page 321) of the API process checklist (see page 198).

After building your API alpha you should have:

- a GitHub repository with a build, pull-request and release pipeline
- a sandbox and possibly an integration test environment
- an API proxy that is secured with error handling which connects to a backend
- an automated testing framework (optional)

7.14.5.2 Prerequisites

These prerequisite steps may be run in parallel to the building your API process steps.

- Learn about the different API deployment environments (see page 596)which are used for testing by external API Consumers.
- Your API has been designed as part of the Designing your API (see page 322) stage.

7.14.5.3 Process steps

1. Create your GitHub repository and deployment pipeline (see page 580).
2. Set up your API sandbox (see page 592)environment. This is a callable API that simulates the API's behaviour and can be used for testing.
3. Set up your other API path-to-live test environments (see page 596).
    - Connect your API proxy to a target server (see page 607). A target server is a backend national system, such as Spine or DPS. Once the proxy and target are linked requests can be sent through your proxy to the target server.
    - Once you have a target server connected you can setup lightweight data transformation or injection on your API proxy (see page 611). This is needed so that API Consumer requests can be converted into a format that your target backend requires.
    - Introduce error handling (see page 624) into your proxy. Errors may occur in your API, either from your proxy or the backend. Errors need to be handled so that the client receives an appropriate error message.
    - Do you need to setup a hosted container (see page 637)? See Transforming data on the APIM platform and containers (see page 371) for more information on whether this applicable for your API.
    - If you are connecting to a target endpoint that is asynchronous see - Setup Async to sync conversion (sync wrap) (see page 647) to build a frontend synchronous endpoint. This will simplify the interaction pattern for the API Consumer.

Once you have built your API and setup the environments, the next steps are:

- Securing your API (see page 774). Securing your API is vital and a requirement for the API Platform.
- Consider setting up automated testing for your API[531]. Testing is important to give you confidence that the API works as expected.

7.14.5.4 Creating a GitHub repository and deployment pipelines for your API

PLEASE NOTE

THIS METHOD IS NOW OBSELETE AND NO LONGER ADVISED.

PLEASE USE PROXYGEN TO CREATE YOUR API.

Follow Getting set up with proxy generator (see page 681).

7.14.5.4.1 Overview

Every API on the API platform has a GitHub repository and there is typically one repository per API proxy. By default, we code in the open - more on that can be found here[532].

That repository contains the source code and config for your:

- Proxy
- Sandbox
- Specification
- Deployment pipelines

There are three Azure DevOps Pipelines, which do the work of getting the contents of your repository onto Apigee (and AWS for hosted containers). These can be controlled by editing the contents of the corresponding pipeline YAML file. By default you have:

Pipeline	File path	Description
Build	azure/azure-build-pipeline.yml	Assembles the contents of your repository into a single file ("artifact") on Azure Devops and pushes any containers to our Docker registry

531 https://nhsd-confluence.digital.nhs.uk/display/APM/Creating+automatic+Apigee+test+applications
532 https://gds.blog.gov.uk/2012/10/12/coding-in-the-open/

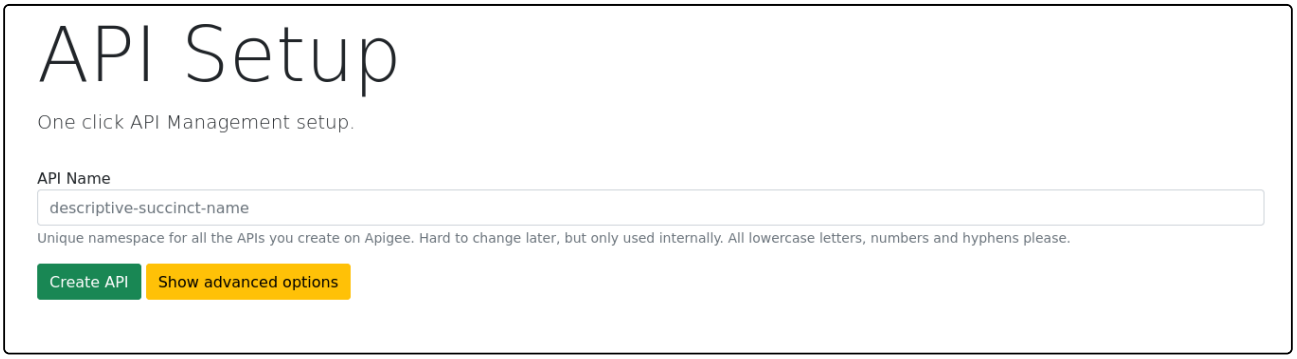
Pipeline	File path	Description
Pull-Request (PR)	azure/azure-pr-pipeline.yml	Deploys ephemeral versions of your proxy/ spec to Apigee (and docker containers on AWS) to internal environments. You can run automated and manual tests against these while you develop.
Release	azure/azure-release-pipeline.yml	Deploys the long-lived version of your pipeline to internal and external environments, typically when you merge to master.

7.14.5.4.2 Step 1: Set up your API

Go to https://setup-api.ptl.api.platform.nhs.uk[533]

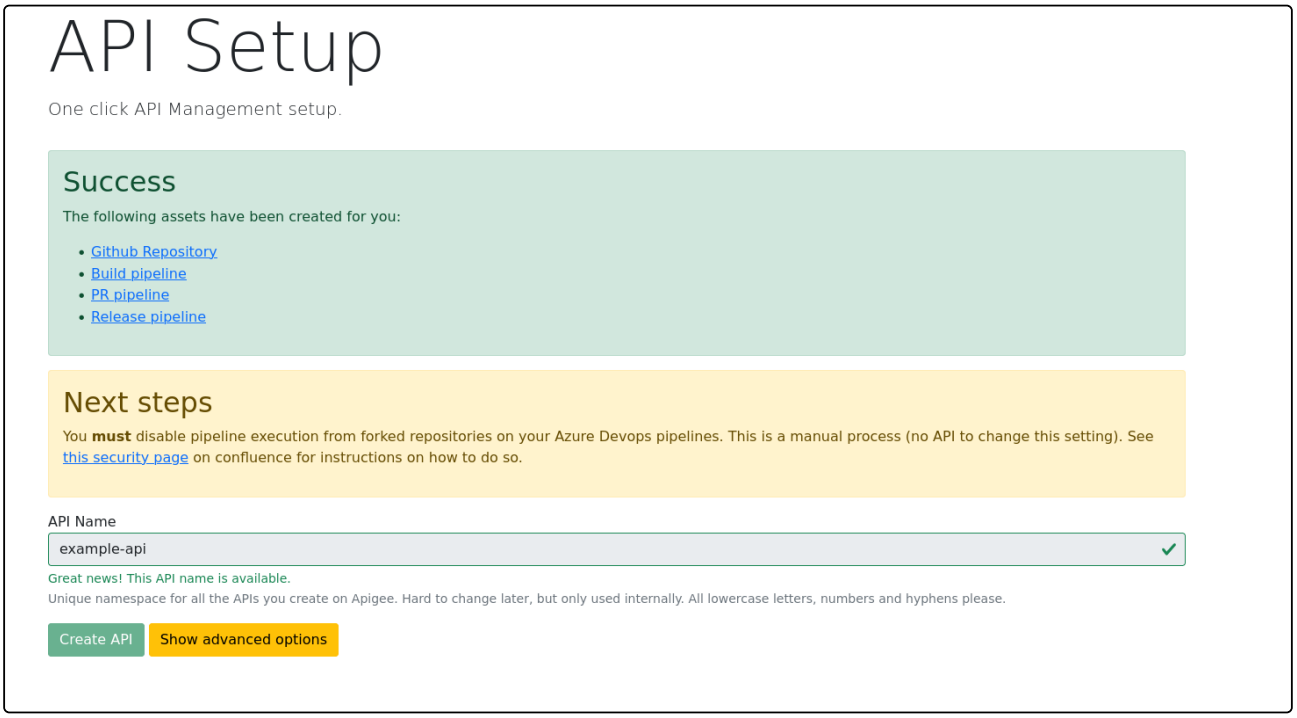
> **AD Group Membership Required**
>
> To access this page you must be a member of the **GRP-AzureNHSD-API-Management-PTL-DL** group. See Get access to Azure AD groups and Splunk (see page 161) for instructions.

Enter the name you would like for your API in the *API Name* box and click **Create API**. You can configure advanced options if needed but the defaults are sensible and in-line with our standard naming conventions.



Once the process is complete, you should see a dialog box telling you your GitHub repository has been created and some pipelines have been created in Azure DevOps. During this process your API you API has also been registered in our deployment database.

533 https://setup-api.ptl.api.platform.nhs.uk/



7.14.5.4.3 Step 2: Enabled YAML triggers and *Disable Forked Repository Builds* for all three your Azure DevOps pipelines

Security Issue

We control who can contribute to our Github repositories, however since we code in the open anyone can create a fork (copy) of our repositories on Github.
By default Azure Devops will execute pipelines from forked repositories as if they were just another branch on our repository.
The build agents which execute our pipelines have access to secret credentials to allow them to create/update/destroy resources on AWS and Apigee.
The nature of these pipelines is they allow for injection of arbitrary code.
Unless you disable this setting, a malicious actor could steal secrets and wreak havoc with the API Platform, disabling live production APIs, stealing personal data, or worse.
Therefore it is vital you do the following steps promptly after repository creation.

```
<rant> Azure Devops Pipelines claim this is a "feature" and does not provide us with a method to disable it by default, nor is it settable via an API call </rant>
```

7.14.5.4.3.1 Follow these steps:

> You will need to do the following steps for **each** of your created pipelines: **build**, **pull-request** and **release**. Reach out in the platforms-apim-producer-support[534] slack channel if you need help.

1. Navigate to your pipelines

Go to https://dev.azure.com/NHSD-APIM/API%20Platform/_build?view=folders and you should see a new folder matching your API name, containing a build, pull-request and release pipeline.

2. Send your pipelines to the support channel and ask them to disable forked repository builds

Copy the link to the folder with your **three** pipelines in and send it with this message to the platforms-apim-producer-support[535] slack channel. Wait for confirmation before moving to the next step.

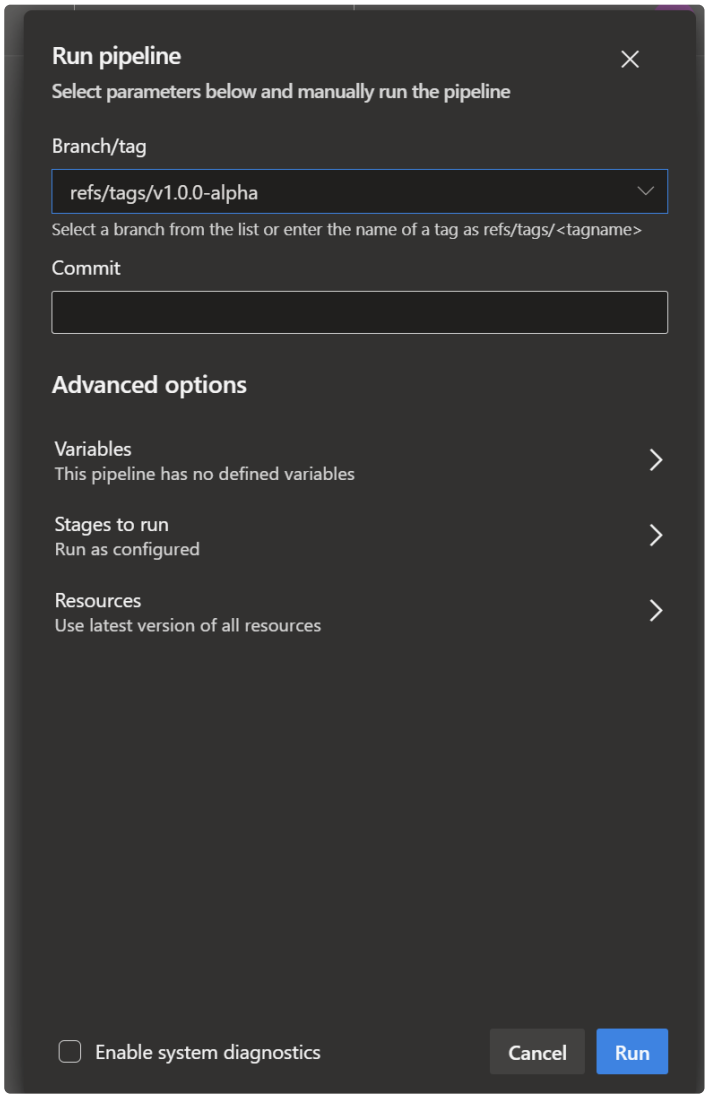
> Please disable forked repository build for my build, pr and release pipelines - <LINK_TO_PIPELINES> - following KOP-089 Disable Forked Repository Builds for Azure DevOps pipelines[536]

7.14.5.4.4 Step 3: Run the build pipeline

You can deploy the template under your own API's name by triggering the build pipeline. This should be run using the release tag created when your GitHub repo was first created. You can find this tag under the releases section on the home page of your repo. It will usually be `v1.0.0-alpha`.

To run the build pipeline click 'Run' then type `refs/tags/<your-tag>` in the Branch/tag section and once selected click 'Run' again.

534 https://nhsd-confluence.digital.nhs.uk/display/APM/Get+access+to+Slack
535 https://nhsd-confluence.digital.nhs.uk/display/APM/Get+access+to+Slack
536 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/608904157/KOP-089+Disable+Forked+Repository+Builds+for+Azure+DevOps+pipelines



7.14.5.4.5 Step 4: "Spec-only" API configuration (optionally)

In case you only need to publish documentation in our developer portal about an API standard, you can configure your API to be "Spec-only" by following the steps in Creating a "spec-only" API - API Management (see page 585).

7.14.5.4.6 Set-up complete

Your GitHub repository and API pipelines are now created and correctly configured.

We recommend you test your pipelines by creating a branch, making some changes and creating a pull request to merge those changes into master.

Opening the pull request should trigger the Build and Pull Request pipelines, merging should trigger the Release pipelines
. If you have any issues with the pipelines not automatically triggering, reach out in the platforms-apim-producer-support[537] channel on Slack and ask them to follow KOP-013 Fixing pipeline triggers[538].

A useful first change

Consider first updating your API specification. This file name depends on what you called your API during the set-up.

For the example shown here "example-api", the specification file would be `specification/example-api.yaml`

7.14.5.4.7 Creating a spec-only API

7.14.5.4.7.1 Introduction

When an API created and released in the platform the spec associated with your API will be deployed onto Apigee along with your proxy. Later on the spec will be consumed by our CMS stack and the spec will be used to autogenerate the documentation associated with your API and optionally publish it onto the API catalogue (API catalogue - NHS Digital[539]). For a detailed explanation about how specifications are published please refer to Documenting your API (see page 445).

APIM will automate the creation of the documentation because:

- It saves a lot of time and resources.
- We believe in the "spec-first" approach when designing and building an API and the OpenAPI Specification as a centralised source of truth.
- Making changes to the specification is controlled by API producers.

7.14.5.4.7.2 What is a spec-only API?

The primary reason a producer will deploy only the specification for an API in the catalogue is because we publish API specification standards between existing APIs.

When creating an API using the method described here (see page 580)and releasing it to one or more environments, our CI/CD pipelines will validate the API and deploy several components, usually:

- An Apigee expanded version of the specification.
- An Apigee proxy.
- One or more Apigee products
- Optionally a hosted target in AWS (for more info visit AWS Hosted Containers Architecture[540])

537 https://nhsd-confluence.digital.nhs.uk/display/APM/Get+access+to+Slack
538 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125414244/KOP-013+Fixing+pipeline+triggers
539 https://digital.nhs.uk/developer/api-catalogue
540 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153364474/AWS+Hosted+Containers+Architecture

In APIM we call a "spec-only API" to an API configured in such a way that no proxy, products nor hosted targets are deployed but only the specification.

7.14.5.4.7.3 Configuration

Once you have created the API following the instructions from Creating a GitHub repository and deployment pipelines for your API (see page 580), you will be ready to configure your repository. You can configure your API to be "spec-only" by:

1. Modifying the ***manfiest_template.yml*** file
2. Pruning the ecs proxy files
3. Modifying the azure pr/release pipeline files
4. Removing any unnecessary code

The following examples corresponds to the "eyecare-e-referrals-service" spec-only API here[541].

Modifying the manifest_template.yml

Follow the below configuration as an example on the changes requires to make to the ***manifest_template.yml*** file.

manifest_template.yml

```
meta:
  api:
    name: eyecare-e-referrals-service # The name of your api created by the API set-
up process
    guid: 9bf88681-af26-4d52-9327-7fa172eb11fe # The guid and spec guid created by
the same process
    spec_guids:
      - bf6758a5-b731-4640-876a-f7ad56e36029
  schema_version: 1
apigee:
  environments: # The two environments for created a test spec (internal-dev) and a
public spec (int)
  - name: internal-dev
    specs:
      - name: eyecare-e-referrals-service-internal-dev
        path: eyecare-e-referrals-service.json
```

541 https://github.com/NHSDigital/eyecare-e-referral-service-api/blob/master/manifest.yml

```
  - name: int
    specs:
      - name: eyecare-e-referrals-service-int
        path: eyecare-e-referrals-service.json
```

In this case we can use the manifest file to deploy the spec called "eyecare-e-referrals-service.json" onto the Apigee environments "internal-dev" and "int" respectively. Since we are only deploying the specification no other entry is needed in the manifest file definition.

When using the manifest file please make sure that the environments listed match the ones listed under azure-pr-pipeline.yml[542] and azure-release-pipeline.yml[543].

Please refer to Manifest.yml reference (see page 896) for a detailed explanation on how to use the manifest file definitions to control your API deployments.

Pruning the ECS proxy files

We now need to delete the ecs proxy files. These files are:

- ***ecs-procies-containers.yml***
- ***ecs-proxies-deploy.yml***

Modifying the Azure pr/release pipeline files

Naturally, azure will attempt to look for a proxy to build which will cause the spec-only builds to fail. As a result, we need to modify the ***azure-pr-pipeline.yml*** and ***azure-release-pipeline,yml*** files.

For ***azure-pr-pipeline.yml*** we are simply adding ***proxy_path: "*** under the parameters of our spec-only config. Below is an example of how the file should look.

azure-pr-pipeline.yml

```
name: "$(SourceBranchName)+$(BuildID)"

trigger: none
pr: none

resources:
  repositories:
    - repository: common
      type: github
      name: NHSDigital/api-management-utils
      ref: refs/heads/edge
```

542 https://github.com/NHSDigital/eyecare-e-referral-service-api/blob/master/azure/azure-pr-pipeline.yml
543 https://github.com/NHSDigital/eyecare-e-referral-service-api/blob/master/azure/azure-release-pipeline.yml

```
      endpoint: NHSDigital
  pipelines:
  - pipeline: build_pipeline
    source: "Eyecare-Referrals-Build"
    trigger:
      branches:
        exclude:
          - master
          - refs/heads/master
          - refs/tags/v*

pool:
  name: 'AWS-ECS'

variables:
  - template: project.yml

extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    proxy_path: ''
    apigee_deployments:
      - environment: internal-dev
```

For ***azure-release-pipeline.yml***, we need to make the same change we made for the previous azure-pipeline file. However, we also need to remove some config relating to apigee_deployments. We want to only specify the environment. An example of how the file should look like can be seen below.

azure-release-pipeline.yml

```
name: "$(SourceBranchName)+$(BuildID)"

trigger: none
pr: none

resources:
  repositories:
    - repository: common
      type: github
      name: NHSDigital/api-management-utils
      ref: refs/heads/edge
      endpoint: NHSDigital
  pipelines:
  - pipeline: build_pipeline
    source: "Eyecare-Referrals-Build"
    trigger:
      branches:
```

```
        include:
          - refs/tags/v*

pool:
  name: 'AWS-ECS'

variables:
  - template: project.yml

extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    proxy_path: ''
    apigee_deployments:
      - environment: internal-dev
      - environment: internal-dev-sandbox
      - environment: internal-qa
      - environment: internal-qa-sandbox
      - environment: ref
        depends_on:
          - internal_qa
          - internal_qa_sandbox
      - environment: sandbox
      - environment: int
        depends_on:
          - internal_qa
          - internal_qa_sandbox
```

Removing any unneccesary code

Finally, you can clean up unnecessary code in your repo by deleting the ***proxy***, ***sandbox*** and ***tests*** folders in your repo.

7.14.5.4.8 One Click Tooling - Service Template Repo

7.14.5.4.8.1 Overview

This page details the service template repo and what comes as standard within the repo.

7.14.5.4.8.2 What you get

/.github

This folder contains templates that can be customised for items, such as opening pull requests or issues within the repo.

/.github/workflows

This folder contains templates for github action workflows such as:

- **pr-lint.yaml**: this workflow template shows how to link Pull Request's to Jira tickets and runs when a pull request is opened.
- **continuous-integration.yml**: this workflow template shows how to publish a Github release when pushing to master.

/azure

This folder contains templates defining Azure Devops pipelines. By default, the following pipelines are available:

- **azure-build-pipeline.yml**: assembles the contents of your repository into a single file ("artifact") on Azure Devops and pushes any containers to our Docker registry. By default this pipeline is enabled for all branches.
- **azure-pr-pipeline.yml**: deploys ephemeral versions of your proxy/spec to Apigee (and docker containers on AWS) to internal environments. You can run automated and manual tests against these while you develop. By default, this pipeline will deploy to internal-dev, but the template can be amended to add other environments as required.
- **azure-release-pipeline.yml**: deploys the long-lived version of your pipeline to internal and external environments, typically when you merge to master.

/azure/templates

Here you can define reusable actions, such as running tests, and call these actions during Azure Devops pipelines.

The tests include:

- setting up pytests.
- running & publishing the full test suite.
- runnning & publishing smoketests.

/proxies

This folder contains files relating to your Apigee API proxy.

There are 2 folders, /live and /sandbox, allowing you to define a different proxy for sandbox use. By default, this sandbox proxy is implemented to route to the sandbox target server (code for this sandbox is found under /sandbox of this template repo).

- **/proxies/default.xml**: defines the proxy's Flows. Flows define how the proxy should handle different requests. By default, _ping and _status endpoint flows are defined. See the APM confluence for more information on how the _ping (see page 1073) and _status (see page 1075) endpoints work.
- **/policies**: populated with a set of standard XML Apigee policies that can be used in flows.

- **/resources/jsc**: snippets of javascript code that are used in Apigee Javascript policies. For more info about Javascript policies see here[544].
- **/targets**: the XMLs within these folders set up target definitions which allow connections to external target servers. The sandbox target definition is implemented to route to the sandbox target server (code for this sandbox is found under /sandbox of this template repo). For more info on setting up a target server see Setting up a target server (see page 607).

/sandbox

This folder contains a template for a sandbox API. This example is a NodeJs application running in Docker. The application handles a few simple endpoints such as: /_ping, /health, /_status, /hello and some logging logic. For more information about building sandbox APIs see Setting up your API sandbox (see page 592).

/scripts

This folder contains useful scripts that are used throughout the project, for example in Makefile and Github workflows. The scripts include:

- **build_proxy.sh**: collect the API proxy and hosted target and deploy to Apigee.
- **calculate_version.py**: calculates a semver-compliant version string from a git repository's commit messages, using commands embedded in commit messages.
- **pre-commit**: checks for any linting errors before code is commited.
- **set_version.py**: reads an openapi spec on stdin and adds the calculated version to it then prints it on stdout.
- **yaml2json.py**: takes yaml on stdin and writes json on stdout, converting dates correctly.

/specification

Creates an OpenAPI Specification to document your API. For more information about developing specifications see Documenting your API (see page 445).

/tests

End to end tests. These tests are written in Python and use the PyTest test runner. The test_endpoint.py file provides a template of how to set up tests, which test your api endpoints. For more information about testing your API see Testing your API (see page 715). The test scenarios are detailed below:

- **test_ping**
- **test_wait_for_ping**
- **test_status**
- **test_wait_for_status**
- **test_app_level0**
- **test_app_level3**
- **test_cis2_aal3**

544 https://docs.apigee.com/api-platform/reference/policies/javascript-policy

Makefile

Contains useful make commands to get you started including; installing dependencies and running smoke tests.

ecs-proxies-containers.yml and ecs-proxies-deploy.yml

These files are required to deploy containers alongside your Apigee proxy during the Azure Devops azure-build-pipeline. In this template repo, we are deploying our sandbox container which is used as the target server for the sandbox proxy.

ecs-proxies-containers.yml

The path to a container's Dockerfile is defined here. This path needs to be defined to allow containers to be pushed to our repository during the azure-build-pipeline.

ecs-proxies-deploy.yml

Here you can define config for container deployment. For more information about deploying ECS containers see Building and pushing docker containers (see page 641).

manifest_template.yml

This file defines 2 dictionaries of fields that are required for the Apigee deployment. For more info see Manifest.yml reference (see page 896).

7.14.5.5 Setting up your API sandbox

7.14.5.5.1 Overview

A sandbox API provides a version of your API with pre-canned responses, it is really important to keep in mind the following principles:

- It is for early consumer testing - it allows them to do initial discovery about your API
- Easy to use
- Callable from using the "Try this API" feature
- Is open access - has no authorisation (unlike the integration test environment) - that's on purpose to make it easy to use
- It is stateless, so it does not actually persist any updates
- It only covers a reduced set of scenarios
- Simple and quick to build

What does this mean in practice when building a sandbox?

- An endpoint might have 5 parameters (which results in many variants), but the sandbox should only look to implement the 2 or 3 common variants

- The mandatory nature of a parameter does not need to be enforced, this is especially important where the parameters are complex
- The "try this now" functionality should have defaults, and this is really important if you choose to keep parameters mandatory
- The sandbox **is not intended to help with Solutions Assurance**

We strongly recommend building a sandbox API - in user testing consumers really like them - it makes the whole learning/early discovery process feel easier (even if the sandbox actually only delivers the same responses that are in your API specification).

7.14.5.5.1.1 Self service

Your sandbox API needs to be self-service - consumers should not have to request access to try out the sandbox.

7.14.5.5.1.2 Building a sandbox API

How you build your sandbox API is up to you. We do not dictate the language or functionality but recommend it reflects some of the basic functionality of your API, to allow discovery for potential consumers.

If you want to build your sandbox from scratch, there are tools out there that convert an OAS file into a mock server. We've not found one that works for every API. We tend to use the OpenAPI tool for one off conversions to create a mock server.

If you have used the api-management-service-template[545] to create your GitHub repo there will be a sandbox template[546] created, which you can use as a starting point. This is a NodeJS app inside a Docker container with a Dockerfile[547] and health checks[548] which will be useful when deploying the API.

Really keep in mind the principles above, it is very easy to allow scope creep. The main area is to keep the sandbox to a reduced set of scenarios - these are implemented as stubs that return static data to simulate a happy path interaction and simulate a few common failure conditions e.g. 404 resource not found or 400 bad request (for invalid or missing parameters or headers).

*Keep it **light weight** - don't over complicate the Sandbox. It is easy to keep adding to it until you have nearly replicated your entire API* 🙂

Adding your endpoints to the sandbox NodeJS app

The template sandbox app will come with some default endpoints eg: `/_ping`, `/_status`, `/health`, `/hello`. To use the "Try this API" feature, you will need to a selection of your endpoints to the app, so that developers can sample the basic functionality of your API.

545 https://github.com/NHSDigital/api-management-service-template
546 https://github.com/NHSDigital/api-management-service-template/tree/master/sandbox
547 https://github.com/NHSDigital/api-management-service-template/blob/master/sandbox/Dockerfile
548 https://github.com/NHSDigital/api-management-service-template/blob/master/sandbox/app.js

You can achieve this by declaring a function for each endpoint in sandbox/handlers.js, which serves the appropriate JSON response and adding the reference to this at the bottom of the sandbox/app.js to route calls to this function. Example: `app.all("/hello", handlers.hello);`

Recommended sandbox features

When building your sandbox, then consider the following features:

- **Core functionality** - ensuring your sandbox supports at least one call for every endpoint, to help show potential consumers what they can expect from using your API.
- **Errors** - Understanding error handling at the discovery phase can also be very useful for potential consumers. We recommend you implement an easy way for the user to trigger canned errors, so they can test their error handling.
- **Health checks** - Monitoring the sandbox with health checks is important for ensuring availability for consumers trying the API (and also mimics what would be required when the API is live). See Adding status monitoring to a service (see page 1065) for how to implement.
- **Logging** - As above, good logging is useful for maintaining the availability of the sandbox and debugging potential issues.

Apart the "core functionality" all the items are in the template NodeJS app.

7.14.5.5.1.3 Deploying a sandbox API

You can create the mock backend either by using the template NodeJS app that is included in the repo or by using the stack you like the most.

Deploying the template NodeJS app

By default, the sandbox environments are not deployed. To enable them, uncomment the corresponding config lines in the following files in your repo:

Environment	manifest_template.yml *	azure/azure-pr-pipeline.yml**	azure/azure-release-pipeline.yml**
Sandbox	- display_name: Sandbox name: sandbox		- environment: sandbox proxy_path: sandbox
Internal Dev Sandbox	- display_name: Internal Development Sandbox name: internal-dev-sandbox	- environment: internal-dev-sandbox proxy_path: sandbox	- environment: internal-dev-sandbox proxy_path: sandbox

Environment	manifest_template.yml *	azure/azure-pr-pipeline.yml**	azure/azure-release-pipeline.yml**
Internal QA Sandbox	- display_name: Internal QA Sandbox name: internal-qa-sandbox		- environment: internal-qa-sandbox proxy_path: sandbox
Internal QA***	- name: internal-qa display_name: Internal QA has_mock_auth: true product_visibility: private		- environment: internal-qa post_deploy: - template: ./templates/run-tests.yml parameters: full: true

***** uncomment lines under `APIGEE_ENVIRONMENTS`

****** uncomment lines under under `apigee_deployments` (near the bottom of the file)

******* the Sandbox environment is dependent on the Internal QA environment, hence why this environment also needs to be enabled

The sandbox environment deploys a hosted container behind the proxy (also called ECS proxy). The "try this API" connects to the sandbox environment.

Deploying using your own Docker image with APIM Hosting

If you have built your own sandbox server, you can use APIM to host that behind your sandbox, by following the steps detailed via link below:

- Creating a GitHub repository and deployment pipelines for your API (see page 580)
- Create a branch in the repo
- Replace contents of the sandbox folder, with your own Dockerized sandbox, and Dockerfile;
- Un-comment the lines related to environment **internal-dev-sandbox**, in the following two files:
  - azure/azure-pr-pipeline.yml
  - azure/azure-release-pipeline.yml
- Change the path of the Dockerfile, if needed in file: **ecs-proxies-containers.yml** (optional)
- Add your sandbox environment variables, as needed in file: **ecs-proxies-deploy.yml**
- Enable deploy environments for **internal-dev-sandbox**, around line 12, in the file: **manifest_template.yml**
- Create a pull request in order for the sandbox to be deployed in ECR and APIGEE

APIM does **not** currently support you deploying from an image an arbitary ECR container.

For more info about how to develop an APIM AWS Hosted Container please read Developing ECS proxies (see page 638).

Hosting and connecting to your own sandbox server

If you have not used any of the options above, you can still get the you OAS "Try this API" to use your own sandbox server.

You can follow steps described via link below but with the proxies/sandbox folder in your repo when adding the target server definition:

- Setting up a target server (see page 607)

7.14.5.5.1.4 Standard test data for sandbox use

We have defined standard test data to use in sandboxes - see Standard test data (see page 606).

7.14.5.5.2 Sandbox environment domain

Sandbox	sandbox.api.service.nhs.uk[549]

For details such as static IPs - see Environments (see page 758).

7.14.5.5.3 API subscription

By default, for a **sandbox** environment, there is no need for developers to subscribe at all, the API does not require authentication.

To hide the sandbox environment from the "Add APIs" list in DOS, ensure the `products.attributes.access` field is set to private for the sandbox environment in the `manifest_template.yml` file in the root of your repo.

7.14.5.6 Path to live environments

7.14.5.6.1 Overview

API consumers need to test their software before it goes live. To help them do this you need to provide externally-facing "path-to-live" environments for our APIs.

(these are separate from your own internal path-to-live environments - that you need for your own testing)

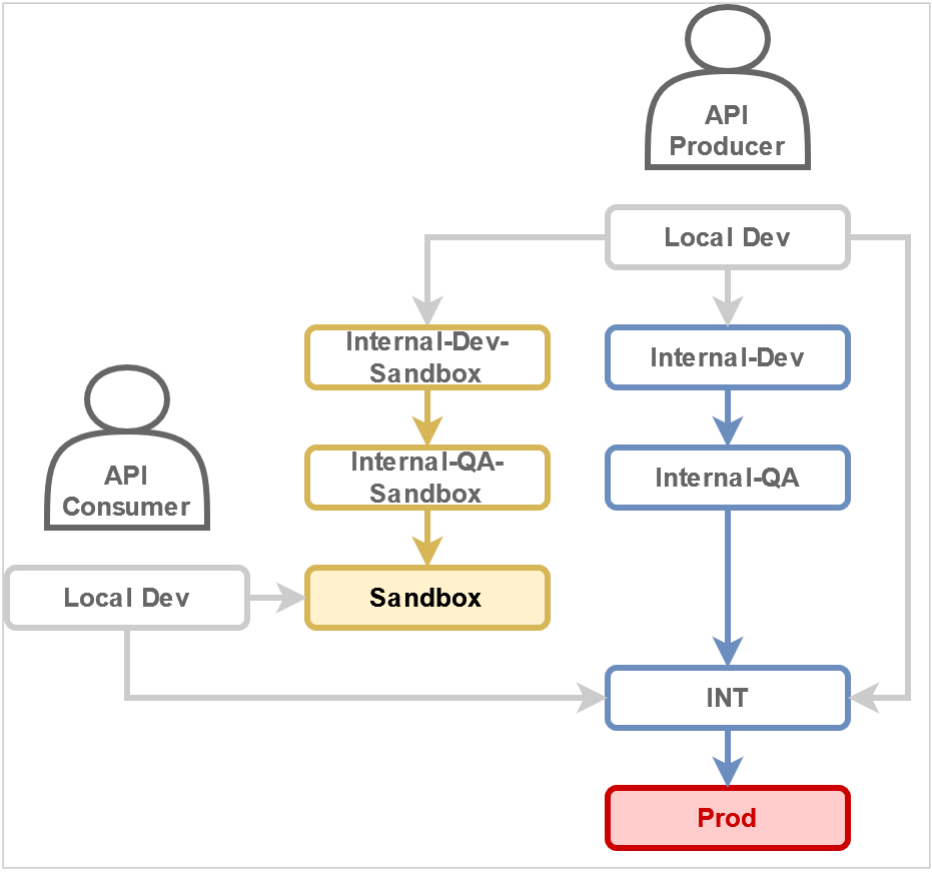
What environments you provide depends on your API, but a common pattern is:

549 http://sandbox.api.service.nhs.uk

- Sandbox environment (see page 598) - a simulation of the API that returns a limited number of canned responses. Generally stateless. Usually hosted in our AWS ECS environment.
- Integration testing environment (see page 598) - a more realistic environment for more formal testing. Generally stateful. For Spine APIs this would connect to the Spine INT environment.

At the moment our policy is **not** to offer further environments - we think it confuses consumers and is unnecessary.

7.14.5.6.2 Environments



Since each environment pulls from the same repo, you may want to add in some per-environment configuration. You can find out more about how to do this in the Per-environment configuration (see page 600) page.

For more details on all environments, such as static IPs and domains - see Environments (see page 758).

7.14.5.6.2.1 Internal PTL environments

Internal dev environment

Internal development of APIs that connect to a "real" back end. This is for development and manual testing. Developers and testers may do whatever they want within this envrionment.

Internal qa environment

Internal testing of APIs that connect to a "real" back end. This is a clean environment for testing deploys.

7.14.5.6.2.2 External PTL environments

Sandbox environment

A sandbox API is a version of your API that s can use for early testing. It allows API consumers to try the API out and discover what shape the request and response needs to be. It should be easy to use, so needs to be open access (**no** authorisation).

We strongly recommend building a sandbox API - in user testing consumers really like them - it makes the whole learning/early discovery process feel easier (even if the sandbox actually only delivers the same responses that are in your API specification).

See Setting up your API sandbox (see page 592) to learn more about sandbox environments.

Integration testing environment

This is where external consumers will do formal integration testing of their software with your API.

It needs to be a pretty good likeness to production, including the security / authorisation pattern for your API.

It will generally be a full fat test environment which uses the same code base as your production environment (as opposed to the sandbox which is just a stub).

And if your API is stateful you're likely to need to think about test data.

Self service

Your integration test API should **ideally** be self-service - consumers should not have to request access to use it.

For APIs that connect through to Spine, that means you might need some code in your proxy to include a dummy ASID in the call through to Spine. For an example of how this works in your proxy, see the PDS FHIR API[550]. For more complex Spine based APIs, the higher level processes are being captured here: Spine based APIs (see page 408). If you need help, contact us.

In some cases, you might want your integration test environment to be "gated" i.e. consumers need to request access. For example, the NHS App API connects to a test environment that, for whatever reason, needs to be controlled in terms of who uses it. This is possible, Apigee does support it out of the box. Contact us and state your case and we'll work with you to set it up.

Deploying an API to the integration testing environment

550 https://github.com/NHSDigital/personal-demographics-service-api/blob/be0ea2aabe28e3786cc82b8f4705ad8da0ee895f/proxies/live/apiproxy/policies/AssignMessage.PopulateAsidFromApp.xml

In order to be able to test your API within the INT environment:

- It must be deployed to that environment
- Spine based APIs need to consider ASIDs and at the right stage update the APIM Generic ASIDs to support the API - see Spine based APIs (see page 408)

See Deploying your API (see page 882) for more information on deploying an API.

Test data

You need to think about what test data consumers will need for formal integration testing.

For testing queries (e.g. searches or retrievals), you could provide standard test data for all API consumers to use. This makes testing easier for API consumers.

If you do this, make sure you align your test data with the test data we use for other APIs. For details, see Standard test data (see page 606).

For testing updates you need to make sure consumers can work independently of one another, so they would need their own private test data, and some way to set it up.

For an example of how to do this, see the PDS FHIR API:

- https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
- https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir/pds-fhir-api-test-data

For Spine path-to-live environments, the Spine Test Data team are available to help with this (testdata@nhs.net[551]).

7.14.5.6.3 API consumers and app setup in the developer portal

To transition from INT to Production, at this stage of the programme, we require API Consumers (and any API Producer test teams) to setup a new App or Apps on Production. This is for a number of reasons:

- To provide an extra layer of protection for Production - that you can't accidently connect your INT environment to Production
- This is achieved because you will have to issue new Client IDs and Secrets against that new App
- Where you use a callback url, this will be different between INT & Production
- If you have JWKS endpoints, that is likely to be different between environments
- For integrations with Spine, different ASIDs will need to be setup

7.14.5.6.4 Non-functional testing

We currently don't have an environment for external consumers to do non-functional testing against our APIs e.g. performance testing.

551 mailto:testdata@nhs.net

7.14.5.6.5 Per-environment configuration

7.14.5.6.5.1 Using the Clear and Secret common config stores

You can pass per-environment config to your application using the clear and secret config stores (implemented as Apigee KVMs), which are managed through the api-management-infrastructure repository[552]. The configuration needs to be a valid JSON string and its values are unpacked and assigned to context variables inside your proxy.

Adding clear config values

Adding a "clear" configuration means that the configuration values are going to be visible in the Apigee UI and also in the deployment pipeline.

Please follow the following steps to add per-environment "clear" config values for your proxy:

- Add the configuration as a valid JSON string under the relevant environment under the "clear-caribou" object in the infrastructure repo here[553].
    - Example: adding config for internal-dev for the proxy named "my-proxy-name"

my-proxy-config

```
environment_kvmaps:
  clear-caribou:
    encrypted: false
    nhsd-nonprod:
      internal-dev:
        {{ API-NAME }}: # This should match the meta/name field in your manifest/
manifest_template.yml
            key1: "value1"
            key2: "value2"
```

- Deploy the changes into Apigee through an infrastructure release.
    - Open a pull request in the infrastructure repo and get in contact with a member of the API Platform team for a code review and a release of Apigee config.

Tip

552 https://github.com/NHSDigital/api-management-infrastructure
553 https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-key-value-maps/vars/main/key_value_maps.yml#L25

Instead of defining the configuration "in-line" like the example shows try creating a new file holding the variables specific to your proxy. Check this example here[554].

Adding secret config values

Adding a "secret" configuration means that the values are **NOT** going to be visible in the Apigee UI or the deployment pipeline.

Please follow the following steps to add per-environment "secret" config values for your proxy:

- Make sure the secret value you want to consume in your configuration exists in Secrets Manager (AWS) and if it does not please get in contact with a member of the API Platform team and ask them to add the secret.
- If decided to use a value retrieved from Secrets Manager then you need to specify the secret_id key.
- If decided to use a raw value (a value that is public in the repo but you want to be secret in the UI and deployment pipeline) then use the raw_value key.
- Add the configuration as a valid JSON string under the relevant environment under the "secret-squirrel" object in the infrastructure repo here[555].
    - Example: adding secret config for internal-dev for the proxy named "my-proxy-name"

my-proxy-config

```
environment_kvmaps:
  secret-squirrel:
    encrypted: true
    nhsd-nonprod:
      internal-dev:
        {{ API-NAME }}: # This should match the meta/name field in your manifest/
manifest_template.yml
            secret_id: "ptl/my_proxy_name/my_secret_json"
```

- Deploy the changes into Apigee through an infrastructure release.
    - Open a pull request in the infrastructure repo and get in contact with a member of the API Platform team for a code review and a release of Apigee config.

Caution

554 https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-key-value-maps/vars/main/identity-service-variables.yml#L19

555 https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-key-value-maps/vars/main/key_value_maps.yml#L57

Remember to deploy the configuration in all environments. If this is not done the proxy will simply fail in the environment with the missing configuration simply because the values will not exist in the context of the proxy.

Using config values in your proxy

Since 9th Sept 2022 this proxy step / policy has been included in the template repository. If you created your repository more recently than that, you can skip this step!

To retrieve a value from your key value map, you'll need to add a series of policies in your API proxy. In the following example, we will retrieve the "clear" config for the "my-proxy-name" proxy.

Please follow the steps:

- Add the following steps to the proxy pre-flow under the request tag.

default.xml

```
<PreFlow name="PreFlow">
    <Request>
        <Step>
            <Name>FlowCallout.GetEnvironmentKeyValueMapData</Name> ## Fetches the
config object from "secret-squirrel" or "clear-caribou" as needed.
        </Step>
    </Request>
</PreFlow>
```

- Ensure the following FlowCallout policy is included in your policies

policies/FlowCallout.GetEnvironmentKeyValueMapData.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.G
etEnvironmentKeyValueMapData">
  <DisplayName>FlowCallout.GetEnvironmentKeyValueMapData</DisplayName>
  <Parameters>
    <!-- {{ SERVICE_NAME }} should match the meta/name field in your
manifest_template.yml file. -->
    <Parameter name="service_name">{{ SERVICE_NAME }}</Parameter>
  </Parameters>
```

```
  <SharedFlowBundle>GetEnvironmentKeyValueMapData</SharedFlowBundle>
</FlowCallout>
```

This policy calls a SharedFlow which grabs the data from clear-carribou/secret-squirrel key value maps, and unwraps the JSON key-value-pairs into individual variables.

Key Value Map	Encrypted?	Apigee Proxy Variable	Masked in Apigee Trace
clear-carribou	No	environment_kvm_data	false
secret-squirrel	Yes	private.environment_kvm_data	true

The value can be accessed in javascript policies[556] or in any XML policy that allows message templates[557]. See next section for a worked example.

Worked example: gp-registrations-management-information

> *Problem:*
>
> *An API key is used to secure Apigee's access to the backend (in addition to IP restrictions).*
> *The value is different per-environment.*
> *The value must not be in the clear in any repositories.*
>
> *Solution:*
>
> *Encrypted Key Value Map entry*

Secret uploaded to AWS secretsmanager

A JSON formatted secret was uploaded to secretsmanager with the secret-id: `ptl/gp-registrations-management-information/secret-squirrel/internal-dev` The secret was JSON formatted with the following format.

556 https://docs.apigee.com/api-platform/reference/policies/javascript-policy
557 https://docs.apigee.com/api-platform/reference/message-template-intro#using-message-template-functions

secret content

```
{"apikey": "<my-secret-backend-api-key>"}
```

Key Value Map yaml file in infrastructure-repository

The secret entry was mapped from secretsmanager into the `secret-squirrel` encrypted KeyValueMap by updating the infrastructure repository.

```
  secret-squirrel:
    encrypted: true
    nhsd-nonprod:
      internal-dev:
        gp-registrations-management-information:
          secret_id: "ptl/gp-registrations-management-information/secret-squirrel/
internal-dev"
```

pro tip

Don't forget to ensure this is deployed to all required environments once merged

3. GetEnvironmentKeyValueMapData shared flow called in proxy PreFlow

proxy/default.xml

```
<ProxyEndpoint name="default">
  <PreFlow name="PreFlow">
    <Request>
      <Step>
        <Name>FlowCallout.GetEnvironmentKeyValueMapData</Name>
      </Step>
    </Request>
  </PreFlow>
```

policies/FlowCallout.GetEnvironmentKeyValueMapData.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.G
etEnvironmentKeyValueMapData">
  <DisplayName>FlowCallout.GetEnvironmentKeyValueMapData</DisplayName>
  <Parameters>
    <Parameter name="service_name">gp-registrations-management-information</Parameter>
  </Parameters>
  <SharedFlowBundle>GetEnvironmentKeyValueMapData</SharedFlowBundle>
</FlowCallout>
```

API key called in encrypted key value map used in backend healthcheck request

Apigee message templating is used to inject the secret API key into the message request.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ServiceCallout async="false" continueOnError="true" enabled="true" name="ServiceCall
out.CallHealthcheckEndpoint">
    <DisplayName>ServiceCallout.CallHealthcheckEndpoint</DisplayName>
    <Properties/>
    <Request clearPayload="true" variable="healthcheckRequest">
      <IgnoreUnresolvedVariables>false</IgnoreUnresolvedVariables>
      <Set>
        <Headers>
          <!-- "apikey" was the name of the key stored in secretsmanager -->
          <Header name="X-API-Key">{private.environment_kvm_data.apikey}</Header>
        </Headers>
      </Set>
    </Request>
    <Response>healthcheckResponse</Response>
    <HTTPTargetConnection>
      <URL>{{ TARGET_URL }}/_status</URL>
    </HTTPTargetConnection>
    <Timeout>20000</Timeout>
</ServiceCallout>
```

7.14.5.6.6 Standard test data

7.14.5.6.6.1 Overview

To ensure consistency across APIs, and to make it easier for API consumers who are testing across multiple APIs, we define standard test / example data to use across all APIs.

Primarily we are talking about using standard test **patients** i.e. the same NHS numbers and basic details (name, date of birth etc).

We do this standardisation primarily in two environments:

- Sandbox (including specs)
- Integration test

For context on path-to-live environments, see Path to live environments (see page 596).

7.14.5.6.6.2 Sandbox (including specs)

If you are doing things right, you'll be using the same test requests and responses in your sandbox as you present in your API specification. So this section applies to sandboxes **and** specs.

Test patients

Patients are mastered in PDS. So it's useful if other APIs use the same test patients that PDS uses.

The standard patients we use in the PDS spec and sandbox are:

NHS number	Given names	Family name	Date of birth	Description
9000000009	Jane	Smith	22 Oct 2010	Standard "happy path" patient Patient has multiple related people
9000000025	Janet	Smythe	22 Oct 2010	Patient is restricted (aka 'S' flagged) Patient has no related people
9000000033	<none>	<none>	<none>	Patient with very virtually no data populated (good for testing adding info when doing PDS updates)
9000000017	Jayne	Smyth	22 Oct 2010	Patient with a single related person

NHS number	Given names	Family name	Date of birth	Description
9111231130	N/A	N/A	N/A	Patient that does not exist in PDS
9000000000	N/A	N/A	N/A	Invalid NHS number

For more details, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir.

Example

For an example of how this has been applied to another API, see:

- https://digital.nhs.uk/developer/api-catalogue/immunisation-history-fhir

7.14.5.6.6.3 Integration test

Test patients

Patients are mastered in PDS. So it's useful if other APIs use the same test patients that PDS uses.

We have a suite of PDS test data packs[558] set up in the Spine INT environment.

When creating test data for other APIs, use patients that also appear in the PDS test data packs.

Example

For an example of how this has been applied to another API, see:

- https://digital.nhs.uk/developer/api-catalogue/immunisation-history-fhir
- https://digital.nhs.uk/developer/api-catalogue/immunisation-history-fhir/immunisation-history-fhir-api-test-data

7.14.5.7 Setting up a target server

7.14.5.7.1 Linking up to an existing target server

If your backend is hosted inside Spine, DPS, or another large national system, chances are that we already have a target server / connection definition to it. In this case you only need to find out its name and link it up to your proxy.

558 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir/pds-fhir-api-test-data

7.14.5.7.1.1 Finding out the name of a target server

Global target server configuration is stored at https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-keystores-refs-targetservers/vars/main/target-servers.yml under `target_servers`.

Once you find the server you want, copy its name.

> ⚠ Make sure the target server you want to connect to is available in all the environments you're deploying to.

7.14.5.7.1.2 Adding the target server definition to your proxy

To connect to the target server, you'll need to set up a target definition under `proxies/live/apiproxy/targets` in your api repository.

For an example, take a look at https://github.com/NHSDigital/personal-demographics-service-api/tree/master/proxies/live/apiproxy/targets

For reference documentation on how to do this, check out https://docs.apigee.com/api-platform/reference/endpoint-properties-reference.

7.14.5.7.1.3 Connecting your proxy to the target server definition

Once your target server definition is created, you'll need to tell your proxy to route to it.

Here's an example from the PDS FHIR API.[559]

7.14.5.7.2 Adding a new target server

To add a new target server, go to https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-keystores-refs-targetservers/vars/main/target-servers.yml and add the information for your server there (you will need to be logged into github and a member of the NHSDigital github organization to be able to access the link).

To use the target server, you will need to:

1. Create a pull request and get it reviewed by APIM support by using the slack workflow "Infrastructure PR" in the platforms-apim-producer-support slack channel.
2. Ask for a release of infrastructure into apigee nonprod and prod ptl
3. Follow Linkinguptoanexistingtargetserver (see page 607) to link your proxy to the target server

559 https://github.com/NHSDigital/personal-demographics-service-api/tree/master/proxies/live/apiproxy/targets

7.14.5.7.2.1 Adding an existing target server to more environments

Follow the same process as above (include setting up TLSMA if you need to) for any other environments you need. You can submit your changes all at once.

7.14.5.7.3 Setting up TLSMA (aka mTLS, TLS Mutual Authentication) on your target server

7.14.5.7.3.1 Obtain or generate certificates

Working with an existing system

If you are working with an existing backend that required TLSMA, you will need to speak to whoever is in charge of issuing certificates for that system. Keep in mind that this may be different entities between production and ptl!

New system

If you are creating a new system, you will generally want to generate your own certificates.

Generating self-signed certificates

To generate a self-signed certificate, you will need to first generate a CA certificate with which to sign your TLS certificates.

This is a basic overview of how to do this.[560]

Getting certificates signed (or signing them with your own CA)

To create a new certificate to use for TLS-MA, you will first need to generate a key of suitable strength (4096 bit RSA is a good minimum, but you may also want to look at Ed25519, ECDHE, and other algorithms).

You can do this with a command such as:

```
ssh-keygen -t rsa -b 4096 -m PEM -f my.key
```

From there, you can generate a CSR (certificate signing request) to be signed by the owner of the CA cert.

Make sure you put together a cert config file[561] first!

560 https://gist.github.com/fntlnz/cf14feb5a46b2eda428e000157447309
561 https://www.switch.ch/pki/manage/request/csr-openssl/

```
openssl req -new -in my.key -nodes -sha256 -out my.csr -config my.conf -extensions
v3_req
```

7.14.5.7.3.2 Upload cert and key to secrets store

Next, ask in #platforms-apim-producer-support to upload your certs to secrets manager.

You will want to ask for secret ids of `ptl/client/<fqdn>/{crt|key}`, and send your key and cert over in an encrypted zip to the triage engineer by email. Send the password to the zip by a different channel, such as slack.

7.14.5.7.3.3 Create a keystore and reference your key and cert

To create a keystore, go to https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-keystores-refs-targetservers/vars/main/tls-keystores.yml and add an entry under prod and/or ptl that looks something like the following:

```
    - name: my-service-name
      type: client
      cn: <fqdn/cn used when creating secret>
      refs:
        my-service-name:
          - environments-i-want-my-keys-to-be-available-in
```

7.14.5.7.3.4 Associate keystore and cert with target server

To associate your cert and key with the target server, you will need to go to https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-keystores-refs-targetservers/vars/main/target-servers.yml, find your target server, and add something simialr to the following:

```
      - name: my-target-server
        host: "my.hostname"
        sSLInfo:
          clientAuthEnabled: "true"
          keyStore: "ref://key-store-name-from-earlier"
          keyAlias: "cert"
```

7.14.5.7.3.5 Request an infra deploy

To use your new setup, you will need to:

1. Create a pull request and get it reviewed by APIM support (via platforms-apim-producer-support slack channel)
2. Ask for a release of infrastructure into apigee nonprod and prod ptl

PTL Infrastructure releases can be performed on-demand. Production infrastructure releases happen periodically, or by arrangement if you need it more urgently.

7.14.5.7.4 Testing connectivity

Once your target server is linked up to your proxy, you can send a request through your proxy to test the connection.

7.14.5.8 Passing information to your API backend

7.14.5.8.1 Overview

The façade / firebreak design pattern (see Architecture[562]) is central to helping provide consistency to your API Consumers. However this means that incoming requests might not contain specific information your backend needs, or data is not in the format that your particular API backend requires. We are also wanting to hide, where easy, any internal NHS Digital complexities.

*APIM does **not** provide a standard template for the data to pass onto your backend. Any Apigee variable is available to you, as well as any data from the OAuth flow - you choose what to pull down and pass on*

Key types of information, you might need to pass onto the backend, each of these correspond to a subsection below where you can find more infomation and examples:

- **Identity** - If the user is logged in with a smart card you might want to pass on their identity, type of authentication onto the backend
- **Apigee Application** - Each consumer connects to their own Apigee Application, and custom data can be attached (e.g. Spine ASIDs or the connecting party ODS code). You might want to pass some of this data downstream to your backend
- **Apigee Developer** - The Apigee App is linked to the developer, so further information (or custom attributes) can be accessed
- **Correlation IDs** - Tracing requests (including aiding in debugging situations) becomes complex very quickly. The platform will support multiple specific tracing concepts, overlapping with the FHIR standard. For more details see: Supporting API consumers with tracing unique identifiers (see page 1104)

562 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97747483/Architecture

- **Other request data** - All the request data is available, much of which remains attached to the request . However, a specific header might need to be added / removed, or information that needs to be placed in a different part of the request

7.14.5.8.2 Injecting meta-data into the request

Once the meta-data is available in your proxy flow, it needs to be injected into the outgoing request to your backend. Possibilities:

- Header:
    - Fields or custom fields
        - Recommend that you base64 encode any JSON to avoid any complications with characters and the underlying HTTP standard
    - JWT - recommended as adding claims into the JWT doesn't change the HTTP structure itself, providing a much looser coupling between stages.
        - Could also be signed
    - If you do use headers - the guidance for incoming headers is still relevant and might help you HTTP headers for your API (see page 395)
- Updating or adding Query Parameters
- Body - updating the payload (not recommended)

Early investigation on how this could have been standardised: Spine Core HL7 FHIR API Standards[563].

7.14.5.8.2.1 Meta-data that is useful for your backend

For Proxygen deployments you can:

- Pick up on pre-defined claims from NHS login / CIS2 Auth and send as headers - see Target Identity Headers[564]
- Configure sending extended attributes (see page 806) as headers see: Configuring target attributes (see page 708)

> Remember it is best practice to only send the information that you need.

How to identify the calling API Consumer

563 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/132317841/Spine+Core+HL7+FHIR+API+Standards
564 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#overview--openapi-specification-constraints-and-extensions

There is a strategic piece of work to create a new identifier (ProductID) to help, with this situation by:

- Having a human readable code attached to an assured piece of software
    - Which enables the legal owner of the connection to be identified
- Six alpha numeric digits - formatted as: "P.GH7-4TY"
- Same in all environments
- This will be embedded in Apigee Logs, Digital Onboarding Service (DOS), ServiceNow to help teams work together

At present, ProductIDs cannot be issued until further work is completed by CPM. A manual workaround has been provided for certain early adopters but is not sustainable for a wider release. If this causes any inconvenience, reach out to us via the platforms-apim-producer-support[565] channel for further discussions.
If you want to read more: Products and Product IDs (see page 308)

Flow variables that are similar to Product ID, but are only to be used in logging (**never** configuration):

- developer.app.id - underlying Apigee App ID, different in different environments.
    - In a DR situation **this number will be** changed
- developer.app.name[566]
- apigee.client_id

Commonly used meta-data

Example type of common meta data you **might** want to pass onto the backend API. Consider how important it is to store it at your end, and whether access to the APIM Splunk logs will actually meet the requirements.

Data	Apigee flow variable	Notes
Application	developer.app.name[567]	e.g. "My Test App" - **not** guaranteed to be unique, but makes it much easier to trace problems though. Consider using ProductID
Client	client.ip	Actual API Consumer IP address, which will normally be a server IP. For user restricted, when looking at the identity service logs, this will be browser IP.

565 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY
566 http://developer.app.name
567 http://developer.app.name

Data	Apigee flow variable	Notes
Spine	app.ASID	For Spine requests, this is the ASID attached to the request.
Tracing IDs	messageID & X-Correlation-ID	See Supporting API consumers with tracing unique identifiers (see page 1104) for full details, but an API Produer should concatenate, the unique Apigee Message ID and (if sent) the Correlation ID into the header "NHSD-Correlation-ID" for passing onto the backend. The backed system should then log the entire value to make end to end tracing easier.

Identity and authorisation meta-data

As the APIM Platform is using the OAuth flow, the identity information is associated with the Access Token issued by Apigee. This data is **never** sent to the client, and is static information made available in flow variables after the **VerifyAccessToken** policy is called. Example values:

Variables	Description
accesstoken.id_token	The entire ID Token
accesstoken.id_token-subject	This is the unique identifier from the identity provider.
accesstoken.auth_user_id	If this is for a CIS2 Auth authentication, this contains the user id - but for NHS login this contains the NHS Number of the user.
NHS CIS2 ID Token	If you don't send the entire ID token to your backend, then you might want to consider sending specific claims or extracting information: • AAL level used in the authentication - from the "authentication_assurance_level" • selectedRole For the full list of claims see NHS CIS2 on NHS Digital site[568].

568 https://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance/scopes-and-claims

Variables	Description
NHS login ID Token	NHS login, from a security perspective, uses the Vectors of Trust model[569]and not acr/ amr. The simplified result is contained in a custom claim: • identity_proofing_level The ID Token also contains important person claims: nhs_number, family_name, birthdate. For the full list of claims see NHS login claims[570].
accesstoken._first_issued	When the ID Token was issued
accesstoken.auth_grant_type	This is a processed value for reporting purposes. The reports need simple values to make it easier, and so this is the right field to use if the "token_exchange" grant type needs to be used for logical processing
accesstoken.auth_type	This is a processed value for reporting purposes. The reports need simple values to make it easier, and so this is the right field to use to determine if an incoming request is App ("app") or User ("user") restricted.

You can read more about token variables for the Identity Service here (see page 810).

Apigee App and developer data

Apigee allows you to tie together every request to the Developer, as long as an API Key is involved. By securing your proxy, with **VerifyAPIKey** and **VerifyAccessToken,** you gain access to data about : the Apigee app that was created; the associated Product (although this is a one to many relationship), and the developer who owns the app.

Some example variables available in the proxy:

Variables	Description
developer.app.name[571]	e.g. "My Test App" - not guaranteed to be unique, but makes it much easier to trace problems though
developer.app.id	Unique app ID - logging only use ProductID

569 https://nhsconnect.github.io/nhslogin/vectors-of-trust/
570 https://nhsconnect.github.io/nhslogin/scopes-and-claims/
571 http://developer.app.name

Variables	Description
apigee.client_id	The consumer key (aka API key or app key) attached to the Apigee App that was used for authorisation - logging only
apigee.developer.email	Email address of the developer
App custom attributes	Current ones (naming still in flux): • "asid" or "asid_code" • "ods_code"

For a list of built-in Apigee flow variables that are set once either of those policies are used see:

https://docs.apigee.com/api-platform/reference/policies/verify-api-key-policy#variables

Correlation and request IDs

There is a lot of detail around how APIM is going to support API Consumers with different options around Correlation IDs and Request IDs - see Supporting API consumers with tracing unique identifiers (see page 1104)

Request information

In addition to the above meta-data, the request itself has a massive amount of information available (e.g. the start time of the request, client certificate information).

See for the full list here: https://docs.apigee.com/api-platform/reference/variables-reference

7.14.5.8.2.2 Spine APIs and how meta data is transmitted

Spine APIs currently only use individual custom field headers to send information to the backend. In the future, certain Spine APIs may use the JWT (see discussions about Strategic metadata transmission (see page 620)) but this is to be decided. For clarity, the status of how the backend APIs are connected to Apigee are:

- Headers: PDS FHIR, TES Alert, SCR FHIR, EPS
- To be decided: RA Flag FHIR, CP-IS, PSIS API, NRL

Example meta data transmission, using the Personal Demographics Service Spine backend

For the PDS implementation, the following headers are used to pass meta-data to Spine:

Header	Example	Policy	Notes

NHSD-Identity-UUID	910000000001	AssignMessage.AddUseridHeader[572]	Extracted from ID Token subject ("sub") claim. CIS2 Auth is the current name for for NHS Identity. UUID (Unique User ID) is the primary user identifier in Spine
NHSD-Session-URID	98765400091		Supplied by the API Consumer, where the Clinician picks the role they are undertaking see Authorisation[573]
NHSD-ASID	200000000421	AssignMessage.AddAsidHeader[574]	Sometimes referred to as the API Consumer ASID or FromASID. See Spine and IGWAY3 security[575] for an overview. See APM-1275[576] - ASID per application - happy path DONE details, and see environments[577] for the fallback ASIDs.
NHSD-Correlation-ID	14d68733-bd4f-4aae-b5ca-a3f783301b91..rrt-936033110107781752-b-geu2-18919-10347056-2	Assi	See Supporting API consumers with tracing unique identifiers (see page 1104)

572 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.AddUserIdHeader.xml
573 https://digital.nhs.uk/developer/user-restricted-restful-apis#user-completes-authentication-and-authorisation
574 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.AddAsidHeader.xml
575 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141201006/Spine+and+IGWAY3+security
576 https://nhsd-jira.digital.nhs.uk/browse/APM-1275?src=confmacro
577 https://nhsd-confluence.digital.nhs.uk/display/APM/Architecture#Architecture-environments

		gnMessage.Swap.RequestHeaders[578]	
NHSD-Request-ID	14d68733-bd4f-4aae-b5ca-a3f783301b91	AssignMessage.Swap.RequestHeaders[579]	See Supporting API consumers with tracing unique identifiers (see page 1104) supplied by API Consumer
NHSD-Identity-IdP	https://am.nhsspit-2.ptl.nhsd-esa.net:443/openam/oauth2/realms/root/realms/oidc[580]	AssignMessage	Extracted from ID Token JWT issuer ("iss") claim. Not used within Spine

578 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.Swap.RequestHeaders.xml
579 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.Swap.RequestHeaders.xml
580 https://am.nhsspit-2.ptl.nhsd-esa.net/openam/oauth2/realms/root/realms/oidc

		.AddIssuerHeader[581]	
NHSD-NHSLogin-User	`p9:9693632192`	AssignMessage.AddPatientAccessHeader[582]	Extracted from the NHS Login JWT. The Proofing Level and NHS Number of the NHS Login user. Relevant in Patient Access mode.

Note - There is no data injected into the payload, it is received by Spine as sent by the API Consumer.

Spine Core FHIR APIs

Being drafted - see Spine Core HL7 FHIR API Standards[583]

7.14.5.8.3 On behalf of other Product IDs

There are a number of situations where one service (A), using APIM for incoming requests might call another service, also behind APIM. Most of the time the service (B) behind APIM isn't interested in that system identity, but in some situations this is very important for auditing or authorisation.

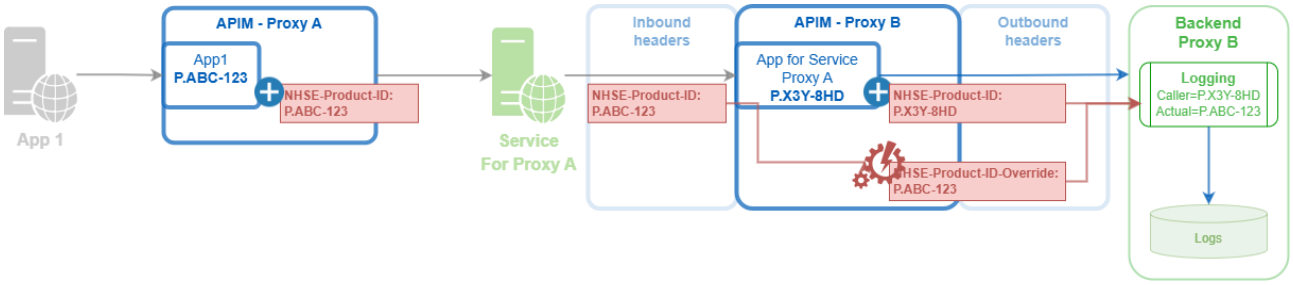
581 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.AddIssuerHeader.xml
582 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.AddPatientAccessHeader.xml
583 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/132317841/Spine+Core+HL7+FHIR+API+Standards

This is where the NHSE Product ID becomes very useful, and the pattern to use here is:

- External third party (App1) calls Proxy A
- APIM injects App1 Product ID into an outbound header NHSE-Product-ID
- Service A connects to APIM Proxy B as normal
    - Attaches NHSE-Product-ID as a header (becoming an inbound header to APIM)
- APIM Proxy B:
    - Maps the incoming value in NHSE-Product-ID to an outbound header NHSE-Product-ID-Override
    - APIM injects Product ID for Service A into an outbound header NHSE-Product-ID

This forms a small chain of two proxies, but the pattern can then go further.

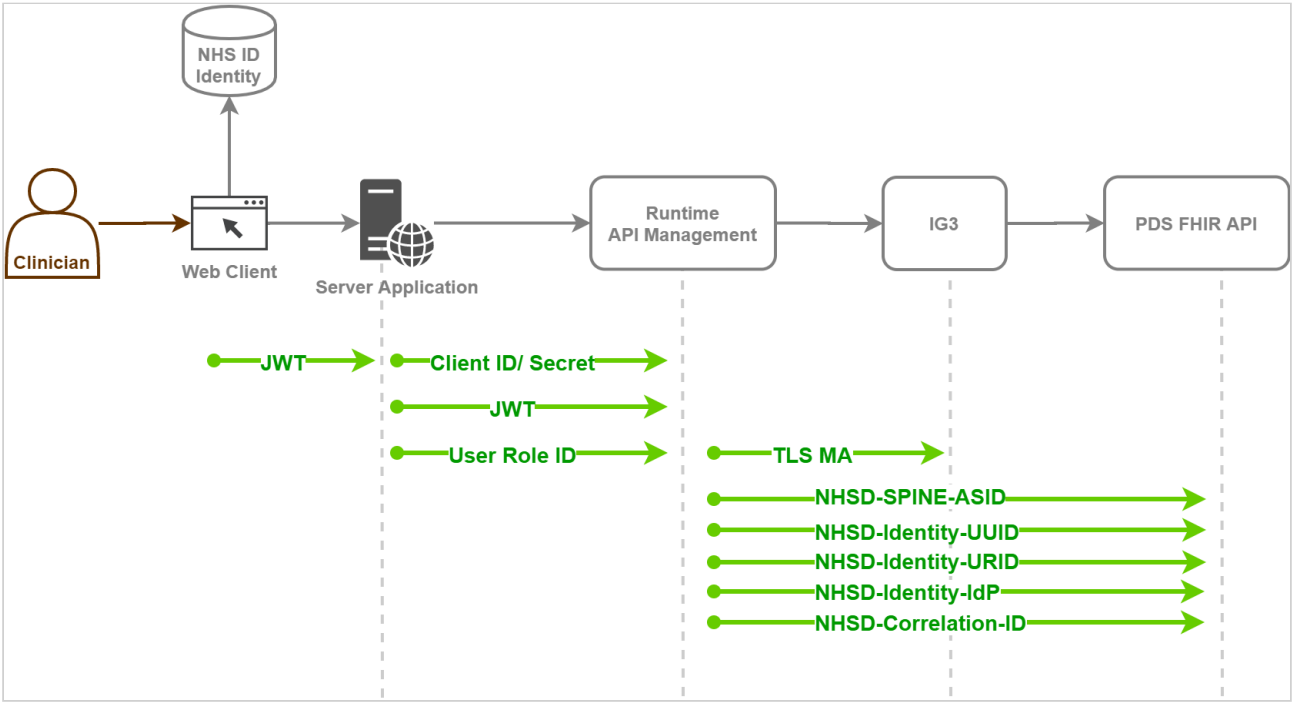
The following visualisation tries to show this:



7.14.5.8.4 Strategic metadata transmission

> ⚠ This approach hasn't gained momentum - although recent work with National Record Locator (NRL) Futures indicates that this might be used as an option that API Producers might choose to use.

This page is to capture the current thinking on the type of meta data that an API backend might be interested in.



7.14.5.8.4.1 API Management Platform Proxied Request

The backend APIs will often need key information, and (depending on the API) some of it might be mandatory. The platform will look to provide a standard set of claims (attributes) and the backend API can choose (or not) to use it.

Consideration will be made, If is easy to implement, or the data size starts to rise too much, that the platform could have a configuration per API that delivers the required information in different ways (e.g. custom header, or JWT).

Claim/ Attribute	Example	Notes
Application unique identifier		Every connection to Apigee will be associated with an Application registered on the platform. This unique identifier will be passed on, however, some API producers might need that value mapped to their internal application ID, and if so Apigee will send that in this attribute

Identity		Holds the identity information, to be determined if this will be as a single entity (e.g. signed JWT) or multiple attributes Conceptually the signed ID Token issued by the IDP would be passed through to act as trusted piece of data for audit purposes - however token size, lifetime (and other) issues might mean this is not practical.
Session Role		Optional, in some situations the end users specifically sets their role for the session. For NHS ID this role is selected via the Identity Agent from a national set. For SPINE this must contain the User Role ID (where selected) for audit. The parent of the User Role ID is the Organisation, and above that the User themselves. However, it should be noted that this can changed during the session, and currently there is no strategy to make this data available via NHS ID (e.g. in the ID Token)
Authentication Method		E.g. Smart Card used
Security Level		e.g. level 3 access
Step Up Auth		If a specific action needs an additional MFA action (e.g. enter code from app) because it is more sensitive - this will hold that information
Tracing ID(s)		Tracing requests (including aiding in debugging situations) becomes complex very quickly. The platform will support two specific tracing concepts: • Apigee creates / manages a Correlation ID • API Consumer can supply an arbitrary string that will not be changed (and might be the same between multiple calls to Apigee) Where ever the backend API carries out logging, these values should be included in that log
Unattended		(?) Lack of identity implies that the request is unattended in nature - need to examine use cases to determine if there is one where identity is supplied for unattended access

Authorisation Key Pairs per API		This is the **most** relevant proposal, and **will** allow Digital Onboarding onboarding leads to configure API Consumer applications with values. The sequence would be:
		• Per API, per assessment set of key value pairs defined (maybe from OAS file?)
		• Dig. Onb. user interface updates to allow viewing & editing (maybe micro front-end?)
		• Mechanism to push key value pairs to Apigee for each API consumer
		• Versioning to support different configs in INT to Prod
		• Shared flows to:
		  • Make KV pairs available in flow (should be working)
		  • Create JWT for transmission

For a breakdown of the existing Spine JWT definition see: Spine Core HL7 FHIR API Standards[584]

Alpha and start of private beta

For the initial work, the above will be cut down to just a few values. It was decided to implement them as a set of custom headers for speed.

Each value will be added as a custom header:

Header	Example	Notes
NHSD-Identity-UUID	910000000001	Extracted from ID Token subject ("sub") claim.
NHSD-Session-URID		Alison Kinloch / Unknown User (extbc1) - have you got an example value to use here?
NHSD-Identity-IdP	https://am.nhsspit-2.ptl.nhsd-esa.net:443/openam/oauth2/realms/root/realms/oidc[585]	Extracted from ID Token JWT issuer ("iss") claim

584 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/132317841/Spine+Core+HL7+FHIR+API+Standards
585 https://am.nhsspit-2.ptl.nhsd-esa.net/openam/oauth2/realms/root/realms/oidc

NHSD-ASID	200000000421	Hard coded to a single value for the environment - see environments[586] Anticipated that this value will vary with each application to hold the third parties ASID
NHSD-Correlation-ID		Created and managed by Apigee
NHSD-NHSLogin-User	`p9:9693632192`	Extracted from the NHS Login JWT. The Proofing Level and NHS Number of the NHS Login user. Relevant in Patient Access mode.

***NOTE**: Unattended access is out of scope for start of private beta*

For the API Consumer view of headers see: Solution Aspect: API Consumer#headers[587]

7.14.5.9 Handling errors

7.14.5.9.1 Overview

Requests through your API proxy may result in errors, either from the API Proxy itself, or the back-end system(s). Unless explicitly handled these errors will be sent through to the client unmodified.

This default behaviour is analogous to leaving a Web Application in debug/development mode, where HTTP 5xx may errors contain sensitive information (such as stack traces).

This page details:

- APIM Platform error principles
- The error handling features available in Apigee and related concepts
- A pattern to handle and reformat errors
- A specific example of the PDS API Proxy where errors are re-formatted into the FHIR-compliant 'OperationOutcome'
- APIM pattern for non-FHIR APIs error format - which is also a FHIR-compliant 'OperationOutcome'

7.14.5.9.2 Key Concepts / Principles

This link takes you through the 'Cliffs Notes' on how to deal with errors in Apigee: Fault Handling Documentation[588]

586 https://nhsd-confluence.digital.nhs.uk/display/APM/Architecture#Architecture-environments
587 https://nhsd-confluence.digital.nhs.uk/display/APM/Solution+Aspect%3A+API+Consumer#SolutionAspect:APIConsumer-headers
588 https://docs.apigee.com/api-platform/fundamentals/fault-handling

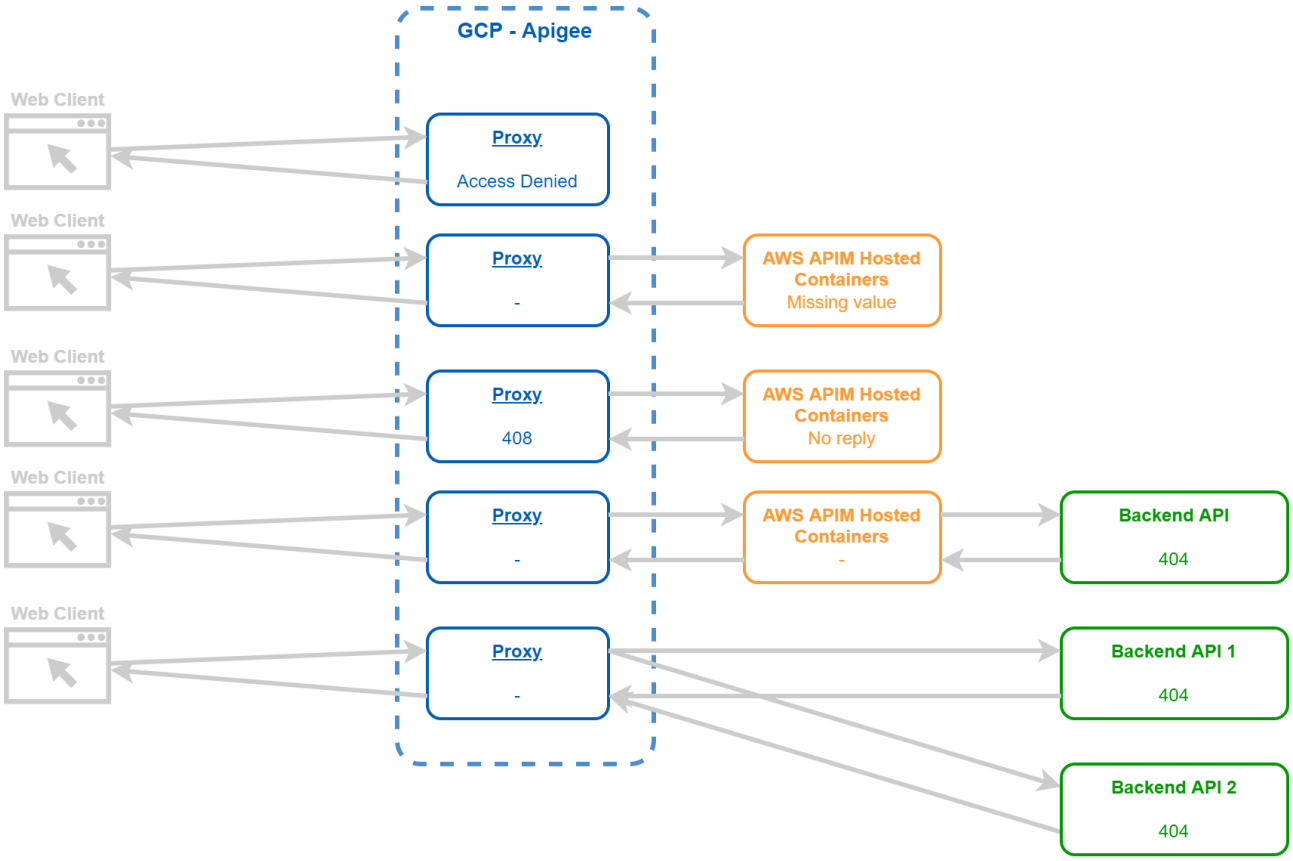
APIM Platform Error Principles

- If you advertise that your API adheres to a certain messaging standing (e.g. FHIR) then *all* interactions should be faithful to that format **including error responses**
- All error responses must include a payload (4xx and even 5xx errors) to aid consumer diagnosis
- The error response text must help the API Consumer to solve problems where they can - reducing contact with NHS Digital
- There is a central repository of APIM codes[589], and these should be used in preference to creating a new one for a specific API
    - Remember there other codes sets (as well as the APIM generic error codes), consider using those as well before your own
- The JSON format in use will be the FHIR-compliant 'OperationOutcome' set out below - even for non-FHIR APIs
- **Don't** over engineer the solution, the aim here is a empower API Consumers, reduced confusion and a reduction in duplicate codes

For an the full set of API design principles see Designing your API (see page 322)

Where errors are generated, the following places are where it could be generated:

589 https://simplifier.net/guide/nhsdigital/home/fhirassets/allassets/profiles/nhsdigital-operationoutcome.guide.md?version=current



7.14.5.9.3 APIM Standard error format

We wanted a rich error message format to improve self serve and reduce contact with NHS Digital. The choice of FHIR for the platform wide error format (unless your API has been agreed that it can use another standard) was still complex decision but came down to:

- By being selective with the field choice - FHIR JSON wasn't any more complex than other rich error message formats
- Any JSON parser can retrieve the data
- There are no other widely adopted industry standards

For the most up to date specification see: https://simplifier.net/guide/nhsdigital/home/fhirassets/allassets/profiles/nhsdigital-operationoutcome

Key aspects:

Area	Description
Contains one to many errors	Most responses will have a single error. If there are multiple then the primary error should always be first (if easy to do).
Invariant error codes	Use an error code to uniquely identify an error - you can change the description, but the code should not change in underlying meaning (the code itself is immutable. If in doubt create a new code
Standard Codes (System)	There is a an APIM standard code set, (tbc) a FHIR UK Core standard set, and where an API Producer has business specific error codes they publish their can manage their own codes
Security	You should not expose any sensitive (personal or system) details in your error messages However, you **should** make it clear the root problem (e.g. expired access token)
Extendable	You, as an API Producer might choose to use other parts of FHIR to extend the message
Codes from multiple systems	Keep the message format simple for your implementation, for example: • If there is a more specific error code, you don't need to also include the generic code • If your code base / solution automatically adds codes - don't write code to then remove extra codes Overtime, with feedback from API Consumers
Shared flow (future)	APIM has a ticket on the backlog to create a shared flow that will: • Accept a standard set of variables • Apply any special rules that make the message more FHIR friendly • Format the message

7.14.5.9.3.1 Example JSON

```
{
```

```
"resourceType": "OperationOutcome",
"id": "rrt-2959959087381887325-c-geu2-24001-82918062-1",
"meta": { "lastUpdated": "2021-04-14T11:35:00+00:00" },
"issue": [{
      "severity": "error",
      "code": "value",
      "details": {
            "coding": [ {
                  "system": "https://fhir.nhs.uk/CodeSystem/Spine-ErrorOrWarningCode",
                  "code": "INVALID_VALUE",
                  "display": "Invalid value" } ]
            },
            "diagnostics": "(invalid_request) firstName is missing",
            "location": [
                  "/data/attributes/firstName" ]
      } ]
}
```

For the most up to date and full list of field definitions specification: NHS Digital Simplifier Guides - OperationOutcome[590], the table below is providing design context and importance to the different aspects:

Section	Field	Description
issue.details.coding	code	Invariant code - this is **critical to never change**, API Consumers must be able to make logical decisions based on this. In rare circumstances changing this could be breaking change. If in doubt, create a new code.

590 https://simplifier.net/guide/nhsdigital/Home/FHIRAssets/AllAssets/Profiles/NHSDigital-OperationOutcome

Section	Field	Description
issue.details.coding	system	System that the code is from. If the error raising logic is in Apigee, there might be a number of scenarios: • If there is an existing FHIR UK Core code and clearly fits - use it • If it is has an existing APIM code and clearly fits - use it • If it is a code that APIM should have added - talk to us • If there is overlap with the backend / business functions, BUT the error only ever gets raised from Apigee - consider having a code that helps distinguish rather than relying on access to the Splunk logs • Use your backend system / code
issue	code	This is a FHIR value set to code up against, select on that is appropriate but you can't create your own here
meta	lastUpdated	To be compatible with FHIR, date and time field values must follow a sub-set of the ISO-8601[591] standard - see RESTful HTTP design points (see page 365)

7.14.5.9.3.2 Shared flow for Unauthorised / Access Token related errors

TODO - Ticket = APM-2533[592] - Shared flow for Unauthorised / Access Token related errors **WITHDRAWN**

A shared flow (code snippet?) has been created to:

- Encapsulate the following scenarios
- Make APIs more self service
- Improve consistency, especially as non-FHIR APIs also use FHIR JSON format

This logic does **not** apply to the Auth (Identity) Service

See Pen test approach - HTTP 401 403 error responses (see page 986) for details

7.14.5.9.3.3 Systems and code fields in coding array

APIM has agreed that the system uri and the code field formats will be:

591 https://en.wikipedia.org/wiki/ISO_8601
592 https://nhsd-jira.digital.nhs.uk/browse/APM-2533?src=confmacro

- https://fhir.nhs.uk/CodeSystem/**NHSD-API**-ErrorOrWarningCode[593]
- **API**_ERROR_CODE

Currently that "API_" indicator is NOT in the "code" field for the code sets, but in many situations the code is the item most down stream users will see. So the addition if for clarity and speeding up diagnosis in. In terms of the system side currently there are: "Spine", "EPS" and "e-RS"

Publishing your code sets

The code sets source are managed here:

- https://github.com/NHSDigital/NHSDigital-FHIR-ImplementationGuide/tree/master/CodeSystem

Raise a Pull Request against the set to manually update - in the future will look at a more automated process.

7.14.5.9.4 Apigee Components - non Proxygen APIs

7.14.5.9.4.1 Fault Sources

- Policies - see policy error reference[594] for the types of errors policies can throw
- Issues with API Proxy message flow (such as routing errors)
- Backend failures
- System-level failures (e.g. Out of Memory)

7.14.5.9.4.2 FaultRules Execution

```
<ProxyEndpoint name="default">
...
    <FaultRules>
       <FaultRule name="invalid_key_rule">
            <Step>
                <Name>invalid-key-message</Name>
                <Condition>...</Condition>
            </Step>
            <Condition>(fault.name = "FailedToResolveAPIKey")</Condition>
        </FaultRule>
    </FaultRules>
    <DefaultFaultRule name="default-fault">
        <Step>
            <Name>Default-message</Name>
        </Step>
    </DefaultFaultRule>
```

- Can be placed in `<ProxyEndpoint>` or `<TargetEndpoint>` elements
- One or more `FaultRules` each having one or more policies (`<Step>` elements)

593 https://fhir.nhs.uk/CodeSystem/Spine-ErrorOrWarningCode
594 https://docs.apigee.com/api-platform/reference/policies/error-code-reference

- FaultRules have conditions and only execute where the condition evals to true.
    - No condition = true by default
- Steps can also have conditions
- Only the first FaultRule with a condition evaluating to *true* gets executed
- DefaultFaultRule provides a "catch-all" where other FaultRules didn't execute

Caveats / Exceptions

- Order of FaultRule checking differs:
    - `<TargetEndpoint>` **top to bottom**
    - `<ProxyEndpoint>` **bottom to top**
- Even if no steps of a FaultRule actually run (because of step-conditionals) the FaultRule is considered to have executed; no other FaultRules will execute
- The exception to the 'only one fault rule gets executed' rule is where a `<DefaultFaultRule>` has child element `<AlwaysEnforce>true</AlwaysEnforce>`
    - **except** where a FaultRule other than the DefaultFaultRule invokes a RaiseFault policy
- Setting `continueOnError=true` attribute to on a policy will prevent the flow from entering an error state
    - The `{policy_namespace}.{policy_name}.failed` variable will still be set true - you can use this in subsequent policies to handle/recover from the error "in flow"
- The PostClientFlow will still be executed when in error state

7.14.5.9.4.3 RaiseFault Policy

The RaiseFault policy is used to put a flow into an error state when an error wouldn't otherwise have occurred.

This policy can also be used to customise the response.

Subsequent policies in the FaultRule (e.g. AssignMessage) policies may overwrite parts of the response set by the FaultRule. (Or in the case of headers having the same name, they are appended.)

7.14.5.9.4.4 FaultRule Conditions

Two variables are always available when in error state (during FaultRule condition evaluation):

- `fault.name`[595] - an error name, which appears in the default error message. See policy error reference[596] for a list.
- `{policy_namespace}.{policy_name}.failed` - see the flow variables section in each policy's reference

Other available variables:

---

595 http://fault.name

596 https://docs.apigee.com/api-platform/reference/policies/error-code-reference

- The variables of the policy that failed
- HTTP message variables that exist at the point of failure

Use the trace tool and the policy reference to figure out what's available

7.14.5.9.4.5 Setting response headers

Under certain conditions response headers need to be set, for instance:

- FHIR requires the Content-Type header be populated - for most APIs this will simply be: application/fhir+json. See HTTP headers for your API (see page 395) for more details
- 503 errors require the Retry-After header to be set - see RESTful HTTP design points (see page 365)

7.14.5.9.4.6 Best Practices

- Include conditions in all FaultRules
- Include a DefaultFaultRule to reformat at least the payload
- Use the PostClientFlow to log details of the error
- MAYBE, TBD: Don't reformat the status code to mask errors from backend services

7.14.5.9.5 HOWTO: Error Handling Pattern

7.14.5.9.5.1 1. Create a "Catch All" Error Message

policies/AssignMessage.CatchallErrorMessage.xml

```
<AssignMessage async="false" continueOnError="false" enabled="true" name="AssignMessa
ge.CatchallErrorMessage">
    <DisplayName>Catchall error message</DisplayName>
    <Set>
        <Payload contentType="application/json">
          {
            "error": "unknown_error",
            "error_description": "An unknown error occurred processing this request.
Contact us for assistance diagnosing this issue: https://digital.nhs.uk/developer/
help-and-support quoting Message ID",
            "message_id": "{messageid}"
          }
        </Payload>
    </Set>
    <AssignTo createNew="false" transport="https" type="request"/>
    <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
</AssignMessage>
```

This policy defines a default message that will be sent to clients in the event of an error. Note that it only specifies the *payload*; other message attributes (such as the status code) remain unchanged.

It aims to be helpful and provides a link to developers on where to seek assistance, as well as the Apigee messageid[597] which can be used to retrieve this exchange from the logs.

Your error message schema may be different to the above - this one follows the format set out in the OAuth V2 specification.

7.14.5.9.5.2 2. Create 'Default Fault Rules' in Proxy and Target Endpoints

Add the following as a child of each <ProxyEndpoint> and <TargetEndpoint> elements:

```
<DefaultFaultRule>
    <Step>
    <Name>AssignMessage.CatchallErrorMessage</Name>
    </Step>
</DefaultFaultRule>
```

This invokes the "Catch All" error message policy wherever a proxy enters an error state.

At this point, any fault will result in the above response, masking potentially sensitive information from clients.

7.14.5.9.5.3 3. Allow "RaiseFaults" with No-Op FaultRule

If you intend to use the RaiseFault[598] policy to generate custom, conditional error messages you must prevent the DefaultFaultRule from executing.

Achieve this by adding the following 'no-op' conditional fault rule element:

- To the *top* of your ProxyEndpoint Fault Rules
- To the *bottom* of your TargetEndpoint Fault Rules

```
<ProxyEndpoint name="default">
  <FaultRules>
    <FaultRule name="allow_raisefaults">
      <Step>
        <Name>AssignMessage.CatchallErrorMessage</Name>
        <Condition>(fault.name NotEquals "RaiseFault")</Condition>
        </Step>
      <Condition>(fault.name Equals "RaiseFault")</Condition>
    </FaultRule>
    <!-- Other <FaultRule> Elements -->
  </FaultRules>
  ...
```

597 https://docs.apigee.com/api-platform/reference/variables-reference#messageid
598 https://docs.apigee.com/api-platform/reference/policies/raise-fault-policy

See the documentation on FaultRules vs. the RaiseFault policy[599] to better understand whether you might want to use RaiseFault policies.

7.14.5.9.5.4 4. Use a RaiseFault policy to Generate a Conditional Error Message

You can use the RaiseFault policy to conditionally terminate request processing and return a customised error message to the client.

An example policy used to inform the client that a required header is missing:

policies/RaiseFault.MissingFooHeader.xml

```
<RaiseFault async="false" continueOnError="false" enabled="true" name="RaiseFault.Mis
singFooHeader">
  <FaultResponse>
    <Set>
      <Payload contentType="application/json">
          {
            "error" : "invalid_request",
            "error_description" :"Missing header: Foo",
            "message_id": "{messageid}"
          }
      </Payload>
      <StatusCode>400</StatusCode>
      <ReasonPhrase>Bad Request</ReasonPhrase>
    </Set>
  </FaultResponse>
  <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
</RaiseFault>
```

Invoke the header by including a conditional step in a proxy flow:

```
<ProxyEndpoint name="default">
    <PreFlow name="PreFlow">
        <Request>
          <Step>
            <Name>RaiseFault.MissingFooHeader</Name>
            <Condition>(request.header.Foo Is null) or (request.header.Foo Equals
"")</Condition>
          </Step>
        </Request>
...
```

599 https://docs.apigee.com/api-platform/fundamentals/fault-handling#rulesvraisefault

7.14.5.9.5.5 5. Adding FaultRules to Catch Policy Errors

Your API Proxy may enter an *error state* due to the execution of a policy.

For example, a VerifyAPIKey[600] policy in a flow expects the caller to send a valid API key in a request header 'apikey'. You anticipate that there are times when this might be missing or incorrect, and wish to provide a customised error message when this is the case.

First, create an AssignMessage policy to specify the error response:

policies/AssignMessage.InvalidOrMissingApiKey.xml

```
<AssignMessage async="false" continueOnError="false" enabled="true" name="AssignMessa
ge.InvalidOrMissingApiKey">
  <Set>
    <Payload contentType="application/json">
      {
        "error": "invalid_api_key",
        "error_description": "API Key in header apikey is invalid or missing.",
        "message_id": "{messageid}"
      }
    </Payload>
    <StatusCode>401</StatusCode>
    <ReasonPhrase>Unauthorized</ReasonPhrase>
  </Set>
  <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
  <AssignTo createNew="false" transport="http" type="request" />
</AssignMessage>
```

Next create a conditional FaultRule that invokes this policy:

```
<FaultRules>
  ...
  <FaultRule name="invalid_key_rule">
    <Step>
      <Name>AssignMessage.InvalidOrMissingApiKey</Name>
    </Step>
    <Condition>(fault.name Equals "FailedToResolveAPIKey") or (fault.name Equals
"InvalidApiKey")</Condition>
  </FaultRule>
  ...
</FaultRules>
```

600 https://docs.apigee.com/api-platform/reference/policies/verify-api-key-policy

The Policy Error Reference[601] lists the runtime errors that may occur during execution of each policy, and their corresponding *names* (i.e. the value of the 'fault.name' variable).

7.14.5.9.6 Example: PDS FHIR-Formatted Errors

You can view an example of how PDS routes errors here: https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/targets/ig3.xml#L77

7.14.5.9.6.1 Error Response Template

AssignMessage.Errors.CatchAllMessage.xml[602] creates a FHIR-formatted error response.

The content of the errors is controlled by setting the variables: pds.error.code, pds.error.coding.code, pds.error.coding.display, pds.error.diagnostics.

7.14.5.9.6.2 Error Variable Defaults

The AssignMessage.Errors.Default.xml[603] policy sets default values for the four pds.error.* (see above) error variables.

It is the first step in ProxyEndpoint → PreFlow → Request element. This ensures the error response template will have something set where it is invoked.

7.14.5.9.6.3 DefaultFaultRules

A <DefaultFaultRule> element is added to both the ProxyEndpoint and TargetEndpoint elements, invoking the AssignMessage.Errors.CatchAllMessage policy.

At this point, all errors that are otherwise unhandled will be re-formatted to FHIR-format, and the details of the error (which may be sensitive) are hidden.

Other than being in FHIR-format, the errors aren't particularly useful at this point.

7.14.5.9.6.4 Customising the Response for a Specific Error

See the <FaultRule> elements and their <Condition> elements to understand how specific, anticipated errors are handled:

- Each contains an AssignMessage policy that sets the error variables to something appropriate

601 https://docs.apigee.com/api-platform/reference/policies/error-code-reference
602 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.Errors.CatchAllMessage.xml
603 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/policies/AssignMessage.Errors.Default.xml

- They then invoke the AssignMessage.Errors.CatchAllMessage policy (or a similar policy, where further response customisation is required) to set the response.

Errors either happen naturally as the result of a certain policy - in this case the FaultRule condition will be set to detect that specific errored condition.

Alternatively, the proxy is will explicitly be put into a fault state using a RaiseFault policy - for example when a required header is unset.

7.14.5.9.7 Example: Non-FHIR Platform wide standard error format

As a design principle, APIM wants all APIs to return the same error format. Over time, more and more API Consumers will want to connect to more than one API, and will want a standard way to understand returned errors.
The aim here is:

- A simple error message format
- The messages should be clear, unambiguous, looking to make your API self-serve (reducing support contacts with NHS Digital)
- Each business rule in your API should return an error message tailored to that scenario
- Extra fields returned beyond the minimum is no issue

It was agreed that the FHIR JSON format (the OperationOutcome set out above) was simple enough for re-use here, and as an API Consumer I don't have to handle errors differently between APIs. For background and reasoning, see D040 - Non FHIR Platform error format[604]

7.14.5.10 APIM AWS Hosted containers

7.14.5.10.1 Overview

APIM offers API Producers the ability to carry out light weight transforms using Docker images - for a deeper dive when this might be appropriate see Transforming data on the APIM platform and containers (see page 371) hosting containers for use with the APIM platform.

604 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182981712/D040+-+Non+FHIR+Platform+error+format

7.14.5.10.2 Deployment

If you are using this option, then APIM automatically deploys your Docker image into AWS Fargate clusters. This is handled transparently in Deploying your API with Proxy Generator (see page 698), or as part of your less transparently in your GitHub repo.

7.14.5.10.3 Adding extra IAM permissions

IAM permissions for hosted apis in ecs can be seen and edited here: https://github.com/NHSDigital/api-management-infrastructure/blob/master/terraform/modules/apis/proxy-iam.tf

Stop!

Before you do this, check with the Deathstar team that there isn't another way to do what you're trying to do

7.14.5.10.4 Configuring egress to other services

See the egress section on Developing ECS proxies (see page 638) for outbound internet connections and HSCN / LDAP connections

7.14.5.10.5 Scaling hosted containers

There are a few ways to determine scaling for containers, as our recommendation is to use AWS Fargate clusters for you container hosting, it is helpful that all attempts to assess rates and traffic ingress / egress prior to building your scaling options.

Please ensure that cost has been included into you calculation for scaling your containers, along with usage maps that will help drive any downscaling during times of low usage.

7.14.5.10.6 Developing ECS proxies

The Apigee hosted target model wasn't scaling to the automation / flexibility needs of the Tribe, and the design intent from Apigee for Hosted Targets always indicated they might need replacing long term.
See D021 - Translation Layer for Apigee[605] for some of the considerations.

A replacement automated APIM "**AWS Hosted Container**" continuous deployment has been built out and this page details how to build out a Docker container and get it deployed along with the Apigee Proxy.

Details on how all this works together are here: Continuous deployment target model (see page 422)

Sections on this page:

605 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117641137/D021+-+Translation+Layer+for+Apigee

7.14.5.10.6.1 Creating a Docker proxy

Current sandboxes / hosted targets are Node apps, but any language can be used to create a Docker proxy, for the purposes of this document we are using node.

Example node Dockerfile:

```
FROM node:12.18.0-alpine3.10

WORKDIR /app

COPY package*.json ./

RUN npm install --only=production

COPY src/ ./
RUN chmod -R a+x /app

USER nobody

EXPOSE 9000
CMD ["node", "main.js"]
```

In the in this example the repo is structured like this:

- docker
  - <container_name>
    - src
      - main.js
    - Dockerfile
    - package.json
    - package-lock.json

but there's **NO** 'wrong' structure.

7.14.5.10.6.2 Port 9000

While it is possible to have multiple containers within an ECS service, a given ECS service is expected to expose port 9000 and to only expose 1 port.

7.14.5.10.6.3 Service Name / Service ID

It's expected that service names may be quite verbose, e.g. my-extremely-long-connector-service, but due to some character name limits, naming some AWS resources requires a shorter name.

If the <service_name> is less than 22 characters then the <service_id> will be the <service_name> ...

Alternatively either **a friendly 'service_id' can be supplied to the 'create-build-env-vars' step**

or

if the <service_name> is longer than 22 characters, then a <service_id> will be generated.

Generated service IDs will be the first chars of the service name elements, plus as much of the MD5 sum as possible.

e.g. for my-extremely-long-connector-service service_id=melcs-aavbced1234aefaefaadd (or similar)

build and deployment of the ECS proxies is scoped to the 'service_id' ...

this controls the name of the ECS service that can be deployed and the name of the ECR repos that the build user can push to ...

a 'build user' **build-<service_id>** is created automatically, this user will have permissions to push to **<account_id>.dkr.ecr-eu-west-2.amazonaws.com/<service_id>_<container_name>** the associated deployment user will also only have permissions to deploy to ECS services scoped to the service name and environment.

7.14.5.10.6.4 Container build config

Container build takes a configuration snippet (yaml vars) file to know what to build

```
docker_containers:

  - name: <container_name>
    dockerfile: "docker/<container_name>/Dockerfile"
    path: "docker/<container_name>"
```

this specifies the docker container(s) to build, with the location of the Dockerfile and the 'path' being the base directory for the docker build context.

e.g. docker build -t <repo>/<service_name>_<container_name> -f docker/<service_name>/Dockerfile ./ docker/<service_name>

7.14.5.10.6.5 Proxy Configuration

In order to access the ECS proxy it's required to use a HTTPTargetConnection which contains APIM_PROXY_HOSTNAME which will be templated at deployment time

```
<TargetEndpoint name="apim">
  <HTTPTargetConnection>
    <SSLInfo>
      <Enabled>true</Enabled>
    </SSLInfo>
    <URL>https://{{ APIM_PROXY_HOSTNAME }}</URL>
  </HTTPTargetConnection>
</TargetEndpoint>
```

7.14.5.10.6.6 Building and pushing docker containers

integration with the build process is via the api-management-utils repo ... https://github.com/NHSDigital/api-management-utils

here's a sample build pipeline

```
name: "$(SourceBranchName)+$(BuildID)"

resources:
  repositories:
    - repository: utils
      name: NHSDigital/api-management-utils
      endpoint: NHSDigital
      type: github

jobs:
  - job: build
    displayName: Build & Test
    timeoutInMinutes: 10
    pool:
      vmImage: 'ubuntu-20.04'
    steps:
      - checkout: self
        path: s

      - checkout: utils
        path: s/utils

      - ....... (other build steps)

      - bash: "mkdir dist"
        displayName: create dist dir

      - bash: "cp proxies dist/proxies"
        displayName: copy proxies

      - bash: "cp ecs-proxies-deploy.yml dist/ecs-deploy-sandbox.yml"
        displayName: copy proxies deployment config

      - template: az/build-prereqs.yml@utils

      - template: az/create-build-env-vars.yml@utils
        parameters:
          out_dir: 'dist'
          service_name: '<service_name>'    # or use service_id: 'service-slug'

      - template: az/become-build-user.yml@utils
        parameters:
```

```
        env_vars_dir: 'dist'

    - template: az/build-and-push-ecs-proxies.yml@utils
      parameters:
        env_vars_dir: 'dist'

    - publish: dist/
      artifact: <service_name>-$(Build.BuildNumber)
```

7.14.5.10.6.7 Container Deployment Configuration

ECS deployment configuration takes a separate vars file, this specifies the exposed service and other container params .. this supports templating at deployment time

at the time of deployment, the release process will attempt to resolve

`<artifact_dir>/ecs-deploy-$(APIGEE_ENVIRONMENT).yml`

then

`<artifact_dir>/ecs-deploy-all.yml`

(with this it is possible to target an ECS proxy to be deployed only to a single environment or have different configuration per environment if required)

example container deployment configuration:

```
docker_service:

  - name: <container_name>
    expose: true
    port: 9000
    environment:
      - name: NODE_ENV
        value: production
      - name: LOG_LEVEL
        value: "{{ 'debug' if APIGEE_ENVIRONMENT == 'internal-dev' else 'info' }}"
      - name: UPSTREAM
        value: "{{ apigee_uri }}"
    health_check:
      matcher: "200"
      path: "/_ping"
```

7.14.5.10.6.8 Docker secrets

Of course we want to avoid including secret or sensitive values statically in build artefacts, at the time of writing the 'service build user' has the following permissions on SSM

```
    - Sid: ssmget
      Effect: Allow
```

```
      Action:
        - "ssm:GetParameter"
        - "ssm:GetParameters"
        - "ssm:GetParametersByPath"
      Resource:
        - "arn:aws:ssm:{{ aws_region }}:{{ aws_account_id }}:parameter/{{ account }}/
platform-common/*"
        - "arn:aws:ssm:{{ aws_region }}:{{ aws_account_id }}:parameter/{{ account }}/
api-deployment/{{ service_id }}/*"
    - Sid: ssmput
      Effect: Allow
      Action:
        - "ssm:PutParameter"
      Resource:
        - "arn:aws:ssm:{{ aws_region }}:{{ aws_account_id }}:parameter/{{ account }}/
api-deployment/{{ service_id }}/*"
```

and the ECR execution role has the following permissions:

```
  statement {
    actions = [
      "ssm:GetParameter",
      "ssm:GetParameters",
      "ssm:GetParametersByPath",
    ]

    resources = [
      "arn:aws:ssm:${local.region}:${local.account_id}:parameter/${var.account}/
platform-common/splunk/*",
      "arn:aws:ssm:${local.region}:${local.account_id}:parameter/${var.account}/api-
deployment/${var.service_id}/*"
    ]
  }
```

with this approach it is possible for the build user to store configuration values and secrets in SSM for use by the containers.

To consume these values use 'valueFrom' when defining your docker secrets ... e.g.

```
docker_service:

  - name: <service_name>
    expose: true
    port: 9000
    environment:
      - name: NODE_ENV
        value: production
    secrets:
      - name: mysecret
```

```
        valueFrom: '/{{ account }}/{{ service_name }}/{{ APIGEE_ENVIRONMENT }}/
mysecret'
    health_check:
      matcher: "200"
      path: "/_ping"
```

7.14.5.10.6.9 Multiple containers

it is possible to specify multiple containers to be built and deployed within the docker service

e.g.

ecs-proxies-containers.yml

```
docker_containers:

  - name: nginx
    dockerfile: "docker/nginx/Dockerfile"
    path: "docker/nginx"
  - name: translator1
    dockerfile: "docker/translator1/Dockerfile"
    path: "docker/translator1"
  - name: translator2
    dockerfile: "docker/translator2/Dockerfile"
    path: "docker/translator2"
```

ecs-proxies-deploy.yml

```
docker_service:
  - name: nginx
    expose: true
    port: 9000
    environment:
        - ...
    secrets:
        - ...
    health_check:
        matcher: "200"
        path: "/_status"
  - name: translator1
    expose: false
    port: 9001
    environment:
        - ...
    health_check:
        matcher: "200"
```

```
      path: "/_status"
- name: translator2
  expose: false
  port: 9002
  environment:
      - ...
  health_check:
      matcher: "200"
      path: "/_status"
```

Only the exposed container will be accessible via the Apigee proxy, but all containers can communicate with one another over the loopback adapter (in the example above, nginx could call translator1 on localhost:9001).

7.14.5.10.6.10 Outbound calls from Docker images

By default, no calling of arbitrary internet sources is allowed from the uploaded Docker images. There are a number of scenarios that an API Producer will want to do this:

- connections to Spine
- connections to the intended existing API backend
- connections to another API based service to support the lightweight processing

This is currently not configurable by an API Producer but is an easy change to make - contact APIM via the Slack API Producer Support channel. APIM supports the following egress patterns:

- a domain name based allow list
- pre-configured connections to Spine with the Spine TLS MA Certificates all managed by APM

Example

Once you contacted APIM you can confirm the configuration for your service is in place by following these steps.

In the following example, we ask APIM to configure the docker service called "my-service" to be able to egress the platform and hit the host "int.api.service.nhs.uk[606]".

- Confirm that an “egress_proxy” was created for the host you requested access to by looking at this[607]file.
    - e.g if requested access to the host "in.api.service.nhs.uk" you should find an entrance like the following under "egress_proxys"

```
apigee-int = {
```

606 http://int.api.service.nhs.uk

607 https://github.com/NHSDigital/api-management-infrastructure/blob/master/terraform/_shared/vars/ptl.tfvars#L106

```
    port        = 507
    client_cert = null
    ca_certs    = []
    sni         = true
    host        = "int.api.service.nhs.uk"
  }
```

- Confirm your container service is allowed to access the "egress_proxy" by looking at this[608]file.
  - e.g for the service called "my-service" you should see an entry like the following under "allowed_api_egress"

```
  my-service= {
    apigee-int : ["internal-dev", "int"]
  }
```

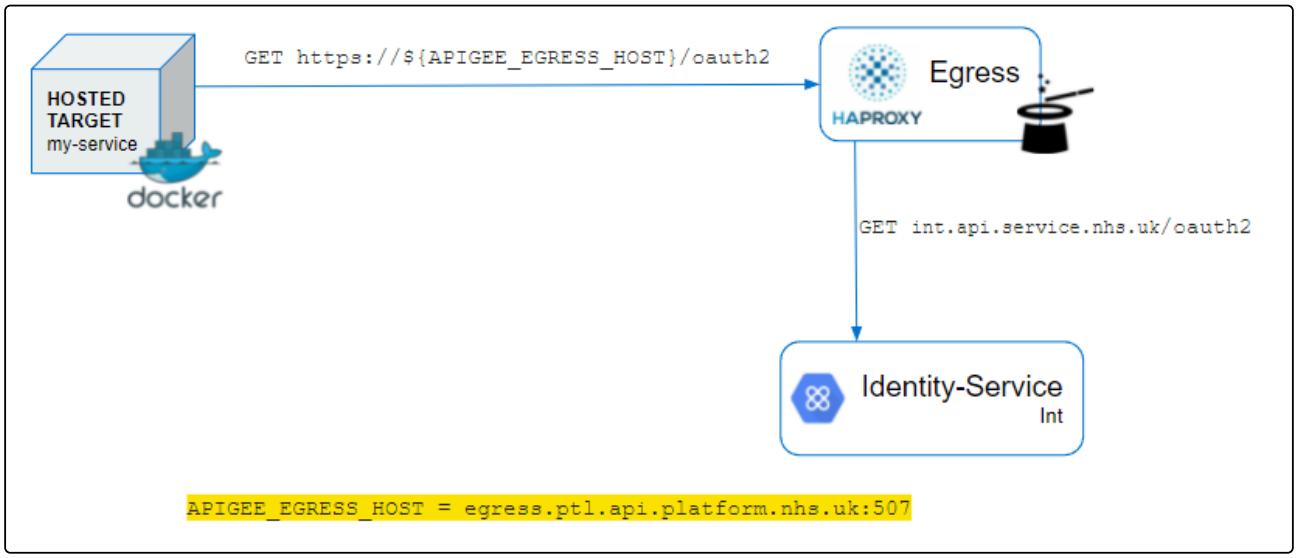
This last piece of configuration is allowing the docker service called "my-service" deployed on "internal-dev" and "int" environments to have access to the host associated with the egress_proxy called "apigee-int" (int.api.service.nhs.uk).

Once this infrastructure configuration is released the pipeline will also create some secrets to facilitate to you the “right host” you should use inside your container. This host will replace the one on your HTTP/HTTPS calls and it will point to our egress ha-proxy on a specific port. Please consume this secret in your service by following step 8 and adding a meaningful name to it.

```
secrets:
 - name: APIGEE_EGRESS_HOST
   valueFrom: '/{{ account }}/{{ service_name }}/{{ APIGEE_ENVIRONMENT }}/apigee-int'
 #The secret name will be equal to the "egress_proxy" name, in this case "apigee-int"
```

APIM platform uses an HA-proxy to control egress traffic from the containers therefore the request from your service should not be pointing to the host you want to reach directly but to a particular port on the egress ha-proxy (the secret resolves to the right host and port for you so you don't have to worry about that). Then the ha-proxy will route your call to the outside world following the host defined on the egress_proxy infrastructure configuration (see image below), in the case you need to use client certs to communicate with the host, once you provide them to the platform our HA-Proxy will use them to communicate with the end host meaning that no logic should be added on your service to handle this certificates.

608 https://github.com/NHSDigital/api-management-infrastructure/blob/master/terraform/_shared/vars/ptl.tfvars#L312



Once the infrastructure configuration is in place and you are using the right host to make the calls in your container your service should now be able to reach the host int.api.service.nhs.uk.

NHS Digital LDAP and HSCN

APIM does have a routing method to HSCN (via the existing AWS Direct Connect, using the DX VPC and DX-HAProxy service) - it is **currently** only configured to work with LDAP and does **not** use the egress proxy routing and config for external internet. If you need HSCN contact APIM Producer Support.

7.14.5.11 Async to sync conversion (sync wrap)

7.14.5.11.1 Overview

Use our "sync wrap" component to present a synchronous endpoint for a backend endpoint that is asynchronous (and uses "short polling"). For context around sync and async APIs, see Interaction patterns (see page 656) and for reference about the HL7-FHIR Asynchronous request pattern visit Async - FHIR v4.0.1 (hl7.org)[609].

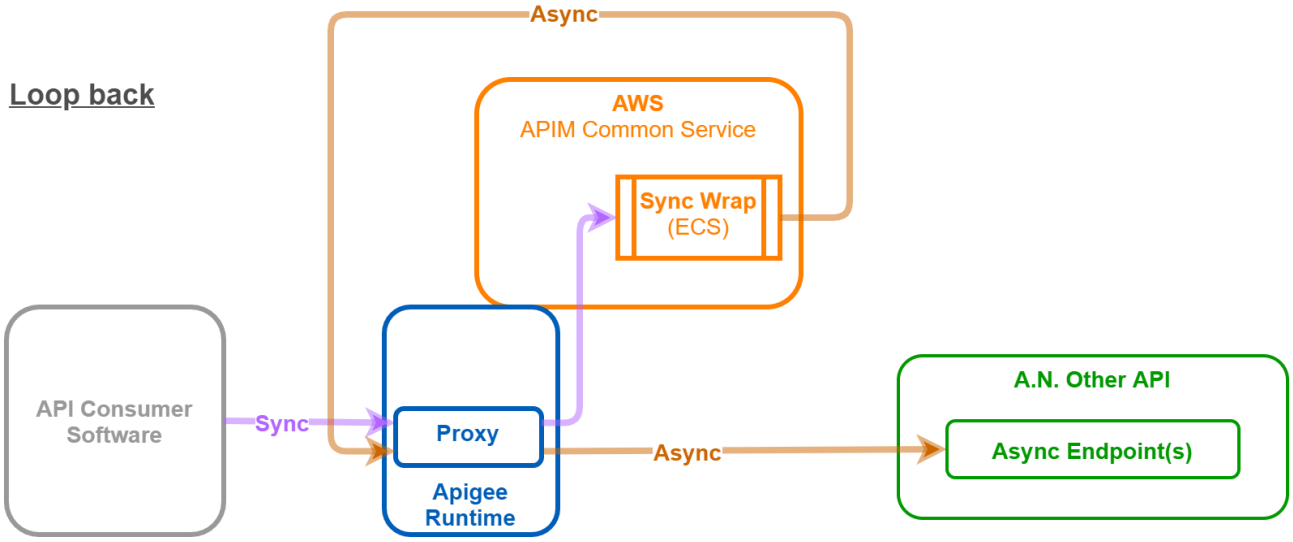
7.14.5.11.2 How the sync-wrap component works

We can simplify the functionality of the sync-wrap component to the following steps:

1. Calling application calls API.
    a. Optionally, it includes a header `X-Sync-Wait` to specify the timeout in seconds (default=5 seconds).

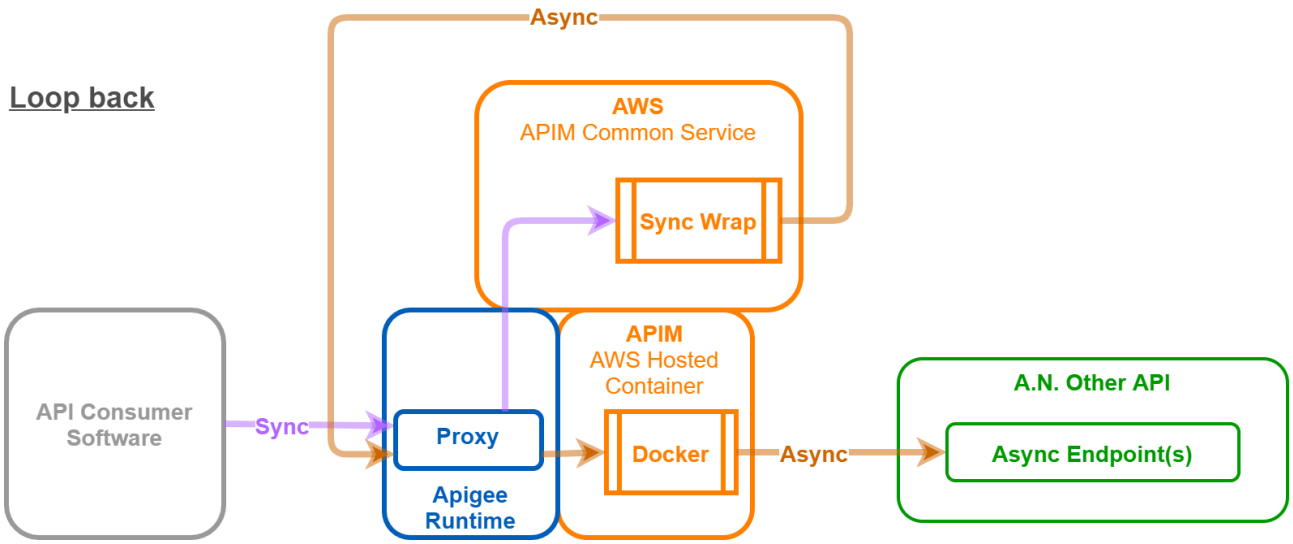
609 https://www.hl7.org/fhir/async.html

2. Because the calling application didn't include a header of `Prefer=respond-async`, the API proxy forwards the request to the Sync Wrap Apigee Proxy, which then routes the request to the Sync Wrap AWS Hosted Container to do the processing
3. Sync wrap calls the API again, triggering async behavior by including a header of `Prefer=respond-async.`
4. API proxy, seeing the Prefer header, calls the asynchronous backend API, and receives a polling URL (or an error) in response
5. API proxy passes the response back to sync wrap
    a. If it receives an error from the backend, it propagates it forward
6. Unhappy path (error response received): sync wrap propagates error back to calling app, via API proxy
7. Happy path: sync wrap polls polling URL (on API proxy) for a response
    a. If it times out, it returns a 504 (Gateway Timeout), which is propagated via API proxy to the calling app
8. When sync wrap receives a response, it propagates it forward, via API proxy, to the calling app



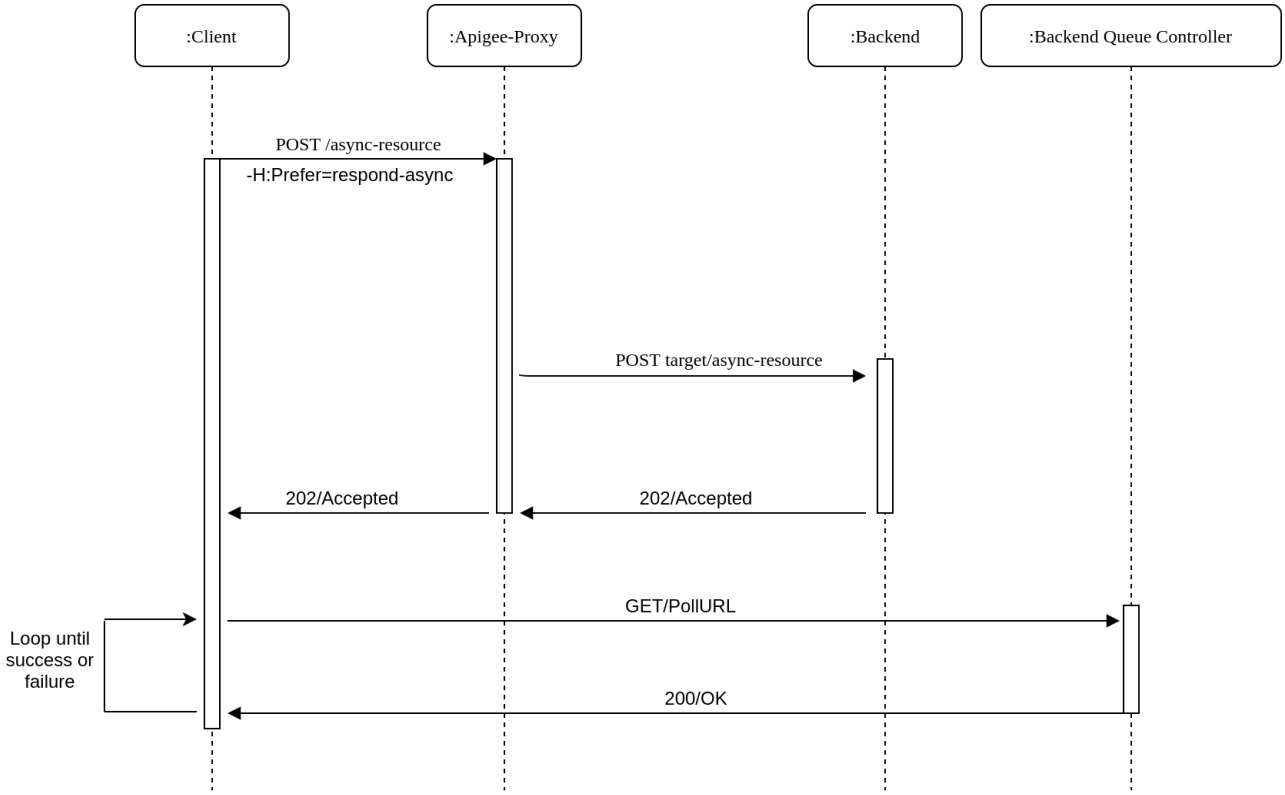
It also possible to use of the APIM platform capability to provide a Proxy with a linked AWS Hosted Container to run a Docker image to process requests. One of the goals of the loop back model was to remove the need for API Producers to build their own Sync Wrap container.

Where the loopback model is used, the Docker image should interact with the backend using the agreed async pattern, and the APIM Sync Wrap solution will handle the Sync to Async conversion. Such a solution would look like:

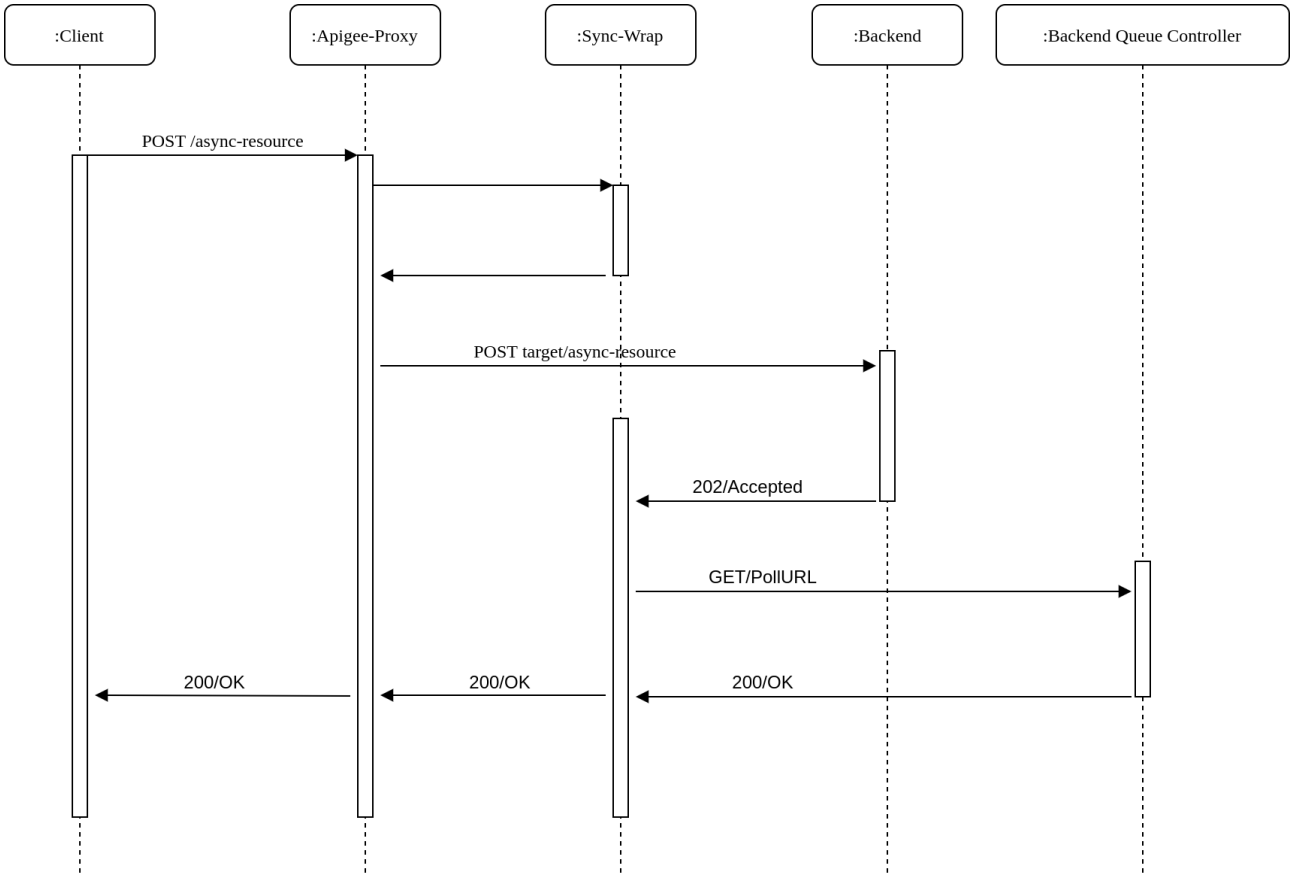


7.14.5.11.3 Interactions with sync-wrap

The client can always choose to not use the sync-wrap component and do the polling by itself. This can be achieved adding the header "Prefer=respond-async" into the call. In this case the proxy will forward the call to the endpoint and the client will receive the 202/Accepted response along with the poll URL. See example below.



When using sync-wrap, the component adds the corresponding header and it takes care of the polling retrieving to the client the corresponding response.



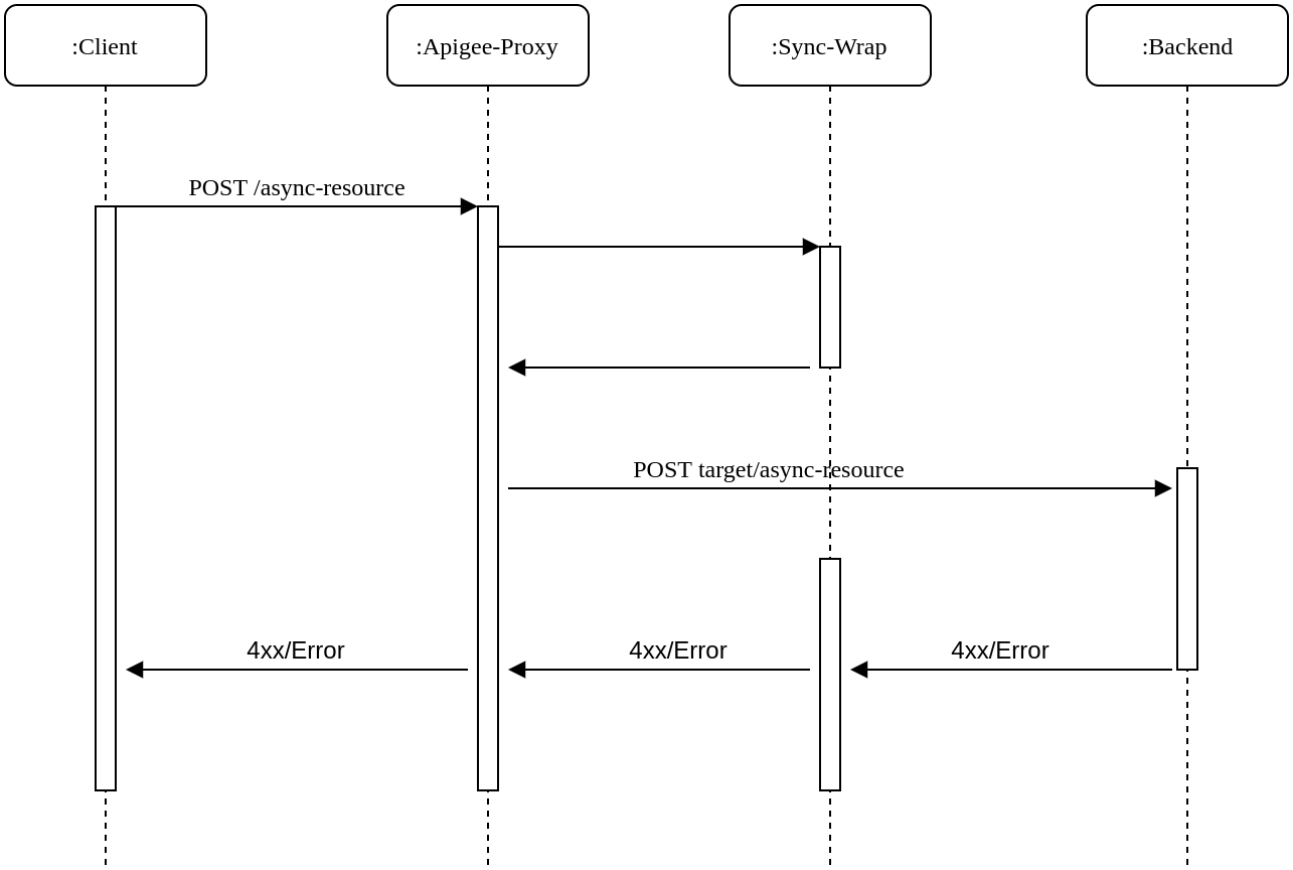
7.14.5.11.3.1 Error handling and Sync-Wrap

In the case of an error occurring during the process, the corresponding status code response will be:

- Default is to just propagate the error to the client through the proxy
- Except when the error can't be corrected by the client - then a **503** with Retry-After header should be used
    - Sync wrap ended up implementing **504**, not 503 - tech debt ticket needed

The use of 5xx instead of 4xx was difficult to agree for guidance and further reading see RESTful HTTP design points (see page 365).

An example of simple propagation:



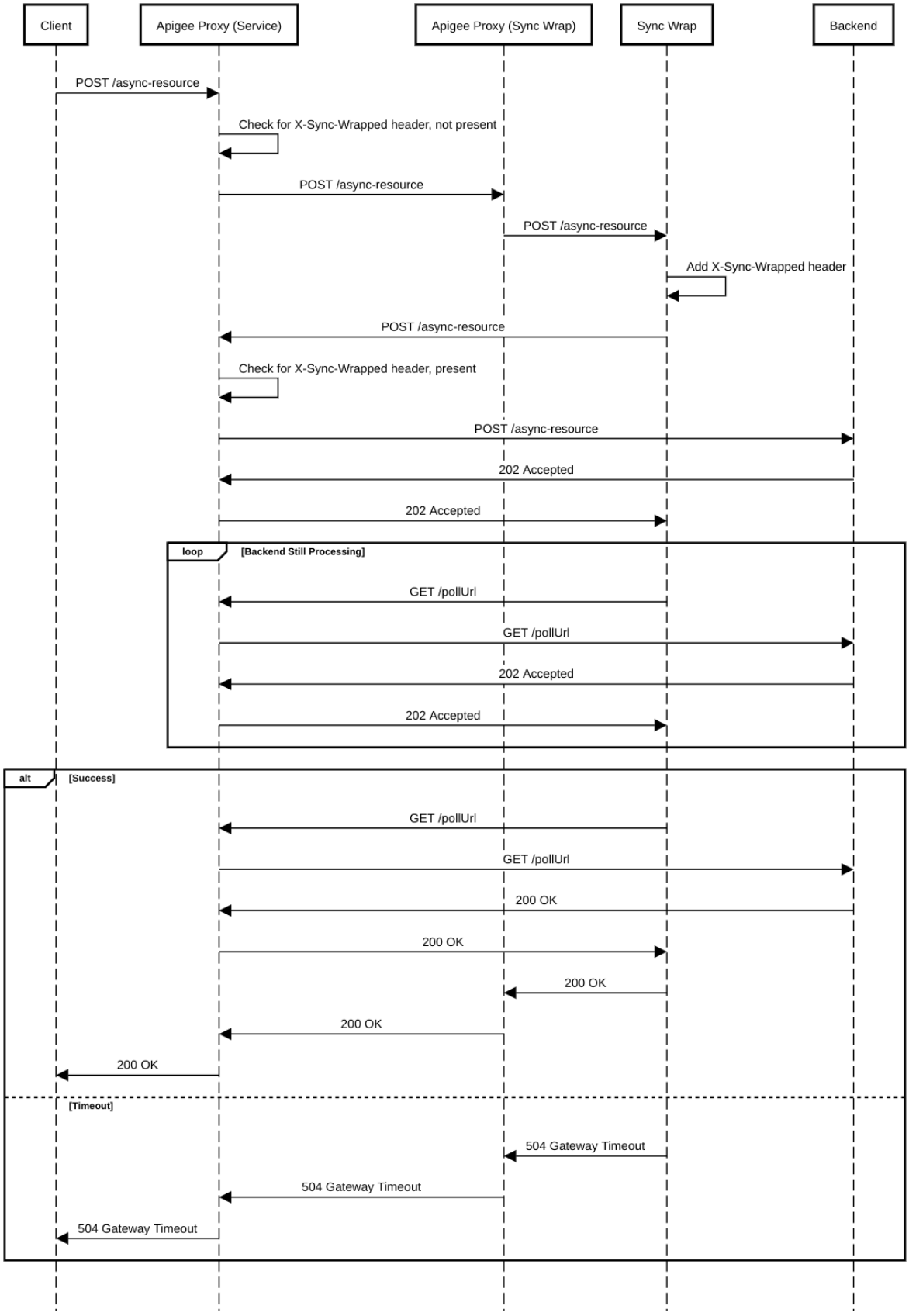
Full interaction diagram

The following sequence diagram covers:

- The full polling sequence

- Timeout example where a 504 is returned

Sync Wrap


Client
Apigee Proxy (Service)
Apigee Proxy (Sync Wrap)
Sync Wrap
Backend
POST /async-resource
Check for X-Sync-Wrapped header, not present
POST /async-resource
POST /async-resource
Add X-Sync-Wrapped header
POST /async-resource
Check for X-Sync-Wrapped header, present
POST /async-resource
202 Accepted
202 Accepted
loop
[Backend Still Processing]
GET /pollUrl
GET /pollUrl
202 Accepted
202 Accepted
alt
[Success]
GET /pollUrl
GET /pollUrl
200 OK
200 OK
200 OK
200 OK
200 OK
[Timeout]
504 Gateway Timeout
504 Gateway Timeout
504 Gateway Timeout

7.14.5.11.4 How to implement sync wrap in your API

7.14.5.11.4.1 Creating a target endpoint

The first step will be to create a new target endpoint for sync-wrap and configure it (see example code below). Note that this will be pointing to the sync-wrap default proxy in Apigee.

sync-wrap.xml

```
<TargetEndpoint name="sync-wrap">
  <FaultRules>
    <FaultRule name="access_token_expired">
      <Step>
        <Name>ExtractVariables.OAuthErrorFaultString</Name>
      </Step>
      <Step>
        <Name>AssignMessage.OAuthPolicyErrorResponse</Name>
      </Step>
      <Condition>oauthV2.OauthV2.VerifyAccessToken.failed</Condition>
    </FaultRule>
  </FaultRules>
  <PreFlow>
    <Request>
      <Step>
        <Name>OauthV2.VerifyAccessToken</Name>
      </Step>
      <Step>
        <Name>Quota</Name>
      </Step>
      <Step>
        <Name>SpikeArrest</Name>
      </Step>
      <Step>
        <Name>AssignMessage.AddSyncWaitHeader</Name>
      </Step>
    </Request>
  </PreFlow>
  <LocalTargetConnection>
    <APIProxy>sync-wrap-{{ APIGEE_ENVIRONMENT }}</APIProxy>
    <ProxyEndpoint>default</ProxyEndpoint>
    <Path>/sync-wrap/{{ SERVICE_BASE_PATH }}</Path>
  </LocalTargetConnection>
</TargetEndpoint>
```

7.14.5.11.4.2 Using sync-wrap in your proxy

Once your sync-wrap target is in place and configured, you can use a RouteRule to route the consumer requests for <your-async-endpoint> to the sync-wrap target. You can implement a condition to do so (see code example below). Then the target will handle the request and will propagate the response accordingly to your proxy.

default.xml

```
   <RouteRule name="sync-wrap">
    <Condition>
      request.verb = "POST" AND (proxy.pathsuffix MatchesPath "/<your-async-
endpoint>") AND request.header.prefer != "respond-async" AND request.header.x-sync-
wrapped != "true"
    </Condition>
    <TargetEndpoint>sync-wrap</TargetEndpoint>
  </RouteRule>
```

7.14.5.11.4.3 Optional configuration

In the PreFlow of your sync-wrap target endpoint, you can optionally set up AssignMessage.AddSyncWaitHeader (see example code below). This policy will add the header "X-Sync-Wait" used by the module to set up the time-out for the asynchronous request. The value of the header "X-Sync-Wait" should be a number between 0.25 and 59.

AssignMessage.AddSyncWaitHeader.xml

```
<AssignMessage async="false" continueOnError="false" enabled="true" name="AssignMessa
ge.AddSyncWaitHeader">
  <Add>
      <Headers>
          <Header name="X-Sync-Wait">59</Header>
      </Headers>
  </Add>
  <IgnoreUnresolvedVariables>false</IgnoreUnresolvedVariables>
  <AssignTo createNew="false" transport="http" type="request"/>
</AssignMessage>
```

NOTE The polling URL used by the sync-wrap component needs to be an absolute URL. Depending on your backend service you may recieve a relative URL, in which case you need to add some logic within your Apigee proxy to convert it.

7.14.5.11.5 Interaction patterns

Modern web standards prefer synchronous communications, and that is the strategy used for API Management. The choice of strategy, though, has implications: infrastructure, scaling and cost.

To help manage and understand the situation, the following two examples are ways of splitting up the problem:

Query	Give me the information about this person
Command	Update this person with the following
Event	Tell me when the person moves home

<30 seconds	Quick (but not for user interfaces)
>30 seconds to a few minutes	Slow
> few minutes	Long running or event style

The use of the word "asynchronous" is ambiguous, and this page focuses on the **first** definition:

- Transport level (HTTP in this context) asynchronous pattern
- Event or messaging style of interaction - core to the FHIR Messaging paradigm

7.14.5.11.5.1 Preferred model - Synchronous

The API Management Platform will expose **synchronous HTTP APIs** for consumption by API consumers. API producers of new APIs should therefore look to design APIs which behave synchronously.

(see: <Need to update D010 - Sync vs Async & Short duration requests[610] with the details from KAD 8362>)

610 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110865159/D010+-+Sync+vs+Async+Short+duration+requests

7.14.5.11.5.2 Hiding asynchronous API behaviour from API consumers

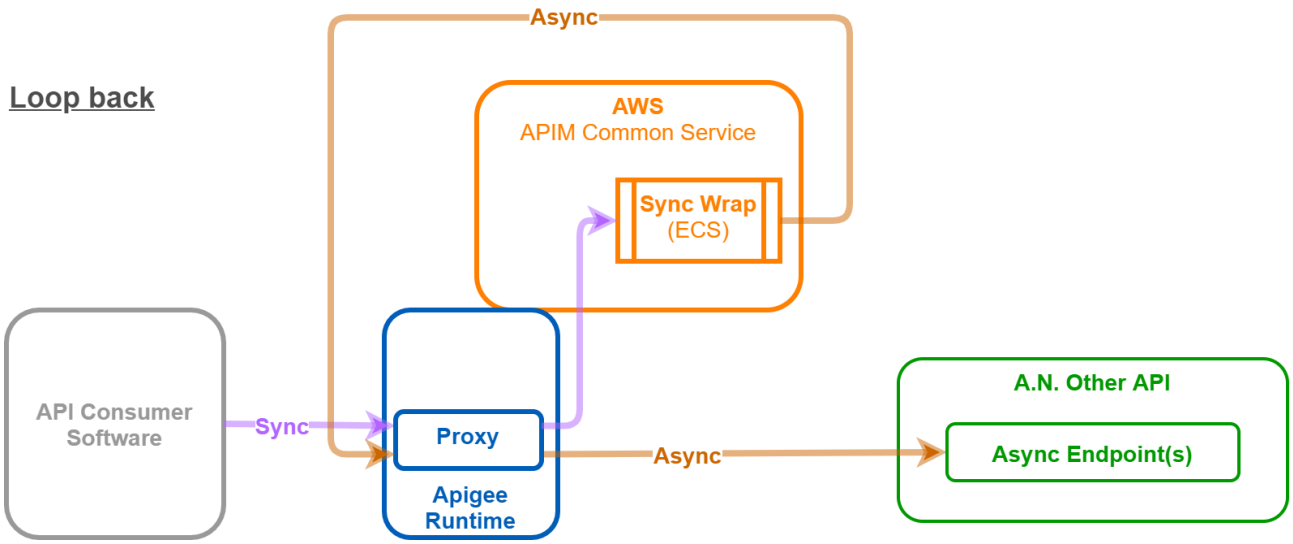
Where an existing API is being migrated onto the API Management platform has asynchronous behaviours, and if is agreed that this asynchronous behaviour of the backend can't be removed (see zzzz for detailed considerations), then an APIM provided loop-back sync wrap pattern will be implemented in order to present a Synchronous API to API consumers.

A common example of an API with asynchronous behaviour is one in which an business transaction or update is implemented in two stages, rather than one. For example, a update operation to an API which exposes a interface to a data repository:

Stage 1: Client submits request to API- An HTTP Request/Response exchange where the response says "Received your update request.."

Stage 2: Once the server has completed the update request, the server initiates a separate client request to the request initiator in order to deliver the result. The client described in stage 1 must stand up an HTTP server to listen for these responses.

The various "synchronous wrapper" described below provides a solution to this common two stage asynchronous behaviour where Stage 2 above is not seen by the API client. The result of Stage 2 is captured by the "wrapper" and provided synchronously in the result of the Stage 1. I.e. Stage 1 becomes "Received your update request, and this is the result.."



API Consumer view

Very small changes in these interactions can have a significant impact on client software, and for Asynchronous interactions there are many existing models. For the exact definitions which will be used across the API Management Platform see:

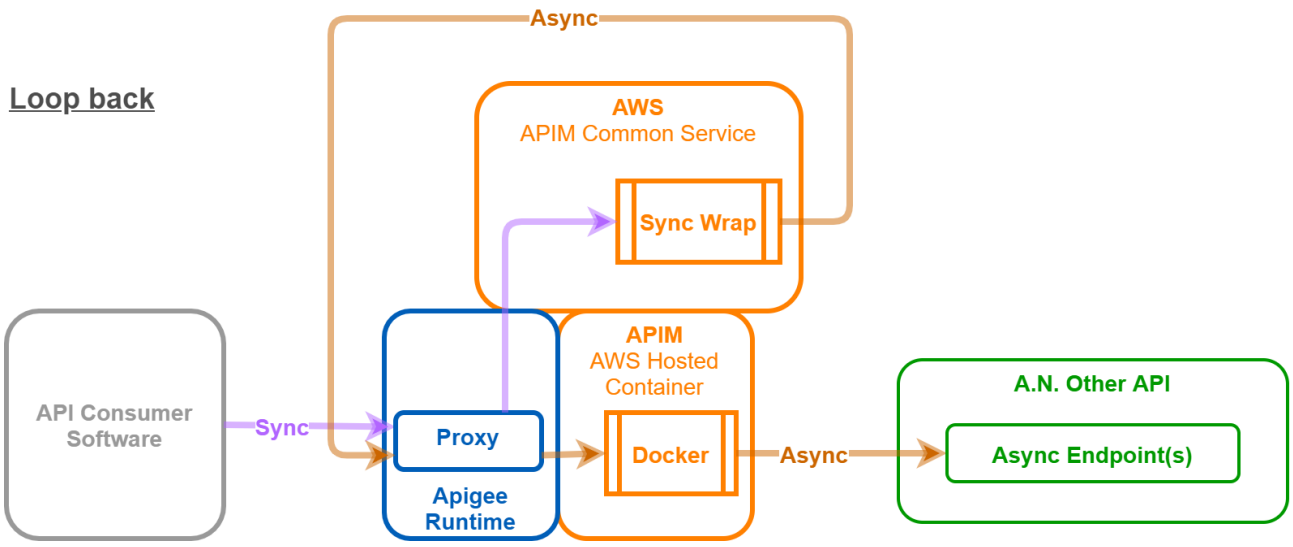
- Synchronous: An HTTP Request/Response exchange as described in the HTTP/2 specification: https://tools.ietf.org/html/rfc7540#section-8.1

- Asynchronous: yyyy

Loopback model, with an APIM AWS Hosted Container

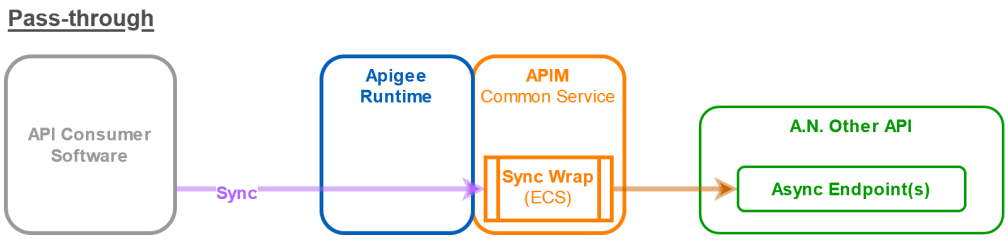
Some API Producers will be making use of the APIM platform capability to provide a Proxy with a linked AWS Hosted Container to run a Docker image to process requests. One of the goals of the loop back model was to remove the need for API Producers to build their own Sync Wrap container.

Where the loopback model is used, the Docker image should interact with the backend using the agreed async pattern, and the APIM Sync Wrap solution will handle the Sync to Async conversion. Such a solution would look like:

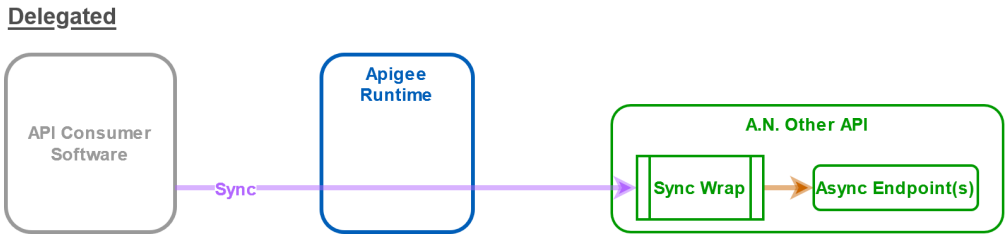


7.14.5.11.5.3 Alternative sync wrap patterns

There are other patterns in use, but APIM and API producers should target the loop back model. The following two patterns are noted here for reference as they do exist currently:



Pass-through is currently used by the SCR API.



7.14.5.11.5.4 Detail of interaction models for both

There are two main synchronous interaction models in the table below, and APIM has ended up on focusing on two other interaction patterns. The API Management support will be for:

- Synchronous (option A or B)
- Asynchronous Request - Reply (option D)

The primary patterns are:

	API Consumer View	Message De-dup / Replay Protection?**	200 possible?	202 possible?	Get previous result	Max time
A	Synchronous	N	Y	N	N	30s (5 default?)
B	Synchronous	Y	Y	N	Option	30s
C	Sync+Async fallback	Y	Y	Y	Y	15mins
D	Async Request - Reply	Y	N	Y	?	30s?

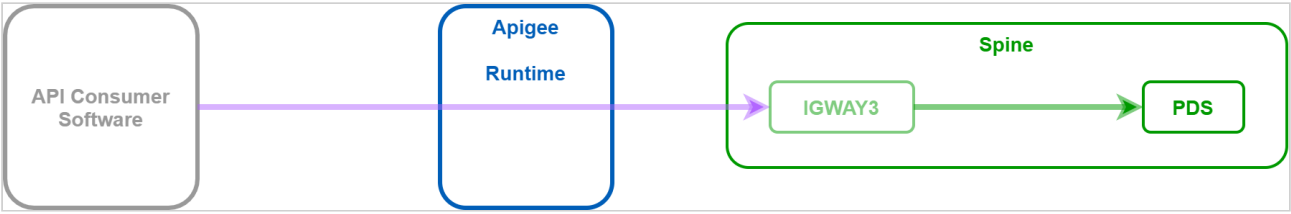
*** Some interfaces will support a client provided unique request ID, which would enable variants around protecting against accidental re-sends, etc. To do this requires infrastructure provided by the API backend, it will not be provided by Apigee.*

7.14.5.11.5.5 Spine implemented solutions

The following variants are being used in Spine - some of these were developed prior to APIM as ways to test synchronous operations against Spine.

Spine provided a truly synchronous operation

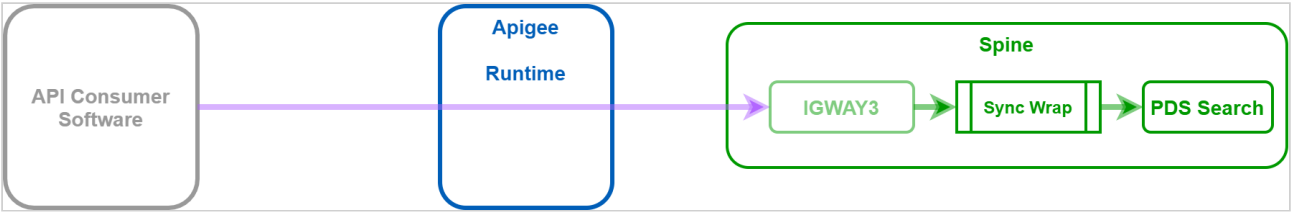
Current usage: PDS Retrieve



Spine provided Sync wrapper around Async

This holds the connection open, uses Spine queues to string a sequence of request together. The overall result of success / fail is fast enough that the the request has a short time frame. The API Consumer simply sees an Synchronous API (Option A or B).

Current users of this: PDS Search; Reasonable Adjustments



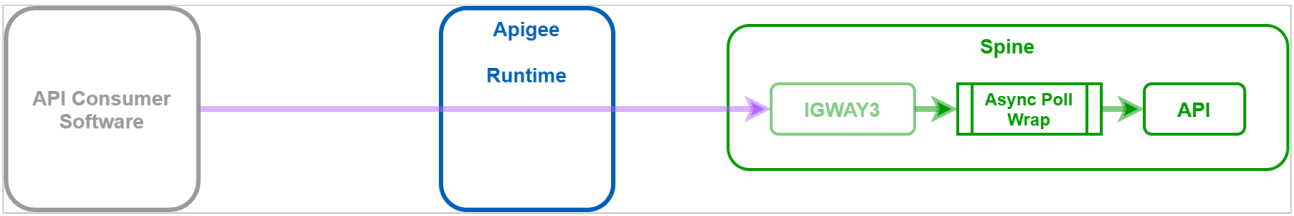
Spine provided Async Request Reply Polling wrapper

This is Option D above:

- See this page for the agreed Async details: RESTful Interface Design[611]
- On the current public facing PDS API the mechanism is also described: API Catalogue[612]
- Currently using Message ID as the unique request key. Keeps this for 56 days. If a subsequent request, with same Message ID is made, then will simply return the original polling URL.
- Uses Spine internal ID to manage the state of the original request. This is the value returned in the polling response

Current usage: PDS Update

611 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97763771/RESTful+Interface+Design
612 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir



7.14.5.11.5.6 Detailed design, options analysis and background

- D035 - Sync to Async Wrapper Pattern[613]
- D010 - Sync vs Async & Short duration requests[614]
- D015 - Long Running Async & Events[615]

7.14.5.12 Using multi-line secrets in the pipeline

7.14.5.12.1 Prerequisites

- a Secret stored in AWS Secret Store

7.14.5.12.2 Problems with using Azure DevOps variable sets

Currently, Azure DevOps does not support multiline secret task variable to be specified. So when we retrieve the variable from AWS secret store and set that variable using the ADO variable set, it will cut all but the first line of the value.

7.14.5.12.3 Workaround

We have added `secret_file_ids`. Putting a secret id in this list will cause it to be downloaded as a file, with the path to that file saved to the variable in Azure DevOps.

For multiline secrets, the secret name is instead the path to a file that contains the secret.

So firstly, specify the multiline secret within the pipeline, for example:

613 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/155880626/D035+-+Sync+to+Async+Wrapper+Pattern
614 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110865159/D010+-+Sync+vs+Async+Short+duration+requests
615 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110872564/D015+-+Long+Running+Async+Events

```
secret_file_ids:
      - ptl/temp/apm-1509
```

During either the build, pull request or release in which the secret is specified Azure DevOps will retrieve the secret and store it within a file.

It is stored within the artifact workspace directory under '/secrets':

```
$(Pipeline.Workspace)/secrets
```

e.g.

```
/secrets
    - multiline_secret_a
    - multiline_secret_b
    - multiline_secret_c
```

For example, the pipeline retrieving the multiline secret will look like:

```
Storing multiline secret...
File name saved to var: apm-1509
```

To use that secret value, you can use the file name variable, in this case, it would be 'APM-1509' to reference the file that the secret stored in and retrieve the value.

For example:

```
MULTILINE_SECRET="`cat $(apm-1509)`"
```

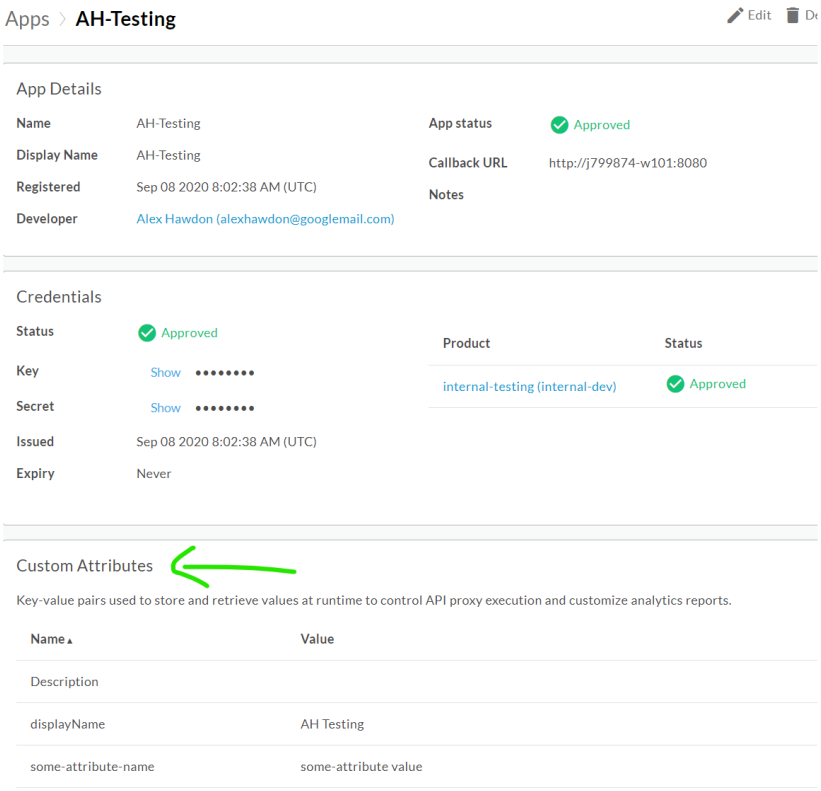
7.14.5.13 ExtendedAttributes shared flow

7.14.5.13.1 Introduction

Apigee has the concept of *custom attributes* , but for a number of reasons the APIM pattern is that you should **not** directly access the custom attributes. Instead only do this indirectly via an APIM shared flow.

For an overview, and reasoning see Configuring extended attributes for API Consumer Apps (see page 806)

You can see the *Custom Attributes* on Applications in Apigee, in the Apigee Edge UI:



7.14.5.13.2 Calling the shared flow

In your proxy or target XML add a step that calls the ***FlowCallout*** immediately after the authorization step e.g. ***OAuth2VerifyAccessToken*** or ***VerifyAPIKey***. This identifies the calling app and exposes app information including its custom attributes.

default.xml

```
<TargetEndpoint name...>
    <PreFlow>
        <Request>
            <Step>
                <Name>OauthV2.VerifyAccessToken</Name>
            </Step>
            <!-- Must be placed after Authentication -->
            <Step>
                <Name>FlowCallout.ExtendedAttributes</Name>
            </Step>
...
```

In your policies folder add an XML file defining the ***FlowCallout***.

7.14.5.13.2.1 FlowCallout policy - OAuthV2 authentication

If you are using an ***OAuthV2.VerifyAccessToken*** policy to authorize users, add the following to your ***policies*** folder.

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
</FlowCallout>
```

7.14.5.13.2.2 FlowCallout policy - VerifyApiKey authentication

If you are using a ***VerifyApiKey*** policy, you will **need to provide its name as a parameter** (the name is defined as an attribute in the root element of your ***VerifyApiKey*** policy).

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
     <Parameters>
      <Parameter name="verifyApiKeyPolicyName">NameOfYourVerifyApiKeyPpolicy</
Parameter>
    </Parameters>
</FlowCallout>
```

Why do I need to provide the name of the VerifyApiKey policy?

Because Apigee assigns the app/product flow variables inside a namespace which includes the policy name for ***VerifyApiKey*** but not for ***OAuthV2.VerifyAccessToken***

7.14.5.13.3 Setting extended attributes on an application

7.14.5.13.3.1 Setting extended attributes

Create a custom attribute with a key/name of `apim-app-flow-vars` and a value in an object format which is valid JSON.

- The top-level key should be your API Proxy's `short_service_name` - typically defined in your API Proxy's `azure/project.yml` file. For example 'pds' in the case of the Personal Demographics FHIR API.
- The maximum size of the entire object is 2KB - keep names and values as short as possible!
- Arrays **are** supported

Example:

apim-app-flow-vars

```
{
    "pds": {
        "ratelimiting": {
            "quota": "1000pm"
        }
    }
}

#array example

{
    "immunisation-history": {
        "authorised_targets":
            ["COVID19", "HPV"]
    }
}
```

7.14.5.13.3.2 Reading extended attributes

In your flow you can now access your unravelled attribute from above like such:

apim-app-flow-vars

```
apim-app-flow-vars.pds.ratelimiting.quota = 1000pm
```

7.14.5.13.4 Error handling

7.14.5.13.4.1 Default error message

If you provide malformed JSON in an Application's `apim-app-flow-vars` the proxy will return as the body of a HTTP 500 response:

InvalidJson

```
{
  "error": "unknown_error",
  "error_description": "An unknown error occurred processing this request. Contact us
quoting the messageId for assistance diagnosing this issue: https://digital.nhs.uk/
developer/help-and-support.",
  "message_id": "{messageId}"
}
```

7.14.5.13.4.2 Customised error handling

To customise the error message implement a conditional fault rule[616] in your calling flow that checks for `javascript.Javascript.SetExtendedAttributes.failed equals "true"`

For example:

```
<ProxyEndpoint name="default">
        ...
        <FaultRules>
           <FaultRule name="extended_attributes_failure">
                <Step>
                    <Name>AssignMessage.ExtendedAttrsCustomError</Name>
                </Step>
                <Condition>(javascript.Javascript.SetExtendedAttributes.failed equals
"true")</Condition>
             </FaultRule>
             ...
        </FaultRules>
```

616 https://docs.apigee.com/api-platform/fundamentals/fault-handling#simplefaulthandlingexample-2createthefaultrulethatwilltriggerthepolicy

7.14.5.14 Storing secrets securely

7.14.5.14.1 Uploading secrets

As you will not have direct access to upload secrets to the secrets store, you will need to have a member of the APIM Deathstar Team do it on your behalf. Please use the 'Create/Update Secret' workflow in #platforms-apim-producer-support. For guidance on how to use workflows please see the pinned message in the channel. (For slack access please see Help and support (see page 2056)).

Note: All secrets need to be JSON formatted - see Per-environment configuration (see page 600) for secret formatting guidance.

7.14.5.14.1.1 Naming your secrets

Shared secrets

If your secret is shared, its name will depend heavily on what its use is, but most usually live in `<account>/platform-common/*` Consult the APIM Support engineers for more information.

Secret for a specific API

The prefix for a given api `foo` would be `<account>/api-deployment/foo/*`

TLSMA Certificates for a Target Server

Refer to the Setting up a target server (see page 607) (section 3.2) for the naming convention for keys and certificates.

7.14.5.14.1.2 Transmitting secrets securely

Encrypted zip

This is the easier but slightly less secure method.

1. Create an encrypted zip with a strong password containing the secrets you wish to send
2. Email the zip to the support engineer
3. Send the password in a separate message via a different channel (e.g. slack) and delete the message after you have confirmed the support engineer has it.

7.14.5.15 EnhancedVerifyApiKey shared flow

7.14.5.15.1 Overview

Apigee includes the VerifyAPIKey policy for authenticating API keys. However, a critical flaw has been identified within this policy: it successfully verifies API keys even if an application has no products subscribed to the API key, allowing access to production API key proxies. Every app starts like this in the developer portal before a consumer assigned a product to their app. We actually rely on this being possible for creating new apps on the developer portal and for cleaning up products. This loophole poses a significant security risk, as it permits unauthorized access to sensitive resources.

To address this issue, we have developed a shared flow designed to enhance the VerifyAPIKey policy. This shared flow acts as a protective wrapper, imposing additional validation checks to ensure that API keys without subscribed products cannot access production proxies. Additionally, it verifies the validity of API keys to prevent access with revoked or invalid keys.

The implementation of this shared flow is paramount for safeguarding the integrity of our production proxies and mitigating the risk of unauthorized access.

7.14.5.15.2 Calling the shared flow

In your proxy or target XML add a step that calls the ***FlowCallout*** during the authentication steps in your proxy.

default.xml

```
<TargetEndpoint name...>
    <PreFlow>
        <Request>
            <Step>
                <Name>FlowCallout.EnhancedVerifyApiKey</Name>
            </Step>
```

The EnhancedVerifyApiKey policy is configured to default to the "apikey" header for authentication. If your header specification differs, such as requiring "X-API-Key" or similar, please manage this customization within your proxy by explicitly setting the "apikey" header with the corresponding value.

In your policies folder add an XML file defining the ***FlowCallout*** as shown below.

7.14.5.15.2.1 FlowCallout policy - EnhancedVerifyApiKey

Add this to your policies folder:

FlowCallout.EnhancedVerifyApiKey.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
nhancedVerifyApiKey">
    <DisplayName>FlowCallout.EnhancedVerifyApiKey</DisplayName>
    <SharedFlowBundle>EnhancedVerifyApiKey</SharedFlowBundle>
</FlowCallout>
```

7.14.5.15.3 Error handling

If you try to access a proxy with an API key which has no products subscribed to it you will see the following error message:

```
{
    "status": 401,
    "error": "no_products",
    "error_description": "No products associated with apikey"
}
```

7.14.5.16 Handling Messaging Response Error code - For Review

A range of error conditions may occur while API proxies process requests from applications. For example:

- Network issues may arise when communicating with backend services.
- Applications might present expired or invalid credentials.
- Request messages could be incorrectly formatted or fail to meet the required standards.

In such instances, if an error occurs after a client application has invoked an API proxy, an appropriate error message will be returned to the client application. It is essential to handle these errors gracefully by interpreting the returned messages and implementing robust error-handling mechanisms within your application.

General Best Practices

1. **Implement Logging and Monitoring**
   Log all request/response details (without storing sensitive data inappropriately) so you can troubleshoot quickly when you encounter error codes.
2. **Use Retries with Exponential Backoff**
   Especially for 5xx errors, implement retry logic to handle transient server or network issues.
3. **Follow Security & Rate Limit Guidelines**
   Always use valid credentials, handle expired or revoked tokens properly, and respect the rate limit[617] specified by API Platform to avoid 401, 403, or 429 errors.

617 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

4. **Engage with Support**
   If errors persist or are unclear, gather diagnostic information (timestamp, request IDs, logs) and contact the NHS API teams or the NHS England API support team.

7.14.5.16.1 4xx status codes

4xx status codes in HTTP indicate that there was an error with the **client's request** to the server. This means the issue lies on the side of the user making the request, not with the server itself and could stem from issues like:

- Incorrectly formatted requests
- Missing or invalid data
- Unauthorized access attempts
- Requesting a resource that doesn't exist

Common examples of 4xx status codes, it causes, how to triage and fix them can be found below:

400: INVALID_VALUE

Cause
A required parameter is invalid (for example, an access token is invalid). Ensure the header parameter is valid and conforms to your API's endpoint specification.

Triage Steps

1. Compare the header parameters in your request against the API documentation.
2. Check that the header content (for example, the token format) matches what the API specification expects.
3. Confirm the request syntax (URL, body payload, etc.) and verify there are no typos or formatting issues.

Fix

- Correct any incorrect parameters or values so they conform exactly to the API specification.
- If you suspect an expired or invalid token, obtain a new token and retry.

400: MISSING_VALUE

Cause
A required parameter (for example, the access token) is missing entirely. Ensure the header parameter is populated and conforms to your API's endpoint specification.

Triage Steps

1. Inspect the headers in your request to identify any missing required fields.
2. Cross-check against the API documentation to confirm all mandatory headers are present.

Fix

- Add the missing header parameter in the correct format.
- Ensure any environment variables (if used) are correctly set and not empty.

401: ACCESS_DENIED

Cause
Authorisation credentials are not valid (for example, a wrong token or invalid client ID/secret). Make sure you follow the instructions in the appropriate security pattern [618]for your API's access mode.

Triage Steps

1. Ensure you are using the correct security pattern for your API's access mode (OAuth2, app-based auth, etc.).
2. Double-check that the credentials (token, keys) have not expired or been revoked.
3. Confirm you are pointing to the correct environment (sandbox vs. production).

Fix

- Update or refresh your tokens/credentials.
- Use the correct client ID and secret or the correct authorisation flow as outlined in the API's security documentation.

401: UNAUTHORISED

Cause
Authorisation credentials are missing or not valid in request. Make sure you follow the instructions in the appropriate security pattern[619] for your API's access mode.

Triage Steps

1. Check the request headers to confirm an Authorisation header is included.
2. Verify your method of sending credentials (Bearer token, basic auth header, etc.) is correctly implemented.

Fix

- Supply the required credentials, typically via an Authorisation: Bearer <token> header or other pattern used by the API.
- Re-test to confirm the credentials are included in every request.

403: ACCESS_DENIED

Cause
You have valid credentials, but do not have permission to perform the requested action. Make sure you follow the instructions in the appropriate security pattern[620] for your API's access mode.

Triage Steps

1. Check if your credentials or user role is restricted from accessing this particular endpoint or resource.
2. Review the API's documentation or contact an administrator to see if additional scopes or roles are needed.

618 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation
619 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation
620 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation

Fix

- Request the necessary permissions or roles from the API provider.
- Update any OAuth2 scopes or client configurations if required.

403: FORBIDDEN

Cause
Similar to the above, the user or application is not permitted to carry out the specific action or operation. Make sure you follow the instructions in the appropriate security pattern[621] for your API's access mode.

Triage Steps

1. Confirm the endpoint you are calling is actually accessible to your user/app role.
2. Look for any custom policy or rule that might be blocking this request.

Fix

- Obtain higher-level privileges or the correct role assignment.
- Check any policy or firewall rule that might be preventing access to that endpoint.

404: RESOURCE_NOT_FOUND

Cause
The requested resource does not exist or cannot be found.

Triage Steps

1. Verify the endpoint URL is correct (including path parameters or query strings).
2. Check whether the resource actually exists; if it's a patient record or prescription, confirm you have the correct ID or reference.

Fix

- Correct the URL, resource ID, or parameters in the request.
- Make sure the resource is present, and you have permission to view it.

409: REC_CONFLICT

Cause
There is a conflict with the current state of the resource, such as a task ID or prescription ID that already exists. E.g request contains a task id and prescription id identical to a record already in the data store.

Triage Steps

1. Ensure you are not creating a duplicate record or reusing a unique ID that has already been used.
2. Check if you must first delete or update an existing record before creating a new one.

Fix

- Use a unique ID or follow the correct workflow to update existing resources.

---

621 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation

- Handle concurrency by checking the current state of the resource before making changes.

429: TOO_MANY_REQUESTS

Cause
Your application has exceeded the allotted rate limit[622].

Triage Steps

1. Review the rate limits specified in the API documentation.
2. Monitor your requests to ensure you are not sending more than the permitted threshold within a given time window.

Fix

- Implement backoff or retry logic, spacing out requests to stay within the limit.
- Where applicable, request a higher rate limit from the API provider, or optimise your calls to reduce frequency.

7.14.5.16.2 5xx status codes

5xx status codes are a class of HTTP response codes that indicate there was a problem with the **server** that is fulfilling the request and could stem from issues like:

- High Traffic/Resource intensive processes.
- Software issues
- Hardware issues
- Network issues
- Security issues
- Maintenance or Server downtime

Common examples of 5xx status codes, it causes, how to triage and fix them can be found below:

500: INTERNAL_SERVER_ERROR

Cause
A generic error occurred on the server, and it could not handle the request properly.

Triage Steps

1. Retry the request after a short delay to see if the issue is transient.
2. Check the API's status page or known issues board (if available) to see if there is an ongoing incident.
3. Inspect your request payload and headers to confirm there is no hidden client-side error.

Fix

- If the issue persists, contact NHS England API support with details (request logs, timestamps).
- Implement general retry logic in your code to handle unexpected server issues gracefully.

622 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

501: NOT_IMPLEMENTED

Cause
The server either does not recognise or support the request method, which it cannot handle. (The only methods that servers are required to support are GET and HEAD.)

Triage Steps

1. Confirm the HTTP method (GET, POST, PUT, PATCH, DELETE) is one that the API actually supports.
2. Review the API specification to ensure you are using the correct operation.

Fix

- Change the request method to a supported method.
- In rare cases where the server truly does not support a feature, you may need to contact the API team for clarification or future support.

502: BAD_GATEWAY

Cause
The server, acting as a gateway or proxy, got an invalid response from an upstream server.

Triage Steps

1. Retry the request, as 502 errors are often transient.
2. Check if there are any network issues between your system and the API gateway.

Fix

- Wait briefly, then retry.
- If the error persists, check for official service announcements or contact support.

503: SERVICE_UNAVAILABLE

Cause
The server is currently overloaded or down for maintenance.

Triage Steps

1. Confirm whether there is scheduled maintenance or a known service outage.
2. Check your application logs for the timing of repeated 503 errors.

Fix

- Use exponential backoff for retries.
- Monitor the service status page or contact support to see when the service is back online.

504: GATEWAY_TIMEOUT

Cause
The server, acting as a gateway or proxy, did not receive a timely response from an upstream server.

Triage Steps

1. Retry the request after a short interval.
2. Investigate whether the endpoint is experiencing performance issues or if a large/complex request is timing out.

Fix

- Optimise your request (e.g., reduce payload size if possible).
- If the request is time-sensitive or cannot be reduced, contact the API team to discuss possible timeout configurations or performance constraints.

Other errors:

Below is a list of 2xx and 3xx HTTP status codes that may occur when a data request is made to the API-M authorisation service via the Apigee proxy. These codes indicate successful (2xx) or redirected (3xx) interactions, rather than errors. However, if the number of 2xx and 3xx responses is unusually low, it can signal overall service issues, potentially accompanied by a surge in 4xx or 5xx errors.

2xx – Successful Codes

1. **200 OK**

- The request was processed successfully, and the resulting data (if any) is returned in the response.
- **201 Created**
- The request was successful, and a new resource (e.g. a record or token) was created on the server.
- **202 Accepted**
- The request was received but has not yet been acted upon. It might be queued or processed asynchronously.
- **203 Non-Authoritative Information**
- The server is returning meta-information that is not from the origin server, but from a local or third-party copy.
- **204 No Content**
- The request was successful, but there is no content in the response body (often returned after a successful update).
- **205 Reset Content**
- The request was successful, and the client should reset the document view (often used in forms).
- **206 Partial Content**
- The server is delivering only part of the resource due to a range header sent by the client (often for large file downloads).

(Other 2xx codes exist, but these are the most common in RESTful APIs.)

3xx – Redirection Codes

1. **301 Moved Permanently**

- The requested resource has been moved to a new URL permanently. Future requests should use the new URL.
- **302 Found (or 302 Redirect)**
- The requested resource is located at a different URL temporarily. The client should continue to use the original URL for future requests (though in practice many clients treat 302 as a 303).
- **303 See Other**
- The server directs the client to retrieve a different URI, typically used to redirect after an operation (like a POST).
- **304 Not Modified**
- The client's cached version is still valid and has not changed. No new content is returned.
- **307 Temporary Redirect**
- The requested resource has been temporarily moved, but future requests should still use the original URL.
- **308 Permanent Redirect**
- The requested resource has been permanently moved, and the client must use the new URL from now on.

(Other 3xx codes exist, but these are the most commonly encountered in modern applications.)

Why 2xx and 3xx Matters for API-M Authorisation Service

- **High 2xx/3xx Count:** Indicates that requests are generally successful or being properly redirected—this is normal and desirable.
- **Low 2xx/3xx Count:** Can suggest that clients are encountering more client-side (4xx) or server-side (5xx) errors. A sudden drop might mean the API-M authorisation service is experiencing configuration, rate-limiting, or connectivity issues.

If you observe a low proportion of 2xx/3xx codes relative to 4xx or 5xx, investigate potential causes such as invalid credentials or tokens (leading to 4xx), or internal server or gateway issues (leading to 5xx).

7.14.5.17 SingleAsidApply shared flow

7.14.5.17.1 Overview

API Platform has introduced a new shared flow, **SingleAsidApply**, to facilitate the transition away from managing end-user organizations with ASIDs, allowing ODS codes to be used directly. As part of this shared flow, we will verify the **NHSD-End-User-Organisation-ODS** header in incoming requests to check whether it is included in the allowed list. If the header passes the verification and its ODS code is present in the list maintained within the APIM extended attributes, the service will proceed. Otherwise, a fault will be raised.

7.14.5.17.2 Steps to be added in the API Proxy for SingleAsidApply Shared Flow

In your proxy or target XML add a steps for FlowCallout.ExtendedAttributes that calls the ***FlowCallout*** immediately after the authorization step e.g. ***OAuth2VerifyAccessToken*** or ***VerifyAPIKey*** . This identifies the calling app and exposes app information including its custom attributes. After that add the step for FlowCallout.SingleAsidApply which will verify the allowed list ODS code.

default.xml

```
<TargetEndpoint name...>
    <PreFlow>
        <Request>
            <Step>
                <Name>OauthV2.VerifyAccessToken</Name>
            </Step>
            <!-- Must be placed after Authentication -->
            <Step>
                <Name>FlowCallout.ExtendedAttributes</Name>
            </Step>
            <Step>
                <Name>FlowCallout.SingleAsidApply</Name>
            </Step>

...
```

In your policies folder add an XML file defining the ***FlowCallout***.

7.14.5.17.2.1 FlowCallout policy - OAuthV2 authentication

If you are using an ***OAuthV2.VerifyAccessToken*** policy to authorize users, add the following to your ***policies*** folder.

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
</FlowCallout>
```

7.14.5.17.2.2 FlowCallout policy - VerifyApiKey authentication

If you are using a ***VerifyApiKey*** policy, you will **need to provide its name as a parameter** (the name is defined as an attribute in the root element of your ***VerifyApiKey*** policy).

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
     <Parameters>
      <Parameter name="verifyApiKeyPolicyName">NameOfYourVerifyApiKeyPpolicy</
Parameter>
    </Parameters>
</FlowCallout>
```

7.14.5.17.3 Setting extended attributes on an application

7.14.5.17.3.1 Setting extended attributes

Create a custom attribute with a key-name pair of `apim-app-flow-vars` and a value in an object format which is valid JSON.

- The top-level key should be your API Proxy's `short_service_name` - typically defined in your API Proxy's `azure/project.yml` file. For example 'ers' in the case of the e referral service API.
- The maximum size of the entire object is 2KB - keep names and values as short as possible!
- Arrays **are** supported

Example:

apim-app-flow-vars

```
#array example

{
    "ers": {
        "allowListodsCode": [
            "X09001",
            "X09002"
        ]
    }
```

```
}
```

7.14.5.17.3.2 Setting the NHSD-End-User-Organisation-ODS

apim-app-flow-vars

```
The NHSD-End-User-Organisation-ODS header is mandatory and must be included in the
request header for validation against the allowed list configured in the extended
attributes.
Currently, the implementation supports only a single ODS code in this header. Support
for multiple ODS codes has not been introduced in this version.
```

7.14.5.17.4 Error handling

If you attempt to access a proxy with an **NHSD-End-User-Organisation-ODS** header that is not included in the allowed list, the response will contain an error message that adheres to the API Platform's standard error structure.

```
{
    "error": "service error",
    "error_description": "ODS code is not valid",
    "message_id": "rrt-1158383327385076491-c-geu2-344098-40386047-3"
}
```

If you attempt to access a proxy without the tag **allowListodsCode** setup in `a pim-app-fl ow-var s`, the response will contain an error message that adheres to the API Platform's standard error structure.

```
{
    "error": "service error",
    "error_description": "Invalid Parameter Values.",
    "message_id": "rrt-1158383327385076491-c-geu2-344098-40386047-2"
}
```

If you attempt to access a proxy with an **NHSD-End-User-Organisation-ODS** header but without `a pim-app-fl ow-var s` setup in the app , the response will contain an error message that adheres to the API Platform's standard error structure.

```
{
    "error": "service error",
```

```
    "error_description": "Invalid Header Values.",
    "message_id": "rrt-1158383327385076491-c-geu2-344098-40386047-1"
}
```

7.14.6 Building and deploying your API alpha with Proxy Generator

7.14.6.1 Overview

This page describes the engineering tasks for building and deploying your API using the Proxy Generator as part of the Alpha phase (see page 321) of the API process checklist (see page 227).

The Proxy Generator uses our custom extension to the OpenAPI specification to define how to render your API specification in the API catalogue for documentation purposes or deploy instances of proxy layers for your API within specific environments, without having to interact directly with our API platform or infrastructure.

After building your API alpha you should have:

- a sandbox and possibly an integration test environment
- an API proxy that is secured with error handling, which connects to a backend
- optionally, an automated testing framework

7.14.6.2 Prerequisites

The following may be run in parallel with building your API process steps.

- learn about the different API deployment environments (see page 596)which are used for development internally by producers, or for testing and integration by external API Consumers
- your API has been designed as part of the designing your API (see page 322) stage

7.14.6.3 Process steps

- getting set up with Proxy Generator (see page 681) - obtaining credentials, other supporting configuration and applications which will allow you to interact with the Proxy Generator API
- setting up your API sandbox (see page 684)
- configuring your API backend (see page 688)
- publishing your specification (see page 697) - publishing your API specification to UAT and the production API catalogue
- deploying your API (see page 698) - deploy directly to an environment using configuration within the API specification file

Once you have built your API and setup the environments, the next steps are:

- securing your API. (see page 856) Securing your API is vital and a requirement for the API platform
- consider setting up automated testing for your API (see page 715). Testing is important, as it gives you confidence that the API works as expected.

7.14.6.4 Getting set up with proxy generator

7.14.6.4.1 Overview

The Proxy Generator uses our custom extension to the OpenAPI specification to define how to render your API specification in the API catalogue (for documentation purposes) or to deploy instances of your API within specific environments, without having to interact directly with our API Platform or infrastructure. To create your API proxy with Proxygen you need:

- An underlying unique name that will be used to manage your API proxy
- Create a public/ private key pair
- Register your API with the APIM team to issue you Proxygen credentials
- Create your API proxy for the first time

7.14.6.4.2 Prerequisites

To use Proxygen for the first time:

- an API name in kebab-case. e.g. `personal-demographics`
- a 4096-bit RSA key (see generating a key pair below)
- a key-ID to represent your RSA key. e.g. `cli-key-1` (see generating a key pair below)

7.14.6.4.2.1 Generating a public / private key pair

You need to generate a private/public key pair for your application in order to make calls to your API with proxy generator. This key pair must be a 4096-bit RSA key pair. The steps to create a key pair are detailed below for production environments.

Essentially, you are creating three files. A private key, a public key both in PEM format and a JWKS file.

1. Decide on a Key Identifier (KID). This KID must have a unique name in order to identify the key pair in use. e.g. cli-key-1
2. Open a BASH shell command prompt. The BASH shell should come as standard on Linux and MacOS. If you are using Windows download Git for Windows[623].
3. Define your KID using the command below.

```
KID={YOUR_KID}
```

4. Run the following commands to create your private and public keys.

```
openssl genrsa -out $KID.pem 4096
```

623 https://gitforwindows.org/

```
openssl rsa -in $KID.pem -pubout -outform PEM -out $KID.pem.pub
```

At this point, you will have two files. *YOUR_KID.pem* which is your **private** key and *YOUR_KID.key.pub* which is your **public** key. Keep your private key to yourself. Never share it with anyone!

7.14.6.4.3 Contact the API platform team

Contact the API platform team through the platforms-apim-producer-support[624] channel to request an API to be created on proxygen. You will need to provide your API name, the **public** portion of the RSA key, and your key-ID, at which point APIM will be able to setup your API and create credentials for you. Once completed you'll be sent the **client_id** (which has been combined with the public key) for authentication.

7.14.6.4.4 Creating your proxy for the first time

7.14.6.4.4.1 Option 1: Using proxygen-cli

The proxygen-cli is a command line interface for the Proxy Generator API, which wraps the API calls in easy to use, human readable commands.

For instructions on how to set up and authenticate with the proxygen-cli see the proxygen-cli user guide.

7.14.6.4.4.2 Option 2: Using your API directly

The Proxy Generator API can be used like any API, directly instead of using the proxygen-cli. To get set up to authenticate with the API, you will need to generate a key pair. See signed JWT authentication[625] page on how to do this. You will then need to authenticate with proxygen. See Authenticating with Proxy Generator (see page 682) on how to do this without using the proxygen-cli. The examples use Python, which requires downloading a python library and runing a basic script. Ensure any URLs are consistent with your ~/.proxygen/credentials file.

Once the token is generated, you can begin calling the API - see the 'Operations' section in the Proxy Generator API[626] specification.

7.14.6.5 Authenticating with Proxy Generator

7.14.6.5.1 Overview

Please ensure you have followed the Getting set up with proxy generator (see page 681) page before attempting to authenticate yourself using the API.

624 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY
625 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication#step-2-generate-a-key-pair
626 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#overview--overview

This page details the steps to authenticate against the proxy generator (either cli or API) using Private Key JWT (JSON Web Token) authentication (aka 'Application Restricted' or 'machine-user' access).

> Username and password login is supported but only when using the proxygen-cli (see section 2 for more details).

7.14.6.5.2 Using the proxygen-cli

See instructions on how to set up the proxygen-cli command line interface to automatically authenticate with Proxygen.

7.14.6.5.3 Using the Proxygen API

7.14.6.5.3.1 Obtaining a Proxy Generator access token: private key JWT Method

Before you can obtain the Proxy Generator access token, you must first generate a key pair (see page 681), which provides:

- a client_id. You should have received a client_id from the API platform team when setting up your API
- the private key file you created to make calls to your API with proxy generator
- the KID of the corresponding public key in the JWKS endpoint

Once you have the above details, perform the following steps to authenticate yourself with Proxygen API:

1. Save the python script (see below) as a file with extension .py.
2. Ensure you edit any URLs, private key paths, KID and your API name placeholders to match your details.
3. Run the script using any terminal or IDE tools, for example, Visual Studio Code.

The script gets the access token required to authenticate with the Proxygen API using details like client ID, KID, private key. At this point, you can use the access token obtained to make requests to Proxy Generator (API) to manage your API.

For more details on the Proxygen APIs and how to call them, refer to the 'Operations' section of Proxy Generator API[627] specification.

Python Code Block - Generate access token

```
import uuid
from time import time
```

627 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis

```
import requests
import jwt  # pyjwt library

REALM_URL = "https://identity.prod.api.platform.nhs.uk/realms/api-producers"
PRIVATE_KEY_FILE = "{your_private_key_path}"  # absolute path
KID = "{your_KID}"

claims = {
    "sub": CLIENT_ID,
    "iss": CLIENT_ID,
    "jti": str(uuid.uuid4()),
    "aud": REALM_URL,
    "exp": int(time()) + 300,
}

with open(PRIVATE_KEY_FILE, "r") as f:
    private_key = f.read()

client_assertion = jwt.encode(
  claims, private_key, algorithm="RS512", headers={'kid': KID}
)

token_response = requests.post(
    f"{REALM_URL}/protocol/openid-connect/token",
    data={
        "grant_type": "client_credentials",
        "client_assertion_type": "urn:ietf:params:oauth:client-assertion-type:jwt-
bearer",
        "client_assertion": client_assertion,
    },
)

access_token = token_response.json()["access_token"]

# Make a request to a protected endpoint
response = requests.get(
    "https://proxygen.prod.api.platform.nhs.uk/apis/<example_api>",
    headers={"Authorization": f"Bearer {access_token}"},
)
```

7.14.6.6 Setting up your API sandbox with Proxy Generator

7.14.6.6.1 Overview

An API sandbox is a controlled environment that simulates the behavior of a production API. It's a crucial tool for developers to test and integrate with your API before it's fully deployed.

It's like a practice ground where developers can experiment, learn, and troubleshoot issues before deploying their applications in a live environment.

A sandbox API provides a version of your API with pre-canned responses, it is really important to keep in mind the following principles:

- it is for early consumer testing - it allows them to do initial discovery about your API
- easy to use
- integrated with the "Try this API" feature on the specification
- is open access - has no authorisation (unlike the integration test environment) - so a consumer can learn how to use the api, not how to authenticate with it
- it is stateless, so it does not actually persist any updates
- it only covers a reduced set of scenarios
- simple and quick to build

What does this mean in practice when building a sandbox?

- an endpoint might have 5 parameters (which results in many variants), but the sandbox should only look to implement 2 or 3 common variants
- the mandatory nature of a parameter does not need to be enforced, this is especially important where the parameters are complex
- the "try this now" functionality should have defaults, and this is really important if you choose to keep parameters mandatory
- the sandbox **is not intended to help with solutions assurance**

We strongly recommend building a sandbox API - in user testing consumers really like them. It makes the whole learning/early discovery process feel easier (even if the sandbox actually only delivers the same responses that are in your API specification).

7.14.6.6.1.1 Self-service

Your sandbox API needs to be self-service. Consumers should not have to request access to try out the sandbox.

7.14.6.6.1.2 Building a sandbox API

How you build your sandbox API is up to you. We do not dictate the language or functionality, but recommend it reflects some of the basic functionality of your API, to allow discovery for potential consumers.

Some tools exist which convert an OAS file into a mock server. We've not found one that works for every API. We tend to use the OpenAPI tool for one-off conversions to create a mock server.

Keep in mind the principles above, it is very easy to allow scope creep. Keep the sandbox to a reduced set of scenarios. These are implemented as stubs that return static data to simulate a happy path interaction and simulate a few common failure conditions e.g. 404 resource not found or 400 bad request (for invalid or missing parameters or headers).

*Keep it **light weight** - don't over complicate the Sandbox. It is easy to keep adding to it until you have nearly replicated your entire API* 🙂

Recommended sandbox features

When building your sandbox, consider the following features:

- **core functionality** - ensure your sandbox supports at least one call for every endpoint, to help show potential consumers what they can expect from using your API
- **errors** - Understanding error handling at the discovery phase can also be very useful for potential consumers. We recommend you implement an easy way for the user to trigger canned errors, so they can test their error handling
- **health checks** - Monitoring the sandbox with health checks is important for ensuring availability for consumers trying the API (and also mimics what would be required when the API is live).
  See Monitoring your API (see page 1115) for how to implement monitoring.
- **logging** - As above, good logging is useful for maintaining the availability of the sandbox and debugging potential issues

7.14.6.6.1.3 Deploying a Sandbox API

Deploying using your own Docker image with APIM Hosting

You can use APIM to host a containerised server behind your sandbox, by following the guidelines detailed below:

Container Constraints

Containerising the sandbox API application gives you flexibility on your underlying software, but in order to ease maintenance and reliability, the following requirements must be satisfied:

- the container must export a plain-HTTP REST API on port 9000
- there should be a working health-check URL. Default path is `/_status`
- the application should log request details and other import information to *stdout* - this will be forwarded to Splunk. See ***More on logging*** below.

Procedure

Once you have a working sandbox API container, see Configuring your API backend (see page 688) for instructions on how to deploy a containerised target using an OAS specification file.

The OAS specification file can be deployed to the sandbox environment using the following command within the Proxygen CLI (see page 681):

```
proxygen instance deploy sandbox <base-path> path/to/specification.yaml
```

The <base-path> parameter will usually be the proxy name created when getting set up, but in some cases you might want to append a descriptor to it (like a PR number etc.).

> For more information on how to manage your API instances with proxygen-cli, see the Proxygen CLI user guide (see page 933).

If you're not using the Proxygen CLI, the relevant API call is documented here[628]

More on logging

Sandbox containers hosted within the API Platform are automatically configured to redirect the stdout stream to our Splunk Cloud instance under the `requests_apm_ptl` index. Logs can be searched for a particular container using a search similar to the one below.

Fill in the missing values - the environment will be either `sandbox`, `internal-dev-sandbox` or `internal-qa-sandbox` based on the proxygen CLI command above)

```
index=requests_apm_ptl source="apm:ecs:api-deployment:ptl:<environment>:<api_name>--
<base-path>--<instance>:<api_name>"

# Or if you're in a rush:
index=requests_apm_ptl source="apm:ecs:api-deployment:ptl:sandbox:<api-name>*"
```

Hosting and connecting to your own sandbox server

If you host your own sandbox server, then this can be set up in the same way as any other 'external target' instance, using the instructions below:

- Configuring your API backend (see page 688)

7.14.6.6.1.4 Standard test data for sandbox use

We have defined standard test data to use in sandboxes - see Standard test data (see page 606).

7.14.6.6.2 Sandbox environment domains

Environment	DNS Name	Used for
sandbox	sandbox.api.service.nhs.uk[629]	Externally accessible sandboxes (e.g. the "Try" functionality in the API catalogue.

628 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#post-/apis/-api_name-/environments/-environment-/instances
629 http://sandbox.api.service.nhs.uk

Environment	DNS Name	Used for
internal-dev-sandbox	internal-dev-sandbox.api.service.nhs.uk[630]	Development and manual testing.
internal-qa-sandbox	internal-qa-sandbox.api.service.nhs.uk[631]	Clean environment for testing deploys.

For full details such as static IPs - see environments (see page 758).

7.14.6.7 Configuring your connection to your API backend with Proxy Generator

7.14.6.7.1 Overview

The API platform[632] acts as the 'front-door' for APIs, providing a layer of authentication, rate limiting and a consistent development experience for API producers and consumers. Requests are processed within the API platform before being forwarded to an API backend, which is managed by producers and contains business logic.

We refer to a specific deployment of an API into an environment as an 'instance' - each instance will separately refer to its own backend server, but multiple instances can use the same backend. Deployment of API instances are covered in Deploying your API (see page 698).

This guide will explain how to configure the different types of backend target servers supported by Proxy Generator using the `target` section within the `x-nhsd-apim` extension of an instance specification - which will allow you to set a backend target for your instance and provide some other associated configuration options. The Proxy Generator specification document provides a complete reference for configuration using the our custom extension[633] to the OAS standard, contains examples of how to define your API instances using the extension, and can be combined with this document to provide a thorough overview of the process.

There are two ways in which you can deploy secrets via proxy generator. Either making REST API calls or b using the proxygen-cli. For a full user guide on how to manage secrets using proxygen-cli, see Proxygen CLI user guide (see page 933).

630 http://internal-dev-sandbox.api.service.nhs.uk
631 http://internal-qa-sandbox.api.service.nhs.uk
632 https://digital.nhs.uk/services/api-platform
633 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#overview--openapi-specification-constraints-and-extensions

7.14.6.7.2 External target backend

An external target is an API backend service that is not hosted within the API platform infrastructure. The service is managed and hosted by the API producer, and should be internet-facing. See our documentation on how to properly secure (see page 856) your API backend.

Configuration of an external target requires that:

- the `target.type` property is set to `"external"`
- the `target.url` property is set to a valid, internet-facing (but secured (see page 778)) URL

An example target definition is:

External target definition

```
target:
  type: external
  healthcheck: /_status
  url: https://example.external.apikey.secured.backend.com
  security:
     # See Security models below
```

The security section should be created based on the security model of the backend server. The authentication models currently supported are API Key and mTLS; both approaches are detailed below.

7.14.6.7.2.1 API key security model

With the API key security model, the API platform authenticates against your backend server using an secret API header, which is included within the header of each request. You need to configure the name of the HTTP header and the value of the API key.

The API key is stored within the API platform as a 'secret', which needs to be deployed using Proxy Generator before you can configure any instances that use it.

1. Deploy secret

REST API

The type parameter should be `apikey` and the body of the PUT request[634] should be the plain text value of the secret.

634 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/environments/-environment-/secrets/-secret_name-

Store an API key secret using the Proxygen REST API

```
PUT /apis/{api_name}/environments/{environment}/secrets/{secret_name}
```

CLI

Store an API key secret using the Proxygen CLI

```
# Note that, the --apikey parameter sets the secret type so it can be referenced as
an apikey secret in the target definition
$ proxygen secret put internal-dev example-api-key --apikey --secret-file ~/tmp/
secret_file.txt
✓ Putting secret example-api-key in internal-dev
{
  "name": "example-api-key",
  "type": "secret",
  "last_modified": "2023-10-26T12:25:29.002000+00:00",
  "environment": "internal-dev",
  "apikey": true,
  "mtls": false,
  "version_id": "30f26d35-aa04-4a7e-893e-440398bcbbeb"
}
```

2. Configure security section

Configure the security section as demonstrated in the example below.

Key details are:

- `type` - always `apikey` for this security model.
- `header` - the name of the HTTP header passed through in requests - make sure this matches what is expected by the backend service.
- `secret` - the name of the secret deployed in step 1. Make sure the secret is available in the appropriate environment.

Configure security section for an API Key secured back-end

```
target:
  type: external
  healthcheck: /_status
  url: https://example.external.apikey.secured.backend.com
  security:
```

```
    type: apikey
    header: X-API-Key
    secret: example-api-key
```

7.14.6.7.2.2 mTLS security model

Mutual TLS provides two-way authentication (assurance that both sides of the connection are genuine) for connections between the API platform and backend target servers. This requires an RSA key and certificate for both ends of the connection, along with a common certificate authority which is used to validate the certificates. Specific details can be found on the Securing your API with proxy generator (see page 856) page.

When using Proxy Generator to deploy an mTLS secured API instance, the "client" (API platform-side) certificate and key are deployed together as a secret, which is then referred to within the `security` section of the target definition.

1. Obtain or generate certificates

1.1 Working with an existing system

If you are working with an existing backend that requires mTLS, you will need to speak to whoever is in charge of issuing certificates for that system. Keep in mind that this may be different entities between production and ptl.

1.2. New system

If you are creating a new system, you will generally want to generate your own certificates.

1.2.1. Generating self-signed certificates

To generate a self-signed certificate, you will need to first generate a CA certificate with which to sign your TLS certificates.

This is a basic overview of how to do this.[635]

1.2.2. Getting certificates signed (or signing them with your own CA)

To create a new certificate to use for mTLS, you will first need to generate a key of suitable strength (4096 bit RSA is a good minimum, but you may also want to look at Ed25519, ECDHE, and other algorithms).

You can do this with a command such as:

```
ssh-keygen -t rsa -b 4096 -m PEM -f my.key
```

From there, you can generate a CSR (certificate signing request) to be signed by the owner of the CA cert.

635 https://gist.github.com/fntlnz/cf14feb5a46b2eda428e000157447309

Make sure you put together a cert config file[636] first!

```
openssl req -new -in my.key -nodes -sha256 -out my.csr -config my.conf -extensions
v3_req
```

2. Deploy RSA key and certificate together as an "mtls" secret

REST API

The type parameter should be `mtls` and the body of the PUT request[637] should contain the private key and certificate files in PEM format, and must be sent as `multipart/form-data` using field names of `key` and `cert` respectively.

Store an mTLS secret using the Proxygen REST API

```
PUT /apis/{api_name}/environments/{environment}/secrets/{secret_name}
```

CLI

Store an API key secret using the Proxygen CLI

```
$ proxygen secret put --mtls-cert client.crt --mtls-key client.key internal-dev
example-mtls-cert
✓ Putting secret example-mtls-cert in internal-dev
{
  "name": "example-mtls-cert",
  "type": "secret",
  "last_modified": "2023-10-27T12:46:37.132000+00:00",
  "environment": "internal-dev",
  "apikey": false,
  "mtls": true,
  "version_id": "d4e150ed-ee2f-4535-970c-5ecd088ae72e"
}
```

3. Configure security section

Configure the security section as demonstrated in the example below.

Key details are:

636 https://www.switch.ch/pki/manage/request/csr-openssl/
637 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/{api_name}/environments/{environment}/secrets/{secret_name}

- `type` - always `mtls` for this security model
- `header` - the name of the combined key/cert secret deployed in step. 1

Configure security section for an API Key secured back-end

```
target:
  type: external
  healthcheck: /_status
  url: https://example.external.mtls.secured.backend.com
  security:
    type: mtls
    secret: example-mtls-cert
```

7.14.6.7.3 APIM AWS hosted containers

If you are using the APIM Hosted Container feature (see page 637)to do some light weight processing for requests, before passing on the request to the backend - then you will need to build a Docker image and Docker File, with some constraints listed below:

7.14.6.7.3.1 Container constraints

1. Expose a plain-text HTTP web service on port 9000

The container (or at least one of a cluster) should be listening for HTTP traffic on the expected port in order for the API platform to forward traffic. TLS is handled by the proxy front-end and is not used at container-level.

2. There should be a working health-check URL.

Default path is `/_status`.

This is important as it's used in two places:

1. by the front-end of the HTTP proxy to ensure that connectivity is available. If it's not working, then requests may not be forwarded to the container.
2. by the container-runtime to ensure that the container is up and running. A failing health check will cause the container to repeatedly restart until the threshold is reached, at which point it will be terminated completely.

3. Log to stdout

The application should log request details and other import information to *stdout* - this will be forwarded to Splunk. Containers hosted within the API platform are automatically configured to redirect the stdout stream to our Splunk Cloud instance under the `requests_apm_ptl` index. Logs can be searched for a particular

container using a search similar to the one below. More details on using logging and other features to monitor you API can be found on the Monitoring your API (see page 1115) page.

```
index=requests_apm_ptl source="apm:ecs:api-deployment:ptl:<environment>:<api_name>--
<base-path>--<instance>:<api_name>"

# Or if you're in a rush:
index=requests_apm_ptl source="apm:ecs:api-deployment:ptl:sandbox:<api-name>*"
```

7.14.6.7.3.2 Deploying a container within the APIM ecosystem

Ensure your API has a container repo

In most cases, a private ECR repo within the API Management infrastructure is allocated to each new API. You can confirm that your API has an ECR repo using the docker-token[638] endpoint, or using the CLI with `proxygen docker get-login`.

1. Authenticate with The API platform container registry

Container images are hosted within a private Elastic Container Registry (ECR) repository under the API Management AWS infrastructure. In order to be able to push and pull images from this registry, you need to obtain authentication details.

REST API

This can be done with a simple HTTP get to the docker-token[639] endpoint on the API.

Obtain a container registry access token using the Proxygen REST API

```
GET /apis/{api_name}/docker-token
```

CLI

638 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/docker-token
639 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/docker-token

Authenticate with ECR using Proxygen CLI

```
# 'docker get-login' command will return the exact command required to authenticate
Docker with ECR.
# You can copy/paste the returned command back into the shell like this:
$ proxygen docker get-login
docker login -u AWS --password <password> <Docker registry URL>
$ docker login -u AWS --password <password> <Docker registry URL>

# Or streamline the whole thing using the BaSH command expansion:
$ $(proxygen docker get-login)
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
WARNING! Your password will be stored unencrypted in <homedir>/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

2. Push your image to the API platform registry

Once you have built your application image locally, you need to tag it with the API platform registry URL and then push to the registry.

The commands below assume you're using the proxygen-cli, but all the required information is also available via the docker-token[640] endpoint.

Tage and push container image

```
$ docker image tag <image id> $(proxygen docker registry)/<api-name>:latest    # Tag
the image with the remote URL
$ docker push $(proxygen docker registry)/<api-name>:latest   # tip: you can use git
commit IDs instead of 'latest' to differentiate between main branch and PRs.
```

3. Add a target section to your OAS specification file

Using the 'Target' section of the specification[641] as a reference, create a target definition within the `x-nhsd-apim` section of your OAS specification file, which will be used as a basis for deploying the proxy.

This step requires the image name and tag from step 2 in order to configure the container tasks.

A short example snippet:

640 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/docker-token
641 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#overview--openapi-specification-constraints-and-extensions

Example x-nhsd-apim section of an OAS file

```
target:
  type: hosted
  healthcheck: /_status
  containers:
    - name: api-name
      image:
        name: api-name
        tag: latest
      environment:
        LOG_LEVEL: info
        NODE_ENV: development
```

7.14.6.8 Creating your spec-only API specification with Proxy Generator

7.14.6.8.1 Overview

An API may not be suitable or ready sit on the API platform but there are still benefits to writing an OAS specification to describe the API and hosting it with the platform. Proxy Generator can be used to automate the creation of an APIs specification and publish it to the API catalogue - NHS Digital[642]. For a detailed explanation about deploying specifications using Proxy Generator see publishing your API documentation (see page 571).

Using the API Platform to generate and host documentation is beneficial because:

- it saves a lot of time and resources.
- we believe in the "spec-first" approach when designing and building an API and the OpenAPI Specification as a centralised source of truth.
- making changes to the specification is controlled by API producers.

7.14.6.8.2 What is a spec-only API?

In APIM we call a "spec-only API" an API configured in such a way that only the specification is deployed and no platform resources such as proxy, products nor hosted backends are deployed.

The primary reason a producer will deploy only the specification for an API in the catalogue, is because we publish API specification standards between existing APIs. If a producer team isn't ready to put their API on the platform but wants the benefits of deploying their documentation through Proxy generator, they can get set up to deploy their OAS spec and can deploy their proxy down the line with only a few extra steps.

7.14.6.8.3 How-to set up a spec-only API with Proxy Generator

The following steps explain how to get set up to deploy a specification with Proxy Generator. Note that the process is the same a non-spec-only set-ups but only the spec resources are deployed.

642 https://digital.nhs.uk/developer/api-catalogue

7.14.6.8.3.1 Get set up with Proxy Generator

Follow the initial steps for getting set up with Proxy Generator here - Getting set up with proxy generator (see page 681).

In summary this involves:

- defining pre-requisites and generating credentials
- providing the pre-requisites and credentials to APIM who will set up the API on the Proxy Generator service
- getting set up to call the Proxy Generator API with the credentials

7.14.6.8.3.2 Write your API Documentation

Before deploying your API documentation you must first write the OAS (OpenAPI Specification). This will be sent as the payload, when calling the Proxy Generator API and will be used to generate the API specificationas displayed on the API catalogue[643].

For details and best practices on writing your API documentation see - Writing your API documentation (see page 451).

7.14.6.8.3.3 Publishing your specification to UAT

Before getting the API live on the API catalogue[644] is a good idea to publish to the UAT version of the catalogue. This allows you to preview how the API documentation will look when it's been rendered by the catalogue. This can also be helpful for discovering and addressing potential errors within the OAS code.

For instructions on publishing a spec to the UAT environment see the UAT section in publishing your API documentation (see page 571)

UAT specifictions must be viewed via the VPN, VDI or on the network in the NHS England office.

In summary this involves:

- creating the UAT specification for an API using Proxy Generator[645]
- get set up with the VDI or VPN if not already (see AWS HSCN VPN access (see page 168))
- post a query in the #platforms-apim-producer-support Slack channel (see page 2056) asking for your spec to be set up in UAT Bloomreach, providing your api_name (as used in the path when calling Proxy Generator) as your spec ID.
- on the VPN or VDI view the spec at:

643 https://digital.nhs.uk/developer/api-catalogue
644 https://digital.nhs.uk/developer/api-catalogue
645 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/spec/uat

```
https://uat2.nhsd.io/developer/api-catalogue/<api_name>    <---- Note api_name is set up by the Bloomreach team so confirm with API Platform team what this should be if any issues
```

7.14.6.8.3.4 Publishing your specification to the API catalogue

Once you're happy with your specifiction and are ready to publish to the API catalogue[646], follow the instructions in publishing your API documentation (see page 571).

In summary this involves:

- creating the specification for an API using Proxy Generator[647]
- post a query in the #platforms-apim-producer-support Slack channel (see page 2056) asking for your spec to be set up in production Bloomreach to be published API catalogue, providing your api_name (as used in the path when calling Proxy Generator) as your spec ID.
- view your published spec on the API catalogue[648].

7.14.6.9 Deploying your API with Proxy Generator

7.14.6.9.1 Overview

The Proxy Generator API deploys platform resources for an API in a few calls. This is based on the `paths`, `servers` and `x-nhsd-apim` extension described in the OpenAPI Specification used as the payload. A collection of these resources is referred to as an instance. Intances are posted in platform environments (such as Internal Development, Integration and Production. See API deployment environments (see page 596)for more information on this) and include:

- a **proxy** - this holds the configuration for auth, monitoring and rate-limiting
- one or more **products** - for controlling scopes and access to your API
- *optionally* a **hosted container** - if we are hosting your containerised backend for you

7.14.6.9.2 Instance dependencies

Before instances can be deployed there are some required dependancies:

- get set up with Proxy Generator (see page 681)
- write your OpenAPI Specification (see page 451)
- if you are deploying an instance for the first time, you will need to first post a secret to set up the target. Read up on how to do this while configuring your api backend[649]

646 https://digital.nhs.uk/developer/api-catalogue
647 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/spec
648 https://digital.nhs.uk/developer/api-catalogue
649 https://nhsd-confluence.digital.nhs.uk/display/APM/Configuring+your+API+backend+with+proxy+generator

7.14.6.9.3 Deploying instances

As the Proxy Generator is an API itself, deploying instances of your API is flexible and quick to use. The API calls can be slotted in to your own deployment cadence and is not restricted to any specific technology when making those deployments.

This section will break down:

- naming instances (see page 699)
- using the OpenAPI Specification as the instance payload (see page 700)
- deploying with the Proxy Generator API (see page 704)

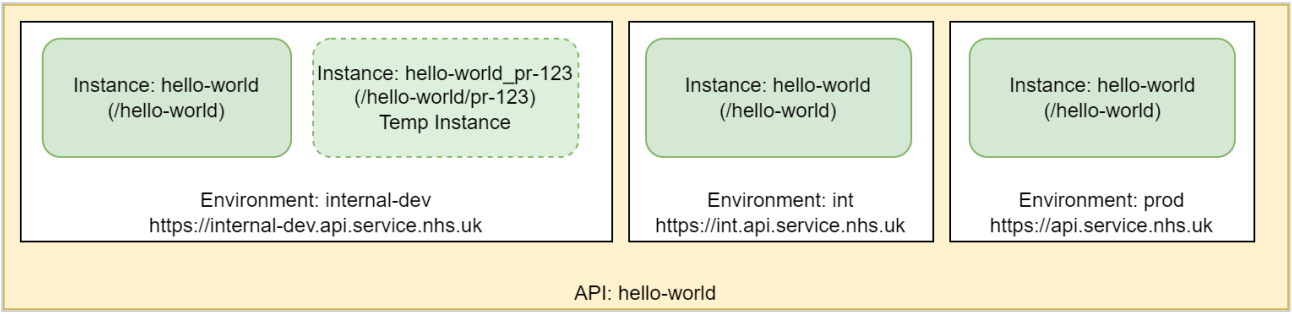
7.14.6.9.3.1 Naming instances

An instance should be thought of as a version of your API hosted in an environment on the API Platform. Creating an instance will deploy an internet facing proxy connected to your target backend, which provides security, monitoring and rate-limiting.

Naming an instance is important because it sets the base path for the url used to access your API (see Setting the base path with the servers field (see page 702) section for how to set this).

The base url the proxy breaks down like so:

```
# Production environment
https://api.service.nhs.uk/{instance_name}

# Non production environments
https://{environment}.api.service.nhs.uk/{instance_name}
```



The instance name is used as the base path to the API proxy and so should:

- be or include the API name
- be kebab-case: hello-world

Muti-segment base paths

Depending on the design of the API, a base path may have multiple segments separated by forward slashes.

- There are some considerations for your paths - see Proxy base path vs API base path (see page 364)

In this case:

- When providing the instance name as the base path in the servers field[650] you should use the forward slashes e.g. `hello-world/FHIR/R4`
- When using the instance name in the path to GET/PUT/DELETE an instance (see page 704), though, you need to replace `/` with `_` for a path-safe version e.g. `hello-world_FHIR_R4`

7.14.6.9.3.2 Using the OpenAPI Specification as the instance payload

The Proxy Generator API uses the APIs OpenAPI Specification (OAS) as the payload to deploy an API on the API Platform.

This is beneficial because:

- **we believe in a specification-first approach.** Properly documenting an API is crucial to making integration easier for consumers.
- **only one payload is needed for deploying both the OpenAPI specification and the proxy instances.** This means all platform resources are easily managed with one source of truth.

While the same OAS is used as the payload for both the API specification (documentation) and the instance specification (platform resources), different parts of the OAS are used or ignored by the Proxy Generator API to deploy the relevant resources.

When publishing the API specification to the API catalogue, the x-nhsd-apim section will be ignored. For more see publishing your API documentation with proxy generator (see page 571).

When publishing the instance specification several sections are important and used to deploy the platform resources.

Expand for a full example of an instance payload and see below for a breakdown of important sections to consider:

Instance payload example

```
openapi: 3.0.3
info:
  version: 1.0.0
  title: Hello Platform As A Service
  description: |
    And Example spec for demoing Proxy Generator
  contact:
    name: API Management Support
    email: api.management@nhs.net
servers:
  - url: 'https://internal-dev.api.service.nhs.uk/hello-paas'
components:
  securitySchemes:
```

650 http://nhsd-confluence.digital.nhs.uk#Settingthebasepathwiththeserversfield

```
    nhs-login-p9:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/nhs-
login-p9
    nhs-cis2-aal3:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/nhs-
cis2-aal3
    app-level3:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/app-
level3
paths:
  /hello/nhs-login:
    get:
      security:
        - nhs-login-p9: []
      responses:
        200:
          description: Successful request
  /hello/cis2:
    get:
      security:
        - nhs-cis2-aal3: []
      responses:
        200:
          description: Successful request
  /hello/application:
    get:
      security:
        - app-level3: []
      responses:
        200:
          description: Successful request
x-nhsd-apim:
  monitoring: true
  temporary: false
  target:
    type: external
    healthcheck: /_status
    url: https://mtls-testing-target.ptl.api.platform.nhs.uk
    security:
      type: mtls
      secret: my-mtls-secret
  access:
  - title: Hello Platform As A Service (Internal Development)
    grants:
      nhs-login-p9: []
      nhs-cis2-aal3: []
      app-level3: []
  ratelimiting:
    proxy:
      limit: 5
      timeunit: minute
    app-default:
```

```
      limit: 10
      timeunit: minute
```

Setting the base path with the servers field

The `servers` section of the specification functions differently for API spec payloads and instance spec payloads.

Instance payload `servers` should be a list of one dictionary including the key of `url` and the value of `https://<ENVIRONMENT>.api.service.nhs.uk/<BASE_PATH (see page 698)>.`

Environment should be the environment you are posting the instance in (see Environments (see page 596)) or left blank with the trailing fullstop removed for production.

Base path should be the base path for your API used across all environments (see Proxy base path vs API base path (see page 364)).

Note: When you deploy an instance using proxygen-cli[651] it will modify the `servers` component automatically to conform to the instance name and environment specified. When using the API you will need to modify the OAS payload before submission.

Instance specs: servers

```
servers:
  - url: 'https://internal-dev.api.service.nhs.uk/hello-paas'
```

As apposed to API spec payloads, where this should be the list of URLs you want advertised to API consumers:

API spec: servers

```
servers:
  - url: 'https://sandbox.api.service.nhs.uk/hello-paas'
    description: Sandbox Environment
  - url: 'https://int.api.service.nhs.uk/hello-paas'
    description: Integration Environment
  - url: 'https://api.service.nhs.uk/hello-paas'
    description: Production Environment
```

Setting API security schemes

651 https://github.com/NHSDigital/proxygen-cli

Proxy generator requires API instances to be secured with appropriate security schemes. See How to configure API consumer security patterns using proxy generator (see page 875).

(API Specifications don't require this, but you must correctly configure this on your API instance before it will deploy.)

Configuring platform resources with the x-nhsd-apim OAS extension

`x-nhsd-apim` is a custom OAS extension that is used to configure and deploy platform resources.

The extension allows you to configure:

- `monitoring` - *default true* - For more info see Monitoring your API (see page 1115)
- `temporary` - *default false* - For more info see Temporary instances (see page 706)
- `target` - For more info see Configuring your API backend (see page 688)
- `access` - For more info see How to configure API products using proxy generator (see page 879)
- `ratelimiting` - For more info see Rate limiting (see page 1117)

Example x-nhsd-apim

```
x-nhsd-apim:
  monitoring: false
  temporary: true
  target:
    type: external
    healthcheck: /_mybackendhealthcheck
    url: https://mtls-testing-target.ptl.api.platform.nhs.uk
    security:
      type: mtls
      secret: my-mtls-secret
  access:
  - title: Hello Platform As A Service (Internal Development)
    grants:
      nhs-login-p9: []
      nhs-cis2-aal3: []
      app-level3: []
  ratelimiting:
    proxy:
      limit: 5
      timeunit: minute
    app-default:
      limit: 10
      timeunit: minute
```

7.14.6.9.3.3 Deploying with the Proxy Generator API

There are two methods to manage API instances. Either by making REST API calls or by using the proxygen-cli. For a full user guide on managing API instances via proxygen-cli, see Proxygen CLI user guide (see page 933).

Note: whether using the API directly or using the proxygen-cli make sure you are set up and authenticated to call the API (see page 681) *and that* targets have been pre-configured[652]

POST instance

Once your OpenAPI Specification is written and ready, POST it using the Proxy Generator API:

Create instance using the API

```
POST /apis/{api_name}/environments/{environment}/instances
```

If using the proxygen-cli[653]:

Create instance using proxygen-cli

```
proxygen instance deploy --no-confirm <environment> <base_path> <path_to_spec>
```

See Proxy Generator OpenAPI Specification for more detail on creating API instances[654]

PUT instance

After your instance has been created for the first time, PUT it using the Proxy Generator API:

Update instance using the API

```
PUT /apis/{api_name}/environments/{environment}/instances/{instance_name}
```

If using the proxygen-cli[655]:

652 https://nhsd-confluence.digital.nhs.uk/display/APM/Configuring+your+API+backend+with+proxy+generator
653 https://github.com/NHSDigital/proxygen-cli
654 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#post-/apis/-api_name-/environments/-environment-/instances
655 https://github.com/NHSDigital/proxygen-cli

Update instance using proxygen-cli

```
proxygen instance deploy --no-confirm <environment> <base_path> <path_to_spec>
```

See Proxy Generator OpenAPI Specification for more detail on updating API instances[656]

GET instance

To retrieve the OpenAPI Specification for a deployed instance, GET it using the Proxy Generator API:

Retrieve instance using the API

```
GET /apis/{api_name}/environments/{environment}/instances/{instance_name}
```

If using the proxygen-cli[657]:

Retrieve instance using proxygen-cli

```
proxygen instance get <environment> <base_path>
```

See Proxy Generator OpenAPI Specification for more detail on retrieving API instances[658]

DELETE instance

If you need to remove a deployed instance, DELETE it using the Proxy Generator API:

Remove instance using the API

```
DELETE /apis/{api_name}/environments/{environment}/instances/{instance_name}
```

If using the proxygen-cli[659]:

656 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/environments/-environment-/instances/-instance_name-
657 https://github.com/NHSDigital/proxygen-cli
658 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/environments/-environment-/instances/-instance_name-
659 https://github.com/NHSDigital/proxygen-cli

Remove instance using proxygen-cli

```
proxygen instance delete <environment> <base_path>
```

See Proxy Generator OpenAPI Specification for more detail on removing API instances[660]

GET list of deployed instances

To see a list of deployed instances, GET using the Proxy Generator API:

Remove instance using the API

```
GET /apis/{api_name}/environments/{environment}/instances
```

If using the proxygen-cli[661]:

Remove instance using proxygen-cli

```
proxygen instance list --env <environment>
```

See Proxy Generator OpenAPI Specification for more detail on retrieving a list of deployed instances[662]

7.14.6.9.4 Temporary instances

API resources are limited across the platform in order to keep deployed resources manageable. API producer teams are allocated a limited number of concurrently deployed API instances (see API config section (see page 707) below for what those limits are). Once this limit is reached, deployments will fail. In order to keep this manageable for producers, there are a number of batch jobs which delete API instances which are marked as `temporary` periodically, based on their `last_modified` date if that is larger than 3 days.

Temporary instances are limited to Internal path to live environments. See environments (see page 596) section for more information on environments.

Examples of when the `temporary` attribute would be appropriate include, but are not limited to:

- on creation of a pull request (PR) - these instances are often used for automated testing
- during penetration testing
- during load testing

660 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#delete-/apis/-api_name-/environments/-environment-/instances/-instance_name-
661 https://github.com/NHSDigital/proxygen-cli
662 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#get-/apis/-api_name-/environments/-environment-/instances

7.14.6.9.5 API config

Certain aspects of deploying instances are limited by API config. Some config is set across the board and some can be configured on a per-API basis:

<table>
<tr><th>Config</th><th>Use</th><th>Defaults</th><th>Configurable per-API</th></tr>
<tr><td>Instance limits</td><td>The API platform has limits on some resources. If instances were not limited then those limits could be reached easily</td><td><table>
<tr><th>Environment</th><th>Instances</th></tr>
<tr><td>prod</td><td>1</td></tr>
<tr><td>int</td><td>5</td></tr>
<tr><td>sandbox</td><td>5</td></tr>
<tr><td>dev</td><td>5</td></tr>
<tr><td>ref</td><td>5</td></tr>
<tr><td>internal-qa-sandbox</td><td>5</td></tr>
<tr><td>internal-qa</td><td>5</td></tr>
<tr><td>internal-dev-sandbox</td><td>10</td></tr>
<tr><td>internal-dev</td><td>100</td></tr>
</table></td><td>Not currently supported</td></tr>
<tr><td>Disallowed environments</td><td>For disallowing deployment to a certain environment. Primary use case is for prod to be turned off until ready to go live.</td><td>prod</td><td>Yes</td></tr>
</table>

7.14.6.10 Configuring target attributes

7.14.6.10.1 Overview

In some cases, APIs require to know additional information about the calling client (Apigee application). Information about the app can be:

- Stored on the Apigee app:
  - Primary way is to add / read from the extended attributes (see page 806) (key value pairs)
  - Platform level data (e.g. jwks url, product-id, ASID) is stored on the limited number of custom attributes
- Apigee flow variables

The target attributes configuration feature allows you to select any of the extended attributes (see page 806) or platform custom attributes and send them to your target server as headers.

This can be done using the x-nhsd-apim section of your OAS spec. Defining the target attributes will configure the instance of the proxy to pull out the specified attributes from an application for every request and pass them through to the backend as a header. These are **not** encrypted so target attributes should not be used for sensitive data, unless your backend server header handling is configured for handling them securely.

If the attribute is required by the backend then the attribute should be marked as required. This will configure the proxy of the the instance to throw an error informing that the required attribute is not set on the application.

Here is an example of configuring target attributes for the x-nhsd-apim section of the OAS spec used when deploying an instance:

```
x-nhsd-apim:
  access:
    - title: Application Restricted
      grants:
        app-level0: []
  target: # Only one target (omit for NO backend)
    type: external
    healthcheck: /_mybackendhealthcheck
    url: target/url
    security:
      type: apikey
      header: apikey
      secret: rowan-cpm-test-8
  target-attributes:
    - name: testing-purpose-value
      required: false
      header: Testing-Purpose-Header
    - name: cpm.first.ods-code
      required: true
```

For each attribute the following should be set:

- `name` - The name of the attribute (extended or custom) the proxy will pull down from the app for each request and forward to the backend
- `required` (defaults to `true`) - Whether or not the request should error if the attribute is not available or cannot be found on the app
- `header` (defaults to `attribute-{name}`) - The name of the header on the request made by the proxy. For recommendations on header names see HTTP headers for your API (see page 395)

7.14.6.10.2 Possible name options

The name field of the target attribute can either be the string representation, or a path through an object.

In the example in the overview you can see the name cpm.ods-code. This means on the app the attribute name is expected to be `cpm` which will be a json object that can be traversed and only the value of the traversed path will be sent on the header.

We do not recommend using paths for values that are set as arrays as it may not have the required result you expect at this point in time.

7.14.6.10.3 Forwarding the attributes

In order to get the attributes forwarded to the target backend they will be attached to the headers on the request made by the proxy after going through a few steps.

The header will:

- Have the name `attribute-{name of target-attribute}` if `header` is not specified
- Contain a non base64 encoded string of the value set on the app

7.14.6.10.4 Example case

In this example scenario we have an app with attributes set:

testing-purpose-value	hello, commas
array	["hello", "i", "am", "array"]
cpm	{ "first": { "ods-code": "RJ11"} }
addtional-attribute	"hello value"

The configuration of the x-nhsd-apim section of the oas spec sent to deploy an instance through Proxygen could look like this:

```
x-nhsd-apim:
  access:
    - title: Application Restricted
      grants:
        app-level0: []
  target: # Only one target (omit for NO backend)
    type: external
    healthcheck: /_mybackendhealthcheck
    url: target/url
    security:
      type: apikey
      header: apikey
      secret: rowan-cpm-test-8
  target-attributes:
    - name: testing-purpose-value
      required: false
      header: Testing-Purpose-Header
    - name: cpm.first.ods-code
      required: true
    - name: array
      required: false
```

When a consumer then makes a request to the proxy, the following headers will have been added onto the request to the backend after passing through the proxy:

attribute-array	["hello",i,am,"array"]
attribute-cpm.first.ods-code	RJ11
Testing-Purpose-Header	hello,commas

Note that it is valid to have attributes on the app that aren't sent to the backend by not specifying them in the oas spec sent to Proxygen (in this case the attribute 'additional-attribute').

7.14.6.10.5 How target attributes support Product ID with Proxygen

Using this functionality it is possible to pull the ProductID custom attribute from the app and forward to the backend.

For more information on Product IDs, see Lifecycle - Product and Product Version[663]

Product ID Availability

663 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/684558838/Lifecycle+-+Product+and+Product+Version#LifecycleProductandProductVersion-ProductIDs

At present, ProductIDs cannot be issued until further work is completed by CPM. A manual workaround has been provided for certain early adopters but is not sustainable for a wider release. If this causes any inconvenience, reach out to us via the platforms-apim-producer-support[664] channel for further discussions.

7.14.6.10.5.1 Example

Proxygen Instance Spec

```
x-nhsd-apim:
  access:
    - title: Application Restricted
      grants:
        app-level0: []
  target:
    type: external
    healthcheck: /_mybackendhealthcheck
    url: target/url
    security:
      type: apikey
      header: apikey
      secret: secret-name
  target-attributes:
    - name: product-id
      required: true
      header: NHSE-Product-ID  # This is the recommended header name for Product ID
```

Example custom attribute set on the app:

product-id	P.GH7-4TY

Header forwarded to the backend after passing through the proxy:

NHSE-Product-ID	P.GH7-4TY

664 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY

There is also discussion of having a sub-division of Product, which is environment specific and holds unique data relating to that. MNS being the first use of that:

- custom attribute: product-device-id
- header to forward: NHSE-Product-Device-ID
- the values will **not** contain the ProductID, just the CPM issued DeviceID

It is not clear if this will be a full pattern or not.

7.14.6.11 Configuring Target Identity Headers

7.14.6.11.1 Overview

In some cases, API targets require additional information about the user to make the request to the API. This information can be passed to the target backend by configuring Target Identity Headers.

The Target Identity Headers can be set on the x-nhsd-apim section of your OAS spec. Each object within the array relates to an individual target identity header. Specifying the `name` field as one of the available options will configure the instance of the proxy to pull the required values from the auth token and pass them through to the backend as headers.

This capability is available when a request is authenticated using CIS2 or NHS login.

Here is an example of configuring Target Identity Headers for the `x-nhsd-apim` section of the OAS spec used when deploying an instance:

```
x-nhsd-apim:
  target:
    type: external
    healthcheck: /_mybackendhealthcheck
    url: https://mtls-testing-target.ptl.api.platform.nhs.uk
    security:
      type: mtls
      secret: <secret_name>
  target-identity:
    - name: cis2-uuid
      header: "custom-user-id-header-name"
    - name: cis2-urid
      header: "custom-role-id-header-name"
    - name: cis2-acr
      header: "custom-auth-level-header-name"
    - name: nhs-login-nhs-number
      header: "custom-nhs-number-header-name"
    - name: nhs-login-identity-proofing-level
      header: "custom-identity-proofing-level-header-name"
    - name: id-token
      header: "custom-id-token-header-name"
```

```
access:
- title: Testing product
  grants:
    nhs-cis2-aal3: []
    nhs-login-p9: []
```

The `name` and `header` fields must be strings. `name` is a required field and must be one of the following values: `cis2-uuid` , `cis2-urid` , `cis2-acr`, `nhs-login-nhs-number`, `id-token` .

The `header` field is optional and specifies a custom header name. When `header` is not present, the following defaults will be used:

- cis2-uuid: `NHSD-Session-UUID`
- cis2-urid: `NHSD-Session-URID`
- cis2-acr: `NHSD-Session-CIS2-ACR`
- nhs-login-nhs-number: `NHSD-NHSlogin-NHS-Number`
- nhs-login-identity-proofing-level: `NHSD-NHSlogin-Identity-Proofing-Level`
- id-token: `NHSD-ID-Token`

7.14.6.11.2 Header Values

The CIS2 UUID, ACR, NHS login NHS number, Identity proofing level and ID token header values are populated using the flow variables on the accesstoken as specified on the Identity Service Token Variables (see page 810) page.

The header containing the role id is populated by making a call to the shared flow UserRoleServiceV2 (see Validate user roles using shared flows (see page 844)). This accepts the header name (provided or default) as a parameter. If being specified on a header by the consumer calling the API this will always come in on a header of `NHSD-Session-URID` . If a custom header name is specified for `cis2-urid` that will be the header that the value will go to the backend as.

7.14.6.12 Deploy an instance into DEP using Proxygen

7.14.6.12.1 Introduction

NHS England has recently completed a review of Spine Core and CIS Path To Live environment provision to analyze the environment based on current demand and utilization. The analysis shows DEV environments are under utilized and hence it will be moved to a minimal support model before being decommissioned on 31 July 2025. For more context, refer this link[665].

665 https://digital.nhs.uk/services/path-to-live-environments

In preparation for this change, we will support hosting APIs on the DEP environment. Outlined below are the required steps to deploy an API instance into the DEP environment using Proxygen. Currently, this will be supported through an interim solution that requires some manual configuration steps before it can be fully enabled through Proxygen.

7.14.6.12.2 Spec Changes and deployment using Proxygen:

At present, the DEP domain is hosted on the Apigee DEV environment. So you will be able to deploy a new instance into the Dev environment and consume it using the host alias for the DEP domain, i.e., 'dep.api.service.nhs.uk[666]'

Make the following changes to your spec, and then deploy a new instance into the Apigee dev environment.

- Ensure the **servers.url** (from which the instance name is derived) includes the suffix '**-dep**', so that it gets deployed as a new proxy within the Apigee dev environment. It should look like: "https://dev.api.service.nhs.uk/<instance_name>**-dep**[667]".
- Update the **target** to point to the required dep backend.
- Set **x-nhsd-apim.access.title** to include the suffix '**dep**'. This ensures consumers can uniquely identify your DEP API on DOS.

At this point, the Product created will have access to your newly created proxy and the OAuth2 authorization service in DEV, i.e., 'identity-service-dev' proxy (oauth2/)

7.14.6.12.3 Manual Configuration (APIM - Deathstar team)

Since this is a temporary fix, there is some manual work required in terms of configuring your Product with the required OAuth2.0 authorization service that connects with CIS2 DEP.

Once the instance is deployed, contact us via the Producer Support Slack channel and we will manually add the '*identity-service-dep-dev*' proxy (oauth2-dep/) to your product.

Note: This manual configuration will be overwritten with every deployment and this will cause downtime for users trying to get the access_token. To re-configure this, please reach out to us via platforms-apim-producer-support[668] Slack channel

7.14.6.12.4 Testing user-restricted auth

In order to test the User restricted Healthcare worker flow in DEP, you need to request a test user in CIS2 DEP as detailed below.

Complete the below online form for a new UUID in Deployment. (Select do not require physical Smartcard if you are going for Windows Hello setup)

---

666 http://dep.api.service.nhs.uk

667 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdev.api.service.nhs.uk%2Fpatient-flags-api-dep%2FFHIR%2FR4&data=05%7C02%7Cviniakshara.kannan1%40nhs.net%7C4a1498a111794308d73608dd9ea0944f%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638841134694425257%7CUnknown%7CTWFpbGZsb3d8eyJFbXB0eU1hcGkiOnRydWUsIlYiOiIwLjAuMDAwMCIsIlAiOiJXaW4zMiIsIkFOIjoiTWFpbCIsIldUIjoyfQ%3D%3D%7C0%7C%7C%7C&sdata=jWL0jPQ260FJpfFltnoMHmids75qt%2B42k4qh%2B%2FHAmKs%3D&reserved=0

668 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY

https://digital.nhs.uk/services/path-to-live-environments/path-to-live-forms/smartcard-request-form-for-the-path-to-live-environments[669]

Once you receive the UUID, drop a mail to itoc.supportdesk@nhs.net[670] to schedule a call with the team and get your Windows Hello setup for DEP authentication.

For other CIS2 authenticator options, refer this[671].

For mock testing (in dev), test users mentioned here (see page 749) can be used.

7.14.6.12.5 Consumer Access

Consumers can access your API on DEP using the below steps.

They need to create an App in the 'Development' environment and look for your DEP API i.e., "<API Name> - <access.title> **DEP** (Development)".

They can access the API using below host alias and base path:

- https://dep.api.service.nhs.uk/oauth2-dep[672] (for auth purpose)
- https://dep.api.service.nhs.uk/<instance_name>-dep (see page 713)

7.14.7 Testing your API

The following resources are available to help you test your API:

- Test Utils 2.0: pytest plugin for APIM (see page 718)
- Testing with mock auth (see page 737)
- Automated quality testing of your API (see page 753)
- INT environment and multiple identity services (see page 755)
- Performance testing (see page 756)
- Environments (see page 758)

669 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdigital.nhs.uk%2Fservices%2Fpath-to-live-environments%2Fpath-to-live-forms%2Fsmartcard-request-form-for-the-path-to-live-environments&data=05%7C02%7Cviniakshara.kannan1%40nhs.net%7C176ea49475c94921d54808dd93b8dfc2%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638829144406468576%7CUnknown%7CTWFpbGZsb3d8eyJFbXB0eU1hcGkiOnRydWUsIlYiOiIwLjAuMDAwMCIsIlAiOiJXaW4zMiIsIkFOIjoiTWFpbCIsIldUIjoyfQ%3D%3D%7C0%7C%7C%7C&sdata=OOp4nugTsKv2oAsl%2BqJ0q4zypz1ptkgRKLzC%2BvTqIeM%3D&reserved=0

670 mailto:itoc.supportdesk@nhs.net

671 https://digital.nhs.uk/services/care-identity-service/applications-and-services/cis2-authentication#authenticator-options

672 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdep.api.service.nhs.uk%2Foauth2-dep&data=05%7C02%7Cviniakshara.kannan1%40nhs.net%7C4a1498a111794308d73608dd9ea0944f%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638841134694447613%7CUnknown%7CTWFpbGZsb3d8eyJFbXB0eU1hcGkiOnRydWUsIlYiOiIwLjAuMDAwMCIsIlAiOiJXaW4zMiIsIkFOIjoiTWFpbCIsIldUIjoyfQ%3D%3D%7C0%7C%7C%7C&sdata=E2J%2BGgc5G6yTdLEtrgDACoggrLppqHlU7Cp7vV7iDA4%3D&reserved=0

Testing Methodology

The testing of the API in this section is done through the use of automated unit tests (unit testing) as opposed to other testing methods

7.14.7.1 Starting point

After using the setup API one click tool (as detailed on Creating a GitHub repository and deployment pipelines for your API (see page 580)) your repo come with some basic tests and uses the NHSD's pytest plugin out of the box.

The example tests can be found in <REPO_ROOT>/tests/test_endpoints.py and cover:

- Test ping (marked as smoketest)
- Test status (marked as smoketest)
- Test auth - app level 0
- Test auth - app level 3
- Test auth - Healthcare worker/CIS2 aal3
- Test auth - Healthcare worker/CIS2 aal2
- Test auth - Healthcare worker/CIS2 aal1

These are some basic examples of what the plugin can do, for more detail please read more here[673], visit the pytest library page[674] or see some more examples here[675].

Out of the box behaviour:

- The tests are ran on the PR pipeline and the release pipeline
- You can use the bool `full` in `apigee_environments` to run either the full test suite (`full: true`) or just the tests marked smoketest (`full: false`) (defined in <REPO_ROOT>/azure/templates/run-tests.yml)
- Example, see <REPO_ROOT>/azure/azure-release-pipeline.yml

Run full test suite for internal-dev

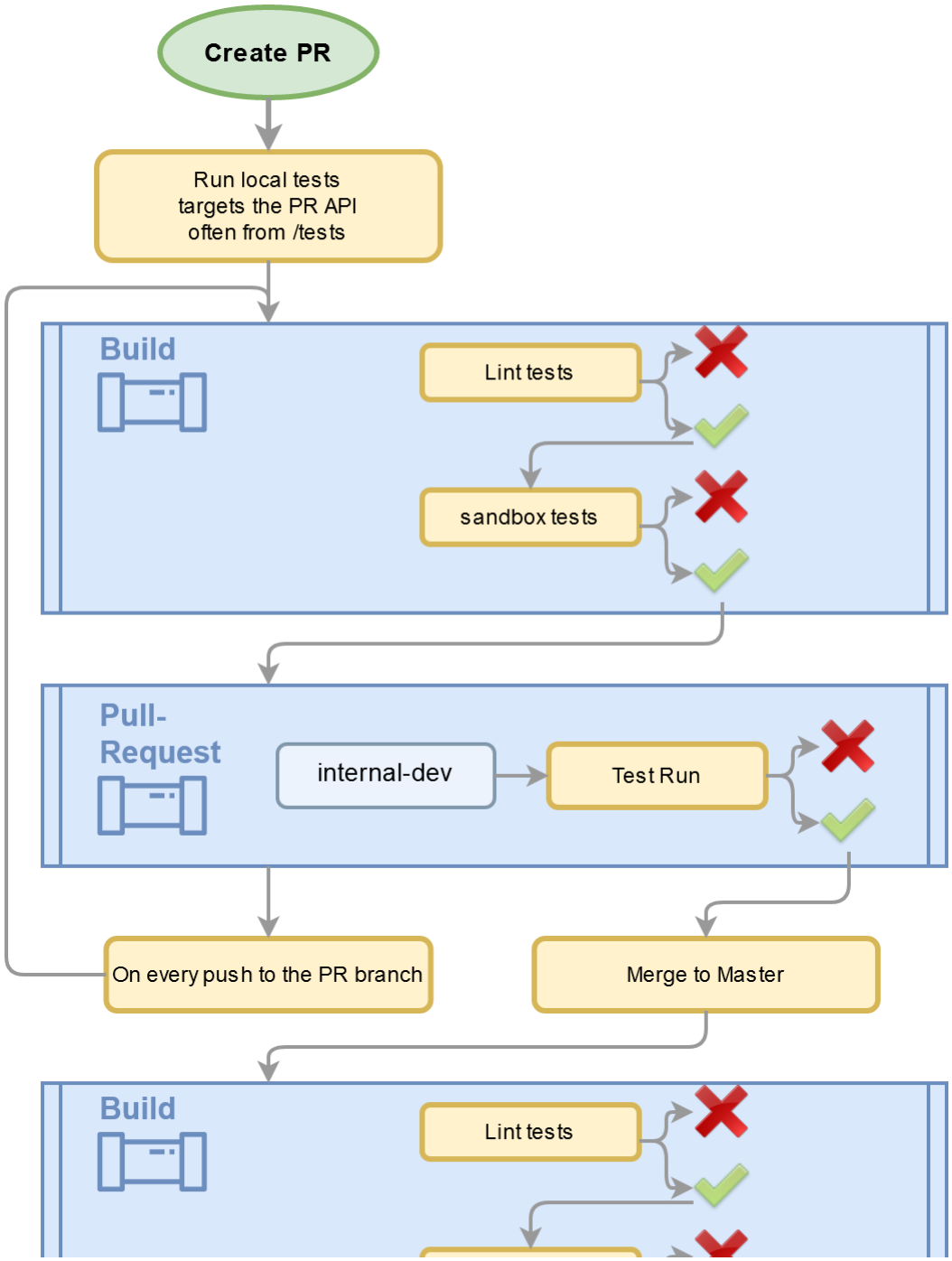
```
extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    apigee_deployments:
      - environment: internal-dev
        post_deploy:
          - template: ./templates/run-tests.yml
            parameters:
```

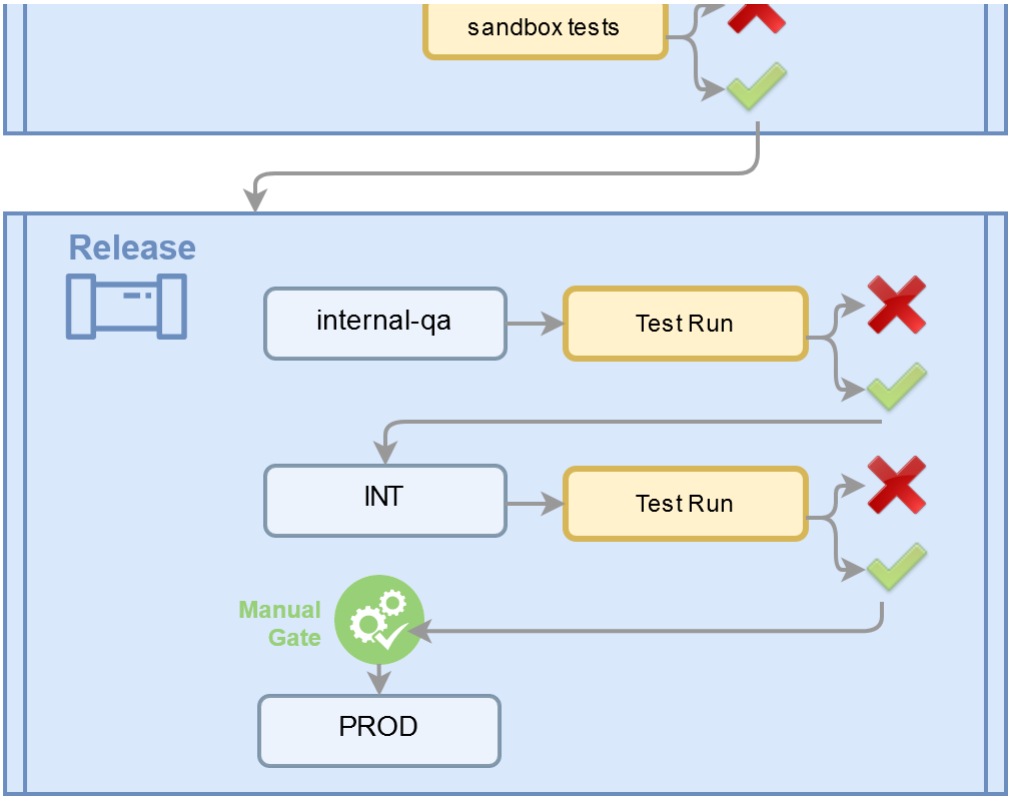
673 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/399181674/old+APIM+Test+Utils+2.0+Plugin
674 https://pypi.org/project/pytest-nhsd-apim/
675 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/tests/test_examples.py

```
12                full: true
```

- There are also make commands: `test` and `smoketest` see <REPO_ROOT>/Makefile

PR Testing process:



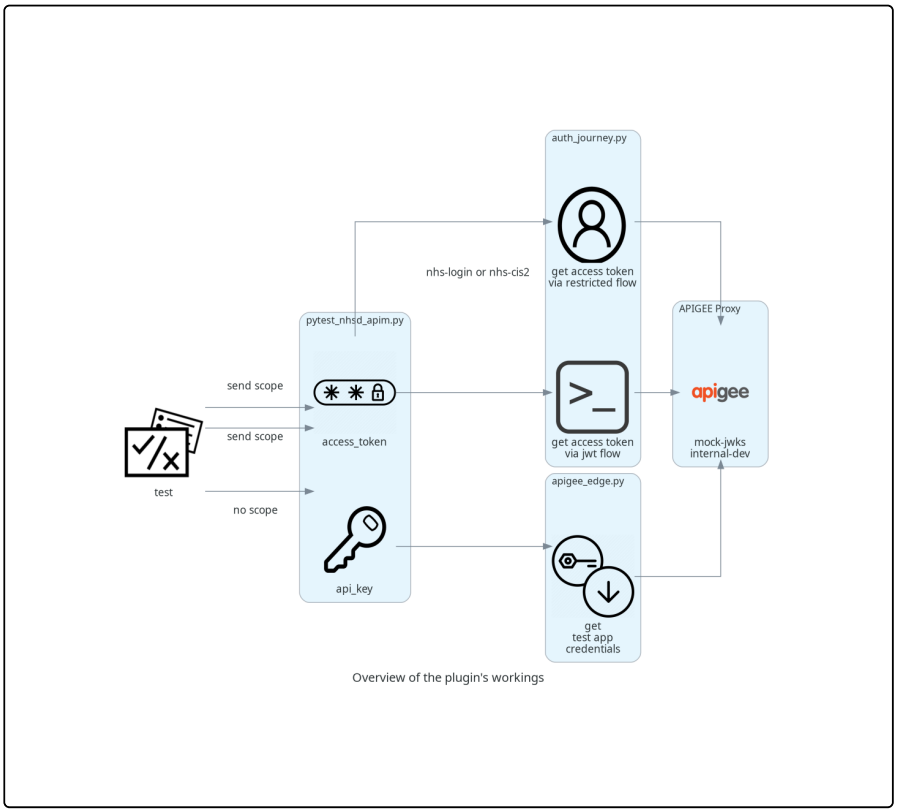


7.14.7.2 Test Utils 2.0: pytest plugin for APIM

Test Utils 1.0 is now deprecated and will be retired on 28th August 2023., this version should be used for new users and producers using 1.0 should migrate.

nhsd-prod org tests

Running tests works best against environments in the nhsd-nonprod org (internal-dev, internal-qa, ref). For infomation on running tests agains nhsd-prod environments (primarily int) see Running tests against nhsd-prod org environments (see page 736)



7.14.7.2.1 Overview

Test Utils 2.0 (pytest-nhsd-apim) is a **pytest plugin** for API Producers.

The source is available on GitHub[676] and maintained by the APIM Platform Team (Deathstar), and published on the Python Package Index (PyPI)[677] as `pytest-nhsd-apim`.

Spend time testing your API/service's unique functionality, not writing boilerplate authentication code. **Get an access token in a single line of Python.** Supports all APIM auth patterns. No need to manage: test Apigee developer accounts, JWT creation, public/private key pairs, NHS Login and CIS2 authorisation, API key credentials.

In addition to this documentation, **working code examples can be found in the repo** in tests/test_examples.py[678]. Test fixtures are defined in src/pytest_nhsd_apim/apigee_edge.py[679], as well as various other useful files.

676 https://github.com/NHSDigital/pytest-nhsd-apim
677 https://pypi.org/project/pytest-nhsd-apim/
678 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/tests/test_examples.py
679 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_edge.py

7.14.7.2.2 Getting started

7.14.7.2.2.1 Installation

It is recommended that you check your package versions in your repo against the versions for the test utils 2.0 to ensure you meet any minimum requirements. You can check the versions we use here[680].

Using your favourite dependency manager. Which is Poetry:

```
poetry add pytest-nhsd-apim
```

Or if unavailable, you can use pip:

```
python -m pip install pytest-nhsd-apim
```

7.14.7.2.2.2 Plugin Configuration

Environment Variables

The plugin requires credentials to interact with Apigee. These are provided through environment variables.

Supply **either**:

An access token returned by the get_token utility[681]. (This token is valid for 24 hours.)

```
export APIGEE_ACCESS_TOKEN=`SSO_LOGIN_URL=https://login.apigee.com get_token`
# if you have logged previously in you can omit the SSO_LOGIN_URL:
export APIGEE_ACCESS_TOKEN="$(get_token)"
```

or your username, password, and OTP Key:

```
export APIGEE_NHSD_NONPROD_USERNAME={my_username}
export APIGEE_NHSD_NONPROD_PASSWORD={my_password}
export APIGEE_NHSD_NONPROD_OTP_KEY={my_otp} # This is the OTP 'seed value' not the
rotating number
```

680 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/pyproject.toml
681 https://docs.apigee.com/api-platform/system-administration/using-gettoken

> If both sets of credentials are defined, the username/password/OTP Key take precedence.

If using Proxy Generator

If you are using the Proxy Generator API then the above environment variables may not be available to you if not using the common Azure pipelines. Proxygen can be used and run from your own pipelines, which means if you have an Apigee account for nhsd-nonprod you can use your own credentials but this is not ideal when running in pipelines

Use Proxy Generator to retrieve a token to supply `APIGEE_ACCESS_TOKEN` as an environment variable either through the API:

```
GET /apis/{api_name}/pytest-nhsd-apim-token
```

Or using the proxygen-cli (see page 933).

```
proxygen pytest-nhsd-apim get-token
```

Command Line Arguments

Set the API Proxy under test using the command line arguments:

- `--api-name`
  The API "service name", as defined in the `azure/project.yml` file in your repo.
- `--proxy-name`
  The name of the API Proxy in Apigee. Typically some combination of *service name* and *environment*, and (for PR-deployed API Proxies) `-pr###`. (If unsure search for your proxy in the Apigee Edge UI.)

7.14.7.2.2.3 Full Invocation Example

```
# Generic format
APIGEE_ACCESS_TOKEN="$(get_token)" poetry run pytest <PATH_TO_TESTS> --api-
name=<API_NAME> --proxy-name=<PROXY_NAME>

# More concrete example - PDS FHIR API, internal-qa environment
APIGEE_ACCESS_TOKEN="$(get_token)" poetry run pytest <PATH_TO_TESTS> --api-
name=personal-demographics --proxy-name=personal-demographics-internal-dev
# PDS FHIR API, PR-deployed
APIGEE_ACCESS_TOKEN="$(get_token)" poetry run pytest <PATH_TO_TESTS> --api-
name=personal-demographics --proxy-name=personal-demographics-pr-871
```

See Test Utils 2.0: Running tests (see page 734) for information on:

- Running tests in pipelines
- Running tests against environments in the `nhsd-prod` Apigee Organization (eg `int`)

7.14.7.2.3 Getting authenticated with the nhsd_apim_authorization marker

The nhsd_apim_authorization marker performs the authorization process on your behalf and provides an access token or API Key.

Pass the following parameters to **configure the desired authorization method**:

- `access`
  Defined the authentication type. One of: `patient`, `healthcare_worker` or `application`
- `level`
  Sets the authorization level
  - P0, P5, P9[682] for Patient access
  - aal3, aal2, aal1 for Healthcare worker access
  - level0, level3[683] for Application-restricted access (corresponds to API Key or Client Credentials with Signed JWT, respectively)
- `login_form`
  For *user-restricted auth methods only*. Provides additional login config, such as the user to be authenticated as.

When the marker is used, the following **fixtures provide the authorization token** to your test method:

- `nhsd_apim_auth_headers`
  A headers dictionary with the access token or api-key pre-populated under its counterpart header name.
- `_nhsd_apim_auth_token_data`
  The complete token data as returned from the authorisation service

We support all types of authorisation flows available from the Authorisation Service:

What the marker does behind the scenes

Given your API name and service name, the plugin interacts directly with Apigee to:

1. Create a Developer account
2. Create an Application
   a. Including a key pair for JWT signing if necessary
3. Subscribe it to the appropriate API Product for your API Proxy and chosen access method
4. Perform authorisation against the second generation mock auth service - logging in as the selected used, or providing an signed JWT credential

682 https://nhsd-confluence.digital.nhs.uk/pages/createpage.action?fromPageId=465645396&linkCreation=true&spaceKey=APM&title=P0%2C+P5%2C+P9

683 https://nhsd-confluence.digital.nhs.uk/pages/createpage.action?fromPageId=465645396&linkCreation=true&spaceKey=APM&title=level0%2C+level3

5. Returns the access token to your test method via a fixture

After tests have executed, the plugin cleans up after itself - deleting the developer account, application etc.

The examples show a mixture of types of use cases to demonstrate how the marker may be used differently inc: single tests, parametrized tests, the marker config passed as an dict, or as parameters etc.

7.14.7.2.3.1 Application Restricted

API Key (level0)

Read more about API Key auth & auth here[684]

API-key nhsd_apim_authorization example

```
@pytest.mark.nhsd_apim_authorization({"access": "application", "level":
"level0"})
def test_app_level0_access(nhsd_apim_proxy_url, nhsd_apim_auth_headers):
    resp = requests.get(
        nhsd_apim_proxy_url + "/test-auth/app/level0", headers=nhsd_apim_a
uth_headers
    )
    assert resp.status_code == 200
```

Signed JWT (level3)

Read more about Signed JWT auth here[685]

Signed-JWT nhsd_apim_authorization example

```
@pytest.mark.nhsd_apim_authorization(access="application", level="level3")
def test_app_level3_access(nhsd_apim_proxy_url, nhsd_apim_auth_headers):
    resp = requests.get(
        nhsd_apim_proxy_url + "/test-auth/app/level3", headers=nhsd_apim_a
uth_headers
```

684 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication

685 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication

```
    )
    assert resp.status_code == 200
```

7.14.7.2.3.2 User Restricted

CIS2 seperate auth (aal3)

Read more about CIS2 separate auth here[686]

CIS2 seperate auth nhsd_apim_auth_headers example

```
@pytest.mark.nhsd_apim_authorization(
    {
        "access": "healthcare_worker",
        "level": "aal3",
        "login_form": {"username": "656005750104"},
        "authentication": "separate",
    }
)
def test_healthcare_access_level_separate_auth(
    nhsd_apim_proxy_url, nhsd_apim_auth_headers
):
    resp = requests.get(f"{nhsd_apim_proxy_url}/test-auth/nhs-cis2/aal3",
headers=nhsd_apim_auth_headers)
    assert resp.status_code == 200
```

CIS2 combined auth (aal3, aal2, aal1)

Read more about CIS2 combined auth & auth here[687]

The example below uses AAL3 as the authentication assurance level. The test will stay the same if using AAL1 or AAL2.

CIS2 combined auth nhsd_apim_auth_headers example

```
@pytest.mark.parametrize(
    ("expected_status_code", "test_path"),
    [
        pytest.param(
            200,
            "/test-auth/nhs-cis2/aal3",
```

686 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

687 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation

```
            marks=pytest.mark.nhsd_apim_authorization(
                access="healthcare_worker",
                level="aal3",
                login_form={"username": "656005750104"},
            ),
        ),
        pytest.param(
            200,
            "/test-auth/nhs-cis2/aal3",
            marks=pytest.mark.nhsd_apim_authorization(
                access="healthcare_worker",
                level="aal3",
                login_form={"username": "656005750105"},
            ),
        ),
    ],
)
def test_health_care_access_level_combined_with_parametrization(
    nhsd_apim_proxy_url, expected_status_code, test_path,
nhsd_apim_auth_headers
):
    resp = requests.get(nhsd_apim_proxy_url + test_path, headers=nhsd_apim
_auth_headers)
    assert resp.status_code == expected_status_code
```

NHS Login seperate auth (P0, P5, P9)

Read more about NHS Login seperate auth & auth here[688]

NHS Login seperate auth nhsd_apim_auth_headers example

```
@pytest.mark.nhsd_apim_authorization(
        access="patient",
        level="P9",
        login_form={"username": "9912003071"},
        authentication="separate",
)
def test_patient_access_level_separate_auth(
    nhsd_apim_proxy_url, nhsd_apim_auth_headers
):
    resp = requests.get(f"{nhsd_apim_proxy_url}/test-auth/nhs-login/P9",
headers=nhsd_apim_auth_headers)
    assert resp.status_code == 200
```

NHS Login combined auth (P0, P5, P9)

688 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation

Combined auth for NHS Login is technically available but not yet supported by NHS Login

Read more about NHS Login combined auth & auth here[689]

NHS Login combined auth nhsd_apim_auth_headers example

```
@pytest.mark.parametrize(
    ("expected_status_code", "test_path"),
    [
        pytest.param(
            200,
            "/test-auth/nhs-login/P0",
            marks=pytest.mark.nhsd_apim_authorization(
                access="patient",
                level="P0",
                login_form={"username": "9912003073"},
            ),
        ),
        pytest.param(
            200,
            "/test-auth/nhs-login/P5",
            marks=pytest.mark.nhsd_apim_authorization(
                access="patient",
                level="P5",
                login_form={"username": "9912003072"},
            ),
        ),
        pytest.param(
            200,
            "/test-auth/nhs-login/P9",
            marks=pytest.mark.nhsd_apim_authorization(
                access="patient",
                level="P9",
                login_form={"username": "9912003071"},
            ),
        ),
    ],
)
def test_patient_access_level_combined_with_parametrization(
    nhsd_apim_proxy_url, expected_status_code, test_path,
nhsd_apim_auth_headers
):
    resp = requests.get(nhsd_apim_proxy_url + test_path, headers=nhsd_apim
_auth_headers)
```

689 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-combined-authentication-and-authorisation

```
37      assert resp.status_code == expected_status_code
```

7.14.7.2.4 Fixtures

Pytest fixtures are functions that are run before tests and can be passed as parameters the return values used in the tests themselves. As well as the previously mentioned `nhsd_apim_auth_headers` and `_nhsd_apim_auth_token_data`, we have a range of custom fixtures for test utils and these fixtures are defined in the pytest-nhsd-apim repo here[690]. Below are a list of useful fixtures that may be useful in your tests and their purpose including example use cases.

> Note: The below examples are fixtures we have found to be useful in testing. Feel free to look at the other fixtures available (imported in this file[691]) and experiment with them in your tests but be aware they were designed to be used by the packages other functions and fixtures so are not formally documented.

7.14.7.2.4.1 nhsd_apim_proxy_url

When you run your tests you are required to pass in the proxy-name you'll be testing against. This fixture returns the base url for your proxy.

nhsd_apim_proxy_url example

```
1  def test_ping(nhsd_apim_proxy_url):
2      resp = requests.get(f"{nhsd_apim_proxy_url}/_ping")
3      assert resp.status_code == 200
```

7.14.7.2.4.2 status_endpoint_auth_headers

This fixture is an object with the apikey header pre-populated with the value needed to hit the _status endpoint of any proxy.

status_endpoint_auth_headers

```
1  def test_status(nhsd_apim_proxy_url, status_endpoint_auth_headers):
2      resp = requests.get(f"{nhsd_apim_proxy_url}/_ping", headers=status_end
point_auth_headers)
```

690 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_edge.py
691 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/pytest_nhsd_apim.py

```
    assert resp.status_code == 200
```

7.14.7.2.4.3 trace

Provides an interface to Apigee Trace functionality[692] allowing you to inspect and make assertions about the execution of your API proxy.

See here[693] for available methods or here[694] for an example usage.

Use the filter functionality[695] to ensure that only your specific API call(s) are being captured.

trace example

```
@pytest.mark.nhsd_apim_authorization(
    access="application", level="level3", force_new_token=True
)
def test_get_header_from_trace(
    self, _nhsd_apim_auth_token_data, nhsd_apim_proxy_url, trace
):
    access_token = _nhsd_apim_auth_token_data["access_token"]

    trace_id = str(uuid4())
    header_filters = {"trace_id": trace_id}

    session_name = str(uuid4())
    trace.post_debugsession(session=session_name, header_filters=header_filters)

    requests.get(
        url=f"{nhsd_apim_proxy_url}/splunk-test",
        headers={
            "Authorization": f"Bearer {access_token}",
            "test-header": "foo bar bar foo",
            **header_filters,
        },
    )

    traced_transaction_ids = trace.get_transaction_data(session_name=session_name)
    trace_data = trace.get_transaction_data_by_id(
        session_name=session_name, transaction_id=traced_transaction_ids[0]
    )
    trace_var = trace.get_apigee_variable_from_trace(name="splunkCalloutRequest.content", data=trace_data)
    splunk_payload = json.loads(trace_var)
```

692 https://docs.apigee.com/api-platform/debug/using-trace-tool-0
693 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py#L844
694 https://github.com/NHSDigital/pytest-nhsd-apim/blob/ca6b7080c3f62911b25537a67a2dc8810ea74aa8/tests/test_examples.py#L418
695 https://docs.apigee.com/api-platform/debug/using-trace-tool-0#refiningmessagecaptureusingfilters

```
30
31      trace.delete_debugsession_by_name(session_name)
32
33      content = splunk_payload["request"]
34      assert content["headers"]
35
36      headers = content["headers"]
37      assert headers["test-header"] == "foo bar bar foo"
```

7.14.7.2.4.4 nhsd_apim_test_app

This fixture is a callable that when called retrieves the current state of the Apigee test app being used in your tests. Can be useful for retrieving information about the app if needed in your tests.

nhsd_apim_test_app example

```
1  def test_get_header_from_trace(
2      self, nhsd_apim_test_app
3  ):
4      app = nhsd_apim_test_app()
5      app_name = app["name]
6      print(f"My current test app is - {app_name}")
```

7.14.7.2.5 Authenticators and Apigee APIs

The marker and fixtures detailed above comprise the main functionality we expect should cover >90% of use cases. Behind the scenes, the marker makes use of a number of classes and fixtures which are also available to be used. **Please consider carefully whether you need to use them directly - we encourage use of the higher-level abstractions wherever possible.**

We include a variety of authenticators including **AuthorizationCodeAuthenticator**, **ClientCredentialsAuthenticator** and **TokenExchangeAuthenticator** as well as interfaces to Apigee APIs which are ready to use and have a fully authenticated Apigee client. These authenticators have been documented under Test Utils 2.0: Authenticators and Apigee APIs (see page 729).

For a more detailed implementation of the rest of the authenticators please refer to the examples here[696].

7.14.7.2.6 Test Utils 2.0: Authenticators and Apigee APIs

The marker and fixtures detailed in the parent page comprise the main functionality we expect should cover >90% of use cases.

696 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/tests/test_examples.py#L308

Behind the scenes, the marker makes use of a number of classes and fixtures which are also available to be used.
Please consider carefully whether you need to use them directly - we encourage use of the higher-level abstractions wherever possible.

7.14.7.2.6.1 Authenticators

We have a variety of authenticators that you can use in case you want to run the tests using a sepcific app already available in Apigee. To use them you will first need to create the configuration object which will valiadate that your configuration is correct on creation using Pydantic, then pass the config to the authenticator and call **get_token()** on it. You can have a look at our examples in our pytest-nhsd-apim[697] repo as well as the complete list of the authenticators available.

AuthorizationCodeAuthenticator

The purpose of this authenticator is to comlete the user-restricted authentication journey for CIS2 or NHS Login using our auth endpoint.

- Read more about CIS2 combined auth & auth here[698]
- Read more about NHS Login combined auth & auth here[699]

test_authorization_code_authenticator

```
def test_authorization_code_authenticator(_test_app_credentials,
apigee_environment):
    # 1. Set your app config
    config = AuthorizationCodeConfig(
        environment=apigee_environment,
        identity_service_base_url=f"https://
{apigee_environment}.api.service.nhs.uk/oauth2-mock",
        callback_url="https://example.org/callback",
        client_id=_test_app_credentials["consumerKey"],
        client_secret=_test_app_credentials["consumerSecret"],
        scope="nhs-cis2",
        login_form={"username": "656005750104"},
    )

    # 2. Pass the config to the Authenticator
    authenticator = AuthorizationCodeAuthenticator(config=config)

    # 3. Get your token
```

697 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/tests/test_examples.py
698 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation
699 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-combined-authentication-and-authorisation

```
    token_response = authenticator.get_token()
    assert "access_token" in token_response
    token = token_response["access_token"]
```

ClientCredentialsAuthenticator

This gets you authenticated using the client_credentials flow in the apim identity service. This method of authentication is also known as 'application restricted' or 'signed jwt'.

Read more about Signed JWT auth & auth here[700]

test_client_credentials

```
def test_client_credentials_authenticator(
    _test_app_credentials, _jwt_keys, apigee_environment
):
    # 1. Set your app config
    config = ClientCredentialsConfig(
        environment=apigee_environment,
        identity_service_base_url=f"https://
{apigee_environment}.api.service.nhs.uk/oauth2-mock",
        client_id=_test_app_credentials["consumerKey"],
        jwt_private_key=_jwt_keys["private_key_pem"],
        jwt_kid="test-1",
    )

    # 2. Pass the config to the Authenticator
    authenticator = ClientCredentialsAuthenticator(config=config)

    # 3. Get your token
    token_response = authenticator.get_token()
    assert "access_token" in token_response
    token = token_response["access_token"]
```

TokenExchangeAuthenticator

The purpose of this authenticator is to get you authenticated using the token exchange flow in the identity service flow.

- Read more about CIS2 seperate auth & auth here[701]
- Read more about CIS2 combined auth & auth here[702]

700 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication

701 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

702 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation

test_token_exchange

```python
def test_token_exchange_authenticator(
    _test_app_credentials, apigee_environment, _jwt_keys, _keycloak_client_credentials
):
    # This one is a bit more interesting since we first need to get a token from
    # keycloak (our mock version of the NHS-Login provider) grab the id_token
    # from that token and then make a call to the identity service to "exchange"
    # the id_token for an Apigee access token.

    # 1. Set yout keycloack config
    config1 = KeycloakUserConfig(
        realm=f"NHS-Login-mock-{apigee_environment}",
        client_id=_keycloak_client_credentials["nhs-login"]["client_id"],
        client_secret=_keycloak_client_credentials["nhs-login"]["client_secret"],
        login_form={"username": "9912003071"},
    )

    # 2. Pass the config to the Authenticator
    authenticator = KeycloakUserAuthenticator(config=config1)

    # 3. Get your id_token
    id_token = authenticator.get_token()["id_token"]

    # 4. Set your token exchange config
    config = TokenExchangeConfig(
        environment=apigee_environment,
        identity_service_base_url=f"https://{apigee_environment}.api.service.nhs.uk/oauth2-mock",
        client_id=_test_app_credentials["consumerKey"],
        jwt_private_key=_jwt_keys["private_key_pem"],
        jwt_kid="test-1",
        id_token=id_token,
    )

    # 5. Pass the config to the Authenticator
    authenticator = TokenExchangeAuthenticator(config=config)

    # 6. Get your token
    token_response = authenticator.get_token()
    assert "access_token" in token_response
    token = token_response["access_token"]
```

7.14.7.2.6.2 Apigee APIs

We also have a varierty of Apigee APIs with the benefit of a fully authenticated Apigee client ready to use if you didnt want to create your own proxy and authenicatate using the examples above.

The APIs we offer at the moment are:

API	Methods	Documentation
DeveloperAppsAPI	here[703]	Overview[704]
ApiProductsAPI	here[705]	Overview[706]
DebugSessionsAPI	here[707]	Overview[708]
AccessTokensAPI	here[709]	Overview[710]

These APIs are defined in apigee_apis.py[711] in the pytest-nhsd-apim repo. You will have access to a number of ready-to-use authenticators such as **ApigeeNonProdCredentials** and **ApigeeProdCredentials**. Both designed to check credentials and retieve info from the API proxy sepcified.

Example use:

apigee_apis_authenticators

```
from pytest_nhsd_apim.apigee_apis import ApigeeNonProdCredentials, DeveloperAppsAPI

config = ApigeeNonProdCredentials()
client =  ApigeeClient(config=config)
apps = DeveloperAppsAPI(client=client)

print(apps.list_aps('{email}'))
```

703 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py#L292
704 https://apidocs.apigee.com/docs/developer-apps/1/overview
705 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py#L575
706 https://apidocs.apigee.com/docs/api-products/1/overview
707 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py#L844
708 https://apidocs.apigee.com/docs/debug-sessions/1/overview
709 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py#L983
710 https://apidocs.apigee.com/docs/oauth-20-access-tokens/1/overview
711 https://github.com/NHSDigital/pytest-nhsd-apim/blob/main/src/pytest_nhsd_apim/apigee_apis.py

7.14.7.2.7 Test Utils 2.0: Running tests

7.14.7.2.7.1 Running tests locally

Once the package is installed, use the following command to get up and running locally testing against nonprod environments (internal-dev, internal-qa, ref).

See environment variables and inputs on the previous page for where to get `get_token` and the inputs

```
export APIGEE_ACCESS_TOKEN="$(get_token)"; poetry run pytest <PATH_TO_TESTS> --api-
name=<API_NAME> --proxy-name=<PROXY_NAME>
```

7.14.7.2.7.2 Running tests on pipelines

Most repos will be configured to use the Azure common pipelines but there are some cases where test-utils may be run in other types of pipelines. The advantage of running tests in the common pipelines is that the APIGEE_ACCESS_TOKEN is scoped to an APIM machine user that has full-privilages in the nhsd-nonprod org. If using your own access token (for local tests or in other pipelines) the token will be scoped to your Apigee permissions. As most producers have APIProducerDev roles in nhsd-nonprod Apigee, they should find they have permissions to do everything needed but it's important to be aware of that distinction.

Azure common pipelines

When running tests with Test Utils 2.0 on the Azure common pipelines[712] the environment variables and inputs are available already in the pipeline.

These tests will usually be configured in the 'run-tests' pipeline in most repos. See the template-repo[713] for how tests are configured by default.

```
    - bash: |
        export PROXY_NAME="$(FULLY_QUALIFIED_SERVICE_NAME)"
        export API_NAME="$(SERVICE_NAME)"
        export APIGEE_ACCESS_TOKEN="$(secret.AccessToken)"
        make test
      workingDirectory: $(Pipeline.Workspace)/s/$(SERVICE_NAME)/$
(SERVICE_ARTIFACT_NAME)
      displayName: Run full test suite
```

Other pipeline uses

If you need to run your tests in other pipelines you can. The example we'll use is Github Actions.

712 https://dev.azure.com/NHSD-APIM/API%20Platform/_build
713 https://github.com/NHSDigital/api-management-service-template

As the APIGEE_ACCESS_TOKEN will not be automatically available to you when you don't have the common pipelines, you will need to generate that yourself in the pipeline. This will mean storing **APIGEE_USERNAME**, **APIGEE_PASSWORD** and **APIGEE_OTP_KEY** in Github secrets.

OTP keys are used to generate MFA codes that will then be used to get the Apigee token.

Create **otp.py** in your repo and add following contents:

otp.py

```python
import sys
import pyotp


def main(username, otp_key):

    otp = pyotp.parse_uri(f"otpauth://totp/{username}?secret={otp_key}")
    print(otp.now())


if __name__ == "__main__":
    username = sys.argv[1]
    otp_key = sys.argv[2]
    main(username, otp_key)
```

An example action for pulling in GitHub secrets, installing dependancies, getting and Apigee access token and running tests. The file would be: **.github/workflows/integration_tests_pipeline.yml**:

.github/workflows/integration_tests_pipeline.yml

```yaml

name: Run integration tests on PR
on: pull_request
jobs:
integration-tests:
runs-on: ubuntu-latest
env:
APIGEE_USERNAME: ${{ secrets.APIGEE_USERNAME }}
APIGEE_PASSWORD: ${{ secrets.APIGEE_PASSWORD }}
APIGEE_OTP_KEY: ${{ secrets.APIGEE_OTP_KEY }}
steps:
- name: Checkout current branch
uses: actions/checkout@v3

- name: install dependencies
run: <run your install scripts>

- name: get otp
id: otp
```

```
19  run: echo ::set-output name=key::$(poetry run python otp.py $
    {APIGEE_USERNAME} ${APIGEE_OTP_KEY})
20
21
22  - name: Install get_token
23  run: |
24  curl https://login.apigee.com/resources/scripts/sso-cli/ssocli-bundle.zip
    -O
25  unzip ssocli-bundle.zip
26
27  - name: Get APIGEE access token
28  id: apigee
29  run: |
30  echo ::add-mask $(SSO_LOGIN_URL=https://login.apigee.com ./get_token -u $
    {APIGEE_USERNAME}:${APIGEE_PASSWORD} -m ${{ steps.otp.outputs.key }})
31  echo ::set-output name=token::$(SSO_LOGIN_URL=https://login.apigee.com ./
    get_token -u ${APIGEE_USERNAME}:${APIGEE_PASSWORD} -m $
    {{ steps.otp.outputs.key }})
32
33  - name: test
34  run: APIGEE_ACCESS_TOKEN=${{ steps.apigee.outputs.token }}; poetry run
    pytest <PATH_TO_TESTS> --api-name=<API_NAME> --proxy-name=<PROXY_NAME>
```

7.14.7.2.7.3 Running tests against nhsd-prod org environments

The pytest package is not designed to be used with nhsd-prod org environments, as that org is a 'clean' environment as we don't want to be making on-demand test apps. Therefor only some of the functionality of the package should be used for nhsd-prod environments. We recommend testing against internal-dev in the nhsd-nonprod org for functionality (as it should be the same across envs) and only doing basic smoke tests against the other environments.

In order to run the package against an environment in the nhsd-prod org (int, sandbox, dev, prod) Apigee you need to supply a test-app id. This is because we do not allow on-demand test apps in nhsd-prod and the nhsd_apim_authorization marker will not work here. It's still possible though to run tests using your own test app and the Authenticators (see page 729) but will need to replace fixtures with your own values (i.e test app credentials and private keys).

- Identify an app in prod Apigee that is authorised against the instance of the API you want to test
- Supply the API key used to authorise calls to the `_status` endpoint of your API (if you're testing the status endpoint).
- Supply the Apigee access token so the package can run but be aware that it is only used to make the package comply.

You can do both of these by using the below test command:

```
export APIGEE_ACCESS_TOKEN="$(get_token)"; poetry run pytest <PATH_TO_TESTS> --api-name=<API_NAME> --proxy-name=<PROXY_NAME> --apigee-app-id=<your-app-id> --apigee-organization=nhsd-prod --status-endpoint-api-key=<key>
```

If your API repo has been generated from the APIM service template, the relevant pipelines are already set up to retrieve the status endpoint API key and supply it to the test command. You need only provide the app id.

```
variables:
  - group: status-endpoint
    {% raw %}${{ if parameters.apigee-prod-org-tests }}:{% endraw %}
      # In order to run tests in prod you must supply the unique ID of an Apigee app
      # that has authorized access to your service proxy.
      - name: test-variables:
      - value: |
          PROXY_NAME=$(FULLY_QUALIFIED_SERVICE_NAME) \
          SOURCE_COMMIT_ID=$(Build.SourceVersion) \
          APIGEE_ACCESS_TOKEN=$(secret.AccessToken) \
          APIGEE_APP_ID= \
          STATUS_ENDPOINT_API_KEY=$(STATUS_ENDPOINT_API_KEY) \
      - name: test-command
      - value: make test-prod
      - name: smoketest-command
      - value: make smoketest-prod
    {% raw %}- ${{ if not(parameters.apigee-prod-org-tests) }}:{% endraw %}
      - name: test-variables:
      - value: |
          PROXY_NAME=$(FULLY_QUALIFIED_SERVICE_NAME) \
          SOURCE_COMMIT_ID=$(Build.SourceVersion) \
          APIGEE_ACCESS_TOKEN=$(secret.AccessToken) \
      - name: test-command
      - value: make test
      - name: smoketest-command
      - value: make smoketest
```

7.14.7.3 Testing with mock auth

7.14.7.3.1 Overview

Our second generation mock authorisation service[714] can be used for testing APIs that use the user-restricted authentication pattern[715]. You can retrieve an access token from the oauth2-mock endpoint and provide the UID of one of our pre-defined test users. Our service uses Keycloak to simulate a sign-in with a smartcard or modern alternative. We have an OIDC provider in place of CIS2 or NHS Login, which simplifies the authentication process. Previously, you would have to manually set up a config and write code to authenticate. But with our service, you can eliminate the need to write any code and use tools like Postman to connect directly to the CIS2 or NHS Login service.

The service can be used by:

714 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#second-generation-service
715 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#user-restricted-apis

- API producer teams, testing in internal environments
- API consumers, testing in the integration test environment

This page explains how to use the service as an API producer team.

7.14.7.3.2 Testing options

When testing with the second generation mock authorisation service you should consider your testing needs:

1. Testing with code using the test utils 2.0 package[716]
    - Integrate authorisation into your API's automated or functional tests.
    - Avoid manual configuration with automatic test apps.
2. Testing using Postman using combined auth (see page 738)
    - Make quick authenticated calls to your API.
    - No code involved.

7.14.7.3.3 Testing with code using the Test Utils 2.0 package

The Test Utils 2.0 package is fully integrated with second generation mock auth. The package makes it straight forward to build functional and automated tests without having to build the authorisation logic.

To test with code, you'll need to:

1. Get set up to use the package[717].
2. Read up on how to test with user-restricted auth using the package[718].
3. Decide on an NHS CIS2 test user[719] or NHS Login test user[720] for simulating a sign in.

7.14.7.3.4 Testing with combined-auth using Postman

Testing with Postman is a great way to make API calls quickly and easily so that you can check responses without code integration. Postman can also be useful for non-technical people to interract with an API without needing to build technical applications.

To test with Postman you'll need to:

- Sign up for Postman[721].

---

716 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?spaceKey=APM&title=APIM+Test+Utils+2.0+plugin
717 https://nhsd-confluence.digital.nhs.uk/display/APM/Test+Utils+2.0%3A+python+testing+package#TestUtils2.0:pythontestingpackage-Gettingstarted
718 https://nhsd-confluence.digital.nhs.uk/display/APM/Test+Utils+2.0%3A+python+testing+package#TestUtils2.0:pythontestingpackage-UserRestricted
719 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#test-users-for-nhs-cis2
720 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#test-users-for-nhs-login
721 https://www.postman.com/

    - Set up Postman for web or download the Postman app (recommended for ease when configuring authorisation).
- Create an Apigee app[722].
- Decide on an NHS CIS2 test user[723] or NHS Login test user[724] for simulating a sign in.

7.14.7.3.4.1 CIS2 vs NHS Login

The only differences in approach for using CIS2 or NHS Login are:

- The scope field: nhs-login for NHS Login or leave blank for CIS2
- The test user used. This should be valid for provider type. See the test users section for some examples.

7.14.7.3.4.2 Step 1: set up authorisation

Just like when authenticating with a smartcard using Postman, you will need to configure the 'Authorization' tab to use OAuth 2.0. By following the example you will:

- Set up to authenticate with the second generation mock authorisation service[725]
- Get an access token using Keycloak
- Use the access token to retrieve information on the test user from the `/userinfo` endpoint.

Use the config below or download our template Postman collection[726] to get started.

On the 'Authorization' tab of your request set the 'Type' to OAuth 2.0 and and fill out the fields of 'Configure New Token' to get an access token from Keycloak:

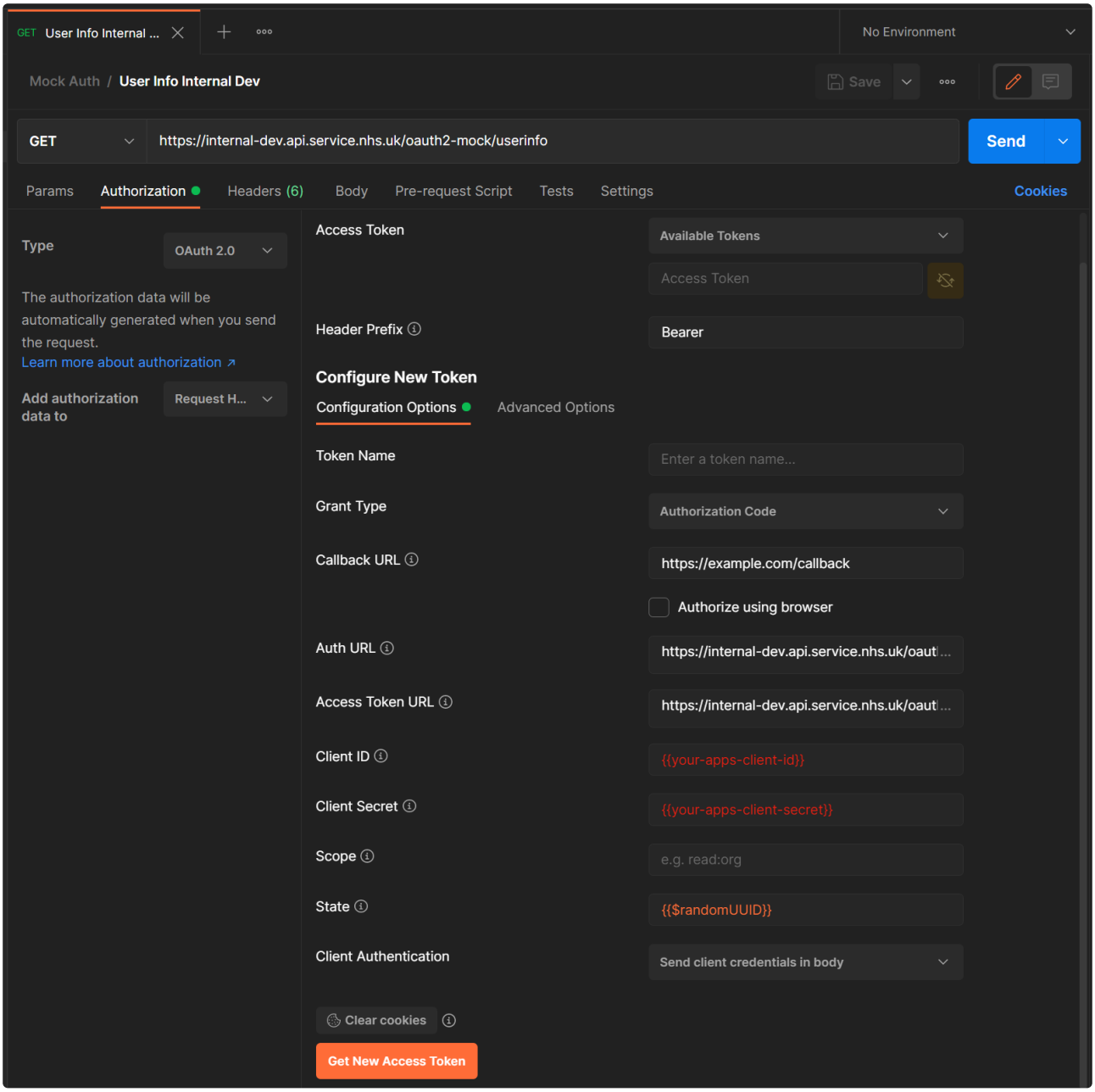
Field	Info	Example
Token Name	Name your token for use with later requests. *Optional*	`Mock Auth Token`
Grant Type	The way an application gets a token. Must be 'Authorization Code'	`Authorization Code`

722 https://digital.nhs.uk/developer/guides-and-documentation/testing#integration-testing-with-our-restful-apis
723 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#test-users-for-nhs-cis2
724 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#test-users-for-nhs-login
725 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#second-generation-service
726 https://app.getpostman.com/run-collection/11953223-2afe541b-8b52-4d4b-972a-7b54335e7f93

Field	Info	Example
Callback URL	Must be a valid callback url and must match the callback url on your Apigee app. If 'Authorize using browser' is ticked you must set the provided url as the callback of your app. Using the postman app is recommended for customising this field.	`https://example.com/callback`
Auth URL	The authorize endpoint of the client you are authenticating with. The client should respond to the environment you are testing in. Our available clients are `internal-dev`, `internal-qa` and `int`	`https://<environment>.api.service.nhs.uk/oauth2-mock/authorize`
Access Token URL	The token endpoint of the client you are authenticating with. The client should respond to the environment you are testing in. Our available clients are `internal-dev`, `internal-qa` and `int`	`https://<environment>.api.service.nhs.uk/oauth2-mock/token`
Client ID	The client ID of your Apigee App.	N/A
Client Secret	The client secret of your Apigee App.	N/A
Scope	Scope can determine the kind of authorisation pattern you are using. Leave blank for CIS2 combined auth, which is currently the only pattern supported. *Optional*	Leave blank for CIS2 or `nhs-login` for NHS Login
State	State is used to keep track of the authentication across the 'authorize' and 'callback' requests that happen under the hood. To avoid clashes, it's best to use something unique like a random UUID.	Postman has a random UUID function available - `{{$randomUUID}}`

Field	Info	Example
Client Authentication	Our authorisation service expects for client credentials to be sent in the body.	`Send client credentials in body`

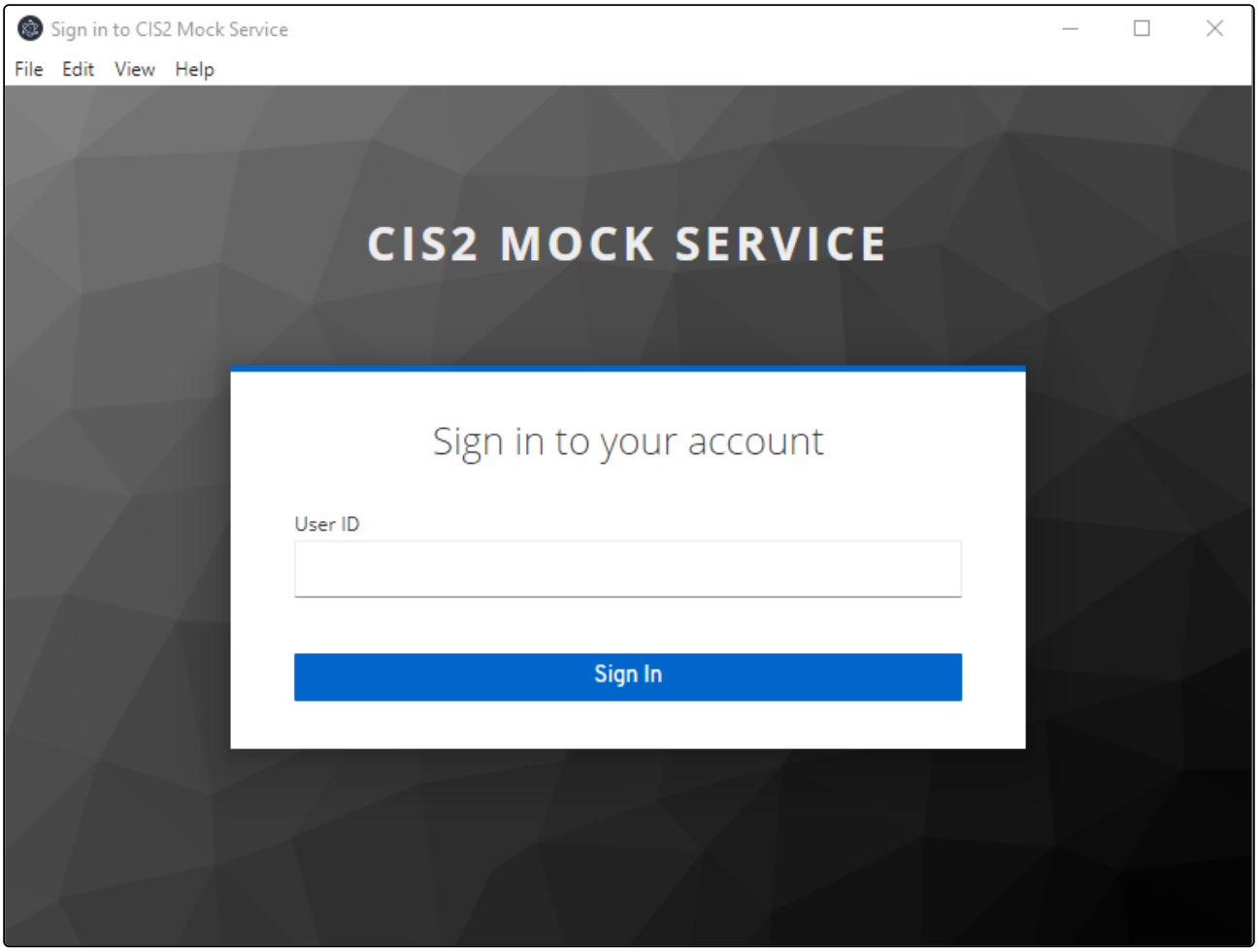
Use the screenshot below for reference.



7.14.7.3.4.3 Step 2: get an access token with Keycloak

After filling in the fields of the 'Authorization' tab, click 'Get New Access Token'.

The CIS2/NHS Login browser window will appear. This is you authenticating with keycloak.



Provide the User ID of the test user you'd like to authenticate with. After clicking 'Sign In' you will receive an access token, which you can use to make calls to user-restricted APIs that allow the combined auth pattern.

For a list of users available see Test Users (see page 749).

7.14.7.3.4.4 Step 3: (CIS2 only) hit the userinfo endpoint with your access token

You can use the `/userinfo` endpoint to check your authorisation has been successful or to retrieve information on the user you authenticated with when getting your access token.

When you've received your access token click 'use token' to apply it to your request or add it as the following header: `Authorization: Bearer <access_token>`.

Send a GET request to `https://<environment>.api.service.nhs.uk/oauth2-mock/userinfo` replacing <environment> with the the one you're testing in (internal-dev, internal-qa or int).

A successful response will return the info of the user you used to signed in, much like the real CIS2 user info pattern.

Example userinfo response for user 656005750107


```
{
    "sub": "656005750107",
    "gmp_id": "GMP123",
    "consultant_id": "TestConsultant",
    "title": "Mrs",
    "uid": "656005750107",
    "rcn_id": "RCN123",
    "nhsid_nrbac_roles": [
        {
            "person_orgid": "656008045105",
            "person_roleid": "555254242105",
            "org_code": "Y51",
            "role_name": "\"Clinical\":\"Clinical Provision\":\"Clinical
Practitioner",
            "role_code": "S8000:G8000:R8000"
        }
    ],
    "nmc_id": "NMC123",
    "middle_names": "Shirley",
    "selected_roleid": "555254242105",
    "email": "shirley.bryne@nhs.net",
    "gdp_id": "GDP123",
    "nhsid_useruid": "656005750107",
    "nhsid_org_memberships": [
        {
            "person_orgid": "656008045105",
            "org_name": "THE NORTH MIDLANDS AND EAST PROGRAMME FOR IT (NMEPFIT)",
            "org_code": "Y51",
            "gnc": "GNC123"
        },
        {
            "person_orgid": "656005753107",
            "org_name": "NHS CONNECTING FOR HEALTH",
            "org_code": "X09"
        }
    ],
    "gdc_id": "GDC123",
    "initials": "S",
    "nhsid_org_roles": [
        {
            "person_orgid": "656008045105",
            "org_name": "THE NORTH MIDLANDS AND EAST PROGRAMME FOR IT (NMEPFIT)",
            "org_code": "Y51"
        },
        {
```

```
            "person_orgid": "656005753107",
            "org_name": "NHS CONNECTING FOR HEALTH",
            "org_code": "X09"
        }
    ],
    "gmc_id": "GMC126",
    "display_name": "User3 Internal-Test",
    "given_name": "User3",
    "nhsid_user_orgs": [
        {
            "org_name": "THE NORTH MIDLANDS AND EAST PROGRAMME FOR IT (NMEPFIT)",
            "org_code": "Y51"
        },
        {
            "org_name": "NHS CONNECTING FOR HEALTH",
            "org_code": "X09"
        }
    ],
    "name": "User3 Internal-Test",
    "idassurancelevel": "3",
    "family_name": "Internal-Test",
    "gphc_id": "GPhC123"
}
```


7.14.7.3.5 Testing with seperate-auth using Postman

When testing with seperate authorisation and authentication using mock auth it's authorisation part that is mocked. Keycloak (our mock OIDC provider) is replacing CIS2 or NHS login in the authorisation part of the journey but the authentication part of the journey is still the same. **The following steps should be used to replace step 5 on the below documentation. All other steps should remain the same.**

- User-restricted RESTful APIs - NHS Care Identity Service 2 separate authentication and authorisation[727]
- User-restricted RESTful APIs - NHS login separate authentication and authorisation[728]

7.14.7.3.5.1 Step 1: Obtain keycloak credentials

To generate the mocked id token you must obtain the credentials for keycloak. To obtain these credentials you can make a call to our mock-jwks service, the application you are using to access the proxy must be subscribed to the mock-jwks product in the same environment you are testing in. mock-jwks is a proxy on the platform like any other and behaves the same way. It can be added through the portal on your app or manually through Apigee (in nhsd-nonprod if you have access) like any other product. Once this product is associated with your application, you can call the mockjwks API's keycloak-client-credentials endpoint with the API Key of the application replacing the environment with the environment you're working with:

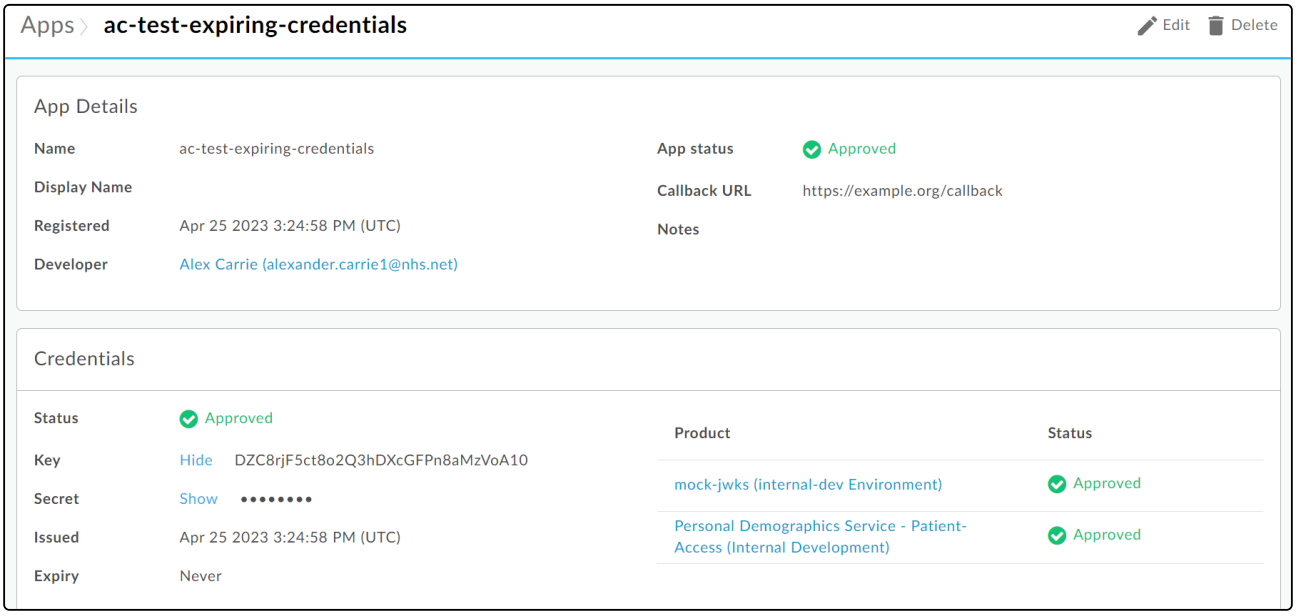
727 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

728 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation

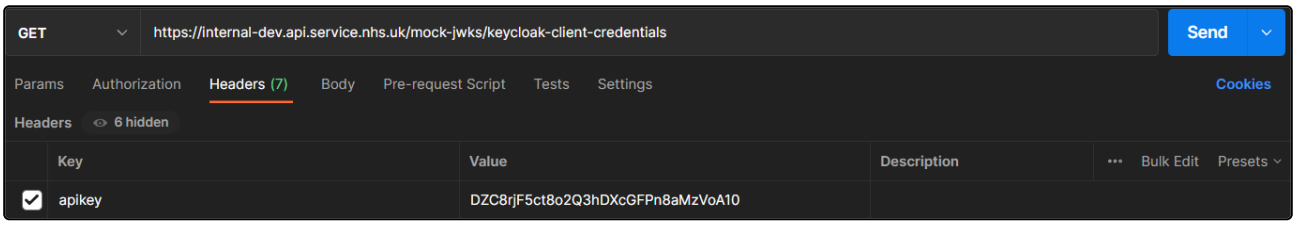
`https://<environment>.api.service.nhs.uk/mock-jwks/keycloak-client-credentials`

You need to include the following headers in your call:

Header	Value
`apikey`	The API Key (aka client ID) from your application



Apigee app with api key



Using the apikey to call the mock-jwks

If the request is valid, the mock-jwks API responds with a HTTP status 200 response. This response contains the following set of parameters for both CIS2 and NHS Login. Use the relevant ones to the provider you're mocking.

Parameter	Description
client_id	The client identifier issued to the mocked authorisation provider client

Parameter	Description
client_secret	The secret assigned to the mocked authorisation provider client.
redirect_uri	`https://example.org`

7.14.7.3.5.2 Step 2: Call the /authorize endpoint of keycloak from the browser

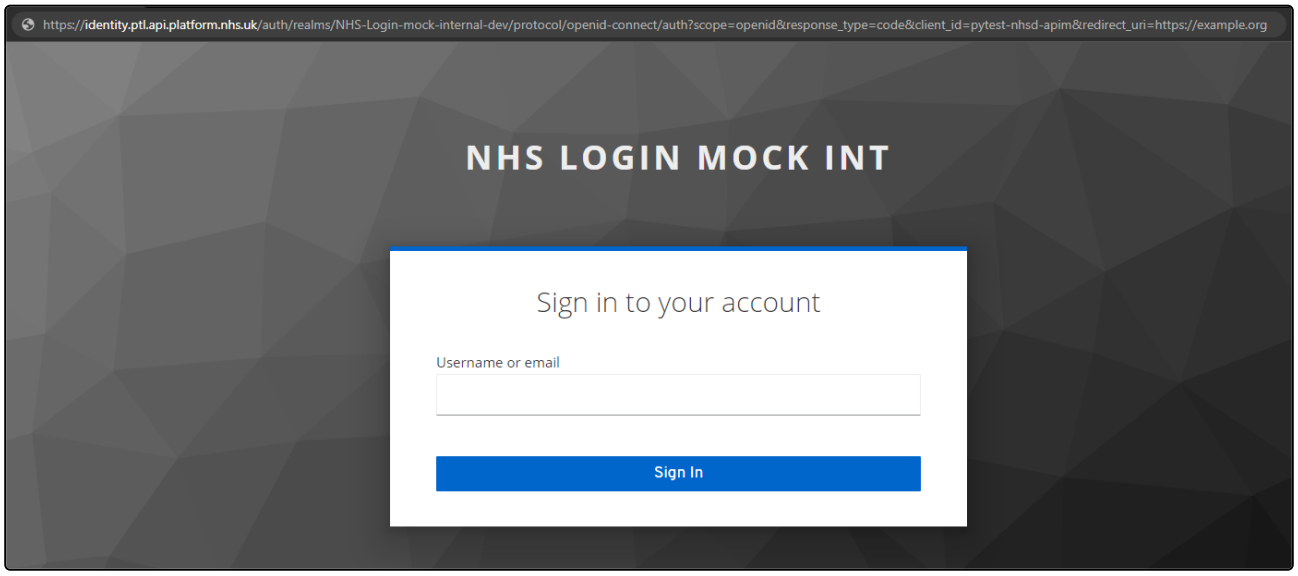
> This GET request must be copied out into the browser due to a bug in Postman's callback set up

Call our authorize endpoint to authenticate via browser in exchange for an authorization code. This call is an HTTP GET to one of the following endpoints (depending on the ODIC provider you're mocking) replacing the environment with the environment you're working in:

- `https://identity.ptl.api.platform.nhs.uk/realms/NHS-Login-mock-<environment>/protocol/openid-connect/auth`
- `https://identity.ptl.api.platform.nhs.uk/realms/Cis2-mock-<environment>/protocol/openid-connect/auth`

You need to include the following query parameters in your call:

Parameter	Description
scope	`openid`
response_type	`code`
client_id	the `client_id` returned from the call to the `/keycloak-client-credentials` endpoint
redirect_uri	`https://example.org`



Enter a valid test user (see Test Users (see page 749)) and click 'Sign In'.

7.14.7.3.5.3 Step 3: Grab the code from the browser redirect

If the request is valid, the OAuth 2.0 authorisation server responds with a HTTP status 200 and anormal redirect to a login form from the auth endpoint. Upon entering a valid set of credentials for a user the browser will navigate to the redirect_uri provided in the authorization call, with an authorization_code as a query parameter. You can find the code parameter in the url when using the browser:

7.14.7.3.5.4 Step 4: Post to keycloaks /token endpoint to retrieve ID token

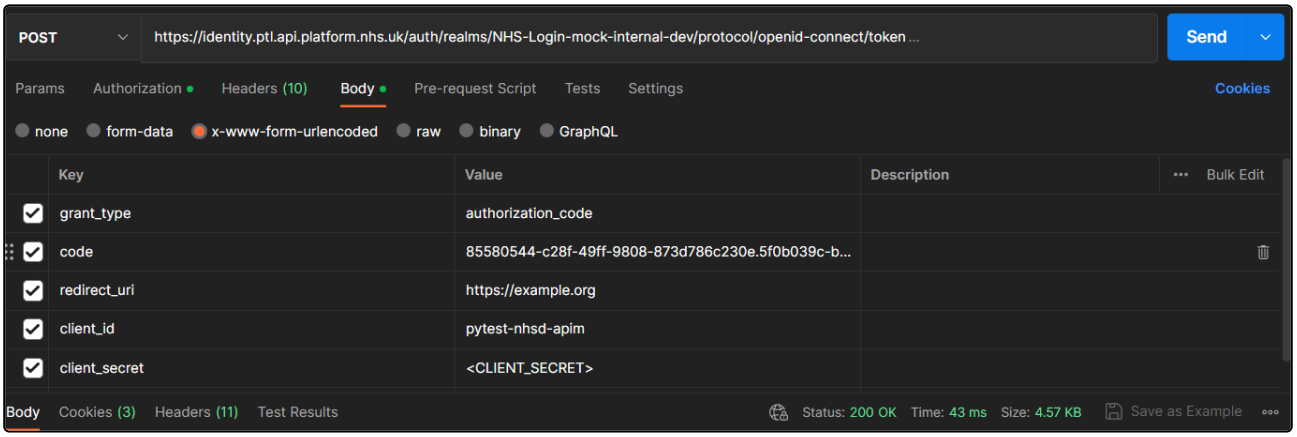
Call keycloaks token endpoint to exchange the credentials and authorization code for an CIS2/NHS Login ID token. This call is an **HTTP POST** one of the following endpoints (depending on the ODIC provider you're mocking) replacing the environment with the environment you're working in:

- `https://identity.ptl.api.platform.nhs.uk/realms/Cis2-mock-<environment>/protocol/openid-connect/token`

- `https://identity.ptl.api.platform.nhs.uk/realms/NHS-Login-mock-<environment>/protocol/openid-connect/token`

You need to include the following as x-www-form-urlencoded parameters in your call:

Form Param	Description
grant_type	`authorization_code`
code	the authorization code returned from the call to the authorization endpoint in Step 3
redirect_uri	`https://example.org`
client_id	the `client_id` returned from the call to the `/keycloak-client-credentials` endpoint
client_secret	the `client_secret` returned from the call to the `/keycloak-client-credentials` endpoint



On a successful response you will recieve back an access token to use as your ID token. Use this ID token and continue with the proccess of exchanging the token for an access token for your proxy:

- User-restricted RESTful APIs - NHS Care Identity Service 2 separate authentication and authorisation[729]

729 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

- User-restricted RESTful APIs - NHS login separate authentication and authorisation[730]

7.14.7.3.6 Test users

7.14.7.3.6.1 CIS2 test users

User UID	Name	National RBAC Coding	Notes
		Job Role (R) Codes	
656005750107	Bryne, Shirley	R8000 (Clinical Practitioner Access Role)	
656005750104	Kommareddy, Surekha	R8008 (Admin/Clinical Support)	
555260695103	Perry, Leo	R8000 (Clinical Practitioner Access Role)	Functional testing user for e-RS
103283499984	Wornin, Denise	R8010 (Clerical Access Role) R0050 (Consultant)	"E-RS CONSULTANT" user
555031998104	BJSS TEST-FIVE	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner)	Functional testing user for e-RS
555031999105	BJSS TEST-SIX	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner) R5170 (End Point Administrator)	Functional testing user for e-RS

730 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation

User UID	Name	National RBAC Coding	Notes
		Job Role (R) Codes	
555032000100	BJSS TEST-SEVEN	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner) R5170 (End Point Administrator)	Functional testing user for e-RS
555032006106	BJSS TEST-THIRTEEN	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner) R5170 (End Point Administrator) R5090 (Registration Authority Agent)	Functional testing user for e-RS
555073103100	BJSS Team	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner) R5170 (End Point Administrator) R5080 (Registration Authority Manager) R0008 (Demographic Supervisor)	Functional testing user for e-RS
555043300081	Saubers, Peter	R0260 (General Medical Practitioner)	
555083334100	Langley, Alex	R8010 (Clerical Access Role) R0050 (Consultant)	Functional testing user for e-RS

User UID	Name	National RBAC Coding	Notes
		Job Role (R) Codes	
555083343101	Langley, Alex	R8010 (Clerical Access Role) R0050 (Consultant) R0260 (General Medical Practitioner) R8015 (Systems Support Access Role)	Functional testing user for e-RS
656005750109		R8000 (Clinical Practitioner Access Role)	AAL2 level authentication with MS Authetication TOTP
656005750110		R8000 (Clinical Practitioner Access Role)	AAL1 level authentication ith Username/ Password
656005750111		R0260 (General Medical Practitioner)	AAL2 level authentication with MS Authetication TOTP
656005750112		R8008 (Admin/Clinical Support)	AAL2 level authentication with MS Authetication TOTP

7.14.7.3.6.2 NHS Login test users

User UID	Last Name, First Name	Identity Proofing Level	Notes
9912003073	USER3, TEST	P0	InvalidUserLogin on retrieve (non-prod apigee)
9912003072	USER2, TEST	P5	InvalidUserLogin on retrieve (non-prod apigee)
9912003071	USER, TEST	P9	Successful self-retrieve (non-prod apigee), vot: P9.Cp.Cd
9900000285	WILDE, Martin	P9	Successful self-retrieve (non-prod apigee), vot: P9.Cp.Cd

User UID	Last Name, First Name	Identity Proofing Level	Notes
9472063845	Egan, Dennis	P9	Successful self-retrieve (before vot check in code), no vot. P9
9472063810	Gorman, Nathan	P9	Not found (non-prod apigee), P9 only, no vot
9472063802	Hannay, Jeff	P9	Not found (non-prod apigee), P9 only, no vot
9462978182	GREENE, Andrea	P9	Not found (non-prod apigee), P9 only, no vot
9462767386	WRENCH, SHERI	P9	Not found (non-prod apigee), P9 only, no vot
9462767270	LOGAN, KIERAN	P9	Not found (non-prod apigee), P9 only, no vot
9446041481	DENIS, Forest	P9	Not found (non-prod apigee), P9.Cp.Cd
5900110915	NICHOLSEN, GARFIELD	P9	Not found (non-prod apigee), P9 only, no vot
5900068196	SMITH, Cp	P9	Successful self-retrieve (non-prod apigee), vot: P9.Cp
5900069176	SMITH, Cp Ck	P9	Successful self-retrieve (non-prod apigee), vot: P9.Cp.Ck
5900071332	Smith, Cm	P9	Successful self-retrieve (non-prod apigee), vot: P9.Cm
9449304130	TWITCHETT, Stacey Marisa	P9	Tim Coates requested for EPS testing
9730675929		P9	Chris Young request for Validated Relationships Service

User UID	Last Name, First Name	Identity Proofing Level	Notes
9730675988		P9	Chris Young request for Validated Relationships Service
9730676046		P9	Chris Young request for Validated Relationships Service
9731974830		P9	Chris Young request for Validated Relationships Service

7.14.7.3.6.3 Adding new test users

If you need a test user with different attributes, please contact us on the producer support channel[731] providing the details required to build the user

See our external documentation[732] for more information on test users.

7.14.7.4 Automated quality testing of your API

7.14.7.4.1 Overview

There are a significant number of tools available, and with the sensitivity of data that NHS Digital handles, it is essential to use every mechanism at hand to reduce accidental mistakes, poor coding, and keep using the latest code to remove vulnerabilities.

This pages sets out how an API Producer might use those tools to:

- Improve quality by working inside the NHS Digital Software Engineering Quality Framework[733]
- Meet the requirements in the "Health and Social Care Cloud Security - Good Practice Guide"
- Ensure there are continuous security tooling running, this will:
    - Significantly reduce risks - for example regularly updated configuration, results in a network port becoming visible to the internet
    - Reduce scope of external pen tests
- Speed up delivery by reducing the times when an external pen test is carried out

731 https://nhsdigitalcorporate.enterprise.slack.com/archives/C016JRWN6AY
732 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-apis-with-our-mock-authorisation-service#test-users-for-nhs-cis2
733 https://github.com/NHSDigital/software-engineering-quality-framework

7.14.7.4.2 Types of automated tooling

The automated scanning comes in various flavours / levels:

- Static - prior to deployment:
  - Check for secrets left in your code
  - Dependency scanning - finding where an external library has been exploited
  - Code vulnerability scanning
  - Code smells
- Runtime:
  - Network ports visible on the internet (or internal VPN)
  - Vulnerability scanning - looking at what is deployed and confirming that is not currently exposing a known vulnerabilities
  - URL / endpoint security - especially where there are complex authorisation rules

Tools that are in use, or being considered for use are:

Area	Type	Tools
Secrets	Static	NHSD pre commit hook scanning tool on the SEQF[734]
Dependency	Static	Dependabot for GitHub Snyk[735] SonarCloud[736] Sonatype Nexus AWS - image dependency scanning on upload to ECR
Code vulnerability	Static	CodeQL - GitHub SonarCloud[737] Sonatype Nexus Snyk[738]

734 https://github.com/NHSDigital/software-engineering-quality-framework/tree/main/tools/nhsd-git-secrets
735 https://snyk.io/
736 https://www.sonarsource.com/products/sonarcloud/
737 https://www.sonarsource.com/products/sonarcloud/
738 https://snyk.io/

Area	Type	Tools
Network port scanning	Runtime	NMap ...
Vulnerability scanning	Runtime	Amazon Inspector - AWS ...

Specific team approaches:

- NHS login:
  - Secrets scanning: Git Secrets - https://nhsd-confluence.digital.nhs.uk/display/NL/Secret+Scanning
  - Code & dependency scanning: Sonatype Nexus IQ - https://nhsd-confluence.digital.nhs.uk/display/NL/Developer+how-to+guides#Developerhowtoguides-AddingscanningtoCI/CD
  - MRs are scanned using AWS code guru
- APIM are using the the SEQF secret scanning, and Dependabot as the core items, but it is really the API Producers who setup their own tooling.

7.14.7.5 INT environment and multiple identity services

To keep the total number of environments down, and make it easier for API Consumers to carry out early testing it was agreed that the INT environment would be unusual and have three Identity Services:

- Identity Service (/oauth2) - full integration with NHS ID INT environment, supports smart cards, Yubikeys and Window Hello
- Identity Service (/oauth2-mock) - Mocked test environment with test users and mocked auth (via keycloak)

Key design points:

- They are inter-changeable: you can get an access token from either and use it on any INT API
- Access tokens are validated when hitting an API, so being issued an access token does not guarantee access to the subsequent API
- Whilst the services are inter-changeable, you **cannot** mix and match, e.g.
  - You can't start an authorisation code flow on one endpoint and call the /token on the other
  - If you need to call the /userinfo endpoint, you can't use an access token from one service to obtain userinfo from the other
  - Warning signs - if the **calling** application includes entering URLs to both endpoints, then there is a design issue

Benefits of using oauth-mock service:

- Provides a way to have one less environment, since there is a significant rise in agility if smart cards don't have to be issued to every developer / tester
- NHS CIS2 requires, for DEV and INT that individual user accounts are created for every developer / tester - slowing down agility, and self-service approach
- Code isolation - this is an entirely separate proxy and the configuration does not interact with the Identity Service itself, removing entirely, any risks of unauthorised access code "leaking" into Production

There is a large disadvantage that the mock service can / will get out of sync with the backend services such as NHS CIS2.

For history behind the decision see 20200520 - WDR Meeting Minutes[739], and APM-844[740] - NHS ID sandpit closing - analysis / spike DONE.

7.14.7.6 Performance testing

7.14.7.6.1 Overview

Before you go live, you need to be sure that your API can:

- handle the expected volumes (aka **throughput**)
- respond quickly enough (aka **response time**)

To do this, you'll need to do **performance testing**.

You'll be asked to provide evidence of performance testing as part of the service transition process - see Service transition for APIs (see page 947).

7.14.7.6.2 Private beta - low usage option

If you are starting with a private beta, you can avoid (or delay) doing performance testing by:

- using the default low rate limit (e.g. 10tps) to your API
- stating clearly in your API specification that
    - you're in a limited beta, meaning that...
    - the API has a low rate limit, shared across all private beta participants
    - you cannot guarantee response times

To find out how to set rate limits on your API see Rate limiting shared flow (see page 1039).

7.14.7.6.3 How to test

Your API consists of multiple components:

- the API proxy
- the API back-end

739 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114910919/20200520+-+WDR+Meeting+Minutes
740 https://nhsd-jira.digital.nhs.uk/browse/APM-844?src=confmacro

- possibly, some bits in the middle

Your API proxy **does** need performance testing - although it's based on an out-of-the-box product (Apigee), it is possible to configure your proxy with rules that don't perform well.

We recommend you perform a **full front-to-back performance test of your API in the INT environment**, testing the API proxy, the back-end, and all the bits in the middle together. For smaller services you can incorporate your testing into your pipelines.

Where your back-end receives traffic from multiple sources (Spine is a good example of this), you should aim to include appropriate load from other sources as part of the test.

If this is going to be tricky, an alternative would be to performance test your components separately, using stubs to stub out back-end behaviour.

7.14.7.6.3.1 Ramping up for busier services

Apigee is a very high performance platform, with (generally) single digit millisecond processing time, and you don't need to involve APIM in detailed performance discussions and testing. Simply carry out your testing by changing your rate limits in the INT environment.

However, if your service could have bursts of over **500 TPS** (Transactions / requests per second), then APIM should be aware and you need to consider:

- Asking for access to our REF environment for performance testing
- User-restricted APIs will be using the APIM Authentication Service
- You don't need to continually run performance tests with APIM involved, but any changes in authentication or patterns of API calling should be tested
- Providing APIM access to your performance test results so that in live service situations, APIM can demonstrate or reference these results

APIM Hosted Containers are standard AWS scalable services, but "out of the box" are a much lower TPS. Consider that your Docker solution will not scale as well as Apigee (it's there to do more complex processing).

7.14.7.6.4 Environments

7.14.7.6.4.1 API proxy

To performance test your API proxy, use the INT environment, or possibly the APIM REF environment. The standard deployment pipeline deploys (for your API) do this environment by default, although you will need to configure it to point to an appropriate environment for performance testing your API back-end.

7.14.7.6.4.2 Spine API back-ends

To performance test a Spine API back-end, use the Spine REF environment. You'll need to speak to the Core Spine Delivery Lead (currently Mark Burton) to request a slot.

7.14.7.6.5 Tools

Our current preferred performance test tool is:

- Locust - open source python based

Other tools in use in NHSD which could be considered for use, if there's a strong enough reason - but not supported by APIM

- JMeter - can be steep learning curve.
- Gatling - requires Scala knowledge

7.14.7.6.6 Developer-facing path-to-live environments

We do not currently provide a performance test environment to external developers wanting to performance test their apps. Our advice to them is to stub out our APIs for their performance testing. This approach might change in the future if we think we can "make integration easier" by providing an externally-facing performance test environment.

Consequently, developer-facing PTL environments (sandbox, integration test) are not intended to be subjected to high volumes of traffic.

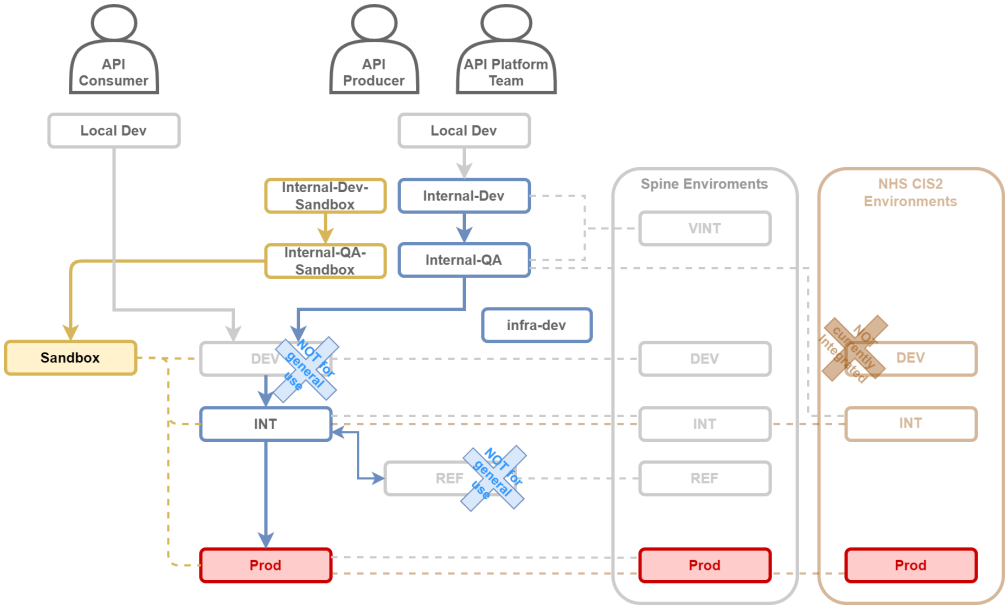
Therefore:

- we recommend setting a low rate limit (e.g. 5tps) in your PTL environments, as explained in Rate limiting shared flow (see page 1039)
- if you do this, you do not need to performance test your sandbox

7.14.7.7 Environments

See Architecture[741] for more details:

741 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/97747483/Architecture#Architecture-environments



7.14.7.7.1 Apigee Environments

	internal-dev	internal-dev-sandbox	internal-qa	internal-qa-sandbox	ref	dev	int	sandbox	prod
ApigeeOrg	nhsd-nonprod	nhsd-nonprod	nhsd-nonprod	nhsd-nonprod	nhsd-nonprod	nhsd-prod	nhsd-prod	nhsd-prod	nhsd-prod

EnvironmentProxyType	live	sandbox	live	sandbox	live	live	live	sandbox	live
Purpose	Development and manual testing. Developers and testers may do whatever they want.	Development and manual testing. Developers and testers may do whatever they want.	Clean environment for testing deploys.	Clean environment for testing deploys.	Reference environment for non-functional testing.	Extra environment for external developers where there is a heavier integration process.	Main integration environment for external developers.	Externally accessible sandboxes.	Live services.
Cleanliness	dirty	dirty	clean	clean	clean	clean	clean	clean	clean

Pull Request Deployments?	yes	yes	no	no	no	no	no	no	no

Access	• Producer Developers • Platform Developers • Nonprod Admins	• Producer Developers • Platform Developers • Nonprod Admins	• Platform Developers • NonProd Admins	• Platform Developers • NonProd Admins	• Platform Developers • NonProd Admins	• Prod Admins	• Prod Admins	• Prod Admins	• Prod Admins

Suggested Test Load	• Pull request smoke tests • Pull request full tests • Smoke tests • Full tests	• Pull request smoke tests • Pull request full tests • Smoke tests • Full tests	• Smoke tests • Full tests	• Smoke tests • Sandbox tests	• Smoke tests	• Smoke tests	• Smoke tests	• Smoke tests	• Smoke tests

DeployDependency	n/a	n/a			• internal-qa • internal-qa-sandbox	• internal-qa • internal-qa-sandbox	• internal-qa • internal-qa-sandbox	• internal-qa • internal-qa-sandbox	• internal-qa • internal-qa-sandbox • manual approval
DeploymentFrequency	On demand (PRs and Releases)	On demand (PRs and Releases)	On demand (Releases only)	On demand (Releases only)	On demand (Releases only)	On demand (Releases only)	On demand (Releases only)	On demand (Releases only)	Requires Manual Approval from On-call engineer
SpineEnvironment	VEIT07	VEIT07	VEIT07	N/A	Ref	Dev	Int	N/A	Prod

ApigeeASID	200000001115	200000001115	200000001115	N/A	200000001215	200000000421	200000001215	N/A	200000022885
SpineASID	200000006422	200000006422	200000006422	N/A	200000006422	200000006422	200000006422	N/A	
NHSCIS2Environment	APIM mock	APIM mock	APIM mock+ CIS2 INT	APIM mock	APIM mock	APIM mock	APIM mock+ CIS2 INT	APIM mock	CIS2 PROD
NHSlogin	APIM mock	APIM mock	APIM mock + AOS	APIM mock	APIM mock	APIM mock	APIM mock+ AOS	APIM mock	NHS login PROD

Domain	internal-dev.api.service.nhs.uk[742]	internal-dev-sandbox.api.service.nhs.uk[743]	internal-qa.api.service.nhs.uk[744]	internal-qa-sandbox.api.service.nhs.uk[745]	ref.api.service.nhs.uk[746]	dev.api.service.nhs.uk[747]	int.api.service.nhs.uk[748]	sandbox.api.service.nhs.uk[749]	api.service.nhs.uk[750]
DOS mapping	Development	Sandbox	Integration test	N/A	Production	Development	Integration test	Sandbox	Production

742 http://internal-dev.api.service.nhs.uk
743 http://internal-dev-sandbox.api.service.nhs.uk
744 http://internal-qa.api.service.nhs.uk
745 http://internal-qa-sandbox.api.service.nhs.uk
746 http://ref.api.service.nhs.uk
747 http://dev.api.service.nhs.uk
748 http://int.api.service.nhs.uk
749 http://sandbox.api.service.nhs.uk
750 http://api.service.nhs.uk

Developer Account / Environment Access	https://dos-internal.ptl.api.platform.nhs.uk/	https://dos-internal.ptl.api.platform.nhs.uk/	https://dos-internal.ptl.api.platform.nhs.uk/	https://dos-internal.ptl.api.platform.nhs.uk/	https://dos-internal.ptl.api.platform.nhs.uk/	https://onboarding.prod.api.platform.nhs.uk/	https://onboarding.prod.api.platform.nhs.uk/	https://onboarding.prod.api.platform.nhs.uk/	https://onboarding.prod.api.platform.nhs.uk/

NHS CIS2 and APIM - the only environments that connect to NHS CIS2 are internal-qa, INT and PROD. All the other environments have a dummy OAuth Identity Service to issue an Apigee Access Token. This services does not require any user validation, but does require a callback endpoint. To speed up testing, and to help API Producers get started, there is a test app provided with a single login button that then takes the user to a page with an access token on. See the table below for the App URLs, or to see more about how INT manages two Identity Services see INT environment and multiple identity services (see page 755). The following page may also help to find the appropriate URLs: https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation#environments-and-testing (NOTE: for real CIS2 INT in the internal-qa environment the URL is https://internal-qa.api.service.nhs.uk/oauth2-int/)

Spine environment (**): this isn't a fixed relationship - consider it the **preferred** environment. The Spine Environment listed operates in the following ways:

- Each proxy (API Producer) can **choose** which Spine environment to connect to through the setting of the target server
- The "Spine ASID" is the **default** ASID applied to Apigee Applications that don't have an ASID specified, this is associated with the specified Spine environment.
    - For API Producers not using the "preferred" environment for their proxy - then it is likely that testing cannot use the default ASID (but there is some inter-environment ASID sharing)
    - If there are complications with the ASID permissions contact Producer Support to help

- NHS CIS2 environment is indirectly linked to the Spine environment. A user for CIS2 DEV and INT are only created when you request a user for those Spine environments. This might cause some testing complications
- API Producers can ignore the "Apigee ASID", this is the ASID associated with the Party Key created for the TLS MA Certificates used to secure connections. This isn't transmitted *(tbc - Aubyn to re-check at some point to confirm this is the actual case)*

It has been agreed with the Spine team that API Management will **not** use Spine REF. Spine REF is focused on Spine non-functional testing, and it is unlikely that APIM will require the use of Spine REF.

7.14.7.7.2 Spine endpoints and target servers

Spine endpoint domain names are as follows (if you have access this is the full page - IGWAYv3 Service Names[751]):

- VEIT07 - veit07.devspineservices.nhs.uk[752]
- INT - msg.intspineservices.nhs.uk[753]
- Production:
    - demographics.spineservices.nhs.uk[754]
    - clinicals.spineservices.nhs.uk[755]
    - prescriptions.spineservices.nhs.uk[756]

Target Servers (replacing the ig3 ones) are now fully configured with the appropriate TLS-MA Certificates in Apigee. They have the following names:

- `spine-demographics`
- `spine-prescriptions`
- This is the same for all environments, EXCEPT INT:
    - `spine-demographics-int`
    - `spine-prescriptions-int`

751 https://nhsd-confluence.digital.nhs.uk/spaces/SPINE/pages/102650651/IGWAYv3+Service+Names
752 http://veit07.devspineservices.nhs.uk/
753 http://msg.intspineservices.nhs.uk/
754 http://demographics.spineservices.nhs.uk/
755 http://clinicals.spineservices.nhs.uk/
756 http://prescriptions.spineservices.nhs.uk/

7.14.7.7.2.1 Terminology / glossary

Apigee Org	Apigee organisation in which this environment lives.
Environment Proxy Type	The type of proxy to deploy for a service.

Cleanliness	One of:
	• Dirty
	  • Manual changes are allowed
	• Clean
	  • Automated IaC and deploy changes only

Pull Request Deployments?	Whether pull requests can be deployed on this environment for development and testing.
Access	Who is allowed to modify this environment.
Test Load	Which test suites are run during deploys for each service.
Deploy Dependency	Environments, if any, that must succeed deployment before this environment is allowed to deploy.

7.14.7.7.3 Other configuration details

7.14.7.7.3.1 Static IPs - Apigee to API Backends

The IP address from Apigee to the backend APIs ("southbound" traffic) will often need to be source IP restricted (whitelisted) and to support that Google provide a pair of static IPs per Org:

Org & Env	Static IPs
Prod Org - Prod Environment	34.89.0.111 34.89.69.6

Org & Env	Static IPs
All other combinations	35.197.254.55 35.246.55.143

Because of the traffic isolation pack the prod environment of nhsd-prod organisation has a dedicated Apigee Message Processor pod, all the remaining environments share an Apigee Message Processor pod.

7.14.7.7.3.2 Dynamic IPs - API Consumer to Apigee

There are no static IPs or lists of IPs available for the Apigee runtime public endpoints ("northbound" traffic). Apigee utilises the GCP global network to route traffic. There are, currently, no hard requirements for static IPs going into the NHS Digital APIs.

See Architecture (Environments)[757] for a visualisation of this.

7.14.7.7.4 Testing

For testing see - Testing with mock auth (see page 737)

7.14.7.7.4.1 DEP environment

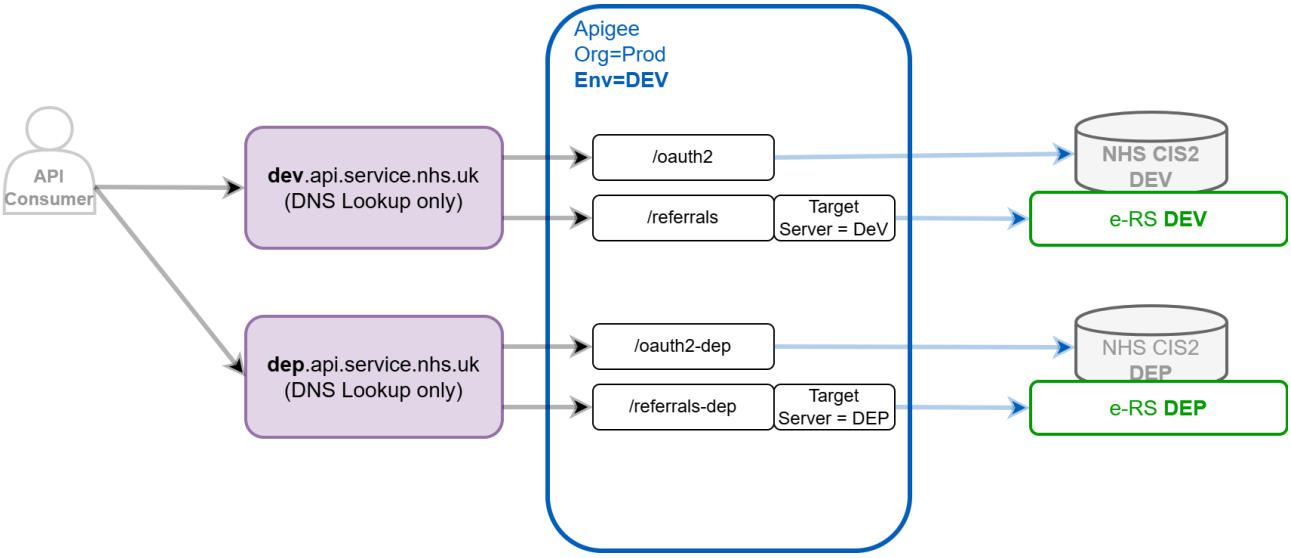
The DEP environment is an alias of the Dev environment. This was created to reflect the Spine Deployment environment on request for the Spine Directory Service API:

- https://dep.api.service.nhs.uk/spine-directory/FHIR/R4/_ping

A DEP version of the identity-service is also now available, which connects to the real Deployment backend for CIS2. This is to support external suppliers train users, e.g. Dedalus.

- https://dep.api.service.nhs.uk/oauth2-dep

757 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/580651965/Architecture+Environments



See Deploy an instance into DEP using Proxygen (see page 713) page for details on how to deploy other APIs to DEP prior to the APIM work to deprecate DEV and fully enable DEP. The agreement for this is documented here: https://digital.nhs.uk/services/path-to-live-environments

7.14.7.7.4.2 NHS login AOS (INT) environment usage

APIM has been issued a Client ID in AOS so that we can do some testing, key details:

Client ID	APIM-1
OIDC Config	https://auth.aos.signin.nhs.uk/.well-known/openid-configuration
User interface	https://settings.aos.signin.nhs.uk/ (i.e. manage your account)
Test Users	Spine PDS Test Data Pack (Integration Environment).xlsx[758]
Redirect URLs	"https://nhsd-apim-testing-int-ns.herokuapp.com/nhslogin/callback", "https://int.api.service.nhs.uk/oauth2/callback", "http://localhost:5000/callback"

758 https://nhsd-confluence.digital.nhs.uk/download/attachments/110873804/Spine%20PDS%20Test%20Data%20Pack%20%28Integration%20Environment%29.xlsx?api=v2&modificationDate=1671102144552&version=1

Private key	NOT SURE but these might the private key: ptl/identity-service/auth.aos.signin.nhs.uk/private_key[759] ptl/truststore/aos.signin.nhs.uk/crt[760]

7.14.7.7.5 Environments and authentication testing

Confirm model is:

- Gain Access Token - by a range of means
- Use Access Token

Problem:

- Automation of positive / happy path tests
- Automation where access is denied can be part automated since it relies on the authentication failing

Existing elements:

- "No Smartcard" used in INT
- In internal dev:
  - On calling / token the resulting ID Token is published by us, signed with an internal private key. This is created with an issuer of CIS2 for INT
  - If we create our own OIDC providers

7.14.7.7.5.1 Production

	Type	Issuer	Positive Testing	Negative
NHS CIS2	Comb		Manual - Smartcard	Automated
	Sep			Automated
NHS login	Sep		?	Automated

759 https://eu-west-2.console.aws.amazon.com/secretsmanager/secret?name=ptl%2Fidentity-service%2Fauth.aos.signin.nhs.uk%2Fprivate_key&region=eu-west-2
760 https://eu-west-2.console.aws.amazon.com/secretsmanager/secret?name=ptl%2Ftruststore%2Faos.signin.nhs.uk%2Fcrt&region=eu-west-2

	Type	Issuer	Positive Testing	Negative
App Restricted				

7.14.7.7.5.2 INT

	Type	Issuer	Positive Testing	Negative
NHS CIS2	Comb		Manual - Smartcard	Automated
	Sep			Automated
NHS login	Sep		?	Automated
App Restricted				
No Smartcard	NHS CIS2			
	NHS login			

7.14.7.7.5.3 Internal-qa

	Type	Issuer	Positive Testing	Negative
NHS CIS2	Might become for e-RS			
	Not avail			
NHS login	Not avail			
App Restricted				
No Smartcard	NHS CIS2			
	NHS login			

7.14.7.7.5.4 Internal-dev

	Type	Issuer	Positive Testing	Negative
NHS CIS2	Not avail			
	Not avail			
NHS login	Not avail			
App Restricted				
No Smartcard	NHS CIS2			
	NHS login			

7.14.8 Securing your API

7.14.8.1 Overview

Broken, exposed, or hacked APIs are behind significant data breaches. They can reveal sensitive medical, financial, and personal data for public consumption. That said, not all data is the same nor should be protected in the same way. How you approach API security will depend on what kind of data is being transferred.

Although in the early stages of the alpha delivery phase you are likely to not implement a fully secured API, you will be thinking of our standards for security for your API. During alpha, you are likely to cover:

- How sensitive is access to my API / service?
- What is OAuth, and how does it work?
- What are the options for API consumers to connect securely?
- Once authenticated, what further authorisation might be applied?
- How do I secure my backend?
- How to manage Dependabot alerts (see page 788)?

When you move into Private Beta you move from design into implementing:

- Configuring the proxy to apply the different access modes
- Working out and applying authorisation
- Make a secure connection to the target server
- Consider wider security issues such as OWASP top 10 API security vulnerabilities (see page 788)

The sooner you prioritize security, the smoother the journey for your API between entering Alpha and exiting Beta to become a live stable API.

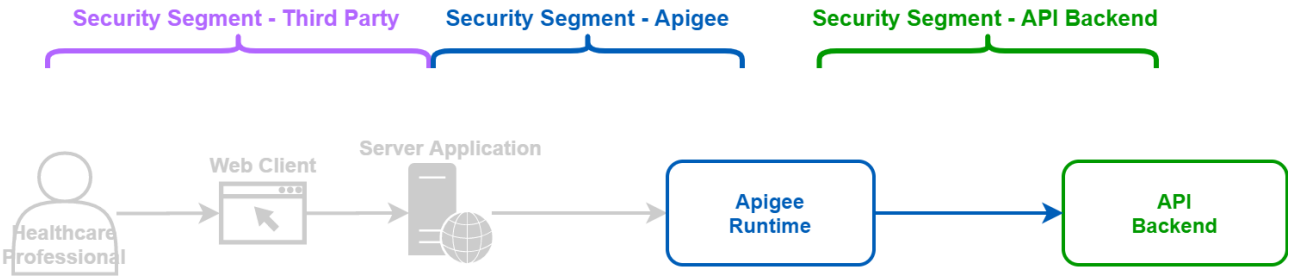
The following section describes the various security implementations and patterns for you to secure your API and API backend.

> **Securing your API with proxy generator**
>
> If using proxy generator see Securing your API with proxy generator (see page 856)

7.14.8.1.1 Security segments

The primary design pattern being followed is the façade pattern, which can also be thought of as a firebreak between the API Consumer and API backend. This is really important to apply this pattern to how you manage security. Your security must be split up into segments, and each one must be loosely coupled to each other. For instance, an access token (itself) should not be used for both Apigee and for the backend.

The security segmentation for most solutions will look like this:



It is important to understand the two key models from the perspective of an API Consumer:

- Application Restricted
- User Restricted

See the consumer documentation: Security patterns for API consumer access (see page 823) or https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation

7.14.8.1.2 How security is applied in a proxy

Each request that comes into Apigee, and into a proxy must be checked that it is authorised (excluding open access). The API Producer **must configure** their proxy to either:

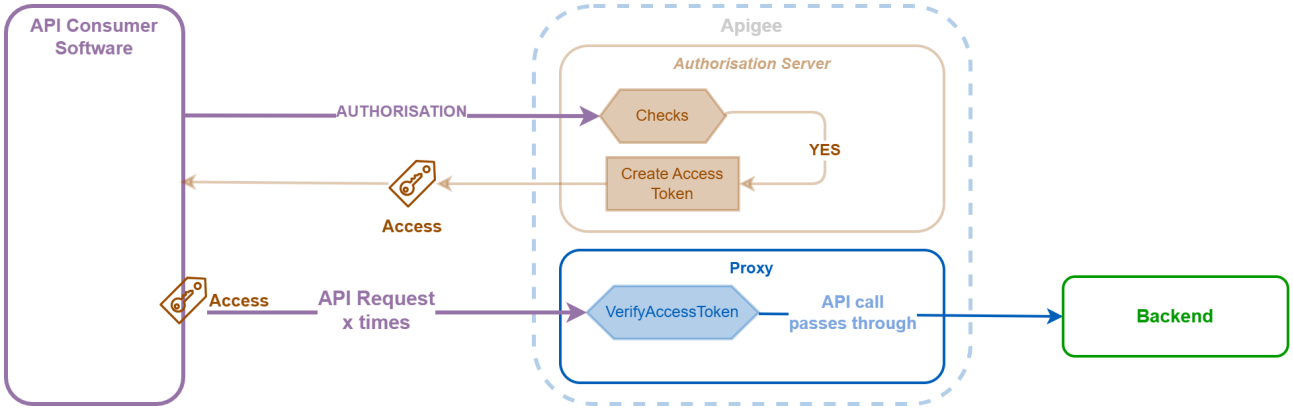
- Validate the **API Key** (using VerifyAPIKey policy)
- Validate the **OAuth Access token** (using VerifyAccessToken policy)

For specific details on how to actually use these in your proxy see: Authentication policies and proxy flows (see page 835)

7.14.8.1.2.1 Securing with OAuth for application and user restricted APIs

This is the **primary** pattern across the platform, and it is a two step process. This two step process allows the platform to offer multiple different OAuth mechanisms, but use the standard Apigee access tokens across all OAuth secure requests.

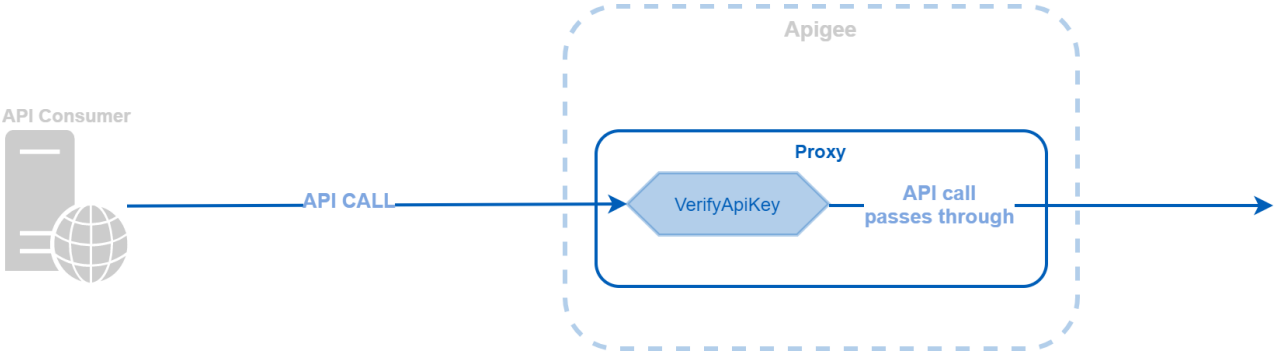
This two stage process/ concept is:



> It is really important to remember that your proxy is disconnected from the method of authorisation.
> Your API should work equally well with the different mechanisms - and should treat combined and separate auth API Consumers identically

7.14.8.1.2.2 Application restricted for lower sensitivity or open APIS - using an API Key

This involves no pre-authorisation, it is simply an HTTP request with the header **apikey**, and the key the developer has set up:



You **must** configure your API Proxy to accept all OAuth access tokens. Many of the lower security APIs will be used in conjunction with more sensitive APIs. You should **not** force API Consumers to switch between OAuth and API Key requests.

7.14.8.2 Wider security approach

This page, and sub-pages, focus on the detailed security measures / aspects that the technical team will implement. However, it is important to consider the wider security picture for API Consumers who are onboarding onto your service. To make this easier for them, and for API Producers to manage this page delves into this:

- Access modes - compliance - security and simplifying running your API (see page 299)

7.14.8.3 Security patterns and implementations

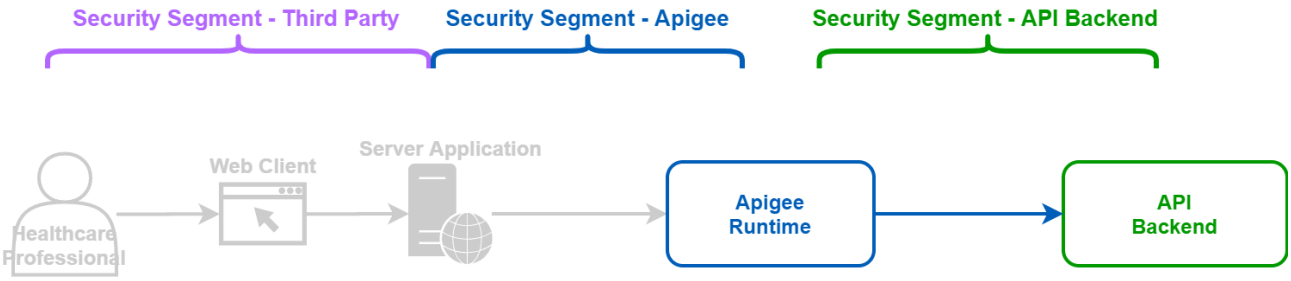
For information on Penetration Testing see: Penetration testing APIs (see page 966)

- Security for API Backend (see page 778)
  - Backend Mutual TLS and DNS (see page 778)
  - Cross-origin resource sharing (CORS) (see page 786)
  - OWASP top 10 API security vulnerabilities (see page 788)
  - Using Dependabot on GitHub Repos (see page 788)
- Security for Third-Party Applications (see page 789)
  - Combined and separate auth differences (see page 790)
  - End to end AWS web application pattern (see page 793)
  - Security information for consumers (see page 794)
- Security for Apigee (see page 795)
  - Accessing the id token claims for smartcards (see page 796)
  - Calling the /userinfo endpoint from AWS hosted containers (see page 798)
  - Client authentication - Private key JWT (see page 800)

- Configuring extended attributes for API Consumer Apps (see page 806)
- HTTP Strict Transport Security (HSTS) (see page 809)
- Identity service token variables (see page 810)
- Managing different security requirements for API Consumers (see page 813)
- OAuth error conditions (see page 816)
- Security patterns for API consumer access (see page 823)
- Validate user roles using shared flows (see page 844)

7.14.8.4 Security for API Backend

Within this section we will be covering security for the API Backend. The Green section in the diagram below.



This area of security will cover the connections made between Apigee and any API backend. Not all APIs are authorised (some do not require any auth!) in the same way, so please read this section carefully as it does not apply to all APIs.

This section will include how to generate keys and CSRs. It also includes how to make use of cross-origin resource sharing and highlights the OWASP top 10 API security vulnerabilities.

- Backend Mutual TLS and DNS (see page 778)
  - Generating keys and CSRs (see page 780)
- Cross-origin resource sharing (CORS) (see page 786)
  - Domains and CORS (see page 787)
- OWASP top 10 API security vulnerabilities (see page 788)
- Using Dependabot on GitHub Repos (see page 788)

7.14.8.4.1 Backend Mutual TLS and DNS

7.14.8.4.1.1 Overview

The API Producer backend service needs to be secured at a similar level to the frontend. For instance an API key secure API can have the backend secured with an API key (not the same one though). Most APIM APIs that expose patient data, will be secured with:

- TLS MA (Mutual Auth)
- Source IP restrictions (although ideally not as it makes the infrastructure more fragile to change)

Adding support for OAuth 2 to connect to your API backend, is a future item on the roadmap.

7.14.8.4.1.2 DNS

As the backend API Provider you could purchase any domain to host your backend on, since API Management (Apigee) is the front door for your API - your backend domain (DNS entries) would never need to be shared internally or externally.

> Where the domain is not going to be shared, the API Producer team should simply use the default DNS names from AWS, Azure or similar.

There are some situations (e.g. a web site) where a custom domain (not auto generated by AWS) is needed, some major benefits for domains in Production:

- Trust of the web site user to see a recognised domain
- The management of nhs.uk[761] is designed to stop malicious entities from taking control or other such attacks

The pattern that NHS Digital development teams have followed previously is to place a service as a sub-domain under **nhs.uk.** To enable team self-management, especially when there are a large number of environments, existing teams have been delegated control of a specific sub-domain.

> The "off-infrastructure delegations" is currently not allowed see the policy on the NHS Digital[762]website. The interim agreement with the DNS Team is:
>
> - A new policy needs to be researched and agreed
> - In the interim, a Lead Technical Architect (TA) needs to be involved to agree exceptions
> - The dev team, if agreed, will be issued a sub-domain under national.nhs.uk[763]
>
> Contact email: dnsteam@nhs.net[764]

761 http://nhs.uk

762 https://digital.nhs.uk/services/networking-addressing/namespace-policy#website-policy-for-the-nhs-uk-domain-www-name-nhs-uk-

763 http://national.nhs.uk

764 mailto:dnsteam@nhs.net

7.14.8.4.1.3 Certificates

Issuing of certificates for nhs.uk is (relatively) straight forward:

- Public facing domains certificates are issued by a Public CA (currently DigiCert), and is done via a ResolveIT[765] ticket with "Infrastructure services" as the affected service
- For "internal" domains (which could include an internet based endpoint), these use the Spine PKI and are supported by the DIR team (dir@nhs.net[766])

7.14.8.4.1.4 TLS MA security

> Support for TLS MA is not consistent across technology platforms, and often very badly documented.
> In an ideal world AWS would support TLS MA on any of their auto generated domains, but the API Gateway service (both REST and HTTP) **require the** use of custom domains. (made clear in their docs on API Gateway[767]). In these situations, this would be a justification for off-infrastructure delegations.

Where an API Producer backend is secured with TLS Mutual Auth, a client authentication certificate needs to be created and installed.

The existing pattern for situations requiring mutual TLS client authentication is that the client (calling) certificate is based on the Spine PKI - and so a CSR (Certificate Signing Request) is sent to the DIR team, against an (often) virtual FQDN (i.e. it is never added to DNS) and they issue the client authentication certificate.

The client authentication certificate does **not** have to be related to the domain / FQDN / certificate that it is connecting to.

Other pages covering this:

- Setting up a target server (see page 607)
- KOP-045 Creation of client certificates for Apigee[768]
- Routing requests to multiple backends (see page 280)

7.14.8.4.1.5 Generating keys and CSRs

Why and when

765 https://resolveit.digital.nhs.uk/
766 mailto:dir@nhs.net
767 https://docs.aws.amazon.com/apigateway/latest/developerguide/rest-api-mutual-tls.html
768 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/328960245/KOP-045+Creation+of+client+certificates+for+Apigee

Why?

Because your TLS certificates have an expiry date

The following are the connections being protected by TLS:

- API Consumer connecting to Apigee Proxies (examples below)
- Connections being protected with TLS MA:
    - Apigee to Spine
    - APIM AWS Hosted Targets to Spine
    - Apigee to AWS Hosted Targets

When?

Before the certs expire, ideally AT LEAST 30 days before their expiry date.

Prerequisites

- Linux-ish env
- OpenSSL command-line tool installed

Easy mode

https://github.com/NHSDigital/api-management-cert-generation

Detailed mode

Create a config file

First up, create an OpenSSL config file. It should look something like this:

```
FQDN = internal-dev.api.service.nhs.uk

[req]
default_bits = 4096
distinguished_name = req_distinguished_name
req_extensions = v3_req
prompt = no

[req_distinguished_name]
C = GB
ST = West Yorkshire
L = Leeds
O = NHS Digital
OU = API Management
CN = internal-dev.api.service.nhs.uk
```

```
emailAddress = api.management@nhs.net

[v3_req]
keyUsage = keyEncipherment, dataEncipherment
extendedKeyUsage = serverAuth
subjectAltName = @alt_names

[alt_names]
DNS.1 = internal-dev.api.service.nhs.uk
DNS.2 = internal-qa-sandbox.api.service.nhs.uk
DNS.3 = internal-qa.api.service.nhs.uk
DNS.4 = internal-portal.developer.nhs.uk
```

Integration with Spine using a TLS MA Certificate

IF a TLS MA Certificate is required, APIM needs to generate a CSR to be passed onto Spine. The Config file **must** be minimal; otherwise, EPS (End Point registration Service) will generate a certificate that won't be right. (i.e. no Common Name in the Subject).

Conf file used previously:

Spine TLS MA Cert

```
[ req ]
default_bits        = 4096
distinguished_name  = req_distinguished_name
req_extensions          = v3_req
prompt = no

[ req_distinguished_name ]
commonName=api.intspineservices.nhs.uk

[ v3_req ]
keyUsage = keyEncipherment, dataEncipherment
extendedKeyUsage = serverAuth
```

Use the same key and CSR generation syntax below.

You can parameterise OpenSSL command to set the CN (commonName)... Could not find the syntax for the .conf file to use the same conf file for multiple FQDNs.

Generate a key and CSR from the config

This code assumes that you have dirs named 'keys' and 'CSRs':

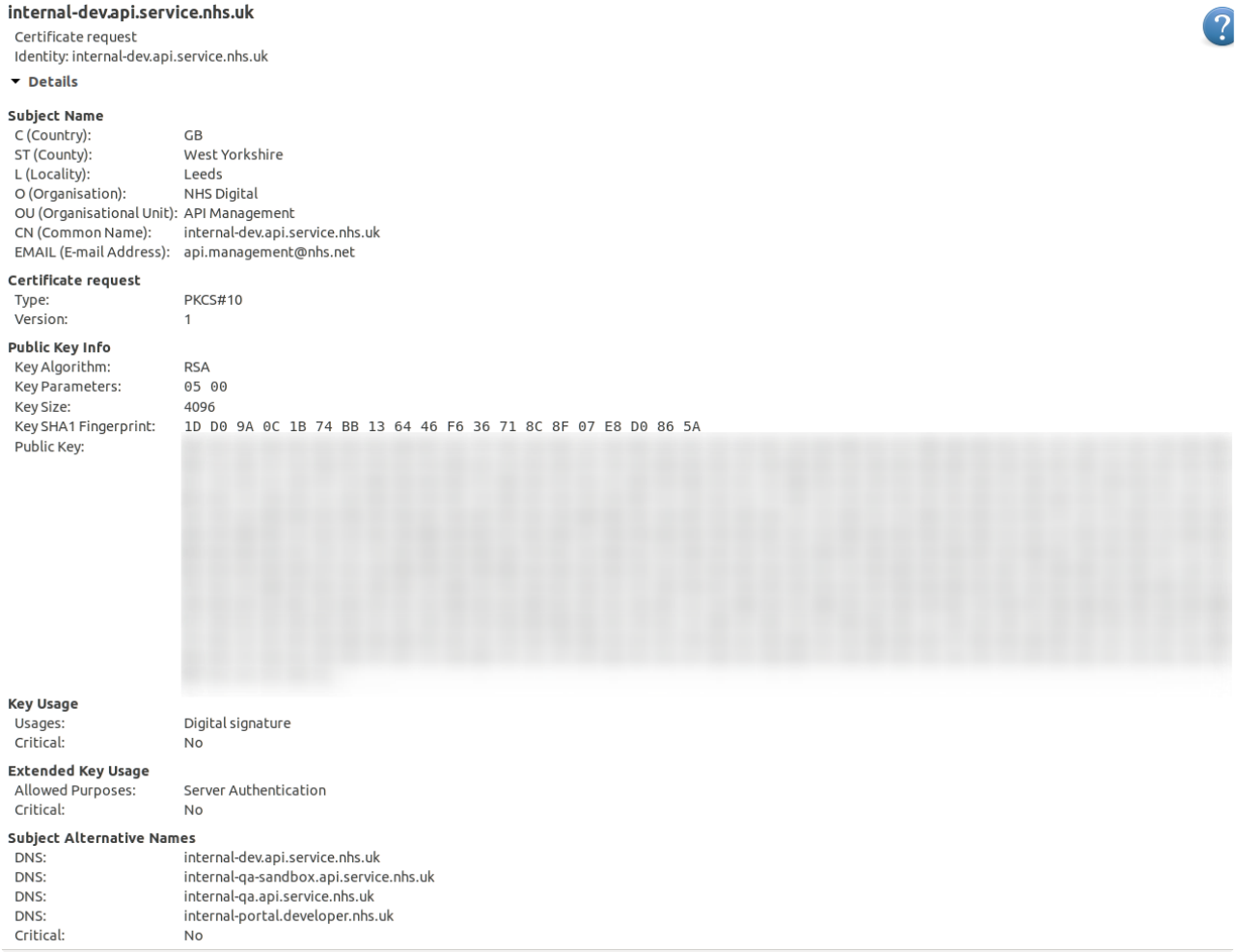
```
$ mkdir keys
$ mkdir csrs
$ openssl req -new -newkey rsa:4096 -nodes -sha256 -keyout keys/internal-
dev.api.service.nhs.uk.key -out csrs/internal-dev.api.service.nhs.uk.csr -config
confs/internal.conf -extensions v3_req
```

This will dump a CSR in 'csrs' and a key in 'keys'.

Verify

If your system supports it, double click on .key and .csr files to ensure that alt names got generated corrected and everything is as you expect.

A healthy CSR based on the config above might look like this:



You can clearly see the alternative names are set, the algorithm is as expected, the usages are correct, and everything in the config has ended up on the CSR.

Or use OpenSSL syntax:

```
openssl req -text -noout -verify -in .\csrs\api.intspineservices.nhs.uk.csr
```

Requirements

For Spine, their certificate requirements are:

- SHA-2...

Getting the CSRs signed

To get the CSRs signed, you'll need to raise a ResolveIT with the subject: 'FAO Infrastructure Team: API Management Cert Signing Request'.

Add a zip of any CSRs to the ticket.

You may need to ask the DNS team to put verification strings in DNS records for you to make this work.

Once the CSRs are done, you will be emailed a link to download your cert.

Download the cert as a single .pem file and put it somewhere safe!

AWS secrets manager

All the private keys and certificates must be stored in AWS Secrets Manager, with no copies residing on local (encrypted) disks beyond creation.

The path in AWS for these will follow this pattern:

- (ptl|prod)/{cert type decriptor}/{cn}/(crt|key|csr)
- See Spine and IGWAY3 security[769] for more details

When creating a new Secret, the following options should be configured:

Setting	Value
Secret Type	Other
Encryption Key	Default

769 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141201006/Spine+and+IGWAY3+security

Setting	Value
Key / Value or Plaintext	Select Plaintext and replace the default contents with the text from the respective file
Tags	Matt Mercer - I noticed you added "AllowedServices1"?
Resource Permissions	None
Configure Auto Rotation	Disable

TLS cert subject alternate names

Cert CN	SANs
api.service.nhs.uk[770]	portal.developer.nhs.uk[771]
int.api.service.nhs.uk[772]	n/a
dev.api.service.nhs.uk[773]	ref.api.service.nhs.uk[774] sandbox.api.service.nhs.uk[775]
internal-dev.api.service.nhs.uk[776]	internal-portal.developer.nhs.uk[777] internal-qa.api.service.nhs.uk[778] internal-qa-sandbox.api.service.nhs.uk[779]

770 http://api.service.nhs.uk
771 http://portal.developer.nhs.uk
772 http://int.api.service.nhs.uk
773 http://dev.api.service.nhs.uk
774 http://ref.api.service.nhs.uk
775 http://sandbox.api.service.nhs.uk
776 http://internal-dev.api.service.nhs.uk
777 http://internal-portal.developer.nhs.uk
778 http://internal-qa.api.service.nhs.uk
779 http://internal-qa-sandbox.api.service.nhs.uk

7.14.8.4.2 Cross-origin resource sharing (CORS)

7.14.8.4.2.1 Cross-origin setup in proxies

The template sandbox proxy (which yours was likely generated from) has a built-in CORS policy which will add CORS headers to any response.

The file in question can be found here: https://github.com/NHSDigital/api-management-service-template/blob/master/proxies/sandbox/apiproxy/policies/AssignMessage.AddCors.xml

Adding new allowed headers

If you need to add headers, edit the `AssignMessage.AddCors.xml` policy in your repo.

Add any new headers in the "Access-Control-Allow-Headers" value, as shown below.

Wildcards are **not** supported in `Access-Control-Allow-Headers` when Authorization or cookie auth is being used!
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Access-Control-Allow-Headers

Exposing response headers in Try this API

In order for response headers (custom or not) to be revealed in responses displayed in Try this API[780], they need to be explicitly enumerated in response Access-Control-Expose-Headers[781].

Example

```
<AssignMessage async="false" continueOnError="false" enabled="true" name="AssignMessa
ge.AddCors">
    <FaultRules/>
    <Properties/>
    <Set>
        <Headers>
```

780 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir.trythisapi#/default/update-patient-partial
781 https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS#access-control-expose-headers

```
            <Header name="Access-Control-Allow-Origin">{request.header.origin}</
Header>
            <Header name="Access-Control-Allow-Headers">origin, x-requested-with,
accept, content-type, nhsd-session-urid, x-my-new-header</Header>
            <Header name="Access-Control-Expose-Headers">origin, x-requested-with,
accept, content-type, nhsd-session-urid, x-my-new-header</Header>
            <Header name="Access-Control-Max-Age">3628800</Header>
            <Header name="Access-Control-Allow-Methods">GET, PUT, POST, DELETE</
Header>
        </Headers>
    </Set>
    <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
    <AssignTo createNew="false" transport="http" type="response"/>
</AssignMessage>
```

7.14.8.4.2.2 Domains and CORS

The production API management domain is: api.service.nhs.uk

Configuring Apigee to work with CORS

See the Apigee Documentation on Adding CORS support to an API proxy[782]

For an example implementation see the Personal Demographics API[783]

Application server configuration hints

Whilst, as an API Producer, you might not need to consume your own API.

TODO

Client browser configuration hints

There is good support for CORS in browsers now, but you might want to consider:

TODO

Why do we only support one domain

Apigee can support multiple domains (virtual hosts inside of Apigee), however it was strategically decided to go with a single domain for the following reasons:

- Apigee best practice is to keep everything in the same host
- Part of the "keep integration" simple pattern we have chosen is to keep it under one domain:
    - There is no confusion about which domain is in use
    - Keeps configuration simpler

782 https://docs.apigee.com/api-platform/develop/adding-cors-support-api-proxy
783 https://github.com/NHSDigital/personal-demographics-service-api

- Reduces platform management overheads as we don't have to manage multiple API Consumer Domain Certificates
- Management of multiple domain names is much more complex and then impacts:
    - DR situations, at the moment we have one domain name to update
    - Future high capacity routing or DDoS protection - if NHS D puts in an extra layer of protection (e.g. Akami) it is significantly to simpler to do
- CORS support is now very strong, and developers everywhere are being forced to use this
- The current CI/CD Pipelines are geared up for this deployment pattern
- Not many of our API Consumers will be browser based, but system to system calls - reducing the number of businesses using this.

7.14.8.4.3 OWASP top 10 API security vulnerabilities

The API vulnerabilities are less technology driven, and so are more subtle and less detectable in a pen test. Current list is here: https://owasp.org/www-project-api-security/

There are many attacks going on, and example of a 70TB of stolen data from Facebook / Parler: https://apisecurity.io/issue-116-facebook-parler-api-vulnerabilities-clairvoyance/

TODO 🙂

7.14.8.4.3.1 Avoiding BOLA

Key recommendations from 42crunch.co:

- Defence in Depth - fine grained authorisation in every control layer
- OAuth are not the solution (e.g. they limit access to an operation not a resource)
- Smaller items, e.g. - Avoid sequential identifiers
- Rate limiting
- TEST

7.14.8.4.4 Using Dependabot on GitHub Repos

All repos created using the one-click set-up API repo tool come with Dependabot security scanning enabled out of the box.

Dependabot is GitHub's vunerability and package manager tool, which scans repos for vunerabilities, alerts on potential issues and where possible raises PRs including the fix.

To find out more about Dependabot see the GitHub Docs: https://docs.github.com/en/code-security/dependabot

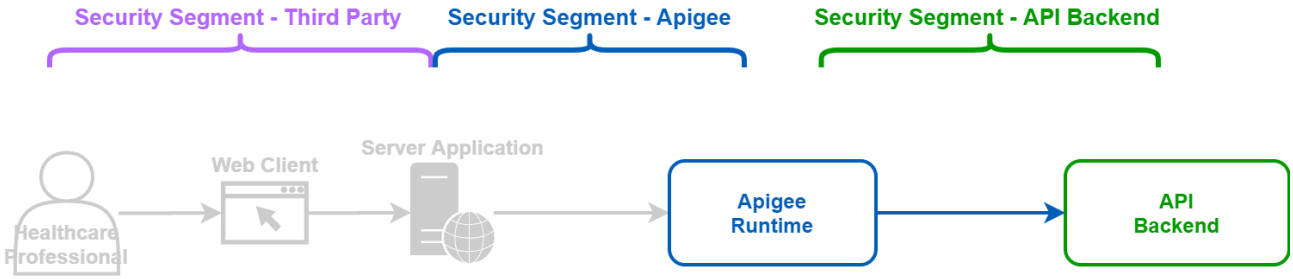
It is the responsibility of the repository owners/maintainers (producer teams) to keep on top of these alerts as they present a security risk to the platform.
This is a task that **needs** to be performed right from the set up of your repo all the way into BAU and should form part of your teams BAU duties.

It is **recommended as a monthly task** for someone to:

- Review Dependabot rasied PRs
    - Review
    - Merge
- Review Dependabout Alerts
    - Action a fix or Dismiss as necessary
- Release to all environments to remove the vunerable code from all envs.
    - Be aware that in Beta and beyond this will require a Production release.

7.14.8.5 Security for Third-Party Applications

Within this section we will be covering security for Third Party Applications. The purple section in the diagram below.



This section covers a large range of security requirements for your third-party application and how it connects to a given server. It includes guidance for making certain API security decisions as well as policies and information that need to be taken into consideration regarding consumer security.

This section will go into how client authentication works and how producers can use the client credentials flow for app-restricted APIs connecting from the server application to the Apigee runtime. This section also covers how to use identity service tokens as well as the combined and seperate auth differences we need to consider when considering an approach.

- Combined and separate auth differences (see page 790)
- End to end AWS web application pattern (see page 793)
- Security information for consumers (see page 794)

7.14.8.5.1 Combined and separate auth differences

The **user** restricted combined and separate auth patterns, are both OAuth flows to obtain authorisation to access the APIs associated with a specific Apigee Application.

Design principles:

- Combined auth is an easier way to integrate with both NHS API Management and CIS2
- Upgrade from combined auth to separate auth is easy
- Separate auth is there to help with more complex integrations

Comparing the two approaches, as an **API Consumer** (and for simplicity focusing on CIS2 - now termed Care Identity Authentication CIA):

	Combined	**Separate**
CIS2 ID Token accessible?	No	Held by consumer
CIS2 Access token?	No	Held by consumer
CIS2 client ID?	No	Setup and held by consumer
CIS2 Selected Role?	No	In CIS2 ID Token as a claim
OAuth Flow	Authorize (3-legged)	Token Exchange (same as client credentials 2-legged)
User Info	Proxied to CIS2 (CIA) Hard coded to one scope (nationalrbacaccess), could be easily changed	Consumer call call direct
Session timeouts	The APIM (Apigee) session is 12 hours, to match CIS2. For both Smartcard and stateless authenticators - the maximum session time must be managed by third partie	CIS2 ID Token is 1 hour, and the APIM recommendation (which is on the separate auth page) is to obtain an APIM access token as soon as an CIS2 ID token is issued so there is no risk the hour expires.

	Combined	Separate
User Info timeout	Limited to 1 hour as the CIS2 access token time expires in one hour	Limited to 1 hour as the CIS2 access token time expires in one hour
Back channel logout (BCL)	Built in	CIS2 recommend that the consumer implements this
Supports refresh token	Yes	Yes
Allow client to specify auth mode	Not currently, need to allow passing of ACR params. On backlog APM-2912[784].	Consumer manages on initial call to CIS2
Allow other OIDC params / scopes to be passed	Originally tied down to make logic simpler and to improve security. No reason why this cannot be changed	Consumer manages on initial call to CIS2
SCAL	Backlog item to plug any gaps in the SCAL for Combined auth (APM-1620[785])	Consumer completes CIS2 SCAL
Confidential Client	APIM to publish guidance an what a confidential client is	APIM to publish guidance an what a confidential client is
Modelling	Hides some complexity from the Consumer, but this could result in confusion later as their solution grows	This models the real world responsibilities of a Consumer, and opens up re-use of the ID Token when other (non NHS Digital) services start to support token exchange
Simulated auth	Supported for combined	ID token is hardcoded
Mock auth	Supported for combined	Supported for separate

Comparing the Producer differences:

784 https://nhsd-jira.digital.nhs.uk/browse/APM-2912
785 https://nhsd-jira.digital.nhs.uk/browse/APM-1620

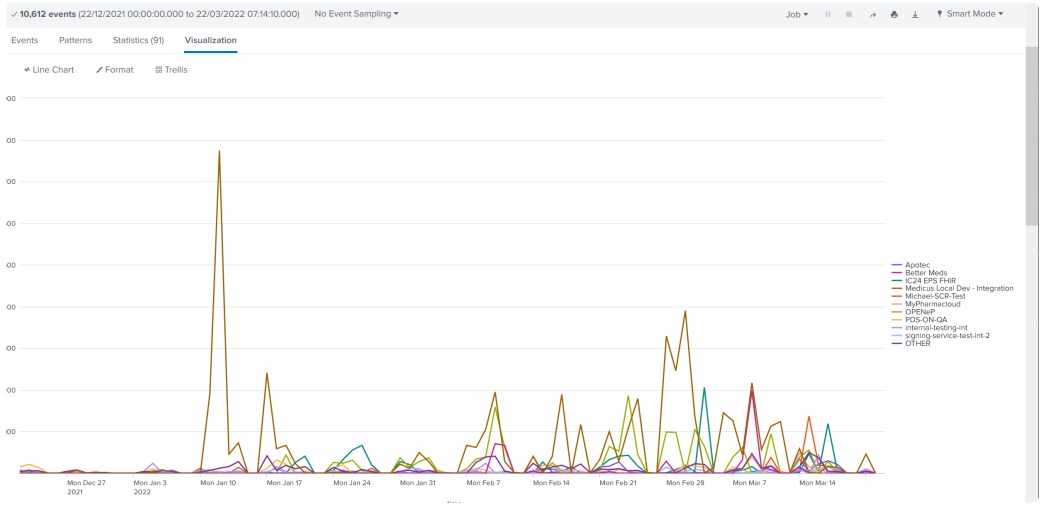
	Combined	Separate
CIS2 ID Token accessible?	Yes	Yes
CIS2 Access token?	Yes	No
User Info	Yes	No
Userinfo usage	EPS SCR	
Validate User Roles Shared Flow (see page 844)	Compatible	Selection logic for the relevant Selected Role ID works Cannot call userinfo, so cannot fully validate any header values passed in. Bug fixing to cope with these scenarios added: • PDS • EPS • SCR • Reasonable Adjustments
User Info timeout	A cache **is** populated, time would need to be set to 12 hours	N/A

7.14.8.5.1.1 Userinfo endpoint usage

The only other API that has openly talked about calling userinfo endpoint is Summary Care Record (SCR) API. Searching in GitHub showed code for the AWS Hosted Container referencing userinfo, and running a query on INT confirmed that at least one app using SCR (Michael-SCR-Test[786] only has one Product) was triggering userinfo requests:

```
index="requests_apm_ptl" sourcetype="apm-apigee-json" meta.env="int" proxy.suffix="/
userinfo" "proxy.name"="identity-service-*"
|  timechart span=1d count by meta.application
```

786 https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/e32c9ee5-ec84-4a83-b3e8-2ac186f29178



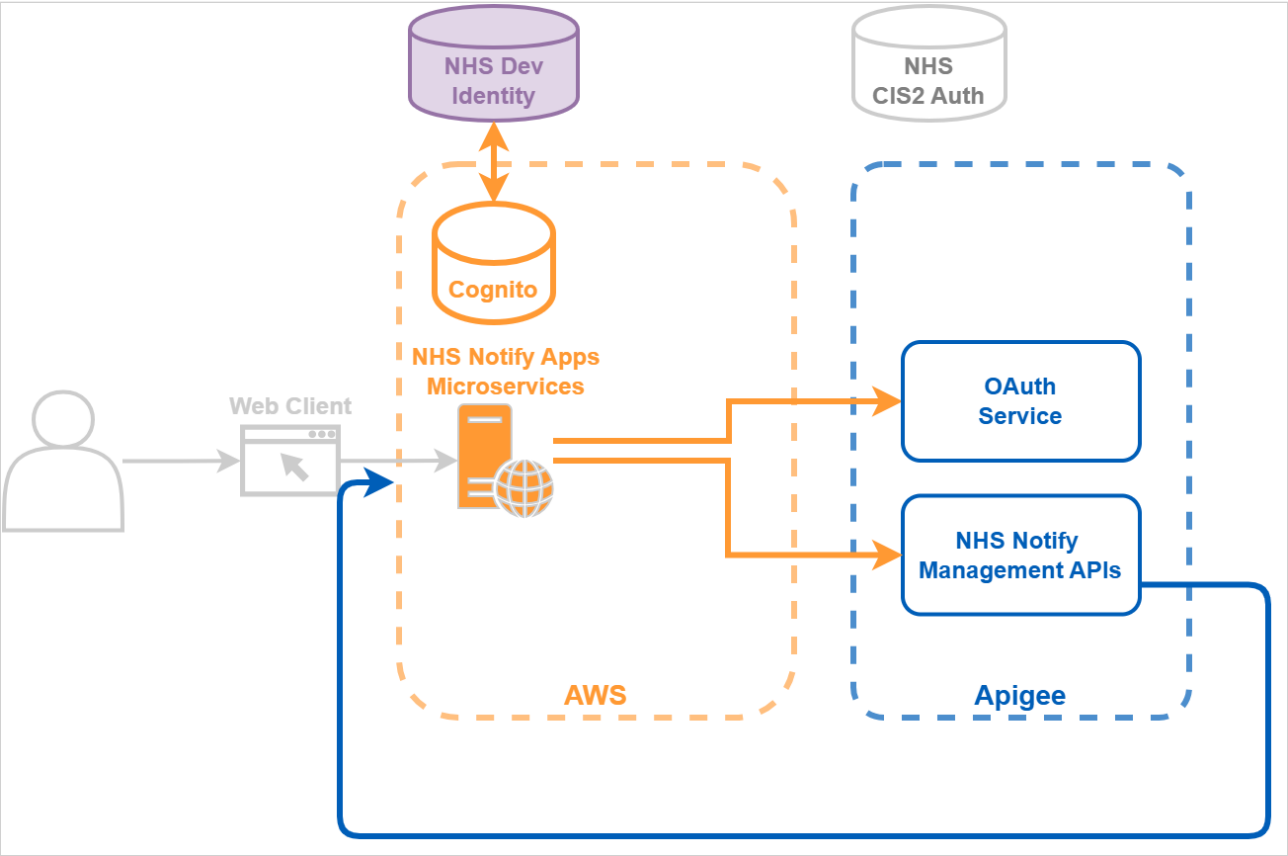
7.14.8.5.1.2 GP IT Futures guidance for New Market Entrants (NMEs)

Paper put together by GP IT to guide NMEs to using Separate Auth - GPITF Guidance on Token Exchange Framework implementation.docx[787]

7.14.8.5.2 End to end AWS web application pattern

Draft page to think about a web application that needs to be secured with CIS2 Auth or NHS login, and how that would interact with application authorisation managed by Cognito:

787 https://nhsd-confluence.digital.nhs.uk/download/attachments/402858600/GPITF%20Guidance%20on%20Token%20Exchange%20Framework%20implementation.docx?api=v2&modificationDate=1647945164642&version=1



7.14.8.5.3 Security information for consumers

As part of good security practice, you should provide contact information for consumers of your API, in order for them to report vulnerabilities in either your API or relevant infrastructure, as well as send any security queries they may have.

Generally, these contacts will be the following:

- cybersecurity@nhs.net[788] - the Cyber Associates Network[789] for NHS Digital and NHS England
- api.management@nhs.net[790] - the mailbox for API management support queries
- https://www.ncsc.gov.uk/information/vulnerability-reporting - the National Cyber Security Centre, where users can report vulnerabilities with UK government services.
- https://hackerone.com/nhs - the HackerOne NHS Programme, where users can submit vulnerabilities in platforms they are registered against.
- https://www.openbugbounty.org/ - OpenBugBounty. Same as HackerOne, different platform

If you have any support or security capabilities as part of your own API, you should include contact details for them also. Ideally this would be in the form of a managed mailbox that appropriate individuals in your team have access to.

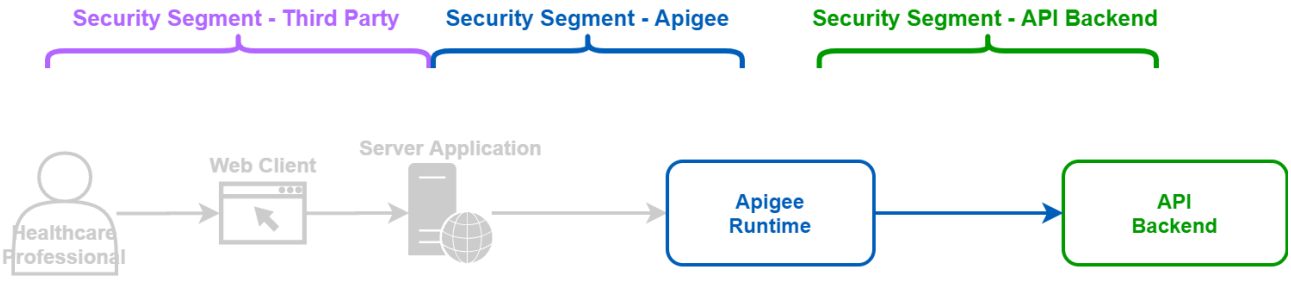
788 mailto:cybersecurity@nhs.net
789 https://digital.nhs.uk/cyber-and-data-security/about-us/cyber-associates-network#contact-details
790 mailto:api.management@nhs.net

In short, make sure consumers know what contacts are available to them, and what the intended purpose of communications is for each option. This information should be provided in the form of a `SECURITY.md` in the top level of your API repository. If you have used our one-click API Setup tool, you should have one of these already.

Otherwise, a good template can be found at https://github.com/NHSDigital/canary-api/blob/main/SECURITY.md; you can add any additional contact information as you see fit.

7.14.8.6 Security for Apigee

Within this section we will be covering security for Apigee. The blue section in the diagram below.



The pages within this section are pages that are directly involved with Apigee itself.

This section includes how to validate user roles using shared flows on Apigee as well as providing information on the OAuth error conditions and the different error responses that can be displayed. This section will also detail how to access id token claims regarding smartcards. The API Consumer Security page has detailed information on application and user restricted APIs and how we use OAuth to handle client authentication. Moreover, there is information on how we can configure API consumer apps in certain situations as well as calling endpoints.

- Accessing the id token claims for smartcards (see page 796)
- Calling the /userinfo endpoint from AWS hosted containers (see page 798)
- Client authentication - Private key JWT (see page 800)
- Configuring extended attributes for API Consumer Apps (see page 806)
- HTTP Strict Transport Security (HSTS) (see page 809)
- Identity service token variables (see page 810)
- Managing different security requirements for API Consumers (see page 813)
- OAuth error conditions (see page 816)
- Security patterns for API consumer access (see page 823)
    - Managing API product access via scopes (see page 827)
    - Authentication policies and proxy flows (see page 835)
- Validate user roles using shared flows (see page 844)

7.14.8.6.1 Accessing the id token claims for smartcards

7.14.8.6.1.1 Overview

The selected role id is an important value issued by the NHS ID service. During authentication with smartcards (this does **not apply** to iOS, Windows or FIDO), this value gets encoded into an id token and attached to the Apigee access token as a custom attribute. This document is about how to access that value.

7.14.8.6.1.2 The Apigee access token

When using a "user restricted API" with the "combined authentication and authorization", a sequence of steps is triggered described in the OAuth broker model (OAuth - Authentication broker model[791]). Then if the authentication and authorization are successful the user is granted an access token.

It is important to understand the difference between an ***access token*** and an ***id token***. In this document, we will call "access token" to the token issued by Apigee and granted to the user on successful authentication and authorization while the "id token" is a JWT token issued by the NHS ID service and attached to the access token metadata during the authentication process.

In other words, the access token that the user receives from the identity service (Apigee) has an id token from the NHS ID service attached as a custom attribute in the metadata of the token.

There are some useful claims from the id token that are already available in the access token, and therefore available in your proxy flow through the VerifyAccessToken policy. The claims are the "acr", "iss" and "sub" and the values are stored in the Apigee context variables 'accesstoken.id_token-acr', 'accesstoken.id_token-issuer' and 'accesstoken.id_token-subject' respectively (see table below).

claim	Apigee variable
acr	accesstoken.id_token-acr
iss	accesstoken.id_token-issuer
sub	accesstoken.id_token-subject

7.14.8.6.1.3 The id token

When decoding the id token from the access token we can see a payload like the following:

791 https://nhsd-confluence.digital.nhs.uk/display/APM/OAuth+-+Authentication+broker+model

id token payload

```
{
  "acr": "AAL3_ANY",
  "sub": "910000000001",
  "aud": "some-client-id",
  "selected_roleid": "555254242102",
  "amr": [
    "N3_SMARTCARD"
  ],
  "iss": "secure",
  "exp": 1616595370,
  "iat": 1616591770,
  "jti": "abf11024-d858-4d19-a172-4841716de547"
}
```

As we can see in the figure above, the id token (and therefore the access token) holds important information about the authentication and in particular the user role used to authenticate (under the filed "selected_roleid").

7.14.8.6.1.4 Getting claims from the id token

There are some useful claims from the id token that are not automatically attached to the access token.

Since the id token is an attribute of the access token, you can always use the DecodeJWT policy (see DecodeJWT policy | Apigee Docs[792]) to decode it and access the claims as needed (see code below).

Decode jwt example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DecodeJWT async="false" continueOnError="false" enabled="true" name="DecodeJWT.id_to
ken">
    <DisplayName>DecodeJWT.id_token</DisplayName>
    <Source>accesstoken.id_token</Source>
</DecodeJWT>
```

Once decoded you will be able to access all the claims and attributes from the id token and custom attributes as context variables in Apigee.

It is important to know that if your proxy is using the UserRoleService shared flow to validate the user role in the request, then all the claims from the id token will be already available in your flow and therefore it will be no need for you to decode the id token. Please refer to Validate user roles using shared flows - API Management (see page 844) for reference on validating user roles.

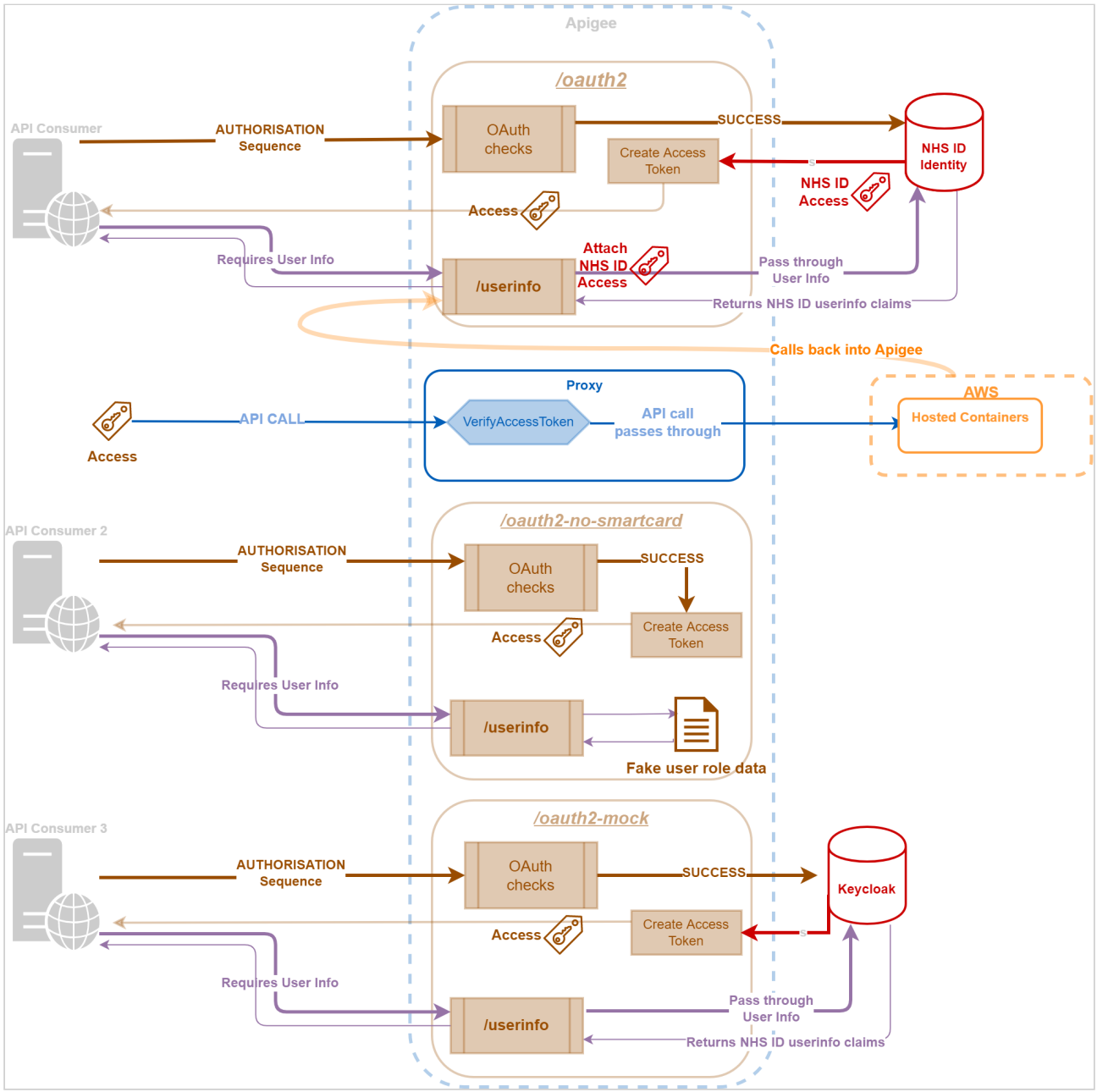
792 https://docs.apigee.com/api-platform/reference/policies/decode-jwt-policy?hl=en

7.14.8.6.2 Calling the /userinfo endpoint from AWS hosted containers

The inter-changeable nature of the services works well, except for the scenarios where an external service (i.e. APIM AWS Hosted Containers) has to call back into Apigee to retrieve the /userinfo data.

Sequence of events:

- Access Token is granted by the no-smart card endpoint
- Request is made to /myapiproxy/endpoint1 with the Apigee access token
- Request passed onto Docker container with Apigee access token
- Docker service needs to get some data contained in the user roles dataset, so it calls back into /<identity-service>/userinfo
    - But how does it know which identity service to call?



7.14.8.6.2.1 Ways to deal with the INT environment

APIM are considering changing the Identity Services on INT to deal with this directly - see APM-1905[793] - Cleaner way to call userinfo endpoint from AWS TO DO , and further details of the key initial discussion see 20210224 - WDR Meeting Minutes[794]

As an API Producer you have full control of your proxy, so there are a few options here:

793 https://nhsd-jira.digital.nhs.uk/browse/APM-1905
794 https://nhsd-confluence.digital.nhs.uk/display/APM/20210224+-+WDR+Meeting+Minutes

Option	Notes
Insist that your developers only use one Identity Service	Not ideal, but for smaller APIs with single API Consumers in testing this works. Ensure that the Identity Service URL or suffix is in an environment based variable so that change overs between endpoints requires no code changes
Set a header in proxy and process in Docker	When a header with a specific value arrives, call the no-smartcard endpoint instead. For the wider Spine meta-data transfers, there is an agreed standard header NHSD-Identity-IdP that should be sent to Spine for logging. In this design pattern, it picks up the NHS ID Token Issuer claim. For the no-smartcard endpoint this value will just be the string "secure" rather than a URN for NHS ID. The AWS Hosted Container application will then know it has to call the no-smartcard endpoint instead. The meta data does not have to be transferred this way, but overall this pattern works reasonably well - with very low risk of code entering Production that would allow access without authorisation.

7.14.8.6.3 Client authentication - Private key JWT

7.14.8.6.3.1 Context

This section specifies how to use the Private Key JWT Authentication method to secure your connection from the server application to Apigee.

7.14.8.6.3.2 Key Terms

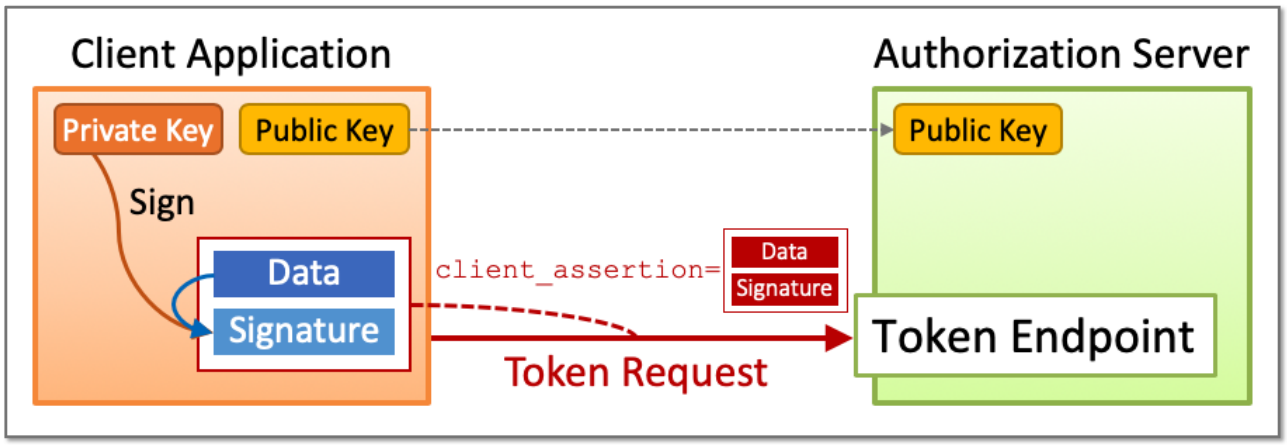
- Client - The Client in this instance is referring to the application that is interfacing with the Apigee server
- Credentials - When we say credentials, we are usually referring to the signature that is sent to the server to prove that the client is who they say they are.

7.14.8.6.3.3 Overview

This method of authentication is used to obtain an access token outside the context of a user. With machine-to-machine (M2M) applications, such as CLIs, daemons, or services running on your back-end, the system authenticates and authorizes the app rather than a user. For this scenario, typical authentication schemes like username + password or social logins don't make sense. Instead, M2M apps use the Client Credentials Flow (defined in OAuth 2.0 RFC 6749, section 4.4), in which they pass along their Client ID and their credentials to authenticate themselves and get a token.

The client application can authenticate by sending either client secret or client assertion. Most of the code examples and libraries for this flow cover the first scenario however, NHSD APIs require the client to pass

client assertion. This point is important when considering the evaluation of different libraries. The diagram below demonstrates how this process works.



7.14.8.6.3.4 Flow

Obtaining an access token consists of one POST request to the `/token` endpoint of the auth server. The body of this
the request must be url-encoded with the following data:

```
{
  "client_assertion": {your jwt}
  "client_assertion_type": "urn:ietf:params:oauth:client-assertion-type:jwt-bearer"
  "grant_type": "client_credentials"
}
```

example curl request

```
$ curl -i -H 'Content-Type: application/x-www-form-urlencoded' -X POST 'https://
internal-dev.api.service.nhs.uk/oauth2/token' -d
'grant_type=client_credentials&client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Acl
ient-assertion-type%3Ajwt-bearer&client_assertion={your jwt}'
```

Creating JWT

A JWT token consists of three parts. A header section, body, and signature. There are several methods of creating and
signing JWTs but before diving into that, first, we need to discuss what information a client has to provide for header
and body section.

Header

The header section contains only one value and that's `kid`. The value must be the same as the one you provided during
API registration.

```
{
  "kid": "kid-value-here"
}
```

Body

The body section contains claims. Please refer to the JWT RFC[795] for further information on claims, and on JWTs in general.

```
{
  "aud": "",
  "exp": 233,
  "iss": "",
  "jit": ""
}
```

Signature

The signature is computed from the header, body, and private key. You shouldn't manually compute this as the process is nontrivial and there are several libraries (see below) available which will do this for you.

Once you have all three elements available you put them together, period-separated, like so.

```
myJwtToken = "{header}.{body}.{signature}"
```

Obtaining access token

Obtaining an access token consists of one POST request to the `/token` endpoint of the auth server. The body of this request must be url-encoded with the following data:

```
{
  "client_assertion": {your jwt}
  "client_assertion_type": "urn:ietf:params:oauth:client-assertion-type:jwt-bearer"
  "grant_type": "client_credentials"
}
```

795 https://datatracker.ietf.org/doc/html/rfc7519#section-4.1

example curl request

```
$ curl -i -H 'Content-Type: application/x-www-form-urlencoded' -X POST 'https://
internal-dev.api.service.nhs.uk/oauth2/token' -d
'grant_type=client_credentials&client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Acl
ient-assertion-type%3Ajwt-bearer&client_assertion={your jwt}'
```

7.14.8.6.3.5 Libraries

OAuth's website has a section with a list of libraries that can be used to perform authentication. This list is not exhaustive and it's completely out of date. We tried some of the items in that list and here is the table below summarises our findings. As mentioned previously most libraries covert the Client Credentials flow by sending the client secret, however, in our APIs we need to create a client assertion which requires creating the signature of JWT using the private key.

Library	Homepage	Used	Comment
DotNetOpenAuth	http://dotnetopenauth.net/	✗	This library is not maintained anymore and is out of data.
DotNetAuth	https://samondotnet.blogspot.com/2012/12/introduction-to-dotnetauth.html	✗	This library is out of date. It doesn't support Client Credentials and it's primarily focused towards ASP.net
Owin Middleware	https://github.com/aspnet/AspNetKatana/	N/A	It's based on the Katana framework and it's focused on ASP.net. We don't know yet if this middleware can be used standalone or not but it's certainly is not a good fit for Client Credentials flow.
IdentityModel	https://identitymodel.readthedocs.io/en/latest/client/overview.html	✗	This library only supports client credentials using client secret.
Microsoft Identity Platform	https://docs.microsoft.com/en-us/azure/active-directory/develop/	✓	This library is by far the most promising way of doing authentication.

7.14.8.6.3.6 Implementation

How to create a certificate from private-key

In order to sign the JWT you need to have a pfx certificate and in this section, we describe how this can be done using your application key-pair. This section assumes your private-key is called "private.key" and public-key is called "public.key" the generated certificate will be called "certificate.pfx".

First, you need to extract the public-key from your private-key. If you have your public-key, then you can skip this step.

```
openssl req -new -x509 -key private.key -out public.key -days 365
```

This should generate public.key file. Now we can generate pfx certificate.

```
openssl pkcs12 -export -out certificate.pfx -inkey private.key -in public.key
```

This will generate the "certificate.pfx" file. We will use this file to sign JWT.

7.14.8.6.3.7 IdentityModel

For the below code you need to get these dependencies:

```
<PackageReference Include="IdentityModel" Version="6.0.0-preview.2" />
<PackageReference Include="System.IdentityModel.Tokens.Jwt" Version="6.14.1" />
```

```
using System;
using System.Collections.Generic;
using System.IdentityModel.Tokens.Jwt;
using System.Net.Http;
using System.Security.Claims;
using System.Security.Cryptography.X509Certificates;
using IdentityModel;
using Microsoft.IdentityModel.Tokens;

class Program
{
    static void Main(string[] args)
    {
        string token = GetToken();
    }

    static string GetToken()
```

```
    {
        string authUrl = "https://internal-dev.api.service.nhs.uk/oauth2";
        string aud = $"{authUrl}/token";
        string tokenEndpoint = $"{authUrl}/token";

        string clientId = "{your-client-id}";

        var certPath = "{path-to-certification.pfx}";
        var cert = new X509Certificate2(certPath);

        // create client_assertion JWT token
        var now = DateTime.UtcNow;
        var token = new JwtSecurityToken(
            clientId,
            aud,
            new List<Claim>
            {
                new Claim("jti", Guid.NewGuid().ToString()),
                new Claim(JwtClaimTypes.Subject, clientId),
            },
            now,
            now.AddMinutes(1),
            new SigningCredentials(
                new X509SecurityKey(cert, "{kid-value}"),
                SecurityAlgorithms.RsaSha512
            )
        );

        var tokenHandler = new JwtSecurityTokenHandler();
        var tokenString = tokenHandler.WriteToken(token);

        var client = new HttpClient();
        // token request - note there's no client_secret but a client_assertion which
contains the token above
        var requestBody = new FormUrlEncodedContent(new Dictionary<string, string>
        {
            {"client_assertion_type", "urn:ietf:params:oauth:client-assertion-
type:jwt-bearer"},
            {"client_assertion", tokenString},
            {"grant_type", "client_credentials"},
        });

        HttpRequestMessage tokenRequest = new HttpRequestMessage(HttpMethod.Post,
tokenEndpoint);
        tokenRequest.Content = requestBody;

        var responseMessage = client.SendAsync(tokenRequest).Result;

        return responseMessage.Content.ReadAsStringAsync().Result;
    }
}
```

7.14.8.6.4 Configuring extended attributes for API Consumer Apps

7.14.8.6.4.1 Overview

As an API Producer, you might have requirements to that mean you need to configure API Consumers differently. This is in addition to the app restricted / user restricted basic authorisation. For instance:

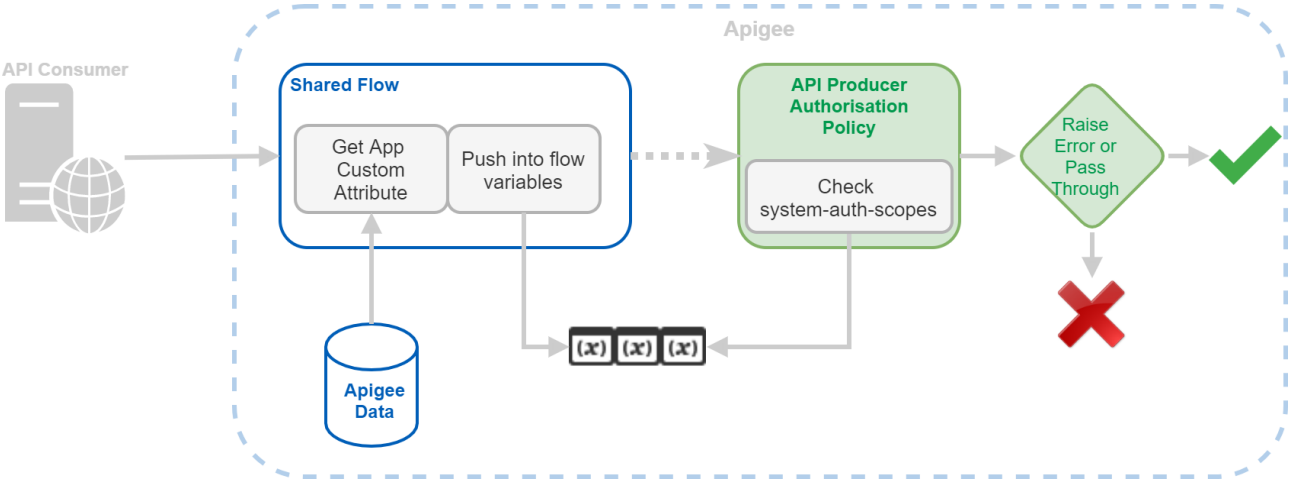
- During onboarding you assure an API Consumer to access a more sensitive data set than other API Consumers
- Simple data that needs to be logged in your backend
- Never store or rely on the Apigee Application ID, use extended attributes
- ***Future**: Digital Onboarding is the best place to allow these key / value pairs to be managed for any API. A UI (and API) will be added so that the data can be mastered there and flow through automatically*

Apigee implementation

Apigee allows the addition of custom attributes, which your proxy logic can take advantage of to achieve the above.

You should **not** directly access the custom attributes, but indirectly via an **APIM shared flow**. To use the shared flow, you need to consider:

- Configuration during onboarding:
  - The values are stored in a JSON object, as simple key value pairs
    - Attributes are segregated into nested-objects named with the short name of your api
  - This object is built and placed in the custom attribute "apim-app-flow-vars"
- Runtime:
  - Shared flow ExtendedAttributes shared flow (see page 662) extracts the JSON and populates the flow variables with the values - they will be prefixed (grouped) by **apim-app-flow-vars**.*
  - The API Producer should add a policy after the VerifyAccessToken to look at the verified Product scopes added and / or the system-auth-scopes
  - Some API Producers will want to send some or all of these attributes to their backend - Passing information to your API backend (see page 611)



APIM has implemented the extended attribute shared flow, because it **overcomes** some Apigee implementation constraints:

- Limit of 18 custom attributes
- Segregates attributes between APIs
- Free text field which is bad for automation, so we use JSON to help us, and you, manage configuration

It is worth noting that Apigee limits custom attribute to 2KB is size - see https://docs.apigee.com/api-platform/reference/limits. In the future, when that limit is exceeded by an API, the shared flow **will be expanded** to deal with more complex storage

7.14.8.6.4.2 Examples

apim-app-flow-vars might contain JSON such as:

```
{"pds":{"app-restricted":{"update":true}}}
```

These scopes are specific to your API, so they don't need to be name spaced, just need to be clear

It is possible to have multiple vars used within the custom attribute. Immunity history has an example of this which can be seen here[796].

7.14.8.6.4.3 Calling the shared flow

In your proxy or target XML add a step that calls the ***FlowCallout*** immediately after the authorization step e.g. ***OAuth2VerifyAccessToken*** or ***VerifyAPIKey***. This identifies the calling app and exposes app information including its custom attributes.

796 https://github.com/NHSDigital/immunisation-history-api#readme

default.xml

```
<TargetEndpoint name...>
    <PreFlow>
        <Request>
            <Step>
                <Name>OauthV2.VerifyAccessToken</Name>
            </Step>
            <!-- Must be placed after Authentication -->
            <Step>
                <Name>FlowCallout.ExtendedAttributes</Name>
            </Step>
...
```

In your policies folder add an XML file defining the ***FlowCallout***.

FlowCallout policy - OAuthV2 authentication

If you are using an ***OAuthV2.VerifyAccessToken*** policy to authorize users, add the following to your ***policies*** folder.

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
</FlowCallout>
```

FlowCallout policy - VerifyApiKey authentication

If you are using a ***VerifyApiKey*** policy, you will **need to provide its name as a parameter** (the name is defined as an attribute in the root element of your ***VerifyApiKey*** policy).

FlowCallout.ExtendedAttributes.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.E
xtendedAttributes">
    <DisplayName>Extract extended attribute</DisplayName>
    <SharedFlowBundle>ExtendedAttributes</SharedFlowBundle>
     <Parameters>
```

```
      <Parameter name="verifyApiKeyPolicyName">NameOfYourVerifyApiKeyPpolicy</
Parameter>
    </Parameters>
</FlowCallout>
```

> **Why do I need to provide the name of the VerifyApiKey policy?**
>
> Because Apigee assigns the app/product flow variables inside a namespace which includes the policy name for ***VerifyApiKey*** but not for ***OAuthV2.VerifyAccessToken***

For more detail on how to implement ExtendedAttributes shared flow click here (see page 662).

Runtime checking an App is authorised

PDS has a more complex way of handling authorisation decisions, but shows how the verified Product scopes are used with extra logic to authorise in different situations. Snippet from https://github.com/NHSDigital/personal-demographics-service-api/blob/master/proxies/live/apiproxy/targets/ig3.xml:

```
 <Step>
<Name>AssignMessage.SetAccessModeUserRestricted</Name>
<Condition>(scope JavaRegex "(.+\ urn:nhsd:apim:user-nhs-id:aal3:personal-
demographics-service\ .+|^urn:nhsd:apim:user-nhs-id:aal3:personal-demographics-
service\ .+|.+\ urn:nhsd:apim:user-nhs-id:aal3:personal-demographics-service$|
^urn:nhsd:apim:user-nhs-id:aal3:personal-demographics-service$|.+\ personal-
demographics-service:USER-RESTRICTED\ .+|^personal-demographics-service:USER-
RESTRICTED\ .+|.+\ personal-demographics-service:USER-RESTRICTED$|^personal-
demographics-service:USER-RESTRICTED$)")</Condition>
</Step>
```

7.14.8.6.5 HTTP Strict Transport Security (HSTS)

7.14.8.6.5.1 Overview

HSTS is a response header that is sent to the browser from a server. Once a browser has received this header, it will prevent future communciations from being sent over HTTP to the given domain. The browser will instead user HTTPS for all communication.

7.14.8.6.5.2 HSTS setup in proxies

We have configured a shared flow to handle HSTS called `AssignMessage.SetHSTSHeader` (link to file[797]). As this has been added in the post proxy flow, you do not need to do anything to enable it and it will automatically apply to all calls made to the proxy.

PostProxySharedFlow/sharedflows/default.xml

```
...
    <Step>
        <Name>AssignMessage.SetHSTSHeader</Name>
    </Step>
...
```

(https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-shared-flows/templates/PostProxySharedFlow/sharedflows/default.xml)

7.14.8.6.6 Identity service token variables

7.14.8.6.6.1 Overview

When a user logs in, their identity and authentication type can be passed to the backend and stored as variables or attributes. The Identity Service API uses OAuth2 flow for identity information. The access token is generated by the Identity Service Proxy API and associated with the access token issued by Apigee. Claims from the user's NHSID JWT are stored with it as attributes. This information is static and never sent to the client. It is made available in flow variables after the VerifyAccessToken policy is called. For more information on passing information to your backend see Passing information to your API backend (see page 611).

From the documentation on Getting Custom Attributes at Runtime[798]:

> *If there are any custom attributes on the token, those custom attributes are made available in a context variable with the name* `accesstoken.{custom_attribute}`.

7.14.8.6.6.2 Variables

After the **VerifyAccessToken** policy is called static information called variables are available to access. In our case this means that following usage of the OAuthV2/VerifyAccessToken policy, subsequent policies in the flow can access the following variables:

797 https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-shared-flows/templates/PostProxySharedFlow/sharedflows/default.xml
798 https://docs.apigee.com/api-platform/security/oauth/customizing-access-tokens#gettingcustomattributesatruntime

Variables	Values	Description	Splunk field
accesstoken.auth_type	"app" or "user"	Primary = reporting** Secondary = authorisation decisions.	auth.**meta**.auth_type
accesstoken.auth_user_id	656005750108 9170445222	NHS CIS2 - user ID. NHS login - contains the NHS Number of the user.	auth.**user**.user_id
accesstoken.auth_grant_type	client_credentials authorization_code token_exchange	This is a processed value for reporting purposes. The reports need simple values to make it easier, and so this is the right field to use if the "token_exchange" grant type needs to be used for logical processing.	auth.**meta**.grant_type
accesstoken.id_token		The entire JWT ID token.	
accesstoken.id_token-subject	656005750108 GUID	The token "subject" is the unique identifier from the identity provider.	
accesstoken.id_token-issuer	URL	Where the token was issued (NHSID info).	
accesstoken.id_token-acr	e.g. AAL3_ANY	CIS2 only - contains the level the user is authenticated to.	
accesstoken.id_token-amr	e.g. N3_SMARTCARD	CIS2 only - the method the user used to authenticate.	
accesstoken.id_token-id-assurance-level	e.g. 3	The level of assurance performed on the user for identity proofing.	
accesstoken.id_token-identity-proofing-level	e.g. P9	NHS Login only - The level of identity proofing performed.	
accesstoken._first_issued		When the ID token was issued.	

Variables	Values	Description	Splunk field
accesstoken.id_token-authentication_assurance_level	e.g. 3	The CIS2 Auth ID Token has the claim **authentication_assurance_level.**	

** The reports need simple values to make it easier to build reports. The small nuances to getting the correct value are all applied. **There is a risk** of using the reporting values for security authorisation, could mean a reporting change causes an issue.

7.14.8.6.6.3 CIS2 Auth only values

CIS2 Auth uses the ACR and AMR values is quite a complex way:

- Authentication Context Class Reference - ACR[799]
- AMR (authentication methods references) claims[800]

These pages have restricted access, and you might need to request access. For convivence, as of Jan 2021, summarised below:

- ACR: NHS ID only supports AAL3, but does include the type of authentication (overlapping with the AMR definition), e.g. AAL3_IOS
  - To use this value, just check that the first four letters match "AAL3"
- AMR: This has the following values:
  - IOS
  - FIDO2
  - N3_SMARTCARD

7.14.8.6.6.4 Personal Demographics Service API example usage

In the Personal Demographics Service API[801] these attributes are accessed in the following policies:

Variable	Description
OauthV2.VerifyAccessToken	Instantiates the `accesstoken` variable
AssignMessage.AddUserIdHeader	Reads the User ID held in the **`accesstoken.id_token-subject`** and creates a HTTP header for consumption by the target service.

799 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/114038063/Authentication+Context+Class+Reference+-+ACR

800 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/117652121/AMR+authentication+methods+references+claims

801 https://github.com/NHSDigital/personal-demographics-service-api

Variable	Description
AssignMessage.AddIssuerHeader	Reads the issuer held in the `accesstoken.id_token-issuer` and creates a HTTP header for consumption by the target service.

7.14.8.6.7 Managing different security requirements for API Consumers

7.14.8.6.7.1 Overview

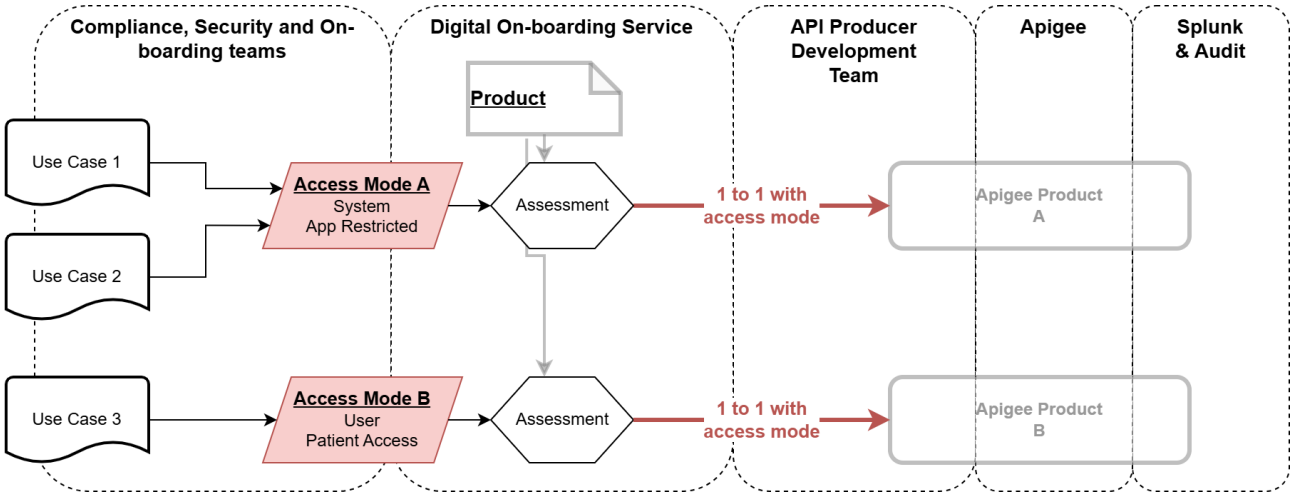
In a simple API you might have very simple authorisation rules, e.g. if it's a patient they can access all the endpoints. As an API develops you might find that a simple yes/no is not enough.

How you apply the security to each request becomes more complex as there are multiple options:

- Access Mode (aka Apigee Products)
- App vs. User-CIS2 vs. User-NHS login
    - Level of authentication e.g. Level 3 or P9
- Extended attributes
- You should allow one or more Apigee applications for each consumer to match their connection topology
- For CIS2, you could consider National RBAC where the healthcare worker selects a role
- Backend fine-grained authorisation

Also see: Access modes - compliance - security and simplifying running your API (see page 299)

Relationships between Product, Access Mode, and Apigee Product:



7.14.8.6.7.2 Definitions

Key pages in the API Producer Zone that will help us understand more:

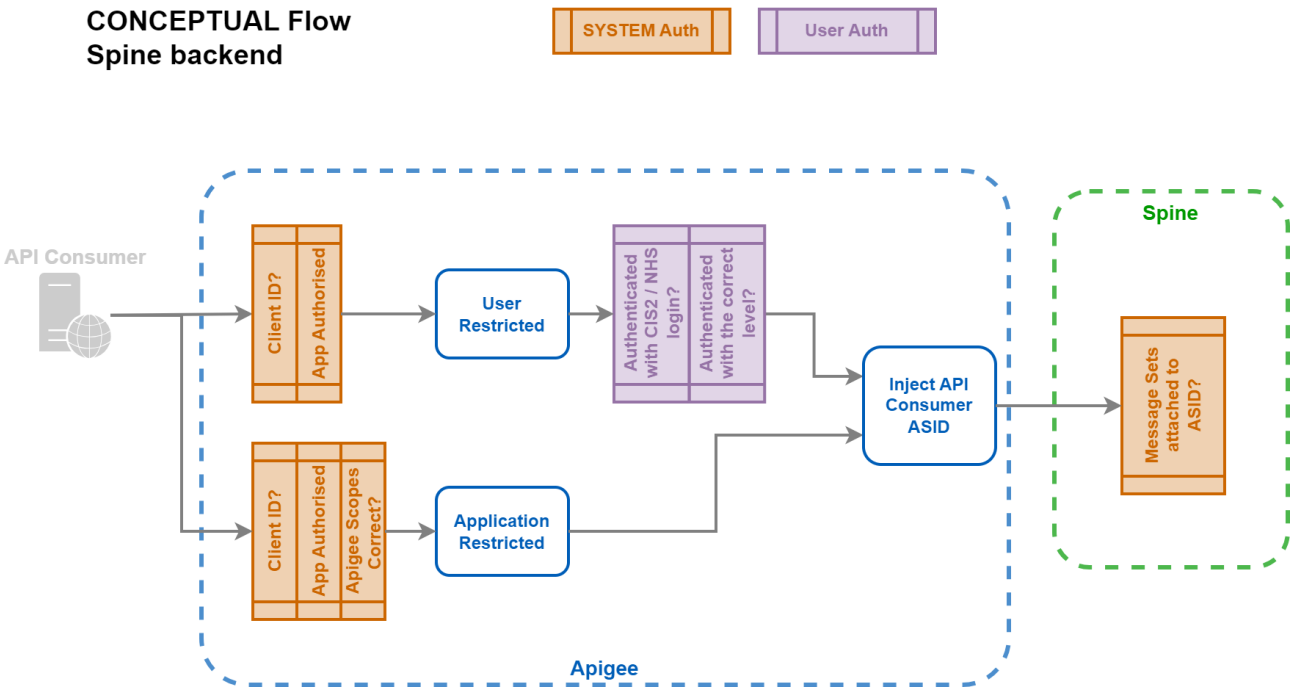
Security	Description	Pages
Access Mode	Assurance for the use of your API changes depending on how the API Consumer uses it. This is managed in the Digital Onboarding Service (DOS) through an interactive journey. For future automation it is **critical** that your API has one Access Mode per Apigee Product	Access modes - compliance - security and simplifying running your API (see page 299)
App vs. User-CIS2 vs. User-NHS login	You can define which endpoints of your API require which type of authentication. This is in-built proxy logic, you don't need to write new authorisation logic.	Managing API product access via scopes (see page 827) How to configure API consumer security patterns using proxy generator (see page 875)
Extended attributes	If you want to only allow some API Consumers to access certain endpoints, but only need one Assurance journey under one access mode. Then you can use extended attributes to store data against the API Consumer Apigee App for use in your proxy logic. Process: • As an API Team you defined the key value pairs you will use to manage this • Your onboarding lead (see page 1000) would request the appropriate values to be used during Production enablement • At runtime, the extended attribute shared flow makes the values available to your proxy logic. You can apply authorisation logic	Configuring extended attributes for API Consumer Apps (see page 806)

Security	Description	Pages
CSI2 National RBAC	APIM does **not apply** any National RBAC rules. Some of the Spine APIs require the use of CIS2 and that the user has "selected a role". This role includes the location the user is working at and what type of role they have (e.g. Doctor or Nurse). No new APIs have adopted National RBAC, as the focus for the new APIs has been narrower and doesn't require this type of IG and Cyber Security approach. We recommend you scope your APIs to avoid this because: • Adds a significant layer of complexity • Increases the onboarding & assurance burden on third party software • The long term strategy around National RBAC is not clear APIM does have a shared flow to do some basic role validation, but this can only work for the combined auth pattern[802]. Where a User Role is mandatory, the same shared flow also handles default role selection and header override logic - which works on both combined and separate auth[803].	Developer Guide on RBAC[804] CIS2 Role Selection[805] Validate user roles using shared flows (see page 844)
Backend fine-grained authorisation	A key principle of the API Platform is to apply simple (coarse) grained security to every request, e.g. have they been authenticated with CIS2 - yes or no. Where more complex security (fine grained) is required, the preferred pattern is for your API backend to implement that logic (e.g. only Doctors can update the record)	Managing different security requirements for API Consumers (see page 813)

802 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation
803 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation
804 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/national-rbac-for-developers
805 https://digital.nhs.uk/services/care-identity-service/applications-and-services/cis2-authentication/guidance-for-developers/detailed-guidance/role-selection

7.14.8.6.7.3 Spine Apigee authorisation concept flow

A conceptual way to look at Apigee and the APIM implementation of authorisation stages when a request is made:



7.14.8.6.8 OAuth error conditions

7.14.8.6.8.1 Overview

In the OAuth flow, there are several steps involving calls to the OAuth server to request information. This document outlines the expected responses and standards for handling various error scenarios. For more details, you can refer to auth0.com/docs[806].

We're primarily concerned with the 'Auth0 Tenant' in the OAuth flow, specifically focusing on :

- step 2: authorization code request to /authorize
- step 6: authorization code + client id + client secret to /oauth/token

7.14.8.6.8.2 OAuth specification

Authorisation error response

806 https://auth0.com/docs/flows/concepts/auth-code

For authorization errors, we follow the OAuth specification outlined in RFC 6749 Section 4.1.2.1[807].

When a request to the authorization server fails due to problems like a missing, invalid, or mismatched redirection URI, or if the client identifier is missing or invalid, the server should notify the resource owner about the issue. It should not automatically redirect the user's browser to an invalid redirection address.

If the resource owner denies the access request, or if there are other issues apart from the redirection URI, the authorization server informs the client by adding certain parameters to the redirection URL in a specific format.

These parameters include:

- `error` (REQUIRED): represents an error code when the request is denied by the resource owner. These codes are:
  - `invalid_request` - missing or invalid parameters, duplicate parameters, or malformed requests.
  - `unauthorized_client` - the client is not authorized for this grant type.
  - `access_denied` - request denied by the resource owner or authorization server.
  - `unsupported_response_type` - authorization server doesn't support this method.
  - `invalid_scope` - requested scope is invalid, unknown, or malformed.
  - `server_error` - unexpected server condition, preventing request fulfillment.
  - `temporaily_unavailable` - server temporarily overloaded or undergoing maintenance.
- `error_description` (OPTIONAL) - provides additional information in human-readable text to help the client developer understand the error.
- `error_uri` (OPTIONAL) - a URL pointing to a web page with more information about the error for the client developer.
- `state` – this is **required** if a "state" parameter was initially present in the client's authorization request. It holds the same value received from the client.

Access token error response

For authorization errors, we follow the OAuth specification outlined in RFC 6749 Section 5.2[808].

When an error occurs, the authorization server responds with an HTTP 400 (Bad Request) status code and includes the following parameters in the response:

- `error` (REQUIRED) - an error code that describes the issue, for example:
  - `invalid_request` - missing or wrong parameters, repeated or malformed.
  - `invalid_client` - authentication failure for the client, like an unknown client or unsupported authentication.
  - `invalid_grant` - the provided authorization grant is invalid or doesn't match the request.
  - `unauthorized_client` - the client is not authorized for this grant type.

807 https://tools.ietf.org/html/rfc6749#section-4.1.2.1
808 https://tools.ietf.org/html/rfc6749#section-5.2

    - `unsupported_grant_type` - unsupported grant type by the server.
    - `invalid_scope` - issues with the requested scope. The error code should only contain specific characters.
- `error_description` (OPTIONAL) - additional text to help the client developer understand the error. It's optional and should use specific characters.
- `error_uri` (OPTIONAL) - a URL to a web page providing more information about the error for the client developer. It should conform to URL syntax and use specific characters.

These parameters are included in the response body using JSON format. The order doesn't matter.

An example of this response:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json;charset=UTF-8
Cache-Control: no-store
Pragma: no-cache

{
   "error":"invalid_request"
}
```

7.14.8.6.8.3 Apigee error responses

Authorisation code request to/authorize

For the 'authorize' step in the OAuth flow, you need to provide certain parameters to be successful in authorising.

Apigee response provides a response body with the format of in a json format:

- ErrorCode
- Error

An example request to our INT /authorize endpoint without the response_type parameter passed:

```
https://int.api.service.nhs.uk/oauth2/authorize?
client_id=yuiaHsalC23LmnaTG6Peqcg0HXY3qWei&redirect_uri=https%3A%2F%2Fwww.google.com&
scope=openid&state=d04a38c3-e508-4c43-9918-76c7203c7ade
```

We receive the error response of:

```
{
    "ErrorCode" : "invalid_request",
    "Error" :"The request is missing a required parameter : response_type"
}
```

These are the expected http responses from Apigee:

Error Condition	Http Response Code	Error Code	Error
Invalid redirect URI	400	invalid_request	Invalid redirection uri http://www.invalid_example.com
No redirect URI	400	invalid_request	Invalid redirection uri
Invalid key	401	oauth.v2.InvalidApiKey	Invalid ApiKey
Missing parameter	400	invalid_request	The request is missing a required parameter : response_type
Missing API key	401	steps.oauth.v2.FailedToResolveAPIKey	Failed to resolve API Key variable request.queryparam.client_id
Invalid response type	400	invalid_request	Response type must be code
Missing response type	400	invalid_request	The request is missing a required parameter : response_type

Authorization code + client id + client secret to/oauth/token

For the token exchange once the client has received the auth code and sends a request to exchange the code for a token.

This also requires some parameters to be sent to verify the transaction in which some error scenarios can arise.

Apigee Reponse provides a response body with the format of in a json format:

- ErrorCode
- Error

An example 400 response when the auth code isn't provided would be:

```
{
```

```
    "ErrorCode" : "invalid_request",
    "Error" :"Invalid Authorization Code"
}
```

Also they respond in the json format of:

- fault
    - fault string
    - detail
        - errorcode

An example of this is:

```
{
    "fault": {
        "faultstring": "Unresolved variable : request.header.Authorization",
        "detail": {
            "errorcode": "steps.basicauthentication.UnresolvedVariable"
        }
    }
}
```

These are the current responses for each error scenario on Apigee currently:

Error Condition	Http Response Code	Error Code	Error
Invalid auth code	400	Bad Request	Authentication Error: not right number of dots, 1
No redirect URI	200	N/A	N/A
Invalid redirect URI	200	N/A	N/A
Invalid grant type	404	invalid_request	Not Found
No API key	400	Bad Request	Incorrect number of callbacks found in JSON response
No API secret	400	Bad Request	Incorrect number of callbacks found in JSON response

Error Condition	Http Response Code	Error Code	Error
No grant type (custom policy)	404	invalid_request	Not Found
No auth code	500	entities.UnresolvedVariable	Unresolved variable : request.formparam.code

Common errors

Some of the common errors that we get are also documented on the consumer facing documentation. You can see the specific error scenarios regarding your application-restricted or user-restricted application by following the Security and authorisation[809] page.

Here are some examples of the common errors:

Error Condition	Http Response Code	Error Code	Error
Client assertion type is missing/invalid	400	invalid_request	Missing or invalid client_assertion_type - must be 'urn:ietf:params:oauth:client-assertion-type:jwt-bearer'
Sub and iss claims don't match or are missing	400	invalid_request	Missing or non-matching 'iss'/'sub' claims in client_assertion JWT
JWT signature is invalid	401	public_key error	JWT signature verification failed
Client secret is invalid	401	invalid_client	client_id or client_secret is invalid
Refresh token is invalid	401	invalid_grant	refresh_token is invalid
Public key misconfigured	403	public_key error	The JWKS endpoint for your client_assertion can not be reached

809 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#top

7.14.8.6.8.4 NHSID error responses

NHSID error responses returned to Apigee

NHSID return errors in a different format than Apigee, so can provide more detail into the error return.

For NHSID the returned response contains:

- code (http error code)
- reason
- message
- failureUrl

An example failure response of a 401 Unauthorized request:

```
{
        "code":401,
        "reason":"Unauthorized",
        "message":"Login failure",
        "detail":{
             "failureUrl":"#unsupported"
         }
}
```

NHSID error responses returned to the calling user

However, if an error occurs between Apigee and NHSID this is an integration error between the two systems.

This results in a error response to the calling user from Apigee:

/callback:

Scenario	status code	error	error_description
State value incorrect	500	unknown_error	An unknown error occurred processing this request. Contact us for assistance diagnosing this issue: https://digital.nhs.uk/developer/help-and-support quoting Message ID
Client_id value incorrect	401	-	-

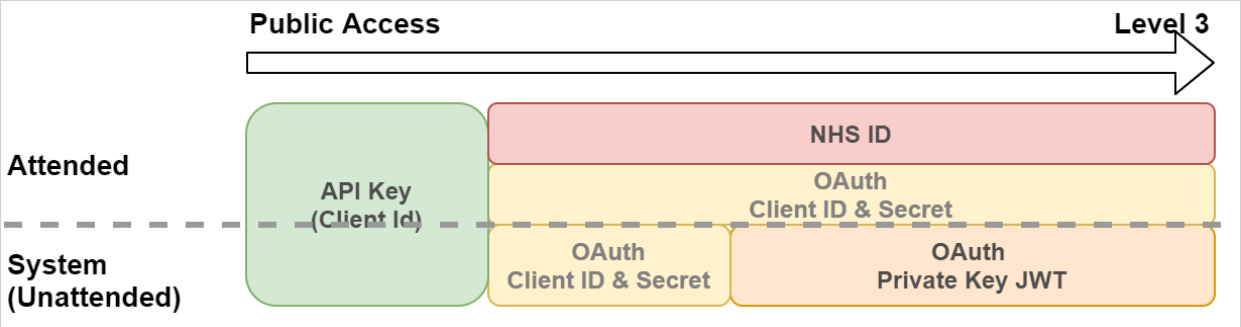
/token:

Scenario	status code	error	error_description
Malformed / unexpected response	500	unknown_error	An unknown error occurred processing this request. Contact us for assistance diagnosing this issue: https://digital.nhs.uk/developer/help-and-support quoting message id

7.14.8.6.9 Security patterns for API consumer access

7.14.8.6.9.1 Overview

There are various security options for connecting to the APIs on the API platform. Overall we are taking a tiered approach to security, keeping it lightweight where possible:



The security options are described to external developers at https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.

This page summarises them and explains which one to use for your API. This is guidance only - you need to decide for yourselves, in collaboration with your security contacts, which model is appropriate.

See:

- Managing API product access via scopes (see page 827) - in terms of guidance on how to implement these on your proxy.
- Guidance for making API security decisions (see page 437) - **draft** guidance on how you might make a decision on determining the appropriate access level
- Authentication policies and proxy flows (see page 835) for details and components for authentication with restricted API's
- *NOTE: This is independent of the security between the API Management Platform and your API backend. Where you, as an API Producer, need the system identity or identity of the authenticated user - you can be provided this in a number of ways, see:* Passing information to your API backend (see page 611). *For details on how security is segmented:* Security Architecture Segments[810]

810 https://nhsd-confluence.digital.nhs.uk/display/APM/Security+architecture#Securityarchitecture-segments

7.14.8.6.9.2 Open access API

> ⚠ APIM Strategy is to never publish a truly "Open API". A key driver for creating API Management was to gain greater visibility of who consumers NHS D data and services.
> The "open API" concept is delivered by setting such APIs to "auto approve in Production". The developer does have to register, but being immediately issued with a key they can use to access the API, should be a low enough barrier to entry that it doesn't discourage usage.
> This option is left in the documentation as non-APIM APIs use it.

Please refer to our developer guidance on open access APIs[811].

Anyone can access an open access API - no control.

- End user not authenticated (or not present)
- Calling application not authenticated
- No way of knowing which users or applications are using the API
- No way of getting in touch with application developers to let them know of changes or issues
- No way of rate limiting per application (you can rate limit overall)

Recommended if:

- Your API doesn't have any sensitive data
- You don't care who uses it
- You don't need to get in touch with developers

7.14.8.6.9.3 Application-restricted API - API key authentication

Please refer to our developer guidance on application-restricted APIs[812], and API key authentication in particular[813].

- End user not authenticated (or system to system where no user is present)
- Calling application authenticated, but not strongly
- No way of knowing which users are using the API
- But you do know which applications are using it
- And you can get in touch with application developers to let them know of changes or issues
- And you can rate limit per application

Recommended if:

- There is low sensitivity for the services or data
- You want to know which apps are using your API
- You don't care which end users are using it

811 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#open-access-apis
812 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#application-restricted-apis
813 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication

- But you don't need them to be strongly authenticated
    - For example, your API doesn't handle personal or sensitive data

7.14.8.6.9.4 Application-restricted API - signed JWT authentication

Please refer to our developer guidance on application-restricted APIs[814], and signed JWT authentication in particular[815].

For detailed background see: OAuth - application restricted access[816]

For an implementer's perspective and technical details, see: Case study - implementing private key JWT application-restricted access (see page 838)

Similar to application restricted standard auth, but:

- This is intended for system to system integrations, where no user is present
- Calling application strongly authenticated

Recommended if:

- Middle level sensitivity data and services
    - For example, your API handles personal or sensitive data
- There is no user present

Deep dive into the design behind system to system authentication: OAuth - application restricted access[817]

Description of implementation in the Identity Service, and how to integrate into an API Proxy: Application Restricted Access / Access Modes & Restrictions (see page 838)

7.14.8.6.9.5 User-restricted - NHS CIS2

Please refer to our developer guidance on user-restricted APIs[818].

- Calling application authenticated
- End user is a healthcare worker strongly authenticated using NHS CIS2

Recommended if:

- The services and data require Identity/Authentication Level 3 security
- The end user is a healthcare professional

We support two patterns:

- Combined auth - see https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-identity-combined-authentication-and-authorisation
    - For this pattern, clients are required to register their application-restricted APIs with us. See: https://portal.developer.nhs.uk/

814 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#application-restricted-apis
815 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication
816 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117664886/OAuth+-+application+restricted+access
817 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117664886/OAuth+-+application+restricted+access
818 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#user-restricted-apis

    - *For detailed background see:* OAuth - Combined Auth[819]
- Separate auth - see https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-identity-separate-authentication-and-authorisation
    - For this pattern, clients are required to register their application with NHS CIS2. The process of obtaining authentication is documented on NHS Care Identity Service 2[820]
    - *For detailed background see:* OAuth - Token Exchange Model - User - Separate auth pattern[821]

If you need to know who the end user is (for example, so you can implement RBAC), or otherwise want more control over authorisation then separate auth is recommended.

Otherwise, combined auth (where authentication and authorisation are handled as part of the same call to our OAuth2.0 auth server) is recommended for a simpler integration. For the combined auth pattern, we have implemented back channel logout (BCL), so you don't have to - see https://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance/session-management#back-channel-logout.

Deep dive into the design behind system to system authentication: OAuth - Combined Auth[822].

7.14.8.6.9.6 User-restricted - NHS login

Please refer to our developer guidance on user-restricted APIs[823].

- Calling application authenticated
- End user is a citizen strongly authenticated using NHS Login

Recommended if:

- You need to know who the end user is
    - For example, access to personal or sensitive data
- The end user is a citizen

Currently only Separate auth is officially available: https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation.

- For this methods, clients are required to register their application with NHS Login. The process of obtaining authentication is documented on NHS Login website[824] and also there is a list of certified OpenID Connect libraries for various programming languages listed here[825]. After doing authentication and obtaining an ID Token the process of exchanging it with an access token is the same as NHS CIS2 separated authentication and authorisation explained on this page.

---

819 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110886099/OAuth+-+Combined+Auth
820 https://digital.nhs.uk/services/nhs-care-identity-service-2
821 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182424707/OAuth+-+Token+Exchange+Model+-+User+-+Separate+auth+pattern
822 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110886099/OAuth+-+Combined+Auth
823 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation#user-restricted-apis
824 https://nhsconnect.github.io/nhslogin/
825 https://openid.net/developers/certified/

- *For detailed background see:* OAuth - Token Exchange Model - User - Separate auth pattern[826]

Combined auth is available in lower environments, but currently is not allowed in Production.

- For this pattern, clients are required to register their application-restricted APIs with us. See: https://portal.developer.nhs.uk/
- *For detailed background see:* OAuth - Combined Auth[827]

7.14.8.6.9.7 Managing API product access via scopes

Overview

You secure your API by restricting requests to certain authentication mechanisms or access levels. We make use of Apigee internal scopes to record and pass onto a proxy what authentication the calling client has undertaken to obtain an access token.

The configuration is done by specifying one or more internal scopes on the API Product in the manifest yaml file:

- Based on what authentication mechanism is used with the Identity Service, the appropriate scopes are assigned to the access token.
- When the VerifyAccessToken policy is used in your API Proxy, it automatically rejects any requests without appropriate scopes.
- If you want to do further checks, or apply this differently per flow - you can customise the behaviour.

> **External Scopes**
>
> We currently do not handle any externally requested scopes during authentication. Scopes are exclusively used internally.

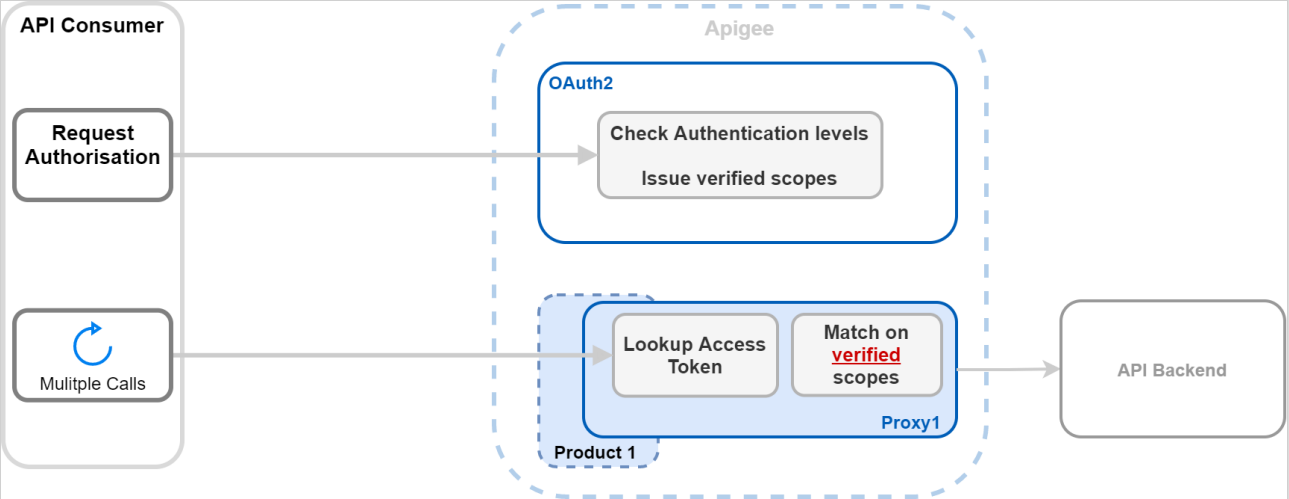
To enforce this within your API **you** must:

1. Configuring products to include scopes (see page 827)
2. Adding VerifyAccessToken policy to your proxy (see page 827)
3. Configuring the VerifyAccessToken policy to restrict access (see page 827)

For a visual overview the three steps above can be viewed as:

826 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182424707/OAuth+-+Token+Exchange+Model+-+User+-+Separate+auth+pattern

827 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110886099/OAuth+-+Combined+Auth



After these steps are completed your API will be secured with the appropriate access levels required to access it.

Coarse grained vs fine grained security

A key principle of the API Platform is to apply simple (coarse) grained security to every request. The implementation pattern on this page, achieves that. Where more complex security (fine grained) is required, the preferred pattern is for your API backend to implement that logic (e.g. only Doctors can update the record). Design points to consider:

- It is sometimes significantly easier / less maintenance to do this in Apigee
- Where the logic occurs in the backend, you must transmit the additional meta-data to the backend. Some of this meta-data might already be in the payload. Where it isn't then a new header, or (preferably) an existing header with JSON format data in should be provided
- To help in your logic - remember that the scopes attached to the Access Token are **verified**, you as an API Producer can trust the value to represent the un-bundling of the ID Token

Higher level details on this topic can be read about here: Access modes - compliance - security and simplifying running your API (see page 299).

Reasons to have more than one Apigee product

The use of more than one Product is not technically required, but APIM do use this to help manage the platform. As an API Producer you should only have more than one Product if:

- You have multiple access modes (e.g. User Restricted and Application Restricted)
- You can map a specific on-boarding SCAL to a specific Product - this makes it easy to see if a specific API Consumer should (or should not) be allowed to use a specific API with a specific access mode
- Where there are multiple OAS files (e.g. one for STU3 and one for FHIR R4)

Read more about access modes and products here: Access modes (see page 299).

Reasons to do more complex authorisation

For the bulk of APIs and to help with compliance management - Products are ideal, they are very visible in the existing Apigee UI. However, overtime, some APIs and their consumers will need further levels of granularity - **where API Consumers will / could be granted compliance exceptions for specific applications** then an API Producer might want to use this model: Configuring extended attributes for API Consumer Apps (see page 806)

Scopes

Scope schema

For the identity service and proxies to validate scopes and allow for specific scopes, we have set a standard way of formatting each scope.

Each scope defined is based on the authentication mechanism used to authenticate, whether that be through application-restricted or user-restricted authentication.

Scope Schema
`urn:nhsd:apim:<identity provider | app restricted>:<security level>:<api name>`

- <identity provider | app restricted>:
  - **user-nhs-id** *(this identifier was created prior to the name change to NHS CIS2)*
  - **user-nhs-login**
  - **app**
- <security level>:
  - For NHS CIS2 user-restricted:
    - aal1
    - aal2
    - aal3
  - For NHS login user-restricted:
    - P0, P5 or P9
  - For app-restricted:
    - level0 (API Key)
    - level1 (OAuth - API key & API secret)
    - level2
    - level3 (Private Key JWT)
- <api name> = the name of your api in kebab case
  - e.g. 'personal-demographics'

Examples:

- urn:nhsd:apim:user-nhs-id:aal3:personal-demographics

- urn:nhsd:apim:user:level3:personal-demographics

> ⚠ If you are using NHS login - there is an assurance focus on ensuring P9 (or P5) has been achieved for the user. However, this is the **identity** assurance level, not the **authentication** assurance level. Technically, within APIM, we should be looking at the Vector of Trust values and Context class values. If your backend services needs to process this, it should either add some logic in the Apigee proxy to process the VoT claim, or pass the VoT claim to your backend.

Configuring products to include scopes

To be able to validate the requested scopes when calling the proxy we must define what scopes are allowed for the specific proxy. The scopes assigned to a particular access token are handled and assigned within the identity service. *(For more detail on how scopes are handled within the identity service:* Authorization Server Scope Handling[828]*)*

The way in which we assign scopes to the product is through defining them on our API's manifest file (`manifest_template.yml` or `manifest.yml`). Please refer to the manifest.yml guidance (see page 896)for further info on this file.

For example, we can add the scopes **urn:nhsd:apim:app:level3:example-api** and **urn:nhsd:apim:user-nhs-login:P9:example-api** like so:

Example Product Definition (Including Scopes) - manifest_template.yml

```
products:
  - name: example-api
    approvalType: auto
    attributes:
      - name: access
        value: public
      - name: ratelimit
        value: 300pm
    description: An example api
    displayName: Example API
    environments: [ {{ ENV.name }} ]
    proxies:
      - {{ SERVICE_NAME }}-{{ ENV.name }}
      - identity-service-{{ ENV.name }}
    scopes:
      - 'urn:nhsd:apim:app:level3:example-api'
```

828 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/178946705/Authorization+Server+Scope+Handling

```
    - 'urn:nhsd:apim:user-nhs-login:P9:example-api'
  quota: {{ ENV.quota | default('300') }}
  quotaInterval: '1'
  quotaTimeUnit: minute
```

When defined in either your **manifest.yml** or **manifest_template.yml** it will then deploy a new version of the product with the specified scopes assigned to it.

Lower security APIs

When an Apigee Access Token is issued the filtering of scopes logic is a simple as possible. The scope urn format includes the "<security level>" component, and the Access Token filter **simply matches** it to the claim in the ID token (depending on which Identity Provider was used).

Whilst this is a simple rule it does allow an API Producer to configure their Products to cope with quite a number of scenarios. For instance, it would be normal for a P5 protected API to also allow a P9 authentication (which is more secure) - to do this the product must be configured in this way:

manifest.yml

```
scopes:
  - 'urn:nhsd:apim:user-nhs-login:P5:example-api-a'
  - 'urn:nhsd:apim:user-nhs-login:P9:example-api-a'
```

For more detailed examples and working through the detailed logic refer to Authorization Server Scope Handling[829].

VerifyAccessToken policy

Adding VerifyAccessToken policy to your proxy

See documentation on the Apigee OAuth2 policy: https://docs.apigee.com/api-platform/reference/policies/oauthv2-policy

To secure your proxy and the flows within it, you need to add the VerifyAccessToken policy to your proxy. This validates any incoming access tokens to determine whether the calling client is authorised to access your API.

The simplest implementation of this is a **OAuth2.VerifyAccessToken.xml** Policy which validates access tokens.

OAuth2.VerifyAccessToken.xml

```
<OAuthV2 async="false" continueOnError="false" enabled="true" name="OAuthV2.VerifyAcc
essToken">
```

829 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/178946705/Authorization+Server+Scope+Handling

```
  <Operation>VerifyAccessToken</Operation>
</OAuthV2>
```

Policy Scope Default

By default the OAuth2 Policy allows for all scopes to be accepted when validating the provided access token.

Once the policy has been added to the proxy, we can then enforce within flows. For example:

default.xml

```
<PreFlow>
        <Request>
            <Step>
                <Name>OauthV2.VerifyAccessToken</Name>
            </Step>
        </Request>
</PreFlow>
```

Configuring the VerifyAccessToken policy to restrict access

Once your verify access token policy has been added to your API proxy you can then add some configuration to that policy to define what access levels can pass that token check.

WARNING

Important warnings about using VerifyAccessToken:

- Due to a limitation in Apigee - you **must replicate** the scopes in your Product definition in your use of VerifyAccessToken.
    - Where your API is used on it's own you won't notice the issue, but when API Consumers incorporate multiple APIs into their App there can be unforeseen implications if you don't follow these rules
    - The knock on impact, is that if you have to change access models, you have to change this list as well
- The configured scopes within the policy must exist within your configured product. If not, access will not be granted.

Apigee documentation sets out that you must specify what scopes are allowed within the **VerifyAccessToken** policy to enforce that the scopes specified are verified. Otherwise, if a scope is assigned to the access token which isn't permitted exists within the 'master' list of scopes assigned to the application and no scopes are specified, then access will be granted. *For more details on this behaviour:* https://docs.apigee.com/api-platform/security/oauth/working-scopes.

To specify what scopes are permitted, add a `<Scope>` tag within the policy containing a space separated list of permitted scopes. *For more details:* https://docs.apigee.com/api-platform/reference/policies/oauthv2-policy#scopeelement

For example, to specify both app and user restricted scopes:

OAuth2.VerifyAccessToken.xml

```
<OAuthV2 async="false" continueOnError="false" enabled="true" name="OauthV2.VerifyAcc
essToken">
  <Operation>VerifyAccessToken</Operation>
  <Scope>urn:nhsd:apim:app:level3:example-api urn:nhsd:apim:user-nhs-id:aal3:example-
api</Scope>
</OAuthV2>
```

In this case only scopes **urn:nhsd:apim:app:level3:example-api** and **urn:nhsd:apim:user-nhs-id:aal3:example-api** assigned to the token will be granted access.

Configuring multiple flows to allow for different access

In some scenarios you might want to restrict access on a per flow basis. For example, if you only require application-restricted authentication on an '/app' endpoint and only user-restricted authentication on a '/user' endpoint.

To be able to do this you must create separate policies for each scenario you would like to restrict, then create separate proxy flows that enforce the appropriate policy.

For example, given the product is defined as such:

manifest.yml

```
scopes:
  - 'urn:nhsd:apim:app:level3:example-api'
  - 'urn:nhsd:apim:user-nhs-id:aal3:example-api'
```

For the '**/app**' endpoint:

We would define the policy:

OAuth2.VerifyAccessTokenAppRestricted.xml

```
<OAuthV2 async="false" continueOnError="false" enabled="true" name="OAuthV2.VerifyAcc
essTokenAppRestricted">
  <Operation>VerifyAccessToken</Operation>
  <Scope>urn:nhsd:apim:app:level3:example-api</Scope>
</OAuthV2>
```

Then we reference that policy within that flow (assuming that the '**/app**' endpoint only accepts GET requests):

default.xml

```
<Flow name="AppEndpoint">
    <Request>
        <Step>
            <Name>OauthV2.VerifyAccessTokenAppRestricted</Name>
        </Step>
    </Request>
    <Condition>(proxy.pathsuffix MatchesPath "/app") and (request.verb = "GET")</
Condition>
</Flow>
```

Then for the '**/user**' endpoint:

We would define the policy:

OAuth2.VerifyAccessTokenUserRestricted.xml

```
<OAuthV2 async="false" continueOnError="false" enabled="true" name="OAuthV2.VerifyAcc
essTokenUserRestricted">
  <Operation>VerifyAccessToken</Operation>
  <Scope>urn:nhsd:apim:user-nhs-id:aal3:example-api</Scope>
</OAuthV2>
```

Then we reference that policy within that flow (assuming that the '**/user**' endpoint accepts GET and PUT requests):

default.xml

```
<Flow name="UserEndpoint">
    <Request>
        <Step>
            <Name>OauthV2.VerifyAccessTokenUserRestricted</Name>
        </Step>
```

```
    </Request>
    <Condition>(proxy.pathsuffix MatchesPath "/user") and ((request.verb = "GET") or
(request.verb = "PUT"))</Condition>
</Flow>
```

Accessing the ID token within your API

Once you have been authenticated and granted a verified access token, you can then access the id token assigned to that access token.

This id token is automatically assigned to the flow variable `accesstoken.id_token` within Apigee once the **OAuth2.VerifyAccessToken** policy has completed.

You can then reference this variable within different policies to access the id tokens value.

For example, you could add the value as header or go on to decode it and verify some of the claims / headers.

7.14.8.6.9.8 Authentication policies and proxy flows

Key concepts

- Apigee and API Products: https://docs.apigee.com/api-platform/publish/what-api-product
- API Producer zone other pages - under Building an API:
  - Securing your API (see page 774)
  - Security patterns for API consumer access (see page 823)
- NHS Digital website: Security and Authorisation Documentation for External Developers[830]

Implementation

> Tip: To accommodate non-authorised responses (such as the ping and status endpoints) the auth-checking policy would typically go in the in the Target Endpoint PreFlow, but it may be appropriate to put it elsewhere depending on your use case.

Application-restricted APIs

API key

Scenario: API Key is being sent in header `apikey`.

Create a policy to check the API Key:

830 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation

policies/VerifyApiKey.FromHeader.xml

```
<VerifyAPIKey async="false" continueOnError="false" enabled="true" name="VerifyApiKey
.FromHeader">
    <APIKey ref="request.header.apikey"/>
</VerifyAPIKey>
```

Invoke that policy in the Target Endpoint's PreFlow:

targets/example_target.xml

```
<TargetEndpoint name="example_target">
    <PreFlow>
        <Request>
            ...
            <Step>
                <Name>VerifyApiKey.FromHeader</Name>
            </Step>
            ...
        </Request>
        ...
    </PreFlow>
    ...
</TargetEndpoint>
```

API Consumers create (or API Producers for test purposes) an Apigee Applications in the developer portal (https://portal.developer.nhs.uk/), which is where they obtain their API Keys. There is no need to use the secret associated with the API Key.

You **must** configure your API Proxy to accept all OAuth access tokens. Many of the lower security APIs will be used in conjunction with more sensitive APIs. You should **not** force API Consumers to switch between OAuth and API Key requests.
See APM-2892[831] - API Key based APIs need to accept any valid access tokens TO DO for details on how to do this.

OAuth: System authentication with private-key JWT

831 https://nhsd-jira.digital.nhs.uk/browse/APM-2892?src=confmacro

For an implementer's perspective and technical details, see: Case study - implementing private key JWT application-restricted access (see page 838)

User-restricted APIs

Overview

All the different OAuth methods have some common functionality that needs to be implemented. Verifying your access token is one one them. In Apigee there is 'OAuthV2' policy which can be used to do the access token verification.

Create a policy to check the access token:

policies/OauthV2.VerifyAccessToken.xml

```
<OAuthV2 async="false" continueOnError="false" enabled="true" name="OauthV2.VerifyAcc
essToken">
  <Operation>VerifyAccessToken</Operation>
</OAuthV2>
```

Invoke that policy in the Target Endpoint's PreFlow:

targers/example_target.xml

```
<TargetEndpoint name="example_target">
    <PreFlow>
        <Request>
            ...
            <Step>
                <Name>OauthV2.VerifyAccessToken</Name>
            </Step>
            ...
        </Request>
        ...
    </PreFlow>
    ...
</TargetEndpoint>
```

OAuth2 Access token is being sent in the `Authorizaton` header using the "`Bearer <token>`" scheme/format. See: https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis

OAuth: User-restricted RESTful API - using NHS CIS2 combined authentication and authorisation

User-restricted RESTful API - using NHS CIS2 - separate authentication and authorisation

For an implementer's perspective and technical details, see: Creating an ID Token to test Token Exchange - API Management - CDT Confluence (digital.nhs.uk)[832]

User-restricted RESTful API - using NHS login - using NHS identity combined authentication and authorisation

User-restricted RESTful API - using NHS login - separate authentication and authorisation

For an implementer's perspective and technical details, see: Creating an ID Token to test Token Exchange - API Management - CDT Confluence (digital.nhs.uk)[833]

Case study - implementing private key JWT application-restricted access

Overview

This documents the implementation of "application-restricted" access to the **Personal Demographics Service API** from a developer perspective.

We detail two components which are logically related but technically distinct:

- OAuth2 *Client Credentials* Grant Type A new grant type, intended to be used in machine-to-machine (M2M) aka unattended aka "application-restricted" scenarios - where a user is not present.
- More granular permissions at the API Proxy level Using implicit *scopes* set at the API Product level to implement more granular security during request processing by the API Proxy.

OAuth2 Client Credentials Grant Type

Usage (Application-Developer Perspective)

A brief summary of public-facing application developer guidance[834] on usage of this OAuth flow.

Setup

The developer's *Application* **must be set up to use this flow**; it will not work "out of the box" like the authorization code flow.

- A **public/private key pair** specific to the application must be created.
    - It is uniquely identified (in the context of their application) by a *key identifier* or "KID" (more on this later)

832 https://nhsd-confluence.digital.nhs.uk/display/APM/Creating+an+ID+Token+to+test+Token+Exchange
833 https://nhsd-confluence.digital.nhs.uk/display/APM/Creating+an+ID+Token+to+test+Token+Exchange
834 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication

- The public key is made available as JSON Web Key Set (JWKS) at an internet-accessible URL. The JWKS contains one or more public keys, with their corresponding KID. (Typically they send this to us and we create and host the JWKS. See KOP-014[835] for the specifics.)
- The URI of their application's JWKS is stored as an attribute on their application called `jwks-resource-url` (This is another manual action performed by us.)

Requesting an Access Token

The Client Credentials grant type is conceptually simple, requiring a single call to the `/token` endpoints of the Identity Service. (No callback with authorization code and subsequent code-for-token exchange as with the authorization code flow.)

1. Create and sign a JWT

   A JWT must be created with the following claims:

   - `iss` (issuer) and `sub` (subject): both set to the *application key*
   - `jti` (JWT token identifier): some unique (in the context of JWTs used by this application) value. (We recommend a GUID.)
   - `exp` (expiry): UNIX timestamp, no more than 5 minutes in the future.
   - `aud` (audience): the full URI of the token endpoint

   And the header:

   - `typ` (token type): `JWT` (standard for all JWTs)
   - `alg` (signing algorithm): `RS512` (our only supported signing algorithm)
   - `kid` : set to the appropriate Key Identifier (see above) for the application, available in the JWKS (see above)

   The token is signed with the appropriate private key.

2. Call the `/token` endpoint. A HTTP POST request is made to the token endpoint. The content type is `application/x-www-form-urlencoded` , and the formdata contains:

   - `grant_type` (the operation being attempted): `client_credentials` (a client credentials grant)
   - `client_assertion_type` (how the client is asserting its identity): `urn:ietf:params:oauth:client-assertion-type:jwt-bearer` (with a JWT token)
   - `client_assertion` : the JWT

3. Receive the access token in the JSON response body. Example:

---

835 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?spaceKey=APM&title=KOP-014+Creating+a+JSON+Web+Key+Set+%28JWKS%29+for+Application-Restricted+Access

```
{'access_token': 'Sr5PGv19wTEHJdDr2wx2f7IGd0cw',
 'expires_in': '599',
 'token_type': 'Bearer'}
```

Implementation

The Client Credentials grant handling is implemented as a *conditional flow* named `Flow.PostTokenClientCredentials` in the Identity Service[836]'s `ProxyEndpoint` (in the `proxies/live/apiproxy/proxies/default.xml` file). The flow is executed when a POST request is made to the `/token` path, with a form param `grant_type` is present and set to `client_credentials`.

A walkthrough of the key elements of the flow:

1. Validation of `client_assertion_type` form param
2. Decoding of the JWT using the DecodeJWT[837] policy. This makes the claims of the JWT available as flow variables, in the namespace `jwt.DecodeJWT.FromClientAssertionFormParam.decoded.claim.<claim>` and `jwt.DecodeJWT.FromClientAssertionFormParam.decoded.header.<header>`. (Note that JWTs are *encoded* but not *encrypted* - we can decode the JWT without needing any keys, which would not be available at this stage of processing.)
3. Validation of the JWT contents; check that all stipulated claims and headers are present
4. Check the JWT expiry is not more than 5 minutes in the future. (We allow an additional ten seconds on this check to mitigate edge cases caused by clock skew.)
5. Determine whether we have seen the JTI before by consulting the an application-scoped cache, where the JTI and Application Key (taken from the JWT's `jti` and `sub` claims, respectively) are used as key fragments.
6. Cache the JTI to prevent re-use.
7. Verify the 'API Key' (aka Client ID) - again taken from the `sub` claim of the JWT. This accesses the developer's App and makes attributes of it available in the namespace `verifyapikey.VerifyAPIKey.FromJWT.<attribute>`
8. The `Javascript.SetServiceCalloutUrl` policy takes the `jwks-resource-url` attribute of the app and uses it to set the target URL and Path of the subsequent policy, `ServiceCallout.ClientCredentialsGrantGetJWKS`. This retrieves the app's JWKS (JSON Web Keyset) containing the public key(s) used to verify the signature of their JWT
9. With the App's JWKS retrieved we are in a position to verify the signature of the JWT with the `VerifyJWT.ClientCredentials` policy. It also validates the `aud` and `exp` claims. The KID is read from the header of the JWT and a corresponding public key is sought in the JWKS from the prior step.

836 https://github.com/NHSDigital/identity-service-api
837 https://docs.apigee.com/api-platform/reference/policies/decode-jwt-policy

10. The `AssignMessage.SetClientCredentials` policy takes the client ID from JWT `sub` claim, and the client secret from the app (see 7) and presents them as if they were submitted as form params. This is necessary for the subsequent generate access token operation.
11. The conditional `AssignMessage.AccessTokenExpiryOverride` policy is used to reduce the access token expiry time by sending an optional formparam `_access_token_expiry_ms`. This is not publicly documented and is used internally for testing purposes.
12. The `OAuthV2.ClientCredentialsGenerateAccessToken` policy generates the access token, but does not send it
13. The `AssignMessage.ClientCredentialsGenerateTokenResponse` is used to customise the token response. It sends a more standards-compliant token format, with many extraneous details removed.

Note that the refresh token is *not* sent to the client, as we don't support that functionality when this flow type is used. This is because it is simpler for the client application to request a new token when it expires, and doesn't inconvenience a user in the same way that having them re-login every time their access token has expired.

Application- / User-Restricted Permissions Handling

The implementation of different access levels happens at the *API Product* level. We leverage 'scopes', but in a manner entirely opaque to client application developers.

We implement the restrictions on functionality in the API Proxy. (As opposed to passing permission details on to the backend system and leaving it responsible for denying/allowing requests based on this information.) Where possible this is implemented using simple Apigee conditional policies. More complex scenarios may require using Java or JavaScript policies to implement the business logic.

Note that the method of obtaining the token (client credentials grant with a JWT client assertion / authorization code flow with a user providing credentials) has no bearing on the permissions model.

Apigee Implicit Scopes Behaviour

API Products have an 'Allowed OAuth Scope' configuration. When an OAuth access token is generated but scopes have not been explicitly requested, the token will carry **all** the scope(s) from the API Products to which the corresponding Application is subscribed.

In our developer-facing documentation we do not list these scopes, nor do we ask developers to specify them explicitly when requesting access tokens.

Setting Up Multiple API Products per API

Optional functionality is available, in the common pipelines, to define multiple Apigee Products for your API. This is really important when you have more than one 'access mode', there must be a one to one mapping to with your Apigee Products - read more: securing your api with access modes[838]

838 https://nhsd-confluence.digital.nhs.uk/display/APM/Privacy+-+compliance+-+security+and+simplifying+running+your+API

You can use this by creating a `products/` directory in your API's repository and adding YAML files defining the products.

The contents of these files are combined with the `default_product` fact in the `ansible/roles/deploy-apigee-product-and-spec/tasks/main.yml` file of the api-management-infrastructure repository. (You do not have to define product attributes that don't differ between products.)

For an example of this in action, see the `products/application-restricted.yml` [839] and `products/user-restricted.yml` [840] files in the personal-demographics-service-api repository. Because of these files, two distinct API Products are made available for developers to subscribe to. One carries a *scope*, the other does not.

Creating (or not) and Naming Scopes

In the PDS example, there are two products, one has a scope, the other does not:

API Product	Scope
PDS: User Restricted	`personal-demographics-service:USER-RESTRICTED`
PDS: Application Restricted	(No scope)

The scope checking (below) process follows a least-privilege model, so the lowest privilege API Product (in our case Application-Restricted offers purely *reduced* functionality from User-Restricted) can have no scope.

A given application may receive scopes from a number of API products, so they should be named in a way that avoids collisions - hence they are prefixed with the API's short 'service name'.

API Proxy Permissions Implementation

Checking for scopes and implementing restrictions for different API Products happens at the API Proxy level. We have established a pattern that splits this process into two stages:

Scope Checking

Your OAuth-secured API Proxy will have an `OAuthV2/VerifyAccessToken` policy at the beginning of the request processing flow. Once this has executed a *flow variable* `scope` is made available. This contains a list of scopes carried by the access token.

We will set a flow variable `access-mode` to some meaningful value that can later be used to implement our restrictions.

---

839 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/products/application-restricted.yml
840 https://github.com/NHSDigital/personal-demographics-service-api/blob/master/products/user-restricted.yml

In `proxies/live/apiproxy/targets/ig3.xml`:

```
<TargetEndpoint name="ig3">
  <PreFlow>
    <Request>
      <Step>
        <Name>OauthV2.VerifyAccessToken</Name>
      </Step>
      <!-- Set flow variable 'access-mode' by inspecting the token's scopes.
      First set to least-privilege (application restricted) then look for scopes
indicating higher privilege, overwriting as appropriate.
      'scope' is a space-separated string of scopes, determined by the Application's API
Product assignments.
      Follow pattern used in AssignMessage.SetAccessModeUserRestricted condition. See
Apigee docs for JavaRegex gotchas (uses '.match()' behaviour matching whole string only).
-->
      <Step>
        <Name>AssignMessage.SetAccessModeApplicationRestricted</Name>
      </Step>
      <Step>
        <Name>AssignMessage.SetAccessModeUserRestricted</Name>
        <Condition>(scope JavaRegex "(.+\ personal-demographics-service:USER-
RESTRICTED\ .+|^personal-demographics-service:USER-RESTRICTED\ .+|.+\ personal-
demographics-service:USER-RESTRICTED$|^personal-demographics-service:USER-
RESTRICTED$)")</Condition>
      </Step>
            ...
```

1. We use the unconditional policy `AssignMessage.SetAccessModeApplicationRestricted` to set the variable to `application-restricted` - the least-privileged of our access modes in this context.
2. Next, the policy `AssignMessage.SetAccessModeUserRestricted` will set it to `user-restricted` **if the correct scope is found**.
3. You could continue to check for other, more highly-privileged modes of access in the same manner

Implementing Restrictions

Between the `<PreFlow>` and `<PostFlow>` elements, are zero or more *Conditional Flows*. We group our access-mode-specific behaviour in conditional flow(s), as required.

From the PDS `proxies/live/apiproxy/targets/ig3.xml`:

```
...
  </PreFlow>
  <Flows>
    <Flow name="application_restricted">
      <Condition>(access-mode Equals "application-restricted")</Condition>
      <Request>
        <!-- Prevent use of PATCH method in application restricted mode -->
        <Step>
          <Name>RaiseFault.PatchMethodNotAllowedAppRestricted</Name>
          <Condition>(request.verb Equals "PATCH")</Condition>
        </Step>
        <!-- If _max-results param is unset, set to allowed value for application-
```

```
restricted mode -->
        <Step>
          <Name>AssignMessage.SetMaxResultsToOne</Name>
          <Condition>(proxy.pathsuffix MatchesPath "/Patient" and request.verb Equals
"GET" and request.queryparam._max-results Is null)</Condition>
        </Step>
        <!-- If _max-results is set to anything other than 1, prevent query -->
        <Step>
          <Name>RaiseFault.MaxResultsInvalidAppRestricted</Name>
          <Condition>(proxy.pathsuffix MatchesPath "/Patient" and request.verb Equals
"GET" and request.queryparam._max-results NotEquals "1")</Condition>
        </Step>
      </Request>
    </Flow>
  </Flows>
...
```

We have a flow `application_restricted` which is conditioned on the `access-mode` variable being set to `application-restricted`.

Each Step within the flow defines some behaviour. For example we use a `RaiseFault` policy to prevent requests attempting to use the PATCH method, and provide an appropriate diagnostic response to the caller.

Caveats

- The method of obtaining the token is entirely independent of the permissions that caller will eventually receive via their access token. Don't be misled into thinking that because you used a client credentials grant you will get permission set X, and authorization code flow will yield permission set Y - this is not the case.
- It is important that developer Apps are subscribed to the correct products. Especially in production, apps should be subscribed to a *single product* for the PDS FHIR API. The above permission model would break down if an app were subscribed to both the user-restricted and application-restricted PDS API Products.

7.14.8.6.10 Validate user roles using shared flows

7.14.8.6.10.1 Overview

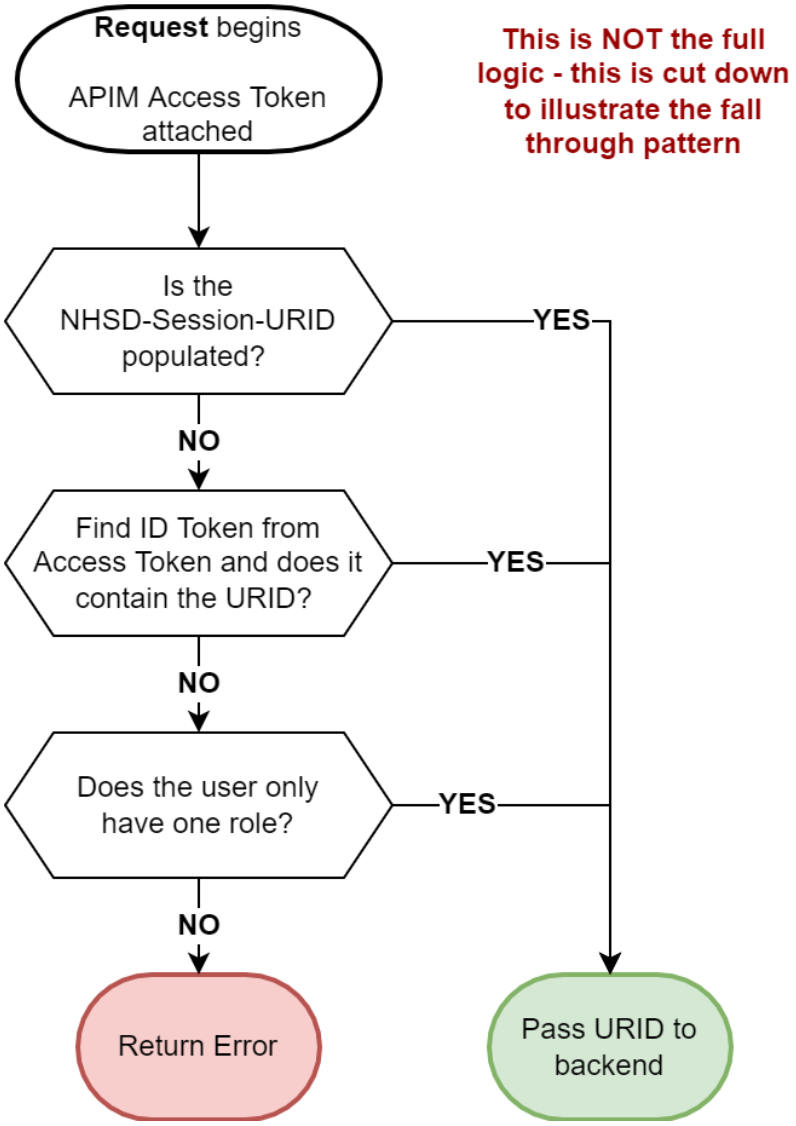
Validating the request user role (which is the role as defined in the National RBAC system) against valid user information is a task that multiple APIs will implement. Because of this, 2 shared flows were created to perform this task and avoid producers having to design and implement a solution by themselves.

These shared flows carry out **two** functional tasks:

- extract the user role id from the multiple sources
- validate that the user role id is actually allocated to the user based on CIS2 data

If your proxy relies on the validation element - it **might** indicate your proxy does not fully support Separate Auth. You need to ensure that your API / Proxy fully supports Separate Auth.

Concept of the logic applied in this shared flow (diagram from another page needs a dedicated version):



7.14.8.6.10.2 How does it work

When using a user restricted API with the combined authentication and authorization[841], a sequence of steps is triggered described in the OAuth broker model (OAuth - Authentication broker model[842]). Then if the authentication and authorization are successful the user is granted an access token.

It is important to understand the difference between an ***access token*** and an ***id token***. In this document we will call "access token" the token issued by Apigee and granted to the user on successful authentication and authorization while the "id token" is a JWT token issued by the NHS CIS2 service and attached into the access token metadata during the authentication process.

In other words the access token that the user receives from the identity service (Apigee) has an id token from the NHS CIS2 service attached as a custom attribute in the metadata of the token (refer to Accessing the id token claims for smartcards - API Management (see page 796)).

The shared flows will read the user restricted token and validate or invalidate the request based on the user-restricted pattern used to authenticate.

CIS2 combined auth

When doing a request, the client can also pass a role id using the "NHSD-Session-URID" header on the request. At this point, they will be two possible values for the user role in the request, one in the token (under the selected_roleid field) and another in the header. The header value will take precedence over the token.

If no role is present in the header or the id token, we can know which role id values are valid for a given user simply calling the /userinfo endpoint within the identity service using the same access token. When doing so we will get a response with information about the user holding the token, in particular, the roles this user is allowed to take under the "person_roleid" field (see image below).

/userinfo response


```
{
    "nhsid_useruid": "910000000001",
    "name": "USERQ RANDOM Mr",
    "nhsid_nrbac_roles": [
        {
            "activities": [
                "Perform Prescription Preparation",
                "View Patient Medication",
                "Amend Patient Demographics",
                "Nurse Prescribers Formulary (NPF) Prescribing"
            ],
            "activity_codes": [
                "B0278",
                "B0401",
                "B0825",
```


841 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation

842 https://nhsd-confluence.digital.nhs.uk/display/APM/OAuth+-+Authentication+broker+model

```
                "B0058"
            ],
            "org_code": "RBA",
            "person_orgid": "555254239107",
            "person_roleid": "555254240100",
            "role_code": "S8000:G8000:R8001",
            "role_name": "\"Clinical\":\"Clinical Provision\":\"Nurse Access Role\""
        },
        {
            "activities": [
                "Manage Workgroups",
                "Independent Prescribing",
                "Execute CDS Extracts (NHS Group Pseud. Data)",
                "Manage Workgroup Membership"
            ],
            "activity_codes": [
                "B0100",
                "B0420",
                "B1510",
                "B0090"
            ],
            "org_code": "RBA",
            "person_orgid": "555254239107",
            "person_roleid": "555254242102",
            "role_code": "S8000:G8000:R8000",
            "role_name": "\"Clinical\":\"Clinical Provision\":\"Clinical Practitioner
Access Role\""
        },
        {
            "activities": [
                "Perform Pharmacy Activities",
                "Verify Prescription",
                "Manage Pharmacy Activities",
                "View Patient Medication"
            ],
            "activity_codes": [
                "B0570",
                "B0068",
                "B0572",
                "B0401"
            ],
            "org_code": "RBA",
            "person_orgid": "555254239107",
            "person_roleid": "555254241101",
            "role_code": "S8000:G8000:R8003",
            "role_name": "\"Clinical\":\"Clinical Provision\":\"Health Professional
Access Role\""
        }
    ],
    "sub": "910000000001"
}
```

Currently Spine allows for the user to change role during any one session. So, a mechanism needs to be in place in the API proxy, to validate that the user role presented in the request corresponds to one of the valid "person_roleid" values presented in the user information get it from the token. This validation is possible using policies in Apigee.

CIS2 separate auth

When using a user restricted API with the separate authentication and authorization[843], userinfo is not available from the APIM authorisation service. Here the role must be provided either in the header or in the selected_roleid field in the id token.

NHS login

While NHS login is a user restricted auth pattern it is used for patient access not healthcare access. In this case, the shared flows are not valid to use as we are not validating user roles.

7.14.8.6.10.3 The UserRoleService shared flow

In order to achieve the validation needed by the different API proxies, a shared flow was created in Apigee to perform this task. This shared flow can be consumed in a simple way by any proxy that needs this functionality as we will see in the following section.

The following figure shows the steps that make up the shared flow and the following table explains each of the policy's functionality (see figure above).

Step	Policy	Function
1	ExtractVariable.Token	Gets the access token from the header and saves it in a context variable used in the following step.
2	GetOAuthv2Info.TokenInfo	Gets information from the access token, in particular the id_token information that is encoded as a custom attribute within the access token.
3	DecodeJWT.id_token	Decode the id_token and populates several context variables with claims and attributes from the id_token.
4	LookupCache.UserInfo	Look at the cache looking for user information related to the incoming access token.

843 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

Step	Policy	Function
5	ServiceCallout.GetUserInfo	If there is no information in the cache, then a service call-out is made to the /userinfo endpoint in the identity service to fetch user information about the user.
6	PopulateCache.UserInfo	If the service call-out was performed, then save the user info in the cache using the token as a key for future requests.
7	AssignMessage.UserInfo	Saves the content of the service call-out response in a variable for future usage.
8	JavaScript.UserRoleValidation	Implement the logic and the validation comparing the default user role (if present in the id_token) or the user role passed by the user (in the NHSD-Session-URID header) and the valid roles present in the user information.
9	RaiseFault.InvalidUserRole	Triggers the error flow within the proxy and populates flow variables variables with error information.
10	AssignMessage.SetHeaderWithValidRole	Set the outcoming valid role in the NHSD-Session-URID request header.

How to consume the UserRoleService shared flow

In order to consume the shared flow you should use the Apigee FlowCallout policy (FlowCallout policy | Apigee Docs[844]). To do this, you will need to create the policy in your proxy and point it to the right shared flow. Then you need to consume the policy as needed.

You can always use the following code example for the FlowCallout.

FlowCallout.UserRoleService.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.UserRoleService">
    <DisplayName>FlowCallout.UserRoleService</DisplayName>
    <FaultRules/>
    <Properties/>
    <SharedFlowBundle>UserRoleService</SharedFlowBundle>
```

844 https://docs.apigee.com/api-platform/reference/policies/flow-callout-policy

```
</FlowCallout>
```

Then you can consume the shared flow as needed.

Step example

```
<Step>
    <Name>FlowCallout.UserRoleService</Name>
</Step>
```

You can also check implementation and testing in a proxy (here[845]) or test the functionality making a request to https://internal-dev.api.service.nhs.uk/shared-flow-testing and tracing it in Apigee to better understand the behavior of the shared flow.

Error handling

If the user role validation performed by the JavaScript.UserRoleValidation policy is not successful, then the raise fault policy will set up the shared flow and therefore the proxy flow where this is being consumed into an error state.

At this point there are some context variables that the raise fault will populate regarding to the cause of the error leaving the responsibility of formatting the error message to the proxy.

The variables populated during error cases are:

Variable Name	Value
validation.statusCode	400
error.reason.phrase	"Bad Request"
error.content	"nhsd-session-urid is invalid" (this content vary depending on the cause of the error)

When your proxy enters into error state, the error flow is triggered. At this point Apigee first check the Fault Rules defined in your proxy and if there is no condition covering the error scenario, then the default Fault Rule is used. This means that you should define either a Fault Rule to handle and format the error messages coming from the shared flow, or at the very least you should be using a default Fault Rule catching all unexpected error that can occur in your proxy (this policy is added by default when creating a proxy from the template example here[846])

Please feel free to use the following Fault Rule in your proxy if needed (FHIR compliant format) and remember that you can format the error messages as needed by the spec of your proxy.

845 https://github.com/NHSDigital/shared-flow-testing/blob/master/tests/test_user_roles.py
846 https://github.com/NHSDigital/api-management-service-template/blob/master/proxies/live/apiproxy/policies/AssignMessage.Errors.CatchAllMessage.xml

FaultRule example

```
<FaultRule name="UserRoleSharedFlowError">
    <Condition>isSharedFlowError = true</Condition>
        <Step>
            <Name>AssignMessage.SharedFlowError</Name>
        </Step>
</FaultRule>
```

AssignMessage.SharedFlowError

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AssignMessage async="false" continueOnError="false" enabled="true" name="AssignMessa
ge.SharedFlowError">
    <DisplayName>AssignMessage.SharedFlowError</DisplayName>
    <Set>
        <Payload contentType="application/json" variablePrefix="%" variableSuffix="#">
        {
            "resourceType": "OperationOutcome",
            "issue": [
                {
                "severity": "error",
                "code": "%error.status.code#",
                "details": {
                    "coding": [
                    {
                        "code": "%error.status.code#",
                        "display": "%error.reason.phrase#"
                    }
                    ]
                },
                "diagnostics": "%error.content#"
                }
            ]
        }
        </Payload>
        <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
        <AssignTo createNew="false" transport="https" type="request"/>
    </Set>
    <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
    <AssignTo createNew="false" transport="http" type="request"/>
</AssignMessage>
```

Once implemented the policies above you will be able to get the following error descriptions for a given error case:

Status code	Reason phrase	Content
400	Bad Request	selected_roleid is misconfigured/invalid
400	Bad Request	selected_roleid is missing in your token
400	Bad Request	nhsd-session-urid is invalid
400	Bad Request	unable to retrieve user info
400	Bad Request	unable to retrieve userRole var
400	Bad Request	no userroles available, please check nhsd-session-urid is valid
400	Bad Request	multiple roles found in user info, please check nhsd-session-urid
400	Bad Request	nhsid_nrbac_roles is misconfigured/invalid

7.14.8.6.10.4 The UserRoleServiceV2 Shared Flow

The UserRoleServiceV2 shared flow is an updated UserRoleService which returns FHIR error messages from the user role id (URID) validation, and can be configured to return the URID on a different header.

Step	Policy	Function
1	ExtractVariable.Token	Gets the access token from the header and saves it in a context variable used in the following step.
2	GetOAuthv2Info.TokenInfo	Gets information from the access token, in particular the id_token information that is encoded as a custom attribute within the access token.

Step	Policy	Function
3	DecodeJWT.id_token	Decode the id_token and populates several context variables with claims and attributes from the id_token.
4	LookupCache.UserInfo	Look at the cache looking for user information related to the incoming access token.
5	ServiceCallout.GetUserInfo	If there is no information in the cache, then a service call-out is made to the /userinfo endpoint in the identity service to fetch user information about the user.
6	PopulateCache.UserInfo	If the service call-out was performed, then save the user info in the cache using the token as a key for future requests.
7	AssignMessage.UserInfo	Saves the content of the service call-out response in a variable for future usage.
8	JavaScript.UserRoleValidation	Implement the logic and the validation comparing the default user role (if present in the id_token) or the user role passed by the user (in the NHSD-Session-URID header) and the valid roles present in the user information. In UserRoleServiceV2 this is also responsible for generating FHIR error payloads.
9	RaiseFault.InvalidUserRole	Triggers the error flow within the proxy and populates flow variables variables with error information.
10	JavaScript.SetHeaderWithValidRole	Set the outcoming valid role in the header specified by the `user_role_header` parameter, defaulting to `NHSD-Session-URID` if not specified.

How to consume the UserRoleServiceV2 shared flow

In order to consume the shared flow you should use the Apigee FlowCallout policy (FlowCallout policy | Apigee Docs[847]). To do this, you will need to create the policy in your proxy and point it to the right shared flow. Then you need to consume the policy as needed.

You can always use one of the following code examples for the FlowCallout.

847 https://docs.apigee.com/api-platform/reference/policies/flow-callout-policy

FlowCallout.UserRoleService.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.U
serRoleServiceV2">
    <DisplayName>FlowCallout.UserRoleService</DisplayName>
    <FaultRules/>
    <Properties/>
    <SharedFlowBundle>UserRoleServiceV2</SharedFlowBundle>
</FlowCallout>
<!-- Calls the flow with the URID sent in the NHSD-Session-URID header -->
```

FlowCallout.UserRoleService.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.U
serRoleServiceV2">
    <DisplayName>FlowCallout.UserRoleServiceV2</DisplayName>
    <FaultRules/>
    <Properties/>
    <Parameters>
        <Parameter name="user_role_header">HEADER_NAME</Parameter>
    </Parameters>
    <SharedFlowBundle>UserRoleServiceV2</SharedFlowBundle>
</FlowCallout>
<!-- Calls the flow with the URID sent in the HEADER_NAME header -->
```

Then you can consume the shared flow as needed.

Step example

```
<Step>
    <Name>FlowCallout.UserRoleServiceV2</Name>
</Step>
```

You can also check implementation and testing in a proxy (here[848]) or test the functionality making a request to https://internal-dev.api.service.nhs.uk/shared-flow-testing and tracing it in Apigee to better understand the behavior of the shared flow.

Error handling

848 https://github.com/NHSDigital/shared-flow-testing/blob/master/tests/test_user_role_service_v2.py

Error handling in UserRoleServiceV2 is the same as the original flow, with the only differences being the response has the header contentType="application/fhir+json", and the payload being generated in the JavaScript.UserRoleValidation callout. This means an error message does not need to be formatted by a flow's FaultRule callout.

The following errors can be returned from the shared flow:

Status code	Reason phrase	payload.issue[0].details.coding[0].code	payload.issue[0].details.coding[0].display	payload.issue[0].diagnostics
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	selected_roleid is misconfigured/invalid
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	selected_roleid is missing in your token
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	no userroles available, please check NHSD-Session-URID is valid
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	multiple roles found in user info, please check NHSD-Session-URID
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	nhsid_nrbac_roles is misconfigured/invalid
401	Bad Request	UNAUTHORIZED	401: The Server deemed you unauthorized to make this request.	NHSD-Session-URID is invalid
500	Bad Request	SERVER_ERROR	500: The Server has encountered an error processing the request.	unable to retrieve user info

Status code	Reason phrase	payload.issue[0].details.coding[0].code	payload.issue[0].details.coding[0].display	payload.issue[0].diagnostics
500	Bad Request	SERVER_ERROR	500: The Server has encountered an error processing the request.	unable to retrieve userRole var

All errors returned have **payload.resourceType:** "OperationOutcome", **payload.issue[0].severity**: "error", **payload.issue[0].code**: "invalid"

7.14.9 Securing your API with proxy generator

Broken, exposed, or hacked APIs are behind significant data breaches. They can reveal sensitive medical, financial, and personal data for public consumption. That said, not all data is the same nor should be protected in the same way. How you approach API security will depend on what kind of data is being transferred.

Although in the early stages of the alpha delivery phase you are likely not to implement a fully secured API, you will be thinking of our standards for security for your API.

During alpha, you are likely to cover:

- How sensitive is access to my API / service?
- What is OAuth, and how does it work?
- What are the options for API consumers to connect securely?
- Once authenticated, what further authorisation might be applied?
- How do I secure my backend service?
- What other security best-practices should I implement?

When you move into Private Beta (see page 194) you move from design into implementing, which involves:

- Securing the connection to my backend service
- Choosing and applying appropriate security patterns for my API consumers
- Restricting access to my API based on these patterns
- Making your service available to API consumers using API products

7.14.9.1 Security segments

The primary design pattern used is the façade pattern, which can also be thought of as a firebreak between the API consumer and API backend. It is really important to apply this pattern to how you manage security.

Your security must be split into segments, and each one must be loosely coupled to each other. For instance, an access token (itself) should not be used for both the API platform and for the backend.

The security segmentation for most solutions will look like this:

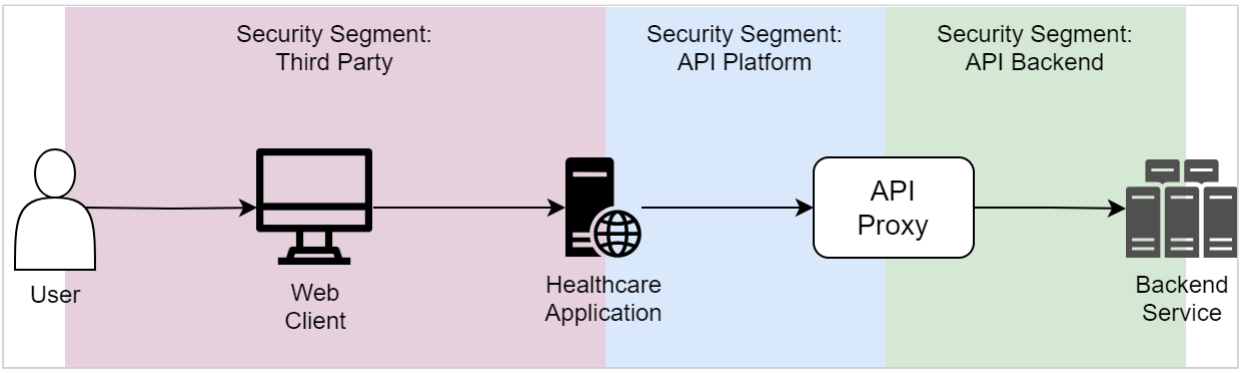


Fig 1: Security segmentation

7.14.9.2 Topics

- API backend security (see page 857)
  - How to implement API key security for your API backend (see page 860)
  - How to implement mutual TLS (mTLS) security for your API backend (see page 865)
- API consumer access security patterns (see page 871)
  - How to configure API consumer security patterns using proxy generator (see page 875)
  - How to configure API products using proxy generator (see page 879)

7.14.9.3 API backend security

This section covers the API backend security segment, the green area of the diagram: securing the connection between the API proxy and backend service, and API-specific security considerations.

Other security best-practices are also discussed, however the overall security posture of your back end service is beyond the scope of this guidance.

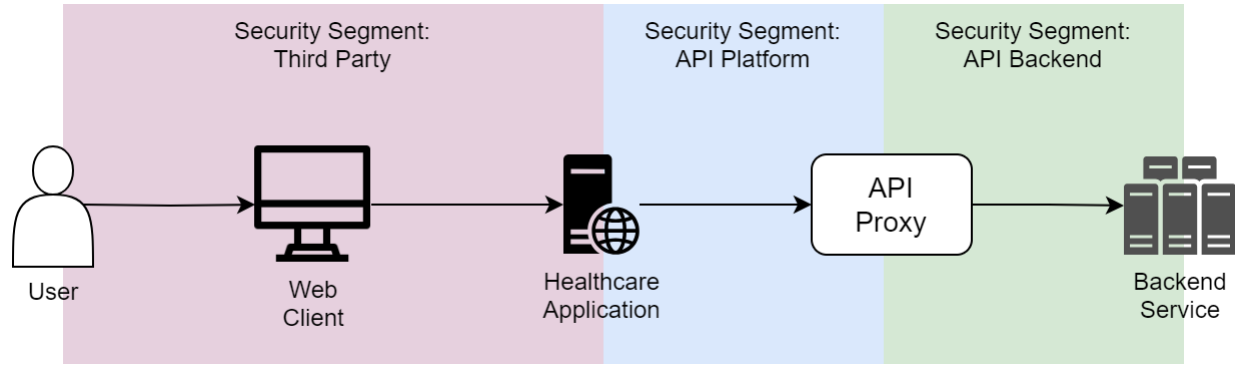


Fig 1: Security segmentation

7.14.9.3.1 Choose the appropriate security method for the API backend

This section only applies to ***external target*** **backends**. If your backend is hosted by the API platform — a *hosted target* — the connection is automatically secured.

Your backend service must be configured to only accept connections from trusted services (in this context: your API proxy.)

> Adhering to the model of loosely-coupled security segments, the methods of authentication (keys, tokens, certificates, etc) are **distinct** from those employed in the API platform security segment. (The ones used by API consumers.)

The API platform supports two methods of authenticating itself to the API backend:

7.14.9.3.1.1 Mutual TLS (mTLS)

Both the API proxy and backend service present TLS keys to one another. Trust is established because the keys are signed by the same certificate authority (CA).

- This is the more secure method
- Zero-downtime rotation of credentials
- More difficult to implement

How to implement mutual TLS (mTLS) security for your API backend (see page 865)

7.14.9.3.1.2 API Key

Trust is established via a 'shared secret' value known to both the API proxy and backend service. The API key is sent in a header with each request from the API proxy.

- This is the less secure method
- Loss of service during credential rotation
- Easy to implement

How to implement API key security for your API backend (see page 860)

The rule of thumb is that the backend service should be secured to the same (or greater) level than to the API proxy:

- APIs that expose patient data should secure their backend service using mTLS
- API key security is only appropriate for APIs using application-restricted API key authentication[849] as the security mechanism for API consumers (the API platform security segment)

849 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication

7.14.9.3.1.3 Source IP Restrictions

You **may** choose to combine either security method with source IP restrictions as an additional layer of security. See Static IPs - Apigee to API Backends (see page 769) for details of the source IPs currently used by the API platform.

However, we recommend **not** doing this, carefully consider the need for source IP restrictions as they make future infrastructure changes more difficult.

7.14.9.3.2 Choosing a DNS name for your API backend

The API platform will dictate the consumer (client) facing address; the DNS name for your back end service is typically a hidden implementation detail.

Where you are using a service provider such as AWS or Azure, it is usually okay to proceed with the automatically assigned DNS name.

Assigning a custom (.nhs.uk[850]) domain name to your back end service may be necessary where:

- Clients will also access it directly (not through the API platform) and would benefit from seeing a trusted name
- Your security posture would benefit from having the DNS managed by nhs.uk[851] to prevent DNS takeover attacks
- It is technically necessary. One example is where AWS API Gateway requires the use of custom domain names to support mTLS[852]

7.14.9.3.3 Additional API backend security considerations

The overall security of your backend service is your team's responsibility, and beyond the scope of this guidance. This list is not exhaustive.

7.14.9.3.3.1 Automatic detection of supply chain vulnerabilities

All teams should be following Products & Platforms best practices - as set out in: https://github.com/NHSDigital/software-engineering-quality-framework.

850 http://nhs.uk
851 http://nhs.uk
852 https://docs.aws.amazon.com/apigateway/latest/developerguide/rest-api-mutual-tls.html

This includes configuring Dependabot[853] or a similar service to automatically detect and mitigate supply chain vulnerabilities - this looks at the libraries your service depends on that are found to be vulnerable.

7.14.9.3.3.2 OWASP API Security Top 10

Review the OWASP API Security Top 10[854] guidance. Determine where it is applicable to your service and apply mitigations as advised.

7.14.9.3.4 How-to guides

- How to implement API key security for your API backend (see page 860)
- How to implement mutual TLS (mTLS) security for your API backend (see page 865)

7.14.9.3.5 How to implement API key security for your API backend

7.14.9.3.5.1 Overview

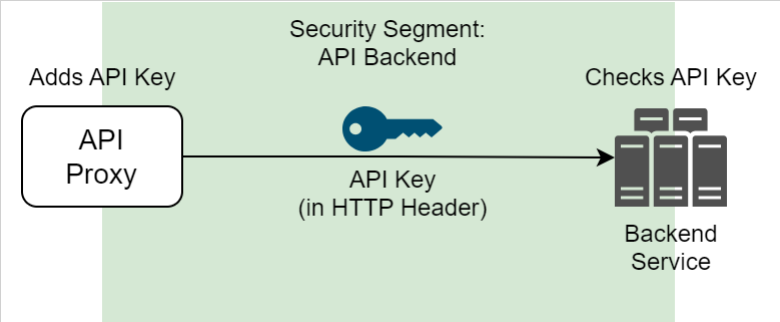


Fig 1: Security segment - API background

Authentication of the API Proxy is achieved by sending a secret value — the API Key — as an HTTP header in every request to the backend service.

The backend service is configured to reject requests not having the correct API Key.

The API Key is a fixed (string) value. It is shared between your API Proxy and the Backend service. This type of security scheme is known as a shared secret.

> ⚠ API Key security is the weaker of the two backend security mechanisms supported by Proxy Generator.

853 https://github.com/dependabot
854 https://owasp.org/API-Security/

It is only appropriate for APIs using application-restricted API key authentication[855] as the security mechanism for API consumers (the API platform security segment).
Unless your back-end supports multiple API keys, key rotation will incur downtime.
Ensure you have read the Choose the appropriate security method for your API backend (see page 858) guidance before continuing to implement API key security.

For a full user guide on how to manage your API's with proxygen-cli, see Proxygen CLI user guide (see page 933).

7.14.9.3.5.2 Implementing API key security

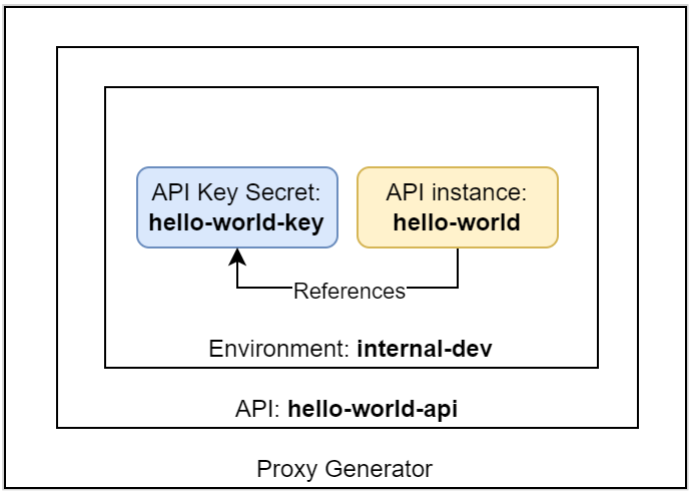


Fig 2: Components of API Key security in the context of Proxy Generator

A facility of Proxy Generator is the storage of **secret values**, such as API keys. These secrets are used in other managed components by referencing them by their name. In this case an API instance is configured to use a named secret as an API key, attaching the secret value to each request to the back end service in a specific HTTP header.

Both secrets and API instances are scoped to particular environments. An environment can contain multiple secrets, each referenced by zero, one, or multiple API instances.

1. Store the API key secret value

Use proxygen-cli to create a secret, specifying the value on the command line:

```
proxygen secret put --apikey --secret-value foo internal-dev hello-world-key
```

855 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication

or reading it from a file on disk:

```
proxygen secret put --apikey --secret-file secret_file internal-dev hello-world-key
```

The format of the command is:

proxygen secret put --apikey [--secret-value <value>|--secret-file <path-to-file>] <environment> <secret-name>

Alternatively call the Proxy Generator API using the PUT /apis/{api_name}/environments/{environment}/secrets/{secret_name}[856] operation with the type=apikey query parameter.

Send the secret value in the request body.

2. Configure your API instance for API key security

API Instance specification file (partial)

```
x-nhsd-apim:
  target:
    type: external
    healthcheck: /_status
    url: https://api.backend.example.com/hello-world
    security:
      type: apikey
      header: nhsd-apim-apikey
      secret: hello-world-key
```

In the `x-nhsd-apim/target/security` section setting `type` to `apikey` enables API key security.

Two other properties are required:

- `header` : the name of the HTTP header used to transmit the API key to the backend service
- `secret` : the name of an API key secret (created in step 1). The secret must exist in the environment you will be deploying your API instance to.

3. Deploy your API instance

Use proxygen-cli to deploy your API instance:

```
proxygen instance deploy --no-confirm internal-dev hello-world oas-specification.yaml
```

See deploying your API with proxy generator (see page 698) for further information about API instance deployment, including how to use the proxy generator API.

Note that the API instance is deployed to the same environment as the secret.

856 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/environments/-environment-/secrets/-secret_name-

Example result

The example above would result in:

- An API Key secret called `hello-world-key` stored in the `internal-dev` environment, having a value of `foo`
- An API instance deployed to the `internal-dev` environment with a base path of `hello-world` referencing this secret
- Requests to the API instance (at `https://internal-dev.api.service.nhs.uk/hello-world`) would be forwarded to the backend server (at `https://api.backend.example.com/hello-world`) and will include a header `nhsd-apim-apikey` with a value of `foo`

7.14.9.3.5.3 Updating (rotating) an API key

> It is good practice to rotate credentials proactively on a periodic basis.
> It is also important to rotate them reactively where a credential has been (or may have been) compromised. Examples include traffic not from the API proxy to the backend service carrying the correct API key, or where an API key has been committed to version control or transmitted over an insecure medium such as instant message, or email.

API key secrets can be overwritten with a new value using the same method as *step 1*, above.

The change will take effect immediately in all API instances referencing the secret value. (It does not require redeployment of the API instance.)

Updating the API key secret created in step 1, changing the value from 'foo' to 'bar'

```
proxygen secret put --apikey --secret-value bar internal-dev hello-world-key
```

7.14.9.3.5.4 An important note about API key security and TLS (HTTPS)

Transport Layer Security (TLS, used in HTTPS) is not a feature of API key security.

You must configure your backend service to use HTTPS, and disallow insecure HTTP-only requests.

If you omit this step the traffic between your API instance and back end service will be unencrypted.

In the future we may drop support for connecting to external target servers not using HTTPS.

7.14.9.3.5.5 Working with API key secrets

View the documentation for the proxygen-cli `secret` subcommand using:

```
proxygen secret --help
```

You can:

- List secrets previously stored
- Describe (get detailed information on) specific secrets
- Delete a secret. This will fail if the secret is in use (referenced by) an existing API instance. The error message will detail which API instances are preventing deletion.

For example:

```
$ proxygen secret list
name             type    environment    apikey    mtls    version_id
last_modified
---------------  ------  -------------  --------  ------
-------------------------------------  ----------------
hello-world-key  secret  internal-dev   True      False   5aad2ab9-2219-48e0-
a2f8-5f44f3d6fdeb  2023-11-06 21:21

$ proxygen secret describe internal-dev hello-world-key
{
  "name": "hello-world-key",
  "type": "secret",
  "last_modified": "2023-11-06T21:21:08.886000+00:00",
  "environment": "internal-dev",
  "apikey": true,
  "mtls": false,
  "version_id": "5aad2ab9-2219-48e0-a2f8-5f44f3d6fdeb"
}

$ proxygen secret delete internal-dev hello-world-key
{
  "name": "hello-world-key",
  "type": "secret",
  "last_modified": "2023-11-06T21:21:08.886000+00:00",
  "environment": "internal-dev",
  "apikey": true,
  "mtls": false,
  "version_id": "5aad2ab9-2219-48e0-a2f8-5f44f3d6fdeb"
}
Delete secret hello-world-key from internal-dev? [y/N]: y
✓ Deleting secret hello-world-key from internal-dev
```

You can also perform these functions directly through the API. See the Proxy Generator API reference[857] for details.

> Secrets stored using Proxy generator are used by the API platform, but they cannot be retrieved (read out) using the Proxy generator API (or proxygen-cli). This is a security-driven design decision.
> If a secret is overwritten or deleted it cannot be recovered. **You are responsible for maintaining secure backups of any secrets you submit.**

7.14.9.3.6 How to implement mutual TLS (mTLS) security for your API backend

7.14.9.3.6.1 Overview

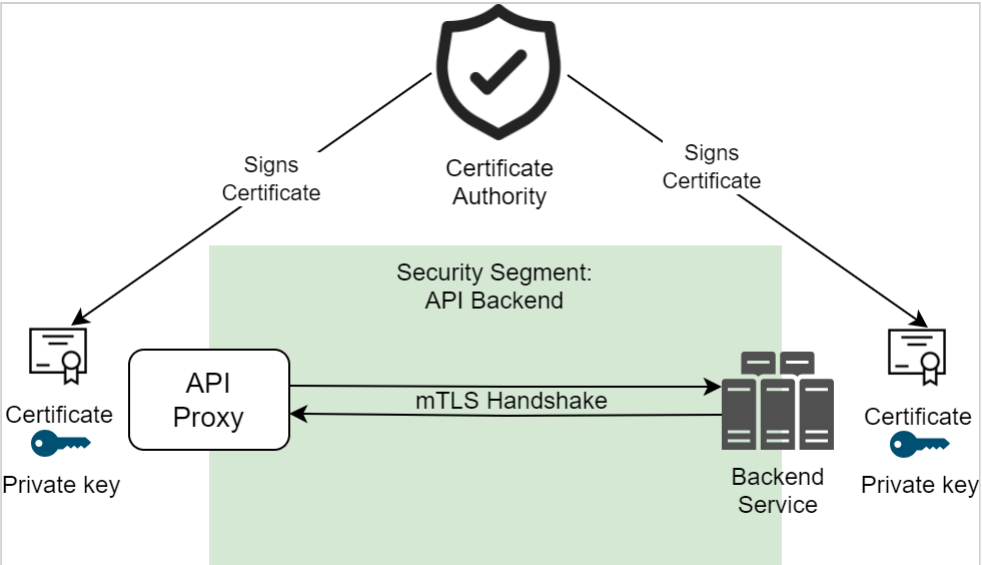


Fig 1: Security segment - API backend

Mutual TLS (mTLS or TLS MA) is the strongest method of securing the connection between your API proxy and backend service. It is suitable for all services, including those exposing patient data.

mTLS is a method for mutual authentication, allowing both parties to verify who they claim to be, and securing the subsequent exchange of information with encryption.

Each party has it's own (not shared):

- **certificate**
  - to establish its identity (who it claims to be)
  - containing its public key

857 https://digital.nhs.uk/developer/api-catalogue/proxy-generator

- **private key**
    - Used for decrypting messages from the other party (encrypted using the public key transmitted with the certificate)

Trust is established by the certificates being signed by the same **certificate authority.**

7.14.9.3.6.2 Obtaining certificates

Obtaining the necessary certificates is beyond the scope of this guide. However, broadly speaking, there are two scenarios:

1. **Connecting to a pre-existing service with an established PKI (Public Key Infrastructure)**: In this case, you will prepare a Certificate Signing Request (CSR) to be signed by the owners of the PKI. This page provides generic guidance on generating CSRs and private keys[858], but the service owner will give their specific requirements. For example the Spine PKI is managed by the DIR Team (dir@nhs.net[859])
2. **Creating your own PKI**: Don't do this, just use the Spine PKI.

If you find yourself in the second scenario, it's important to properly secure the root CA certificates and keys. They are the backbone of your PKI and, if compromised, could put your entire system at risk. A secure server, regular key rotations, and limited access controls are all things to consider when securing your root CA.

To proceed you need a signed certificate and corresponding private key in PEM format:

```
$ head -n3 client-certificate.pem private-key.pem
==> client-certificate.pem <==
-----BEGIN CERTIFICATE-----
MIIFjjCCA3YCFBdZVJAyV5W8wn8sbNV0u/1csKSJMA0GCSqGSIb3DQEBCwUAMIGD
MQswCQYDVQQGEwJHQjEXMBUGA1UECAwOV2VzdCBZb3Jrc2hpcmUxDjAMBgNVBAcM

==> private-key.pem <==
-----BEGIN PRIVATE KEY-----
MIIJQwIBADANBgkqhkiG9w0BAQEFAASCCS0wggkpAgEAAoICAQCN9kXgC45cE58R
wikm1mD85TfeH5kTjJQAaO3Hx/bj6+Ra1s4cX2VKlJCDin4wMIbgMWLQTYVIAaCv
```

7.14.9.3.6.3 Configuring the backend service to validate client certificates

Configuring the backend service to support mutual TLS (mTLS) security is beyond the scope of this guide. It's important to note that AWS API gateway requires the use of custom domains to support the mTLS security pattern: https://docs.aws.amazon.com/apigateway/latest/developerguide/rest-api-mutual-tls.html

Update 29 Nov 2023 AWS have implemented support for mTLS in their ALBs https://aws.amazon.com/blogs/aws/mutual-authentication-for-application-load-balancer-to-reliably-verify-certificate-based-client-identities/

7.14.9.3.6.4 Implementing mTLS security

858 https://help.switch.ch/pki/manage/request/csr-openssl/
859 mailto:dir@nhs.net

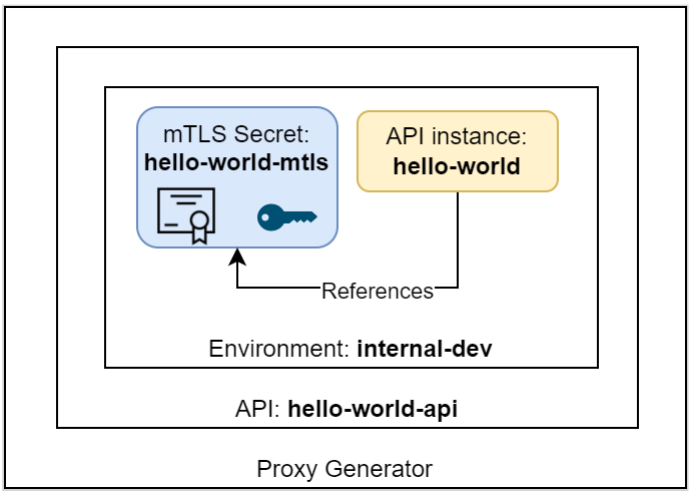


Fig 1: Components of mTLS security in the context of Proxy Generator

A facility of Proxy Generator is the storage of **secret values**, such as mTLS certificates and private keys. These secrets are used in other managed components by referencing them by their name.

In this case an API instance is configured to use a named secret containing an mTLS **certificate** and associated **private key**. When the API instance connects to the backend service it presents the certificate to authenticate itself, and encrypts traffic using the private key.

Both secrets and API instances are scoped to particular environments. An environment can contain multiple secrets, each referenced by zero, one, or multiple API instances.

Store the mTLS certificate and private key as a secret

Use proxygen-cli to create a mTLS secret comprising the mTLS certificate (`client-certificate.pem` file) and corresponding private key (`private-key.pem` file):

```
proxygen secret put --mtls-cert client-certificate.pem --mtls-key private-key.pem
internal-dev hello-world-mtls
```

The format of the command is `proxygen secret put --mtls-cert <cert-file> --mtls-key <key-file> <environment> <secret-name>`

> For a full user guide on how to manage secrets via proxygen-cli, see Proxygen CLI user guide (see page 933).

Alternatively call the Proxy Generator API using the `PUT /apis/{api_name}/environments/{environment}/secrets/{secret_name}`[860] operation with the `type=mtls` query parameter. The request body must be `multipart/form-data` with fields `key` and `cert` containing the private key and certificate data, respectively.

Configure your API instance for API key security

API Instance specification file (partial)

```
x-nhsd-apim:
  target:
    type: external
    healthcheck: /_status
    url: https://api.backend.example.com/hello-world
    security:
      type: mtls
      secret: hello-world-mtls
```

In the `x-nhsd-apim/target/security` section setting `type` to `mtls` enables mutual TLS security.

One additional property is required:

- `secret` : the name of an mTLS secret (created in step 1). The secret must exist in the environment you will be deploying your API instance to.

Deploy your API instance

Use proxygen-cli to deploy your API instance:

```
proxygen instance deploy --no-confirm internal-dev hello-world oas-specification.yaml
```

See deploying your API with proxy generator (see page 698) for further information about API instance deployment, including how to use the proxy generator API.

Note that the API instance is deployed to the same environment as the secret.

Example result

The example above would result in:

860 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#put-/apis/-api_name-/environments/-environment-/secrets/-secret_name-

- An mTLS secret called `hello-world-mtls` stored in the `internal-dev` environment, containing a certificate and private key pair (read from the files `client-certificate.pem` and `private-key.pem`)
- An API instance deployed to the `internal-dev` environment with a base path of `hello-world` referencing this secret
- Requests to the API instance (at `https://internal-dev.api.service.nhs.uk/hello-world`) would be forwarded to the backend server (at `https://api.backend.example.com/hello-world`) and be secured using mTLS

7.14.9.3.6.5 Updating (rotating) mTLS certificate/private key pairs

It is good practice to rotate credentials proactively on a periodic basis.
It is also important to rotate them reactively where a credential has been (or may have been) compromised. Examples include traffic not from the API proxy to the backend service presenting the correct mTLS certificate, or where the certificate or key has been committed to version control or transmitted over an insecure medium such as instant message, or email.

mTLS secrets can be overwritten with a new value using the same method as **step 1,** above.

The change will take effect immediately in all API instances referencing the secret value. (Does not require redeployment of the API instance.)

Updating the mTLS secret created in step 1

```
proxygen secret put --mtls-cert new-client-certificate.pem --mtls-key new-private-key.pem internal-dev hello-world-mtls
```

Working with mTLS secrets

View the documentation for the proxygen-cli `secret` subcommand using:

```
proxygen secret --help
```

You can:

- List secrets previously stored
- Describe (get detailed information on) specific secrets
- Delete a secret. This will fail if the secret is in use (referenced by) an existing API instance. The error message will detail which API instances are preventing deletion

For example:

```
$ proxygen secret list
name              type    environment    apikey    mtls    version_id
last_modified
----------------  ------  -------------  --------  ------
------------------------------------  ----------------
hello-world-mtls  secret  internal-dev   False     True    5d62162c-
f183-4ef5-9e9e-8b5bd2072653  2023-11-21 08:21

$ proxygen secret describe internal-dev hello-world-mtls
{
  "name": "hello-world-mtls",
  "type": "secret",
  "last_modified": "2023-11-21T08:21:41.022000+00:00",
  "environment": "internal-dev",
  "apikey": false,
  "mtls": true,
  "version_id": "5d62162c-f183-4ef5-9e9e-8b5bd2072653"
}

$ proxygen secret delete internal-dev hello-world-mtls
{
  "name": "hello-world-mtls",
  "type": "secret",
  "last_modified": "2023-11-21T08:21:41.022000+00:00",
  "environment": "internal-dev",
  "apikey": false,
  "mtls": true,
  "version_id": "5d62162c-f183-4ef5-9e9e-8b5bd2072653"
}
Delete secret hello-world-mtls from internal-dev? [y/N]: y
✓ Deleting secret hello-world-mtls from internal-dev
```

You can also perform these functions directly through the API. See the Proxy Generator API reference[861] for details.

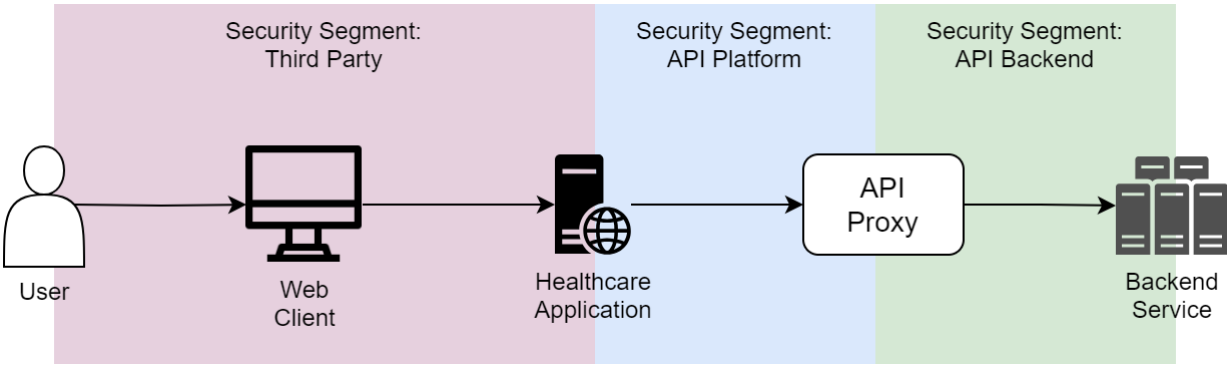
Secrets stored using Proxy generator are used by the API platform, but they cannot be retrieved (read out) using the Proxy generator API (or proxygen-cli). This is a security-driven design decision.

If a secret is overwritten or deleted it cannot be recovered. You are responsible for maintaining secure backups of any secrets you submit.

861 https://digital.nhs.uk/developer/api-catalogue/proxy-generator

7.14.9.4 API consumer access security patterns

7.14.9.4.1 Overview



This section covers the API platform security segment (the blue area of the diagram): securing the connection between the Healthcare application and the API proxy.

Proxy generator supports a number of security patterns (access modes) to ensure only authorised applications and users can access services on the API platform.

The Security and authorisation[862] guide on the Developer hub provides an introduction to this topic.

API Producers are responsible for **choosing the appropriate security pattern** for their service, and **configuring the API proxy** to implement this security pattern.

7.14.9.4.2 Choosing a security pattern

Here we describe the different security patterns that can be implemented using Proxy generator. We provide examples of when each has been used and guidance for consideration but determining the appropriate pattern for your service is beyond the scope of this document.

See Guidance for making API security decisions (see page 437)

The supported security patterns fall into two broad categories: **application-restricted**, and **user-restricted**. User-restricted is further divided into **healthcare worker access** and **patient access** modes.

All security patterns require the client application is known to us by being **registered on the developer portal**.

The registered application must be subscribed to your service by enabling the relevant *API Product (see below)*.

> The codes (eg app-level0 or nhs-login-p9) are used later to configure your API proxy to implement the respective security pattern. In “How to” section we show how these are used as securityScheme values in your OAS file.

862 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation

7.14.9.4.2.1 `app-level0` Application-restricted with API key authentication

In this access mode the calling application authenticates itself by providing an API key in each request[863].

It is the weakest level of security and not suitable for services exposing personal or other sensitive data.

If an end user is present, we do not authenticate them. This is the responsibility of the client application.

The Health Research Data Catalogue API[864] uses this security pattern.

7.14.9.4.2.2 `app-level3` Application-restricted with signed JWT authentication

In this access mode the calling application authenticates itself by sending a signed JSON Web Token (JWT) to our OAuth 2.0 authorisation server[865].

This method is more secure than API key authentication and may be appropriate for services involving personal or sensitive data.

If an end user is present, we do not authenticate them. This is the responsibility of the client application.

The Ambulance Data Submission - FHIR API[866] uses this security pattern.

7.14.9.4.2.3 `nhs-cis2-aal1` `nhs-cis2-aal2` `nhs-cis2-aal3` User-restricted healthcare worker access

In this access mode:

- the calling application authenticates itself to the API platform using a signed JWT
- the end user is authenticated using NHS Care Identity Service (CIS2)

API consumers may choose to implement either combined[867] or separate[868] authentication and authorisation patterns. We do not distinguish between the two methods at the API proxy level.

Where the 'separate' pattern is used the client application is also registered with NHS Care Identity Service (CIS2).

(In path to live environments we provide a mock CIS2 environment[869] for initial development.)

CIS2 supports the concept of authenticator assurance levels[870] which can be used to distinguish between stronger and weaker method of end user authentication:

---

863 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication
864 https://digital.nhs.uk/developer/api-catalogue/health-research-data-catalogue
865 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication
866 https://digital.nhs.uk/developer/api-catalogue/ambulance-data-submission-fhir
867 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-identity-combined-authentication-and-authorisation
868 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-identity-separate-authentication-and-authorisation
869 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-with-our-mock-authorisation-service-using-nhs-cis2---separate-authentication
870 https://csrc.nist.gov/glossary/term/authenticator_assurance_level

- AAL3: a strong two-factor authentication method. Typically using a smart card, or FIDO2 authentication such as Windows Hello or iOS biometric. Intended to allow healthcare workers access to view and update sensitive, clinical information.
- AAL2: A moderate strength security making use of multi-factor authentication such as TOTP. Intended to provide healthcare workers with access to moderately sensitive information, balancing usability and security for tasks that require more protection than AAL1, but without the need for the higher assurance level of AAL3.
- AAL1: weaker, single-factor authentication method (username and password). Intended to enable users to quickly access non clinical systems.

There are currently no services on the API platform that implement healthcare worker access at AAL1 and AAL2 level.

The Reasonable Adjustment Flag - FHIR API[871] implements this security pattern, at AAL3 level.

7.14.9.4.2.4 `nhs-login-p0` `nhs-login-p5` `nhs-login-p9` User-restricted patient access

In this access mode:

- the calling application authenticates itself to the API platform using a signed JWT
- the end user is authenticated using NHS Login

API consumers may choose to implement either combined[872] or separate[873] authentication and authorisation patterns. We do not distinguish between the two methods at the API proxy level.

Where the 'separate' pattern is used the client application is also registered with NHS Login.

(In path to live environments we provide a mock NHS Login environment[874] for initial development.)

NHS Login supports the concept of identity verification levels[875] which can be used to determine the level of user verification:

- P0 (low): Low identity proofing. A user has verified ownership of an email address and mobile phone number.
- P5 (medium): Basic identity information. The user has provided some information that has been checked to correspond to a record on PDS.
- P9 (high): Physical comparison. The user has completed an online or offline identity verification process where physical comparison between the photographic identity and the person asserting their identity has occurred.

There are currently no services on the API platform that implement patient access at P0 or P5 verification levels.

The Immunisation History - FHIR API[876] implements this security pattern, at P9 proofing level.

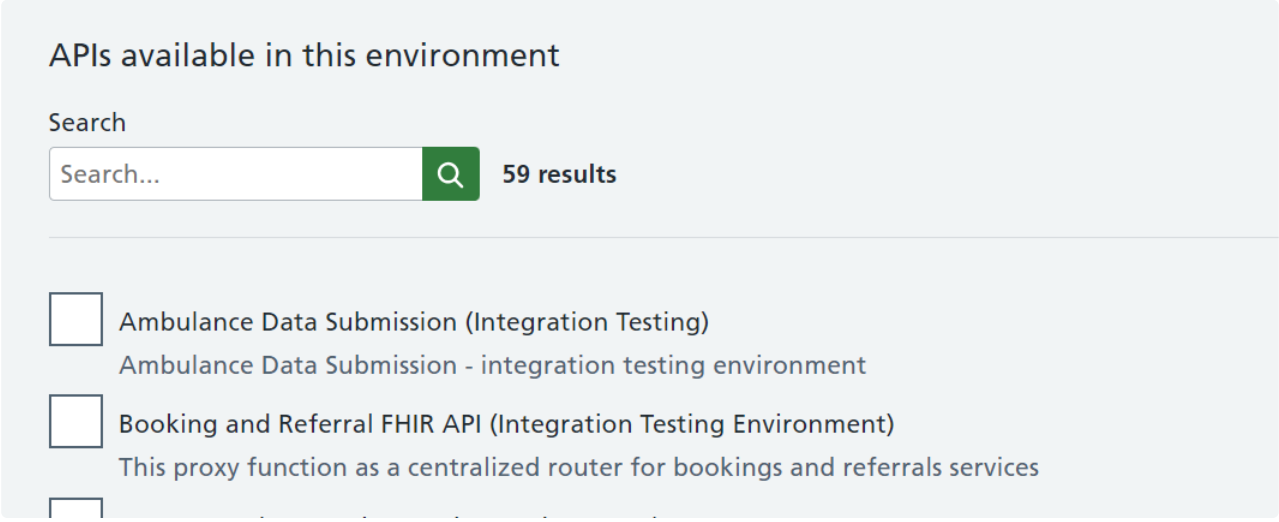
---

871 https://csrc.nist.gov/glossary/term/authenticator_assurance_level
872 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-combined-authentication-and-authorisation
873 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation
874 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/testing-with-our-mock-authorisation-service-using-nhs-login---separate-authentication
875 https://nhsconnect.github.io/nhslogin/vectors-of-trust/
876 https://digital.nhs.uk/developer/api-catalogue/immunisation-history-fhir

> ⚠ In this access mode the end user is a patient. If your service exposes clinical or sensitive information you must implement additional security controls in your API backend to ensure patients can only access their own data.

7.14.9.4.3 API Products

To access our services API consumers must sign in to the digital onboarding service[877] and register their application. The application must then be subscribed to an API before it can be used.

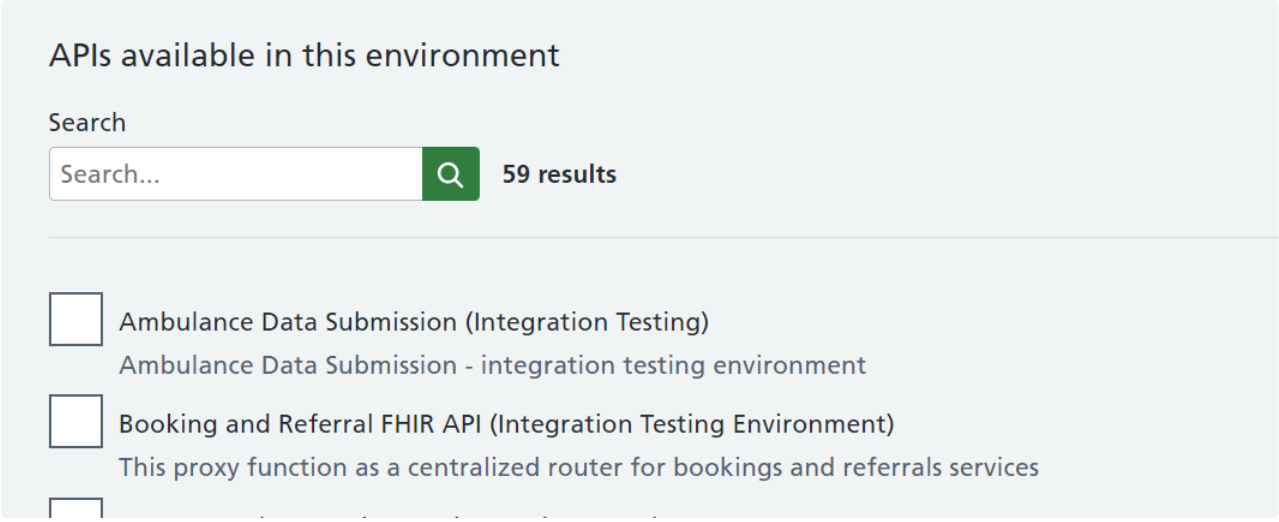


A portion of the details page for an application, showing enabled access to the elective-waiting-list API in the Integration Testing environment

For path to live environments (such as the integration test environment[878]) this subscription is automatically approved (enabled). In the production environment approval is a manual step which happens after completing our onboarding and assurance process. Applications not yet approved show as 'pending'.

The *Add APIs* button lists the APIs (API Products) available in the application's environment:

877 https://onboarding.prod.api.platform.nhs.uk/
878 https://digital.nhs.uk/developer/guides-and-documentation/testing#integration-testing



APIs can implement multiple access modes to support different types of consuming applications and use cases. One example is the Personal Demographics Service - FHIR API[879] which allows Application-restricted access to the Get patient details[880] operation, but requires Healthcare worker access when calling the Update patient details[881] operation.

Assurance requirements may differ depending on the access mode/security pattern used. For example it is not desirable for an application authorised to use the Personal Demographics FHIR API in Application Restricted access mode to also use other supported access modes (such as patient, or healthcare worker access).

To support this flexibility you must explicitly specify API products as part of your API proxy configuration.

7.14.9.4.4 How-to guides

- How to configure API consumer security patterns using proxy generator (see page 875)
- How to configure API products using proxy generator (see page 879)

7.14.9.4.5 How to configure API consumer security patterns using proxy generator

7.14.9.4.5.1 Overview

In common with other aspects of the Proxy generator design philosophy, we leverage OpenAPI v3.0 to implement our security patterns. From the OpenAPI Authentication and Authorization guide[882]:

879 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir
880 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#get-/Patient/-id-
881 https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir#patch-/Patient/-id-
882 https://swagger.io/docs/specification/authentication/

> *Security is described using the* `securitySchemes` *and* `security` *keywords. You use* `securitySchemes` *to define all security schemes your API supports, then use* `security` *to apply specific schemes to the whole API or individual operations.*

Proxy generator supports the following security schemes:

Security Scheme	Access Mode
`app-level0`	Application-restricted API Key
`app-level3`	Application-restricted Private Key JWT
`nhs-cis2-aal1`	Healthcare Worker Access (CIS2) - Weak authenticator assurance level (single factor)
`nhs-cis2-aal2`	Healthcare Worker Access (CIS2) - Moderate authenticator assurance level (multi-factor)
`nhs-cis2-aal3`	Healthcare Worker Access (CIS2) - Strong authenticator assurance level (two factor)
`nhs-login-p0`	Patient Access (NHS Login) - Low identity proofing level
`nhs-login-p5`	Patient Access (NHS Login) - Medium identity proofing level
`nhs-login-p9`	Patient Access (NHS Login) - High identity proofing level

If you are using NHS login - there is an assurance focus on ensuring P9 (or P5) has been achieved for the user. However, this is the **identity** assurance level, not the **authentication** assurance level. Technically, within APIM, we should be looking at the Vector of Trust values and Context class values. If your backend services needs to process this, configure ProxyGen to pass the VoT claim to your backend where you can process it and potentially reject the request.

7.14.9.4.5.2 Example implementation

```
openapi: 3.0.3
info:
  version: 1.0.0
  title: Hello World API
servers:
  - url: 'https://internal-dev.api.service.nhs.uk/hello-world'
components:
# 1 -- Definition of security schemes used in this specification
  securitySchemes:
    app-level0:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/app-
level0
    app-level3:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/app-
level3
# 2a -- Whole API (default) security pattern
security:
  - app-level0: []
  - app-level3: []
paths:
  /hello:
    # Security for this operation not overridden
        get:
      responses:
        200:
          description: Successful request
        # 2b -- Per-operation security specification overries global security value
    post:
      responses:
        201:
          description: Created
      security:
        - app-level3: []
```

Step 1. Defining securitySchemes

> *All security schemes used by the API must be defined in the global* `components/securitySchemes` *section.*

- Use only the Proxy generator supported security schemes (see the above table)
- The value of each must be a reference, of the format `$ref: <https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/><security-scheme` (see page 875)`>`

Step 2. Applying security

> *After you have defined the security schemes in the* `securitySchemes` *section, you can apply them to the whole API or individual operations by adding the* `security` *section on the root level or operation level, respectively. When used on the root level,* `security` *applies the specified security schemes globally to all API operations, unless overridden on the operation level.*

- Use schemes defined in the `components/securitySchemes`
- Use an empty array `[]` as the value of each object.

In our example we specify either `app-level0` (Application restricted API key) or `app-level3` (Application restricted private key JWT) in the root level `security` section. This applies to the `GET /hello` operation as it does not override this. The `POST /hello` operation overrides this, and requires `app-level3`.

7.14.9.4.5.3 Implementation guidance

- If you specify a lower access, assurance or proofing level clients (or users) authenticated at a higher level will be granted access. For example an API specifying `nhs-login-p5` can be used with a P9 NHS Login account.
- Security patterns are disabled entirely in the sandbox environments. (Sandbox environments are intended to be completely open access.)
- You MUST specify either:
  - A default, whole-API security pattern
  - Security patterns for every operation
- The API platform does not support the use of **scopes** for API consumer access patterns.
- From the OpenAPI Authentication and Authorization guide[883]: "*Global security can be overridden in individual operations to use a different authentication type, …,* ***or no authentication at all****"*. Proxy generator managed APIs do not support disabling security.

7.14.9.4.5.4 Next step: configure API products

The API is now configured to enforce appropriate access mode(s) across all operations. However, API consumers must be able to subscribe their application to your API.

Before deploying, please proceed to: How to configure API products using proxy generator (see page 879)

883 https://swagger.io/docs/specification/authentication/

7.14.9.4.6 How to configure API products using proxy generator

7.14.9.4.6.1 Overview

This guide continues on from How to configure API consumer security patterns using proxy generator (see page 875)

Once an API has been secured by applying security schemes (access modes) using the `securitySchemes` component and `security` sections it can be made available to API consumers. This is done by configuring **API products**.

An API product:

- Makes your API available to consumers via the *Add APIs* button in the Digital Onboarding Service
- Controls the access modes (and levels) an API consumer is permitted to use when accessing your API

7.14.9.4.6.2 Example implementation

```
openapi: 3.0.3
info:
  version: 1.0.0
  title: Hello World API
servers:
  - url: 'https://internal-dev.api.service.nhs.uk/hello-world'
components:
  securitySchemes:
    app-level0:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/app-
level0
    app-level3:
      $ref: https://proxygen.prod.api.platform.nhs.uk/components/securitySchemes/app-
level3
security:
  - app-level0: []
  - app-level3: []
paths:
  /hello:
    get:
      responses:
        200:
          description: Successful request
    post:
```

```
      responses:
        201:
          description: Created
      security:
        - app-level3: []
x-nhsd-apim:
  access:
  - title: Application-restricted with API key
    description: Hello world API product description
    grants:
      app-level0: []
  - title: Application-restricted with Private Key JWT
    grants:
      app-level3: []
    visible: false
```

API products are configured in the `x-nhsd-apim/access` section.

This is an array of objects having the following properties:

- `title` Controls how the API Product is displayed in the **Add APIs** list. See below.
- `description` Controls the short description displayed underneath the API product in the Add APIs list.
- `grants` A list of access modes this API Product enables. The items should be present in the `components/securitySchemes` section. Use an empty array `[]` as the value of each object.
- `visible` controls API product visibility. Optional property that defaults to `true`.

API product title

The `title` value specified is combined with:

- the `title` from the `info` section (the API name)
- the name of the **environment** the API instance is deployed in

The format is `{info title} - {access title} ({environment})`

If our example schema were deployed to the `int` environment these would be the entries in the *Add APIs* list:

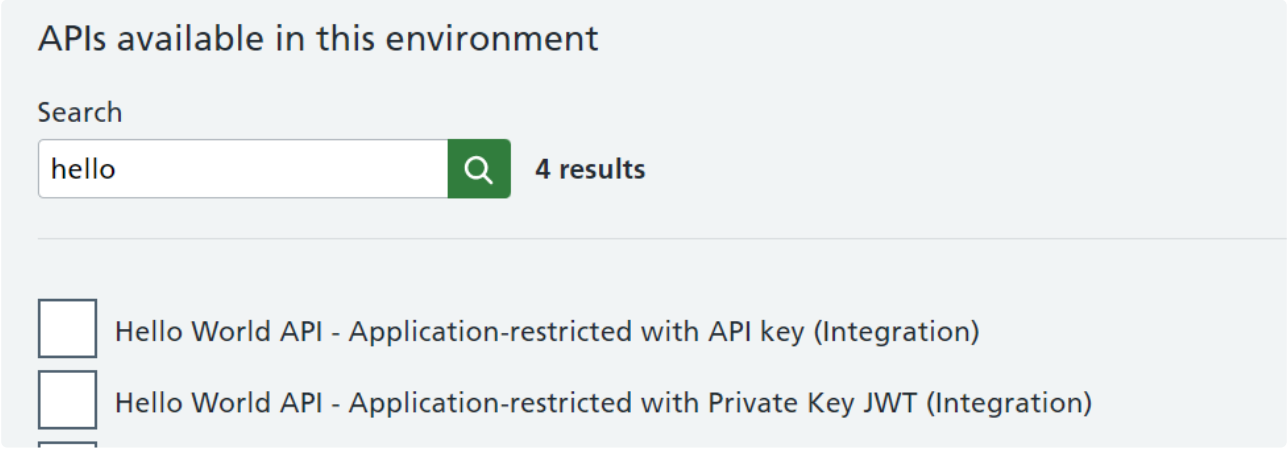


Fig 1: Available APIs

API product visibility

The intention is that setting the `visible` property to `false` creates a private API product. One that does not appear in the *Add APIs* list for general API consumer usage, but is available for limited, private, or internal use cases. (Action from the API Producer teams or the API Platform 'Deathstar' team would be needed to subscribe an application to such products.)

Example result

If the above OAS specification were deployed as an API instance in the Integration environment it would result in two API products:

- **Hello World API - Application-restricted with API key (Integration)** Applications subscribed to this product:
  - Use API Key authentication[884]
  - ✓ Can call the `GET /hello` operation
  - ✗ Cannot call the `POST /hello` operation
- **Hello World API - Application-restricted with Private Key JWT (Integration)** Applications subscribed to this product:
  - Use signed JWT authentication[885]
  - ✓ Can call the `GET /hello` operation
  - ✓ Can call the `POST /hello` operation

884 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication

885 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication

7.14.9.4.6.3 Implementation guidance

⚠ Warning about changing `grants`

API Products are uniquely identified by:

- The API **instance name** (base path)
- Deployed **environment**
- The **grants** list

If you update the grants, you fundamentally change the API product such that the existing one is deleted and a new one is created.

Any Applications subscribed to the old API product will not be subscribed to the new one. Doing this could have unintended consequences, and in production would interrupt service.

We are working on improving this:
APM-4839[886] - SPIKE: API products foot-gun. Identify others, and propose a fix DONE

API product approval

When an API consumer adds a path to live environment API product to their application it is automatically approved.

In production it moves to a “pending” state and must be manually enabled.

This allows self-service access to PTL environments and allows us to control who can access production services.

In the future we may allow API producers to control this behaviour:
APM-4841[887] - Support Approval Type Control on API Products TO DO

7.14.10 Deploying your API

7.14.10.1 Overview

API proxies are deployed to Apigee via Azure pipelines. These pipelines are created and synced with your GitHub repo when you set up your proxy with the setup-api service (see page 580) and use the configuration in your repo to deploy your proxy, products and config to Apigee, as well as a hosted backend or sandbox if you have chosen for us to host them.

There are two types of deployment flow:

- Pull request depolyments, which a triggered every time you push to a branch with an open PR.
- Release deployments, which are triggered on a merge to the main/master branch.

886 https://nhsd-jira.digital.nhs.uk/browse/APM-4839?src=confmacro
887 https://nhsd-jira.digital.nhs.uk/browse/APM-4841?src=confmacro

On each of these Azure will first run a 'Build' pipeline which gathers the config from the repo and builds an artefact. Once the successfully completed, the 'Pull Request' or 'Release' Pipeline will automatically trigger and use the artefact to deploy the config.

The majority of the Azure pipelines are common pipelines that will automatically be used during deployment, however, we do offer some features for improved security (self-serve release gates for production) and efficiency (dependency caching).

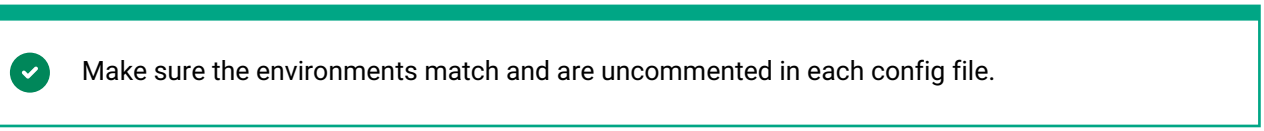
7.14.10.2 Process steps

After following the steps in building your API Alpha[888]

1. Configure the API deployment per environment using
    a. The Azure pipeline config files (see page 884)
    b. The proxy manifest file (see page 896)
2. Add caching (see page 908) where needed to your pipelines
3. Configure self-serve releases and manual approval (see page 913)
4. Test out your deployed API by using it as a consumer would (see page 925)

7.14.10.3 Configuring the API deployment

There are two files that serve as configuration files that will change how the API is released and how it behaves.



7.14.10.3.1 azure-release-pipeline.yml

This defines what environments you release your API to. So we can specify we want to start releasing an API to the INT environment by adding it to the list of Apigee_environments:

```
- environment: int
  depends_on:
    - internal_qa
    - internal_qa_sandbox
```

When your repo is generated this has already been defined in this file but INT has been commented out and can be uncommented when ready. You may find you need

This will then need to be deployed to master where it will tag and create a release which will trigger the release pipeline to run.

888 https://nhsd-confluence.digital.nhs.uk/display/APM/Building+your+API+alpha?src=contextnavpagetreemode

Once deployed, your API will be available for integration testing on the INT domain.

See Managing your API deployments (see page 884) for more information on this file.

7.14.10.3.2 manifest_template.yml

This file sits at the root of your API GitHub repository and aims to supply the deployment configuration to Apigee. Version 1.1 completely specifies all the API products, specs, and API catalog objects.

The manifest file should contain information about the Apigee resources you deploy. This includes (but is not limited to) security patterns on products, rate-limiting config and the visibility of products in the Developer Hub.

The environments are also listed here and it is important they match those in the azure-release-pipeline.yml file.

See Manifest.yml reference[889] for more information about this file and how it alters the deployment of the API.

7.14.10.4 Managing your API deployments

7.14.10.4.1 Introduction

While most of the configuration for deploying the your proxy resources is done by the API Management common pipelines, there are several pipeline files you'll need to configure for which environments are deployed and running integration tests against them.

When a proxy repository is created, 3 pipelines are automatically generated for that proxy in azure folder:

- build Pipeline is configured through `azure/azure-build-pipeline.yml`
- pull Request pipeline, is declared in `azure/azure-pr-pipeline.yml` and is triggered when a new PR is created.
- release Pipeline, is ran when you need to create a release on the merge of a PR and is configured through `azure/azure-release-pipeline.yml`
    - you can read more about releasing to production and self-serve releases in Deploying to production (see page 1120).

7.14.10.4.2 Shared parameters

These parameters are passed into all pipelines and are defined in `azure/project.yml`.

889 https://nhsd-confluence.digital.nhs.uk/display/APM/Manifest.yml+reference?src=contextnavpagetreemode

Name	Type	Default	Description	Example
service_name	string	null	The full name of your service.	`personal-demographics`
short_service_name	string	null	The abbreviated name of your service.	`pds`
service_base_path	string	null	The URL basepath of the API service.	`personal-demographics`

7.14.10.4.2.1 Custom variables

If you want to define custom variables to pass to more than one pipeline file, they can be defined in this file, then passed to a pipeline file using the pattern: `${{ variables.variable_name }}`

7.14.10.4.3 Build pipeline

Each API GitHub repository uses a shared Build Pipeline. The pipelines are defined using the YAML schema[890], stored alongside the code in `azure/azure-build-pipeline.yml`. In addition to building and testing ("Continuous integration" activities), each build pipeline outputs a single versioned artefact suitable for deployment into a number of environments by the release pipeline. By default, this pipeline:

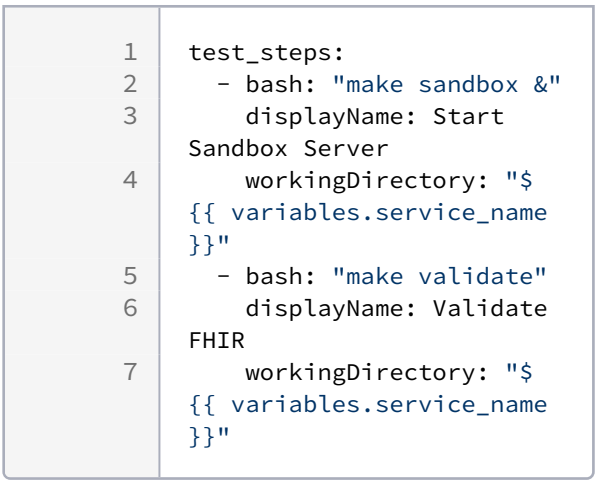
1. installs your project dependencies from the top-level `poetry.toml`.
2. checks that your project dependencies' licenses are compatible. We currently fail on dependencies released under GPL and L/GPL licenses due to compatibility issues with other licenses. There is a ticket open to review this:
   APM-1848[891] - Discuss the current state of license checks in the pipeline DONE .
3. lints your Python (flake8)
4. runs any test steps specified in the `test_steps` parameter.
5. compiles your YAML specification into a versioned JSON specification.
6. compiles your proxies and JSON specification into a deployable artefact.

890 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

891 https://nhsd-jira.digital.nhs.uk/browse/APM-1848?src=confmacro

7.14.10.4.3.1 Parameters

Name	Type	Default	Description	Example
service_name	string	null	See Shared parameters (see page 884)	
short_service_name	string	null	See Shared parameters (see page 884)	
variables	object	[]	A key-value map of variables passed into the pipeline steps. Keys are case-sensitive and should contain only letters, numbers and underscores. Values are free-text and support character escapes.	`variables:` `  PIP_CACHE_DIR: ".poetry"`
secret_file_ids	object	[]	A list of secret files that the pipeline will retrieve from AWS Secrets Manager as part of the pipeline. Secret files are read into a file on the Azure DevOps build instance, and the file path is stored as a var. Newlines are preserved.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/KEYSTORE`

Name	Type	Default	Description	Example
secret_ids	object	[]	A list of secrets that the pipeline will retrieve from AWS Secrets Manager as part of the pipeline. Secrets are read directly into an environment variable, and may not properly preserve multi-line strings.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/CLIENT_SECRET`
config_ids	object	[]	A list of Parameters that the pipeline will retrieve from AWS SSM Parameter Store as part of the pipeline.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/CLIENT_ID`
test_steps	stepList	[]	A list of additional Azure DevOps steps that the build pipeline should run at the Test step. For more information on how to structure Azure DevOps steps, see YAML schema[892].	

```yaml
test_steps:
  - bash: "make sandbox &"
    displayName: Start Sandbox Server
    workingDirectory: "${{ variables.service_name }}"
  - bash: "make validate"
    displayName: Validate FHIR
    workingDirectory: "${{ variables.service_name }}"
```

892 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

Name	Type	Default	Description	Example

post_ecs_push	step List	[]	A list of additional Azure DevOps steps that the build pipeline should run after the ECS Push step.
			For more information on how to structure Azure DevOps steps, see YAML schema[893].
				`post_ecs_push:`
				`  - bash: "scripts/populate_ssm_parameters.sh"`
				`    displayName: Copy secrets from secrets manager to SSM`
				`    workingDirectory: ${{ variables.service_name }}`

7.14.10.4.4 Pull request and release pipeline

Each API GitHub repository uses a shared pull request pipeline and a Release pipeline. The pipelines share the same parameters but the PR pipeline deploys Pull request preview (see page 893) versions of a release environment so you can test changes before merging and deploying to the core environments.

The pipelines are defined using the YAML schema[894].

7.14.10.4.4.1 Parameters

Name	Type	Default	Description	Example	
service_name	string	null	See Shared parameters (see page 884)		
short_service_name	string	null	See Shared parameters (see page 884)		
service_base_path	string	null	See Shared parameters (see page 884)		

893 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema
894 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

apigee_deployment	object	<pre>apigee_deployments:
		  -
		environment:
		internal-dev
		
		make_spec_visible: true
		  -
		environment:
		internal-qa
		
		make_spec_visible: true
		  -
		environment:
		internal-qa-sandbox
		
		proxy_path:
		sandbox
		
		make_spec_visible: true
		  -
		environment:
		sandbox
		
		proxy_path:
		sandbox</pre>
			An object containing the environments that this pipeline should deploy to.	<pre>apigee_deployments:
				  - environment:
				internal-dev
				  make_spec_visible:
				true
				  post_deploy:
				  - template:
				templates/pds-tests.yml
				  parameters:
				
				test_pack_path: ./e2e/
				smoke.json
				  test_type:
				smoke</pre>	

proxy_path	string	live	Path to proxy directory from /proxies at the project root	live	
enable_monitoring[1]	bool	true	Should we attempt to call _ping (see page 1073) to determine service status.	true	
enable_status_monitoring[1]	bool	true	Should we attempt to call _status (see page 1075) to determine service status.	true	

secret_file_ids	object	[]	A list of secret files that the pipeline will retrieve from AWS Secrets Manager as part of the pipeline.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/KEYSTORE`	
secret_ids	object	[]	A list of secrets that the pipeline will retrieve from AWS Secrets Manager as part of the pipeline.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/CLIENT_SECRET`	
config_ids	object	[]	A list of Parameters that the pipeline will retrieve from AWS SSM Parameter Store as part of the pipeline.	`/ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/CLIENT_ID`	
pre_template	object	[]	A list of additional Azure DevOps steps that the build pipeline should run before rendering Jinja templates. For more information on how to structure Azure DevOps steps, see YAML schema[895].	`See test_steps.`	

895 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

post_template	object	[]	A list of additional Azure DevOps steps that the build pipeline should run after rendering Jinja templates. For more information on how to structure Azure DevOps steps, see YAML schema[896].	`See test_steps.`	
pre_deploy	object	[]	A list of additional Azure DevOps steps that the build pipeline should run before deploying your API. For more information on how to structure Azure DevOps steps, see YAML schema[897].	`See test_steps.`	

896 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema
897 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

post_deploy	object	[]	A list of additional Azure DevOps steps that the build pipeline should run after deploying your API.
			For more information on how to structure Azure DevOps steps, see YAML schema[898].
			post_deploy can be used to run your tests against a deployed environment.
			e.g.
				`post_deploy:`
				`- template:`
				`templates/e2e.yml`
				`parameters:test_`
				`type:`
				`smoketest`	

portal_api_requires_callback_url	bool	false	Does the Portal API require a callback URL?		parameters:

jinja_templates	object	[]	Key/values for custom jinja templating	`jinja_templates:`
				`PDS_TARGET_SERVER:`
				`spine-demographics`
				`REQUIRE_ASID: false`	

7.14.10.4.5 Rename your proxy

The externally visible base path for a proxy, the product names displayed are key to helping to make an API understandable. It is important that these terms align with the naming and content in the OAS file rendered in the API Catalogue. However, proxies are often deployed very early on and the name needs to be changed.

If you decide (or are asked) to change the name of you need to be aware that:

898 https://docs.microsoft.com/en-us/azure/devops/pipelines/yaml-schema?view=azure-devops&tabs=schema%2Cparameter-schema

- The "name" of your proxy (or API) is mostly visible to API Consumers in two places, and these are the most important items to change:
    - Name shown in the API Catalogue
    - The URL of your API
- You can't rename a GitHub repo - however, it does not need to match exactly as it's not something your API consumers look at
- The service_name and short_service_name must **never** change as this is the true unique key in the deployed infrastructure

To rename your proxy, you should:

- Change the basepath of your proxy - https://{domain / host}/{**base-path**}/{**resource-path**} (see Proxy base path vs API base path (see page 364))
    - This is stored in the **service_base_path** variable of project.yml, simply update this and the next deployment will carry out the change
- Change the OAS file name
    - You will also need to change the spec_file variable
- You might need to change the API name in the Catalogue, but often the human readable title is only indirectly related to the name used in the URL
- The OAS file will need to have the following changed:
    - Update to the new base path, which also means that the 'Try it now' feature works
    - Update 'Environments and testing' section
    - Update the base path in the sandbox
- Update the automated tests and verify that all tests execute successfully

7.14.10.4.6 Pull request previews

7.14.10.4.6.1 Introduction

To facilitate releasing into PTL early and often, the API Platform's CI/CD uses Pull Request Pipelines. These pipelines build and deploy pipelines into an environment where they can be observed and reviewed, allowing full testing to take place during the PR process in a realistic environment.

The deployed apis are also namespaced (they have a suffix automatically appended to their name) so that they do not interfere with other instances of the same API.

7.14.10.4.6.2 Prerequisites

For PR Previews to work, you must ensure that you have first set up your repository and pipelines using the following tutorials:

- Create a github repository and deployment pipelines (see page 580)
- KOP-013 Fixing pipeline triggers[899]

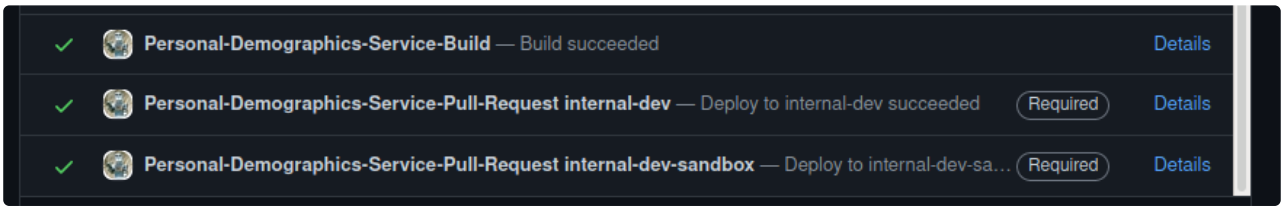
899 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125414244/KOP-013+Fixing+pipeline+triggers

7.14.10.4.6.3 How to deploy a pull request API preview

Deploying a preview is a simple process:

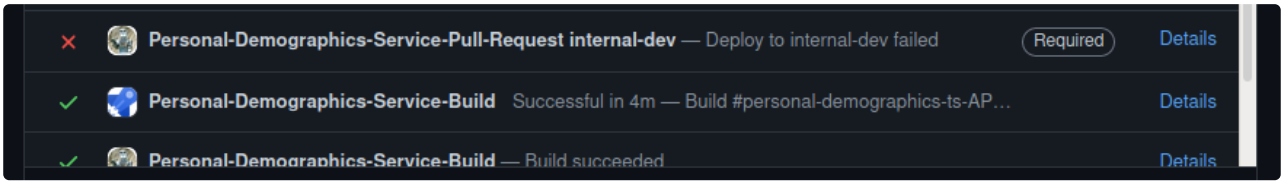
1. Write some code on a branch
2. Open a pull request in github to that branch
3. Wait for the preview to deploy itself.

The build and release for the preview will appear in the status box with green ticks when it has finished successfully:



When a deploy fails

On a failure, you will see a red cross appear in the status box, such as this:



By clicking the 'Details' link, you will be taken to the build in Azure DevOps so you can diagnose the issue.

Where is the preview?

The preview is deployed in one or both of the internal-dev and internal-dev-sandbox environments. If you do not have a sandbox, then the preview will only be deployed to internal-dev.

Where is the code that drives this process?

The code can be found in your repository under `azure/azure-pr-pipeline.yml`. It references code from the utils repository[900].

Apigee Proxy

In Apigee, a new api proxy will be created, named `<YOUR-API>-pr-<GITHUB-PR-NUMBER>`.

You can access this proxy at `https://internal-dev.api.service.nhs.uk/<YOUR-API>-pr-<GITHUB-PR-NUMBER>`.

900 https://github.com/NHSDigital/api-management-utils/tree/master/azure

If you have a sandbox, it can be accessed at `https://internal-dev-sandbox.api.service.nhs.uk/<YOUR-API>-pr-<GITHUB-PR-NUMBER>`.

ECS

If your deploy includes containers, your proxy will be deployed as `<YOUR-API-SHORT-NAME>-pr-<GITHUB-PR-NUMBER>` in the `apis-internal-dev` and/or `apis-internal-dev-sandbox` ECS clusters.

7.14.10.4.6.4 Troubleshooting

I can't see any statuses at the bottom of my pull request

This indicates that the pipelines may not exist, or the triggers may not be working. Check the setup guide (see page 580).

My preview is building, but not releasing

This is usually down to a broken PR pipeline trigger. To fix, follow KOP-013 Fixing pipeline triggers[901].

My preview has disappeared!

There is a regular daily clean-up of previews that deletes any more than five days old. To remedy this, you can rerun your PR's pipelines by commenting `/azp run` in the PR comments.

7.14.10.4.6.5 Best practices

Require the preview deploy to succeed

When setting up your repo's branch protection, require the `<YOUR-API>-Pull-Request` pipeline to succeed in order to allow merging.

This way, you can have more confidence that there are no fundamental errors in your proxy or any attached containers.

Add tests to run during the preview deploy

During the preview deploy is the ideal time to check that any new or changed behaviour has taken effect, and provides evidence at the PR level which can be used when consulting with assurance.

901 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125414244/KOP-013+Fixing+pipeline+triggers

7.14.10.5 Manifest.yml reference

The manifest is a YAML document named `manifest.yml` that sits at the root of your API Github repository.

> **Manifest templates**
>
> You can either provide a fully-written `manifest.yml` , or a `manifest-template.yml` template. See Manifest templates (see page 902) for more info on using templates.

The manifest aims to specify (as much as is securely possible) all the things to Apigee as part of a deployment (and maybe in the further future AWS). Version 1.1 completely specifies all the API products and specs.

The build pipeline validates your manifest (generating a manifest from your template beforehand, if that is the approach you have taken). On successful validation, the file is copied into the build artifact. The pipeline fails on unsuccessful validation.

After the proxy is deployed in the `Deploy Proxy And Products` Azure DevOps pipeline step of a deployment pipeline, if the manifest file is present; then the products and specs from the manifest are deployed.

If you have not provided either a manifest or a manifest template, the deployment pipelines will proceed with their old behaviour, honouring the Azure DevOps Pipeline YAML template parameters. **This implies that if the manifest is present old pipeline flags are ignored!**

See Manifest related ansible playbooks (see page 904) for an in-depth description of the ansible tasks ran as part of the pipeline.

7.14.10.5.1 Manifest.yml specification

Here is an example of a minimal `manifest.yml` to deploy a one `product`, one `spec`, and add a single entry linked to that product/spec to internal-dev `api_portal`.

example manifest

```yaml
meta:
  api:
    guid: 96836235-09a5-4064-9220-0812765ebdd7
    name: canary-api
    spec_guids:
      - 0af08cfb-6835-47b5-867c-95d41ef849b5
  schema_version: 1
apigee:
  environments:
```

```
  - name: internal-dev
    products:
    - approvalType: auto
      attributes:
      - name: access
        value: public
      - name: ratelimit
        value: 5ps
      description: tweet tweet!
      displayName: Canary API (Internal Development)
      environments:
      - internal-dev
      name: canary-api-internal-dev
      proxies:
      - canary-api-internal-dev
      quota: '300'
      quotaInterval: '1'
      quotaTimeUnit: minute
      scopes: []
    specs:
    - name: canary-api-internal-dev
      path: canary-api.json
```

7.14.10.5.1.1 Document structure

At the root level, a manifest is a dictionary with two required fields, `meta` and `apigee`. (As of `schema_version` 1.1 additional keys are allowed and ignored).

Expand/Collapse All (see page 896)

meta

A block that stores some meta-data about your manifest file.

The `meta` block currently has 2 fields:

meta.api

The `api` field contains some ID tags for your API.

meta.api.name

The name of your API must be all lowercase with hyphens separating words. All objects deployed to Apigee must be prefixed with this name.

meta.api.id (DEPRECATED)

> ⚠ As of `schema_version 1.1` the `meta.api.id`[902] is deprecated, use `meta.api.guid` instead. Support for `meta.api.id`[903] will be dropped in `schema_version` 2.

A UUID4 string, unique to your API. This value is required so that if your API ever needs to change its name, there can be continuity with state API management tracking.

meta.api.guid

> ⚠ As of `schema_version 1.1` the `meta.api.id`[904] is deprecated, use `meta.api.guid` instead. Support for `meta.api.id`[905] will be dropped in `schema_version` 2.

A UUID4 string, unique to your API. This value is required so that if your API ever needs to change its name, there can be continuity with state API management tracking.

meta.api.spec_guids

A list of unique UUID4 strings.

Each string corresponds to a spec your API conforms to.

These values are distinct from the `meta.api.id`[906] to allow for a single API to support multiple specs (e.g. version v1/v2/v3).

This value should match the value placed in the API specification[907] custom header `x-nhs-api-spec-guid`.

> ℹ **Relationship between Specs and Products**
>
> NOTE: The agreed relationship between Specs and Products is:
> - One Spec can be on many Products
> - A Product can only relate to one Spec
>
> This should have minimal, if no impact, on the build out of APIs - but makes automation of information and stats outside of APIM (e.g. API Producer dashboards, Aalto EA tool, etc.) really easy.
> However, initial work around Products and Splunk logging ran into some issues. In the current version of the manifest there is an implied one to one relationship between the multiple Spec mentions in the manifest file.

902 http://meta.api.id

903 http://meta.api.id

904 http://meta.api.id

905 http://meta.api.id

906 http://meta.api.id

907 https://nhsd-confluence.digital.nhs.uk/display/APM/API+specification+formatting+guide

meta.schema_version

The schema_version denotes what version of the manifest structure you are adhering to. This is based on semantic versioning, i.e. `<major version>.<minor version>.<patch>` .

For the purpose of validation, you only need to supply a current major version number. Minor version numbers and patch numbers are not validated against.

The current version, documented on this page, is 1.1.6. You can see a list of changes between versions in the relevant section.

apigee

A highly structured object describing what will be deployed to Apigee. Currently, there is only one contained item:

apigee.environments

The `environments` field corresponds to a list of `environment` dictionaries, representing a complete list of what you want to get deployed to Apigee in that particular environment. As of manifest `schema_version` proxies are not yet included in this list.

Each environment object must have:

environment.name

The name of the nhsd-digital Apigee environment you want to deploy to. Must be one of:

- internal-dev
- internal-dev-sandbox
- internal-qa
- internal-qa-sandbox
- ref
- dev
- sandbox
- int
- prod

environment.products

The use of more than one Product is **not** technically required, but APIM do use this to help manage the platform. As an API Producer you should only have more than on Product if:

- You have multiple access modes (e.g. User Restricted and Application Restricted)
- You can map a specific on-boarding SCAL to a specific Product - this makes it easy to see if a specific API Consumer should (or should not) be allowed to use a specific API with a specific access mode
- Where there are multiple OAS files (e.g. one for STU3 and one for FHIR R4)

The `products` field contains a list of dictionaries. Each dictionary in the list is a unique product. The key/value pairs in the dictionary are translated into request body parameters in calls to create/update `products` in Apigee. As such, the meaning and types of fields are given by the Apigee Management API[908] documentation.

As of the manifest `schema_version` 1.1 **all fields on the product are required** (though they may not be by the Apigee API).

The Apigee Management API lists additional request body parameters which are not (yet) implemented.

However, the entire subset that has ever been used in APIM deployments are implemented, and it is easy to add more should the need arise.

Product field	Required	Type	Description
approvalType	Yes	Choice(auto/manual)	How to grant developer requests for their app to use this product.
attributes	Yes	List of `product.attribute`	See `product.attributes` section.
displayName	Yes	String	Name as displayed in Apigee portal.
environments	Yes	List of strings	Must have one entry that matches the `name` of the enclosing manifest.environments object
name	Yes	String	REST identifier must be appropriately namespaced (begin with meta.api.name[909] and end with apigee.environment.name[910]), e.g. canary-api-internal-dev
proxies	Yes	List of strings	Each entry must correspond to a proxy which exists on Apigee
quota	Yes	String	String form of an integer describing how many requests can be made in `quotaInterval` x `quoteTimeUnit`

908 https://apidocs.apigee.com/docs/api-products/1/types/APIProduct
909 http://meta.api.name
910 http://apigee.environment.name

Product field	Required	Type	Description
quotaInterval	Yes	String	String form of an integer (see quota)
quotaInterval	Yes	Choice (minute/hour)	See quota
scopes	Yes	List of strings	Scopes available to access tokens granted by the identity-service

product.attributes

Product attributes are key/value pairs associated with the product, accessible at runtime in your Apigee proxy.

Product attribute field	Required	Type	Description
name	Yes	String	REST identifier for the attribute
value	Yes	String	Value of the attributes

API Management has reserved some product attribute names to enforce design patterns.

As such, some attributes **must** be placed on your product.

Reserved product attributes	Required	Allowed values
ratelimit	Yes	The string form of an integer
access	Yes	public/private

environment.specs

The `specs` field contains a list of dictionaries. Each dictionary is a unique spec which may have the following fields.

Spec field	Required	Type	Description
name	Yes	String	A unique name to identify this spec in Apigee.

Spec field	Required	Type	Description
path	Yes	String	file **relative to the manifest** containing a YAML OpenAPI 3.0.0 schema

Note that YAML is a superset of JSON, so JSON is valid YAML. (The Apigee spec API is currently undocumented, so I cannot point you to it here).

7.14.10.5.2 Manifest templates

The upside of giving a complete specification of every product/spec/portal object to be deployed to Apigee is complete control.

The downside is the manifest grows large quickly. (At the time of writing the manifest required to mimic PDS deployments is 574 lines long!) And will only increase in size as more parts of the APIM platform are brought under into its scope.

While committing such a large file to version control is a perfectly valid way to proceed, making systematic edits to such a (repetitive) file may be a human-error-prone hassle.

Ansible has jinja templating[911] built-in. This provides an out-of-the-box, flexible, and powerful way to specify the manifest more concisely.

Producer squads can use an alternative templating method (e.g. jsonnet) and keep that template under version control. All the build pipeline cares about is finding a `manifest.yml` at the repository root at build time.

7.14.10.5.2.1 `manifest_template.yml` requirements

- A `manifest.yml` must not already exist at the top level of your repository.
- Your manifest template must be named `manifest_template.yml` and sit at the top level of your repository.
- The `manifest_template.yml` **must** have two sections separated by the YAML stream separator `---`
- The first section contains all the variables required for templating. The second section (which will not initially be valid YAML) is then templated, referencing the first section's variables.

The following example `manifest_template.yml` only generates two environments worth of manifest, but could easily be extended to all environments.

911 https://jinja.palletsprojects.com/en/latest/templates/

example manifest_template.yml

```
DESCRIPTION: tweet tweet
APIGEE_ENVIRONMENTS:
  - name: internal-dev
    display_name: Internal Development
    has_mock_auth: true
  - name: internal-qa
    display_name: Internal QA
    has_mock_auth: true
SPEC_GUID: '0af08cfb-6835-47b5-867c-95d41ef849b5'
---
meta:
  api:
    name: canary-api
    guid: 96836235-09a5-4064-9220-0812765ebdd7
    spec_guids:
      - {{ SPEC_GUID }}
  schema_version: 1
apigee:
  environments:
{% for ENV in APIGEE_ENVIRONMENTS %}
{% set TITLE = 'Canary API (' + ENV.display_name + ')' %}
{% set NAME = 'canary-api-' + ENV.name %}
  - name: {{ ENV.name }}
    products:
      - name: {{ NAME }}
        approvalType: auto
        attributes:
          - name: access
            value: public
          - name: ratelimiting
            value:
              {{ NAME }}:
                quota:
                  enabled: true
                  limit: 300
                  interval: 1
                  timeunit: minute
                spikeArrest:
                  enabled: true
                  ratelimit: 600pm # 10 requests per second
              app:
                quota:
                  enabled: false
                spikeArrest:
                  enabled: false
        description: {{ DESCRIPTION }}
        displayName: {{ TITLE }}
        environments: [ {{ ENV.name }} ]
        proxies:
```

```
          - {{ NAME }}
          - identity-service-{{ ENV.name }}
{% if ENV.has_mock_auth | default(false) %}
       - identity-service-mock-{{ ENV.name }}
{% endif %}
        scopes: []
    specs:
      - name: {{ NAME }}
        path: canary-api.json
{% endfor %}
```

7.14.10.5.3 Manifest related ansible playbooks

> The migration of deployment logic from Azure pipelines to ansible means you can deploy the manifest from your machine.

This page contains a breakdown of the individual playbooks used by ansible as part of the build / PR / release pipelines.

Note for the uninitiated: in ansible lingo, a standalone script is called a 'playbook'.

The recommended entrypoint to the playbooks is the `Makefile` in the `ansible` directory of `api-management-utils`.

7.14.10.5.3.1 Validate-manifest

Although manifest validation takes place in several plays. This play is a stand-alone validation script.

```
cd api-management-utils/ansible
DIST_DIR=/path/to/dist/dir make template-manifest
```

The following environment variables can influence this play:

env var	required	usage
`DIST_DIR`	no	The directory ansible looks for the manifest.yml (and spec files).
`SERVICE_NAME`	no	For validation

Some internal consistency checks are applied to the manifest before it is deployed. As of version=0, these are:

a. The `SERVICE_NAME` environment variable must begin with the `meta.api.name`[912] defined at the root of the manifest file. (This is a sanity check that the pipeline is running the right manifest).
b. All product/spec names MUST begin with the `service_name` defined at the manifest root.
c. All product/spec names MUST contain the environment name.
d. All product names in an environment are unique.
e. All spec names in an environment are unique.
f. The `environments` field of a `product` must contain only one entry, which matches the name of the manifest `apigee.environments` dict which contains it.
g. For each `spec.path` attribute, there must be a file at `DIST_DIR/spec.path` containing valid YAML (YAML is a superset of JSON, so JSON is fine too).

7.14.10.5.3.2 Deploy-manifest

Selects an environment from the manifest file and deploys the `products`, `specs`, and `portals` to Apigee.

Usage

```
cd api-management-utils/ansible

DIST_DIR=/path/to/dist/dir \
APIGEE_ORGANIZATION=nhsd-nonprod \
APIGEE_ENVIRONMENT=internal-dev \
APIGEE_ACCESS_TOKEN=$(get_token) \
make deploy-manifest
```

The following environment variables can influence this play:

env var	required	usage
`DIST_DIR`	no	The directory ansible looks for the manifest.yml (and spec files).
`SERVICE_NAME`	no	For validation 0 (see validate-manifest)
`APIGEE_ENVIRONMENT`	yes	Selects the environment to deploy.

912 http://meta.api.name

env var	required	usage
`APIGEE_ORGANIZATION`	yes	The Apigee organization which contains the specified `APIGEE_ENVIRONMENT`
`APIGEE_ACCESS_TOKEN`	yes	A token which authorizes your use of the Apigee Management API[913].
`PULL_REQUEST`	no	A string of the form 'pr-1234', see the section below.

On PULL_REQUEST deployments

As of manifest `schema_version` 1.1, pull request deployments simply deploy what is specified in the `internal-dev`, `internal-dev-sandbox` environments of the `apigee` section of the manifest. The only changes are that all names of proxies, products, specs, beginning with `service_name` have `internal-dev` replaced with `PULL_REQUEST`. e.g. `my-service-internal-dev` -> `my-service-pr-1234`.

As of version 1.1, this simple namespacing substitution means that the proxy deployments to the `internal-dev` and `internal-dev-sandbox` specified by the `azure/azure-pr-pipeline.yml` file **MUST** match those specified in the `azure/azure-release-pipeline.yml` file. Trying to deploy a `product` an entry in `proxies` that does not exist on Apigee will result in a 400 response from Apigee, causing the playbook to error and exit.

7.14.10.5.3.3 Template-manifest

Constructs the `manifest.yml` from a `manifest_template.yml` file and validates it.

913 https://docs.apigee.com/api-platform/system-administration/using-oauth2

Usage

```
cd api-management-utils/ansible
DIST_DIR=/path/to/dist/dir make template-manifest
```

This play can be controlled with environment variables:

env var	required	usage
`DIST_DIR`	no	The directory ansible looks for the `manifest_template.yml` (and spec files).
`SERVICE_NAME`	no	For validation (see validate-manifest)

7.14.10.5.4 Multi spec deployments

You can deploy multiple specs for a single repo. Having multiple OAS specs is generally used for major version changes or if your API has lots of endpoints that warrant separate spec files.

To create a multi spec deployment you need to:

- Create an additional spec file in the "specification" folder in your repo
    - This new file will represent your spec, we encourage you to append a useful identifier (e.g. a major version number) on the end
- Add another "spec" item under the specs block in the manifest/manifest_template file
    - The "name" of the spec item must start with the name of your API
    - The "path" of the spec item must point to your spec in JSON format
- Update the "package.json" lint and publish steps to convert your spec from yaml to json format (if you use the spec linter and yaml2json converter)

An example of how to configure your repo to do multiple spec deployments can be seen here https://github.com/NHSDigital/booking-and-referral-fhir-api/pull/132/files#diff-984628f876f8c9971c75d2185a2b65008abb8bfa04bfb1539307446a540a8481R65-R67

7.14.10.6 Pipeline caching using the s3 cache task

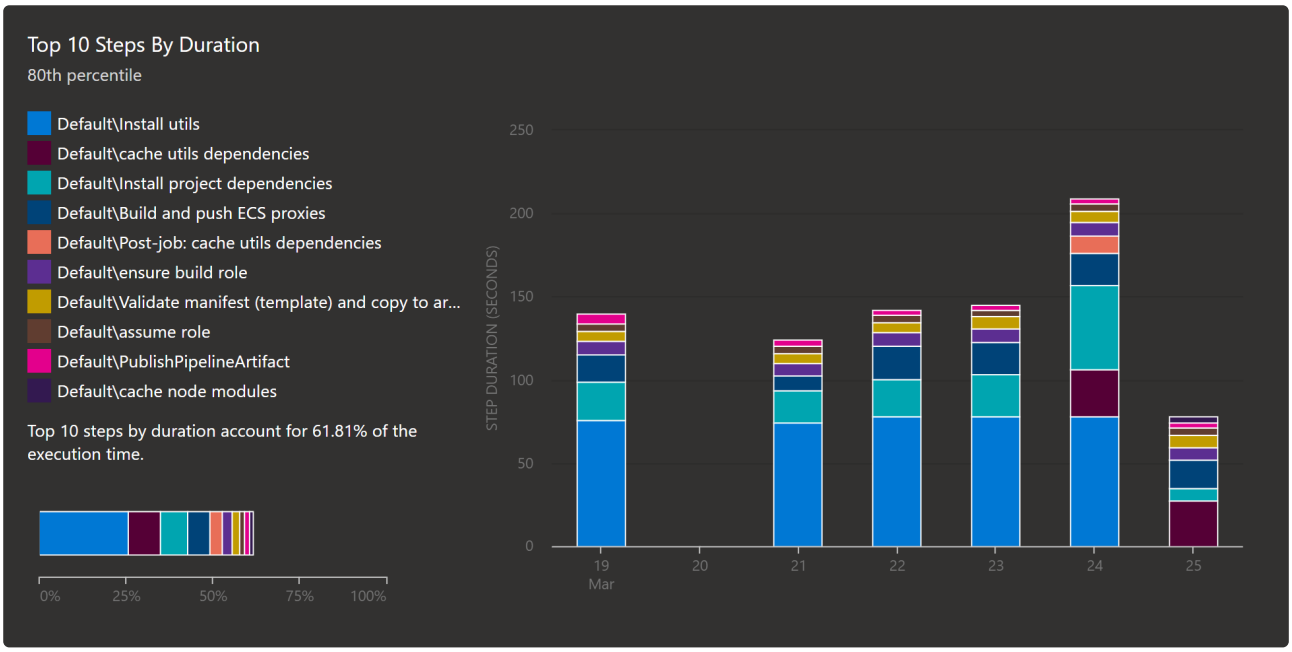
7.14.10.6.1 Overview

The S3 cache action is an Azure Pipelines extension to be used with build pipelines in Azure DevOps to give us better control of our caching dependencies when deploying APIs. The task is not exclusive to dependencies and can be used in any pipeline to cache a given file or folder.

7.14.10.6.1.1 Why use the cache task?

In our build pipelines the 'install-utils' step makes up between 30-60% of the deployment time. Using the cache task to restore dependencies radically decreases this time, saving about a minute per deploy on average.

See below example stats from 7 days of deploys on the 'Immunisation History API' build pipeline. The cache task was introduced on the last day.



7.14.10.6.2 Using the S3 cache task

The s3-cache-action task is designed to add an easy way to provide caching of dependencies. To demonstrate, let's examine the following build definition snippet:

S3 Cache Action Task

```
- task: s3-cache-action@1
```

```
    inputs:
        key: 'node | s3CacheTask/package-lock.json'
        location: 's3CacheTask/node_modules'
    displayName: cache node modules

  - bash: npm install
    condition: ne(variables['CacheRestored'], 'true')
    displayName: Install node dependencies
```

The task will internally hash the key to create a path to give to S3 and check for a cache hit. On a first run of the build the action will report a cache miss (as they have never been cached), the install dependencies step will run and the task will run a post-build job to upload the dependencies to S3. On the second run of the build the task will report cache hit, as the key exists in S3. This will extract the dependencies, use them and skip the install and post build steps.

Note: Using a condition on the install step is optional, as the install will report that the dependencies are already installed. The conditional logic can be used to be more explicit in the pipeline

7.14.10.6.2.1 Inputs

- `key` (required) : Key that will be hashed in S3. Pattern: Id (optional) | working directory | file pattern. Use file pattern to point to package-lock.json or poetry.lock to check changes to dependencies.
- `location` (required) : Path to file or folder to be cached.
- `bucket` (optional) : S3 bucket for cache task to use. By default uses environment to cache to NHSD pipeline bucket.
- `pipelineIsolated` (optional) : Adds pipeline id to S3 to make cache only valid for the pipeline using it. Set false by default.
- `debug` (optional) : Set true to turn on logging information for the task. Set false by default.
- `alias` (optional): This must be used if using the task more than once in a pipeline and you are using the conditional logic for installs. This is appended to the 'CacheRestored' variable and should then be used in the condition for the install step. (Example below).

Multiple S3 cache tasks

```
  - task: s3-cache-action@1
    inputs:
        key: 'node | s3CacheTask/package-lock.json'
        location: 's3CacheTask/node_modules'
        pipelineIsolated: false
        debug: true
        alias: 'NodeModules'
    displayName: cache node modules

  - bash: npm install
    condition: ne(variables['CacheRestored-NodeModules'], 'true')
    displayName: Install node dependencies
```

```
  - task: s3-cache-action@1
    inputs:
        key: 'python venv | ./poetry.lock'
        location: '.venv'
        pipelineIsolated: false
        debug: true
        alias: 'Poetry'
    displayName: cache virtual env

  - bash: poetry install
    condition: ne(variables['CacheRestored-Poetry'], 'true')
    displayName: Install poetry dependencies
```

7.14.10.6.3 Using the S3 cache task in build pipelines

The S3 cache task is used in the common pipelines on the `apigee-build` pipeline. The pipeline uses the task to cache `utils` dependencies but has the capability to cache the dependencies of the API during the build using the `cache_steps` parameter. The build runs the `cache_steps` before installing and a user can pass a cache task or multiple cache tasks for dependencies. The build pipeline will cache these dependencies but will also install any not specified.

7.14.10.6.3.1 Example from Canary API

Canary API build pipeline snippet

```
...
variables:
  - template: project.yml
extends:
  template: azure/common/apigee-build.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    cache_steps:
      - task: s3-cache-action@1
        inputs:
          key: 'node modules | ${{ variables.service_name }} | package-lock.json'
          location: '${{ variables.service_name }}/node_modules'
          debug: true
          alias: 'NodeModules'
        displayName: cache node modules
      - task: s3-cache-action@1
        inputs:
          key: 'poetry | ${{ variables.service_name }} | poetry.lock'
          location: '${{ variables.service_name }}/.venv'
```

```
          debug: true
          alias: 'Poetry'
        displayName: cache python dependencies
```

7.14.10.6.4 S3CacheAction Class

The code for the S3 Cache Task is hosted at the `apim-s3-cache-action`[914] repository and uses the S3CacheAction class for interacting with S3. The class can be used for cache interactions with S3.

7.14.10.6.4.1 Requirements

The class can be instantiated using an `options` object to set up the AWS client and bucket.

If you have your AWS credentials globally configured then s3Client option can be left blank and will use your credentials ambiently.

S3CacheAction class options

```
const options = {
    s3Client: new AWS.S3({
            credentials: {
                accessKeyId: 'test-id',
                secretAccessKey: 'test-secret',
            }
        }),
    bucket: 'example-bucket'
```

7.14.10.6.4.2 Methods

createCacheKey

The S3CacheAction caching interactions must be provided with a key string with a specific format.

The `createCacheKey` method will then hash the key parts an join them separated by '/' characters, which creates a file tree within the provided S3 bucket.

If the key part is a file it will hash the file, otherwise it will hash the string.

The task with prefix the key with 'global' by default or 'pipeline' if the `pipelineIsolated` input is set to `true.`

914 https://github.com/NHSDigital/apim-s3-cache-action

Pattern

Identifier | Working Directory | File Pattern

Examples

Cache Key examples

```
// Input
'"node modules" | $(System.DefaultWorkingDirectory) | package-lock.json'
// Method output
'a08e14f4ae7761bd5b85b22f235f9b2e6a489ed7b71904daf8069c225abf983c/
45549dbb28f29efdb4b8aeb2b69088d8fa34693a2cc3597fe2389eecc8b17742/
dba27c31aad935787bb275c3e5e4e957708f15386de599eff1db476022cd7e4c'
// S3 prefixed by task
'global/a08e14f4ae7761bd5b85b22f235f9b2e6a489ed7b71904daf8069c225abf983c/
45549dbb28f29efdb4b8aeb2b69088d8fa34693a2cc3597fe2389eecc8b17742/
dba27c31aad935787bb275c3e5e4e957708f15386de599eff1db476022cd7e4c'

// Input
'Test data | $(System.DefaultWorkingDirectory) | testData/test.json'
// Method output
'e27c8214be8b7cf5bccc7c08247e3cb0c1514a48ee1f63197fe4ef3ef51d7e6f/
8a4f7378bf9f77ee01e78ff0dc31ff5669358a6fe0198be9a6f859999b9d50f2/8253544304dab00d2a07
0de771567c2bf6c5decc4120424324dfbc8169c4e63a'
// S3 prefixed by task
'global/e27c8214be8b7cf5bccc7c08247e3cb0c1514a48ee1f63197fe4ef3ef51d7e6f/
8a4f7378bf9f77ee01e78ff0dc31ff5669358a6fe0198be9a6f859999b9d50f2/8253544304dab00d2a07
0de771567c2bf6c5decc4120424324dfbc8169c4e63a'
```

createCacheEntry

The `createCacheEntry` method uploads a file or directory as cache entry in S3. It requires a `targetPath` (path to file or directory you want to cache) and a `keyName`, which should the key hashed using the `createCacheKey` method. The method packs the file or directory into a tar and streams the tar to S3.

maybeGetCacheEntry

The `maybeGetCacheEntry` method retrieves a cache entry from S3 using the hashed key it was uploaded with. It requires the hashed `keyName`, which should the key hashed using the `createCacheKey` method. It also requires a `destination`, which is the destination for the cache entry to be extracted to. The method calls the `findCacheEntry` method, which returns a stream of the desired cache entry and then pipes the stream to the destination directory. It will report a cache hit on a success.

Cache miss

If there is no matching cache entry of the provided key the function does not error but reports a cache miss so that the task can decide to then upload the cache entry in this situation.

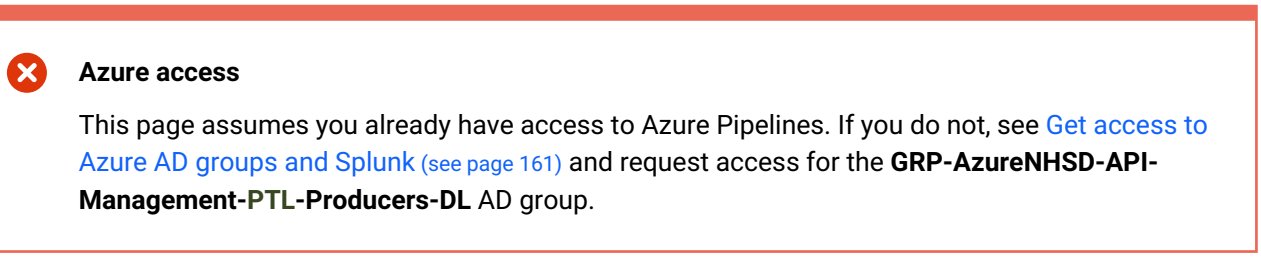
Fix Python env

In the situation returned cache entry is a 'python virtual environment' the `maybeFixPythonVenv` method will check for python files with a 'shebang path' and rewrite the path to specify the path the cache entry has been extracted to.

7.14.10.7 Using manual approval on pipelines

7.14.10.7.1 Overview

Manual approval is used on deployment pipelines to act as a gate for deploying to certain environments. Approval gates can have approvers designated to avoid the wrong person deploying to an environment, which is especially useful for releases to production. By default, production deployments have a manual approval gate set with only API Platform admins as approvers, but this can be configured to allow API producer teams to have their own approver team and therefor allow producers to deploy their own proxies to production. It is also possible to turn manual approval off for production, though this is not recommended. Manual approval can also be set for path-to-live environments with alternate approver groups configured.

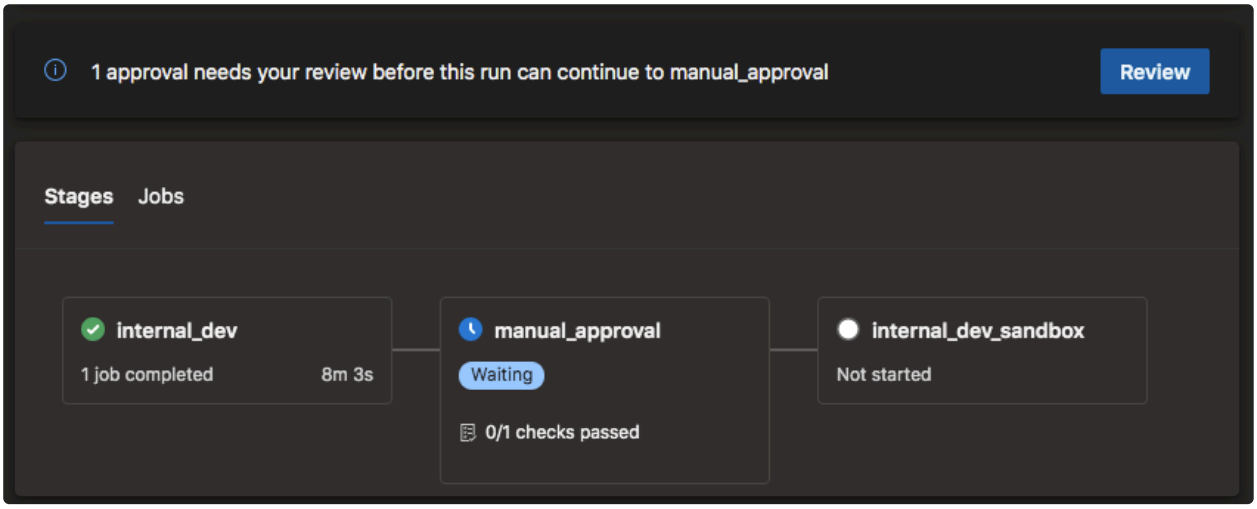


7.14.10.7.2 Approach

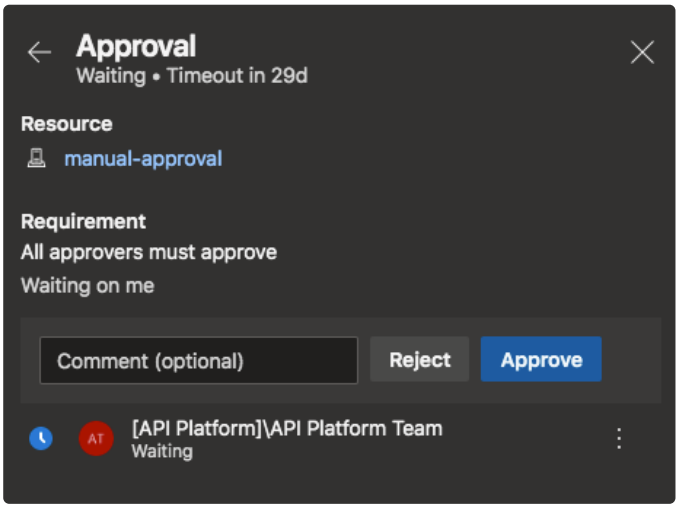
In order to have approval on environments, they must be set directly on the environment in the Azure DevOps UI, which must have designated approvers. The problem with directly setting the approval on the 'prod' environment is that this approval would be set across the board for all APIs and doesn't allow for flexibility.

To get around this problem the 'manual-approval' environment was created, acting as a dummy environment, deploying nothing but serving as a gate between environments. The approver must review in the Azure DevOps UI (User Interface). The approval has a time-out, which is set to 30 minutes but appears as 29 days for some users, which is a UI bug.

7.14.10.7.2.1 Example use for 'manual-approval' gate in Azure DevOps UI



The deployment process is interrupted by the "dummy environment" making the approval necessary to continue with the deployment flow.



The designated approver(s) can add comments; and reject or approve the deployment.

7.14.10.7.3 Manual approval for production

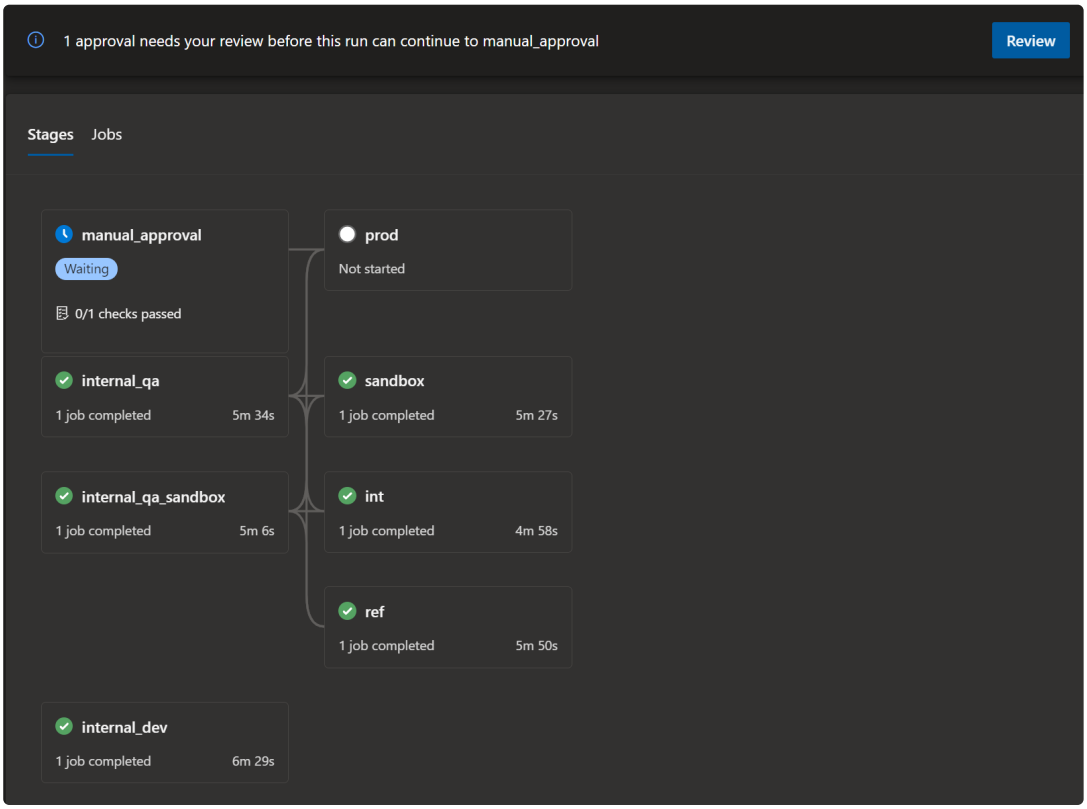
7.14.10.7.3.1 Default behaviour

Manual approval for prod is controlled by the `prod_requires_approval` parameter in the release pipeline config found at `azure/azure-release-pipeline.yml` in your repo.

The parameter is set to true by default, so not specifying it will give you the default behaviour. For example:

```
extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    apigee_deployments:
      - environment: internal-dev
      - environment: internal-qa
      - environment: internal-qa-sandbox
      - environment: sandbox
      - environment: int
        depends_on:
          - internal_qa
          - internal_qa_sandbox
      - environment: ref
        depends_on:
          - internal_qa
          - internal_qa_sandbox
      - environment: prod
        depends_on:
          - internal_qa
          - internal_qa_sandbox
```

Once the release pipeline is run the stage configuration will look like:



This will put a manual approval gate before production and require internal qa and internal qa sandbox to deploy before the production deployment is tarted. Only once those environments have passed the manual approval stage will it require approval. Once approved, production will begin to deploy. If rejected, the production deployment will be skipped.

Turn off default behaviour (optional - not recommended)

To turn off manual approval on prod the `prod_requires_approval` parameter must be set to false in the parameters for the release pipeline.

```
extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    prod_requires_approval: false
```

7.14.10.7.3.2 Self-serve production deployments

By default, the manual approval gate will only have API Platform admins as approvers, meaning only the Platform team can deploy your proxy to production. If you wish to have the capability to deploy your proxy to production yourself, this can be configured with the following steps:

Ask the API Platform team to set up your Azure approver group

Request an approval group by using the 'Self-serve Release Conf' workflow in the *#platforms-apim-producer-support* channel workflow (see Get access to Slack and support channels (see page 151) for access). For guidance on how to use workflows please see the pinned message in the channel.

The workflow will ask you to provide the following:

- The name of your API and a link to your GitHub repo.
- The name and shortcode email of a designated admin of your team (e.g. WXYZ9@hscic.gov.uk)
    - Optionally - a list of names and shortcode emails for other approvers on the team. This can also be managed later by the admin above. See step 4 for more info.

Wait for confirmation your Azure approver group has been set up

It is important to wait until your approver group has been set up before continuing to the next step or you will see errors with your production deployment.

Configure your release pipeline

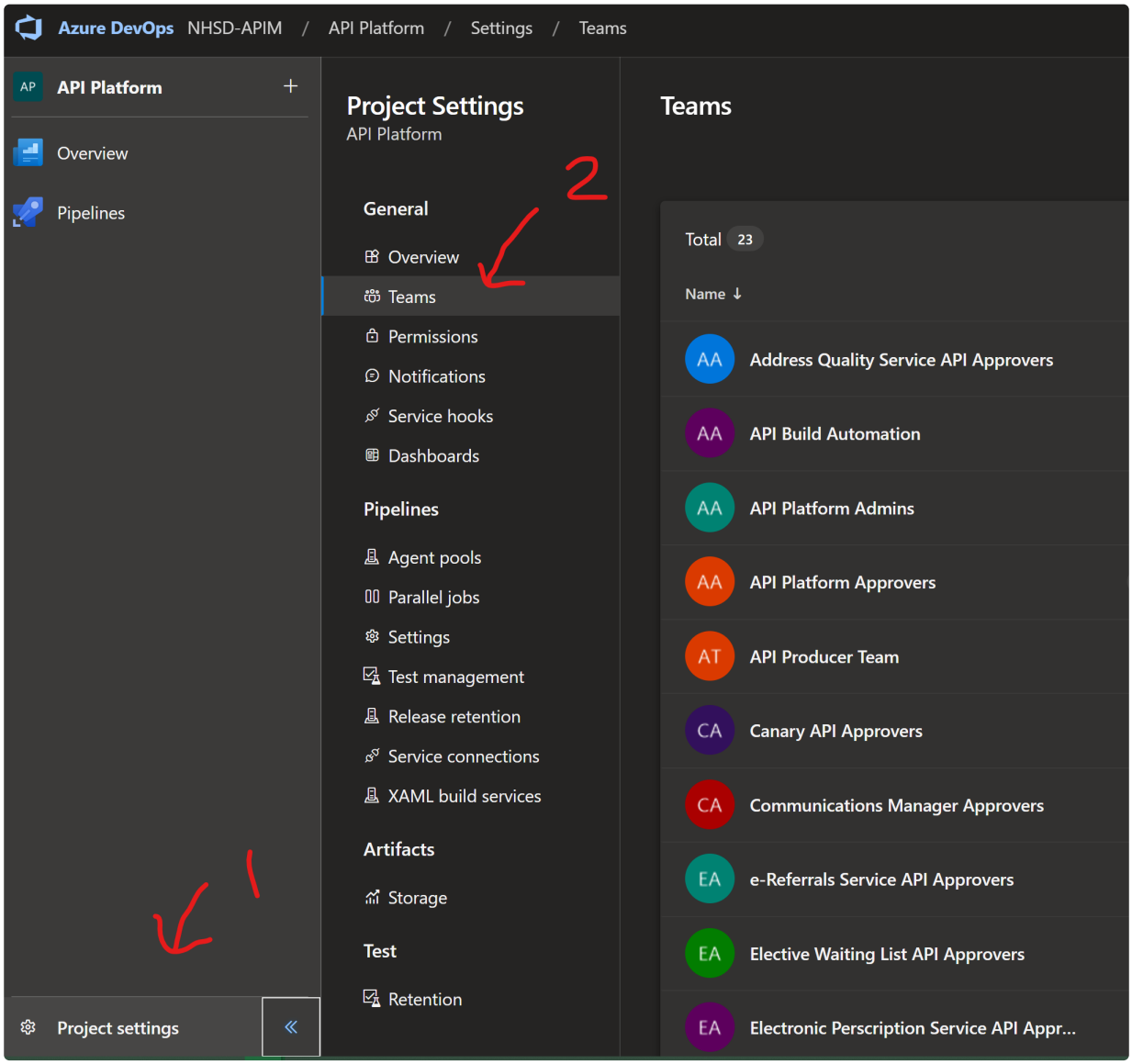
On your release pipeline found in your repo at `azure/azure-release-pipeline.yml,` add `prod_producer_approval` as a parameter set to `true`. This will pass your service name to the common pipelines as the name of your manual approval gate, which the Platform team will have set up your approver team to approve.

```
extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    prod_producer_approval: true
```

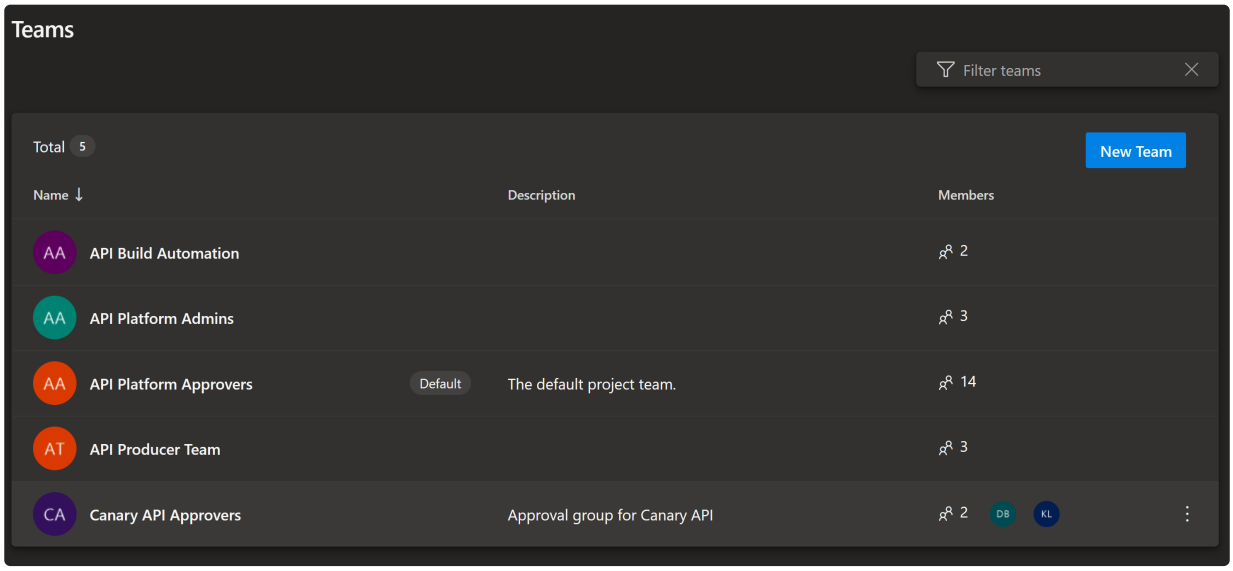
Add your team as approvers

To manage your approval team in Azure.

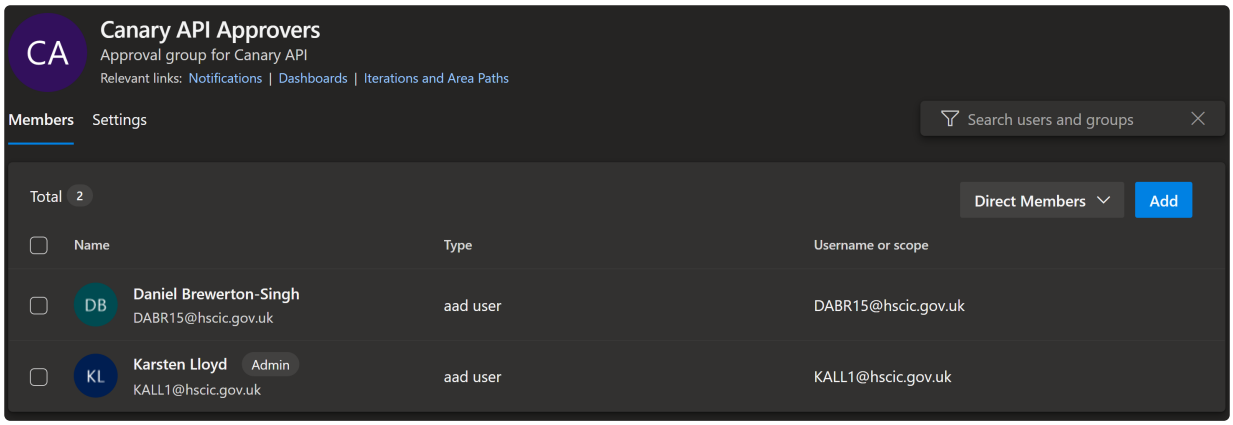
Navigate to Project settings > Teams



Cho426pxose your team. This should be '<Your API name> Approvers'



As an admin of the team you can add new members using their shortcode email or elevate other users to admins

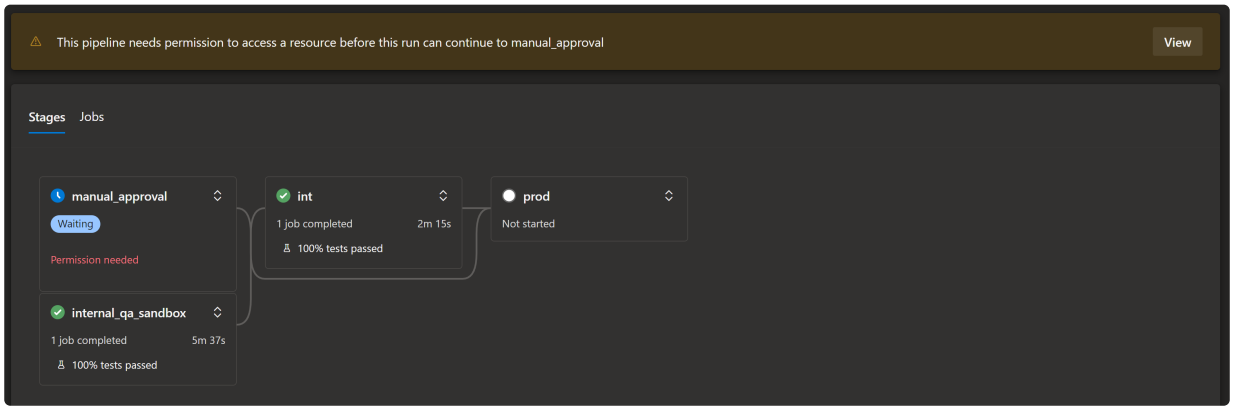


Deploy to production

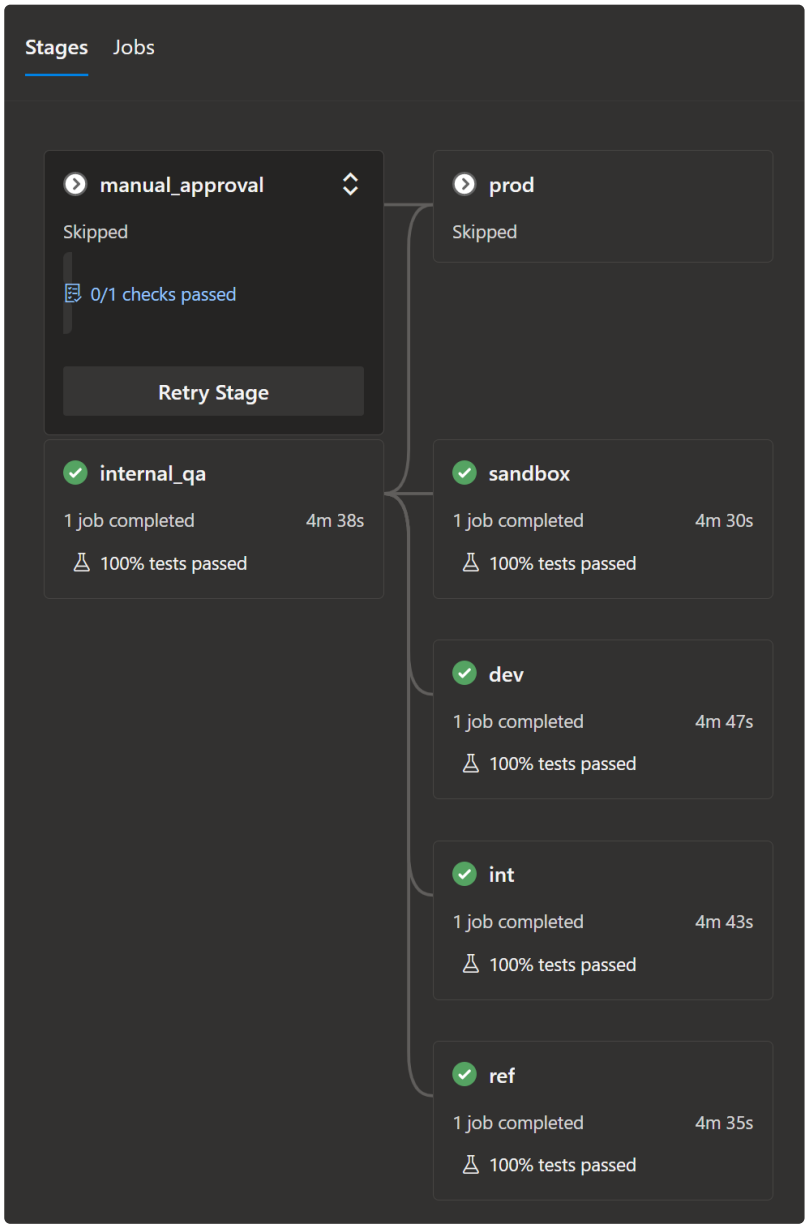
When you merge a pull request on your GitHub repo the release pipeline will automatically trigger and release to your path-to-live environments with production blocked pending on the manual approval gate. As you will likely want to release to production later than the path-to-live environments, the manual-approval will time out after 30 minutes. When it's time to release to production:

Get the environment approved

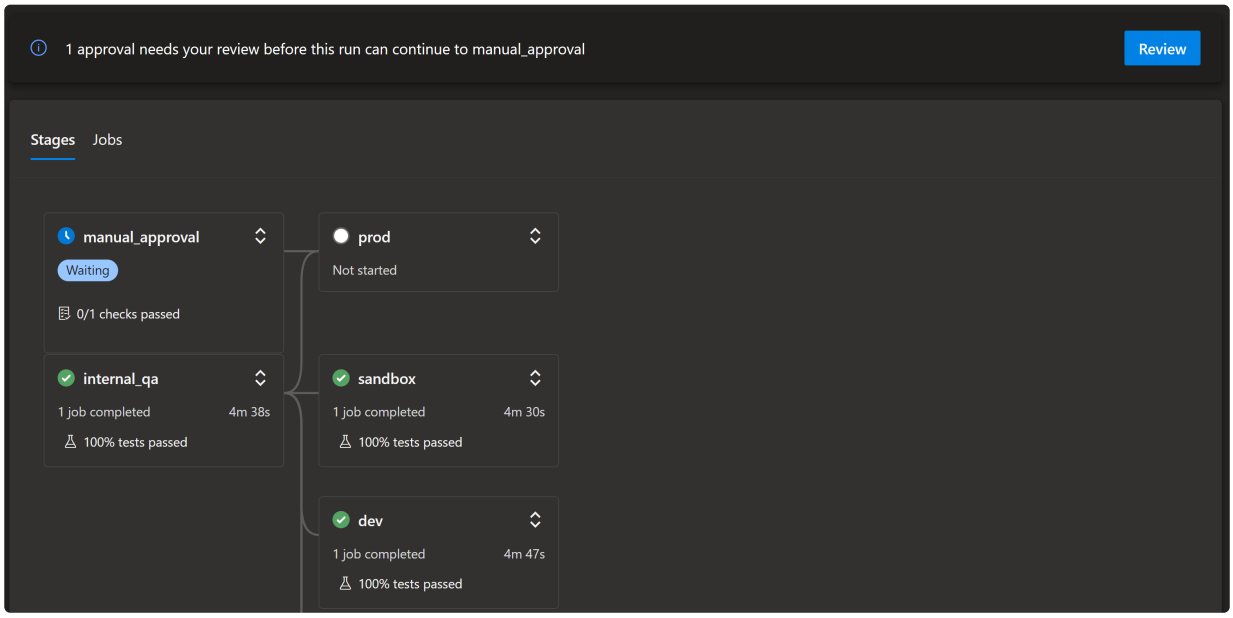
When you first deploy using the environment it needs the permissions granting by an admin. Post in the *#platforms-apim-producer-support* channel (see Get access to Slack and support channels (see page 151) for access) with a screenshot asking for the environment to be enabled to access the resource



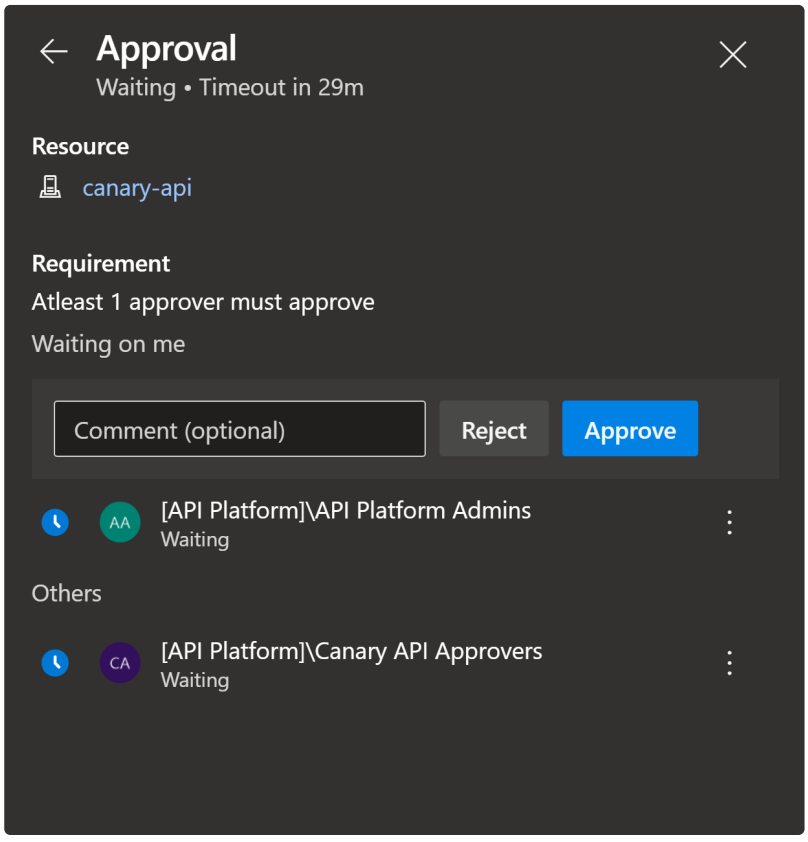
Click 'Retry stage' on the manual_approval gate



Click 'Review'



Click 'Approve'. This will trigger the prod environment to deploy



7.14.10.7.4 Manual approval for path-to-live and multiple environments

Manual approval gates can also be used for path-to-live environments. To implement the feature for a different environment (e.g. 'int') a producer must:

- Define `manual-approval` as a new environment in the `azure/azure-pr-pipeline.yml` or `azure/azure-release-pipeline.yml` file. The value must exactly be `manual-approval`
- Add a `stage_name`. This must use underscores and append the environment which will depend on it e.g. `manual_approval_int`.
- Add `depends_on` to the target environment with the value of the stage name.
- Add the other environments the target environment would usually depend on.

azure-release-pipeline.yml (In example API)

```
apigee_deployments:
      - environment: manual-approval
        stage_name: manual_approval_int
      - environment: int
          depends_on:
            - manual_approval_int
            - internal_qa
            - internal_qa_sandbox
```

This can be done with multiple environments and can also be done with the `prod_requires_approval` parameter set to true.

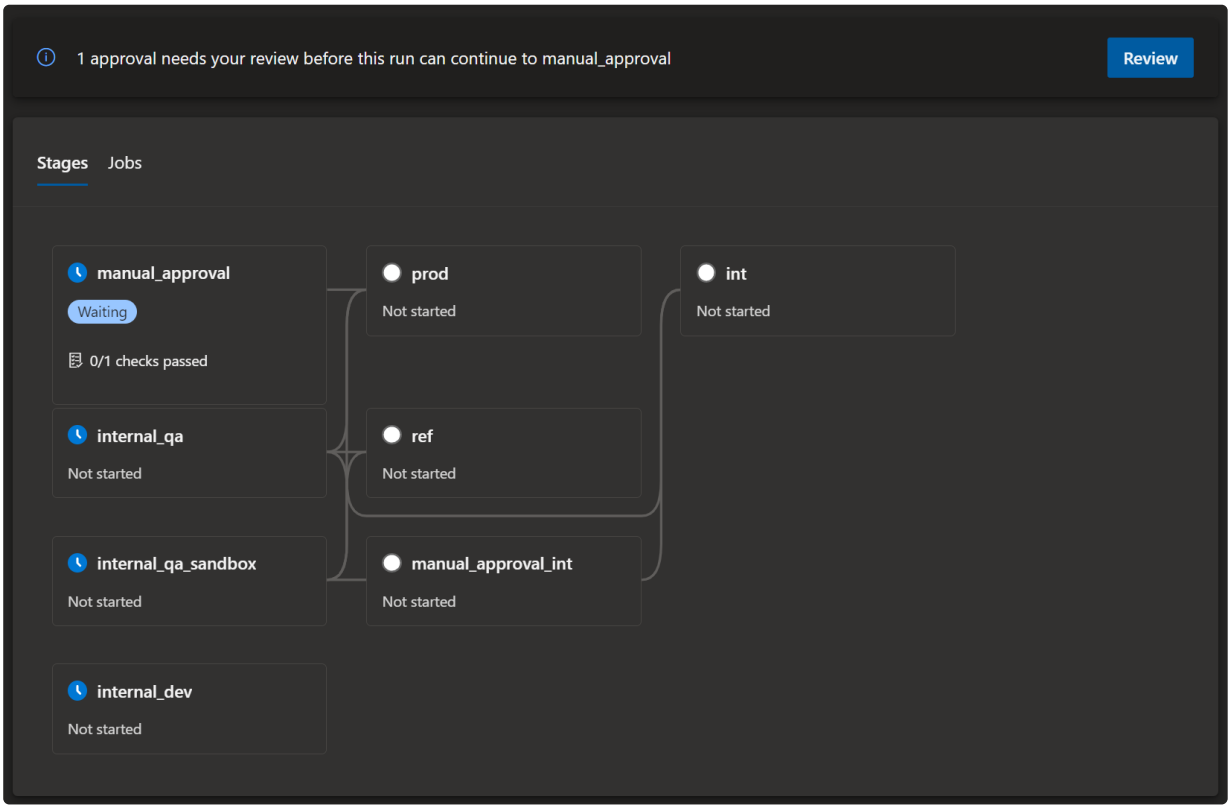
Every time your specify manual-approval for a different environment, the stage name must be unique, which is why it is recommended to append the environment.

azure-release-pipeline.yml (In example API)

```
    apigee_deployments:
      - environment: internal-dev
      - environment: internal-qa
      - environment: internal-qa-sandbox
      - environment: sandbox
      - environment: manual-approval
        stage_name: manual_approval_int
      - environment: int
          depends_on:
            - manual_approval_int
            - internal_qa
            - internal_qa_sandbox
      - environment: manual-approval
        stage_name: manual_approval_ref
```

```
    - environment: ref
      depends_on:
        - manual_approval_ref
        - internal_qa
        - internal_qa_sandbox
    - environment: prod
      depends_on:
        - internal_qa
        - internal_qa_sandbox
```

7.14.10.7.4.1 An example using 'prod_requires_approval' and an addition manual approval for int



7.14.10.7.4.2 Using alternative manual approval groups for path-to-live environments

It is possible to use other manual approval gates other than the common 'manual-approval'. Example of this is where different approvers are needed other than the Platforms team members who can approve the default gate or if you need a separate set of approved for path-to-live and production.

To use another gate you can use the `manual_approval_env` parameter on the pr or release pipeline, which will overwrite the default manual-approval gate. This will only work with gates that exist on Azure DevOps already. For example the 'manual-approval-ptl' gate, can be approved by anyone who has access to the platforms instance of Azure. It is possible to create a new gate if there is a special need for this and in this case you should contact the Platform team via the *#platforms-apim-producer-support* channel (see Get access to Slack and support channels (see page 151) for access).

Example use of the `manual_approval_env` parameter is shown below for the int environment:

Things to note:

- You must still specify `manual-approval` as the environment used for the target environment to depend on.
- You must still specify a unique stage name.
- You must still add the other environments the target environment would usually depend on.

manual_approval_env usage

```
parameters:
  service_name: ${{ variables.service_name }}
  short_service_name: ${{ variables.short_service_name }}
  service_base_path: ${{ variables.service_base_path }}
  enable_status_monitoring: true
  enable_monitoring: true
  manual_approval_env: manual-approval-ptl
  apigee_deployments:
    - environment: internal-dev
    - environment: internal-dev-sandbox
    - environment: internal-qa
    - environment: internal-qa-sandbox
    - environment: sandbox
    - environment: dev
    - environment: manual-approval
      stage_name: manual_approval_int
    - environment: int
      depends_on:
        - manual_approval_int
        - internal_qa
        - internal_qa_sandbox
    - environment: ref
    - environment: prod
```

7.14.10.8 Accessing your deployed API

7.14.10.8.1 Overview

Once you have deployed your API, it's useful to test it out by following the process a consumer follows when getting set up to call your API. This is useful for understanding how a consumer will onboard and then use your API and get you set up for any manual testing you might want to do of your endpoints.

Depending on the environment being accessed and your configuration, access to your API is either granted automatically or is gated (usually for production).

7.14.10.8.2 Access your deployed API as a consumer would

To access your API as a consumer would, head to the developer portal[915] and go through the following steps:

1. If you do not already have one, create a developer account[916].
2. Navigate to my developer account[917] and sign in.
3. It is important when creating an application that these are not owned by an individual, so if you don't have one you will need to create a team first. To create a team, select 'My applications' and 'Manage my teams'. Select '+Team' to create a team and add in the emails addresses of people for your team.
4. If you have an existing team set up then, select 'My applications' and 'Manage your applications', or go directly to 'Manage your applications' if you've had to set up a new team.
5. Select 'New app'. If you have an existing app you can add your new API to that app provided it is for the same environment. In which case you'd select the existing app and go to step 9.
6. Enter details for your application and click 'add application' to create it.
7. Select your application and then 'edit more application details'.
8. Make a note of the App ID and API Key (aka client ID).
9. In the list of APIs, find the API you want to use and activate it by clicking 'Enable' and 'save'.
    a. The list is updated every hour, on the hour.

Depending on the authorisation/authentication pattern your API uses, you can read more about how to access your API on the following pages (found on the developer hub[918])

7.14.10.8.2.1 Application-restricted

- API key authentication[919]
- Signed JWT authentication[920]

7.14.10.8.2.2 Health-care worker (user-restricted)

- CIS2 combined authentication and authorisation[921]

915 https://digital.nhs.uk/developer
916 https://onboarding.prod.api.platform.nhs.uk/Index
917 https://onboarding.prod.api.platform.nhs.uk/Index
918 https://digital.nhs.uk/developer
919 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-api-key-authentication
920 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication
921 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-combined-authentication-and-authorisation

- CIS2 separate authentication and authorisation[922]

7.14.10.8.2.3 Patient access (user-restricted)

- NHS Login combined authentication and authorisation[923] (Currently not supported by NHS Login)
- NHS Login separate authentication and authorisation[924]

7.14.10.8.2.4 External portal vs Internal portal

We have both an external and internal Apigee organisation (each has it's own portal). It can be useful to use each, depending on what environment you wish to manually test your API with.

- Internal portal[925]
    - For producer development and testing
    - For environments:
        - internal-dev
        - internal-dev-sandbox
        - internal-qa
        - internal-qa-sandbox
        - ref
- External portal[926]
    - For public consumer use
    - For environments:
        - prod
        - int
        - sandbox
        - dev

7.14.10.8.3 Manual vs auto approval for consumer applications subscribing to a product in Apigee

When subscribing an app to an API, Apigee provides two options:

---

922 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-cis2-separate-authentication-and-authorisation

923 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-combined-authentication-and-authorisation

924 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/user-restricted-restful-apis-nhs-login-separate-authentication-and-authorisation

925 https://dos-internal.ptl.api.platform.nhs.uk/

926 https://onboarding.prod.api.platform.nhs.uk/

1. Automated approval - When consumers subscribe in the portal they can subscribe by clicking 'Enable' and can start using the API straight away.
2. Manual approval - When consumers subscribe in the portal they can subscribe by clicking 'Request' and must then ask the Platform team to enable their app via their onboarding lead. (If you want to know more about the onboarding process for consumers see - App enabled into production (see page 1025))

By default, we use the following approach for all APIs on the platform:

- For **sandbox**, there is no need for developers to subscribe at all, the API does not require authentication. Therefore we suggest setting the `products.attributes.access` field to private for sandbox products (See Manifest.yml reference (see page 896) for details)
- For **integration test**, approval is **automatic**, so consumers can self-serve
- For **production**, approval is **manual** - we need to do various checks in the onboarding process before allowing access

7.14.10.8.3.1 Manual approval

In most cases, approval for API subscriptions in production needs to be manual. This allows the onboarding process to be executed before granting production access.

In this case, the application will show up as "Pending" in Apigee. We do not get notified about this, the "nominated" onboarding lead will need to raise a producer support request via slack, so we can approve this for you (as described in App enabled into production (see page 1025)).

7.14.10.8.3.2 Automatic approval for production

Some APIs might not need a manual process, for example, if they give access to open (non-personal) data. ODS is a good example - organisation data is in the public domain anyway, so no need to manually approve API access.

If you want your API to have auto-approval in production, get in touch with us in the usual way, quoting KOP-031 Granting prod auto-approval on products for an API[927], and we'll set it up for you.

You **cannot** change to auto approval yourself by updating the approval_type value in the manifest.yml file, as this will be overwritten to manual in the pipeline.

7.14.10.9 Source control

The source code for the APIs is held on GitHub[928], within the NHS Digital organisation. It is publicly visible - we are "Coding In The Open" as recommended in the GOV.UK Service Manual[929].

927 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/227246300/KOP-031+Granting+prod+auto-approval+on+products+for+an+API
928 https://github.com/NHSDigital/
929 https://www.gov.uk/service-manual/service-standard/point-12-make-new-source-code-open

An example is the PDS API. https://github.com/NHSDigital/personal-demographics-service-api

For access and setup on GitHub see Set up your development environment (see page 153).

7.14.10.9.1 Rules of contribution

As this code is visible to the public, several things are vital:

1. Never publish secrets on any branch.
    a. If secrets are published, remove any references in existing code and rotate secrets immediately.
2. Check the licenses of libraries you add.
    a. GPL or Affero GPL libraries may not have compatible licenses with the MIT/OGL License of our repository or may require the GPL to be published along with the repo.
3. Watch your language in issues, pull-requests, code, and commit messages
    a. Imagine everything you write will be read by the writers of popular newspapers

7.14.10.9.2 Important Branches

7.14.10.9.2.1 master

master is the default branch and source of truth and should be protected with merge rules

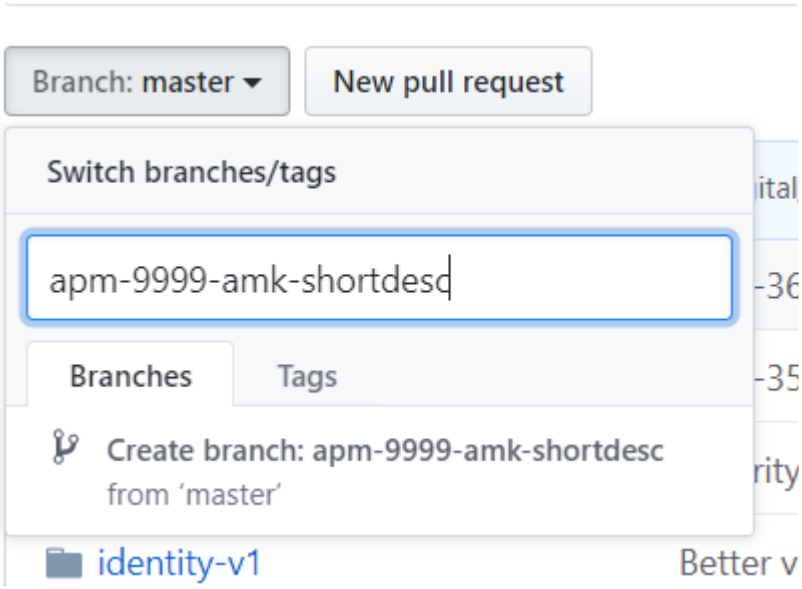
master merge rules

1. Must be reviewed by at least 1 peer
2. Must pass all status checks

7.14.10.9.3 Starting a new branch

All feature/bug/whatever branches should conform to the following convention:

```
<jira reference>-<short-description-here>
apm-123-my-description
```

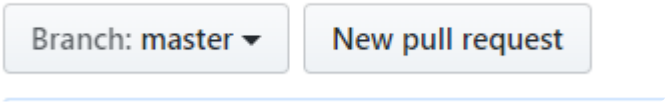


7.14.10.9.4 Commit messages

All commit messages should conform to the following convention:

```
APM-123 A short description of the commit

An optional longer description that
can span over multiple lines.
```

7.14.10.9.5 Pull Requests



By default, a pull request template should appear when raising a pull request. It should look something like this:

```
## Summary* Routine Change
* :exclamation: Breaking Change
* :robot: Operational or Infrastructure Change
* :sparkles: New Feature
* :warning: Potential issues that might be caused by this change

Add any other relevant notes or explanations here. **Remove this line if you have
nothing to add.**

## Review Checklist
```

```
:information_source: This section is to be filled in by the **reviewer**.

* [ ] I have reviewed the changes in this PR and they fill all or part of the
acceptance criteria of the ticket, and the code is in a mergeable state.
* [ ] If there were infrastructure, operational, or build changes, I have made sure
there is sufficient evidence that the changes will work.
* [ ] I have ensured the changelog has been updated by the submitter, if necessary.
```

Make sure you complete this appropriately!

Example:

```
## Summary
* Foo'd the bar
* :exclamation: Broke the build
* :robot: Added sleeps into build process to slow it down
* :sparkles: Rainbow marquees
* :warning: I wrote this while drunk

You shouldn't accept this pull request under any circumstances!

## Review Checklist
:information_source: This section is to be filled in by the **reviewer**.

* [ ] I have reviewed the changes in this PR and they fill all or part of the
acceptance criteria of the ticket, and the code is in a mergeable state.
* [ ] If there were infrastructure, operational, or build changes, I have made sure
there is sufficient evidence that the changes will work.
* [ ] I have ensured the changelog has been updated by the submitter, if necessary.
```

7.14.10.9.5.1 Pull request title

Please include:

1. The ticket name
2. A short but descriptive title

7.14.10.9.6 Reviews

All changes, including both code and documentation, **must** be reviewed before merging into master.

7.14.10.9.7 Utils repository branching strategy

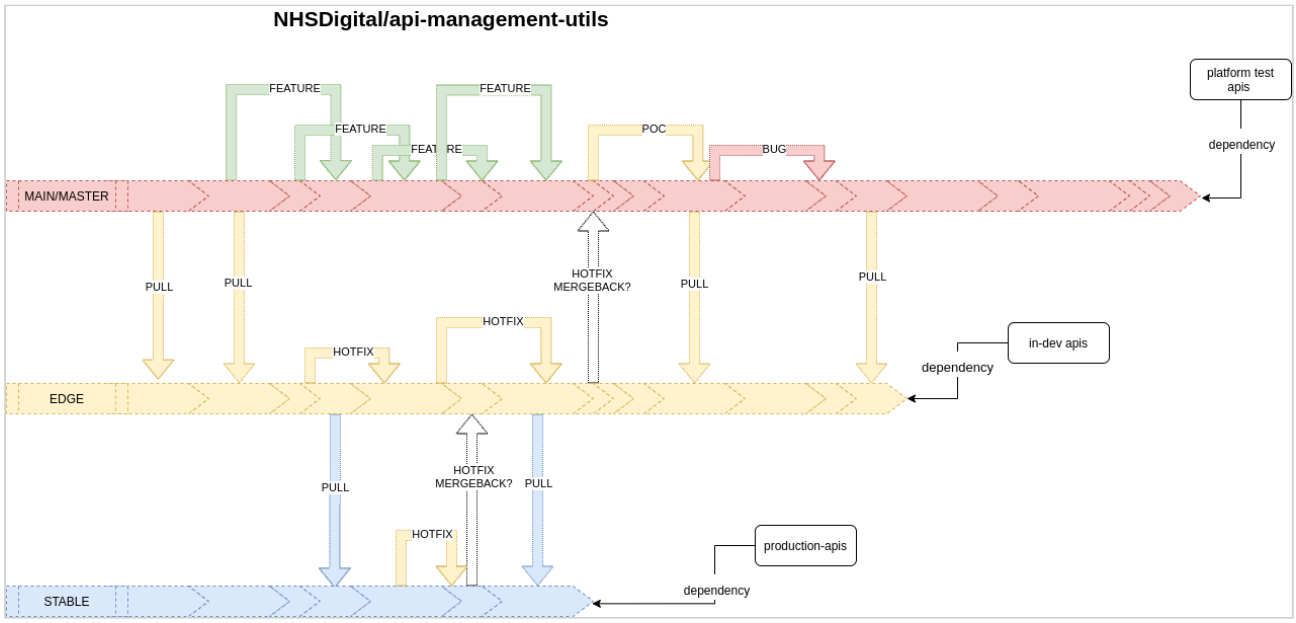
Managed API deployments are use the NHSDigital/api-management-utils (https://github.com/NHSDigital/api-management-utils) repo for build and deployment orchestration

In order to support change and stability for existing consumers we propose to introduce the following branching / tagging strategy,

7.14.10.9.7.1 Branches

branch	purpose	breaking changes	experimental / poc / features	immutable	can hotfix / cherrypick
main / master / develop (name change ?)	primary integration branch, avoid long running feature branches	yes	yes	no	n/a
tags/vX.X.X	tagged for every change to the 'main' branch	yes	yes	yes	no
edge	merged from master / cherry picked after successful testing and feature completeness	yes	no	no	yes
stable	merged from edge after successful testing or bedding period for features	only after deprecation warnings for a period?	no	no	yes

7.14.10.9.7.2 NHSDigital/api-management-utils - branch flows



7.14.11 Alpha exit review

The last step of completing Alpha is to hold an exit review with your producer liaison (currently Jonathan Eagle). This will likely be conducted during a regular catch up review meeting.

This is to:

- conduct a peer review of the activities you've done during alpha - as per Alpha checklist (see page 198)
- agree you've done everything necessary to exit alpha
- decide whether to proceed to beta

To request a review, contact us - see Help and support (see page 2056).

7.14.12 Proxygen CLI user guide

7.14.12.1 Overview

The Proxygen CLI is a dedicated command-line interface tool designed to streamline the interaction between producers and the Proxy Generator service. It provides producers with a convenient and intuitive way to deploy API instances, manage API specifications, manage secrets, and deploy them within specific environments without needing to directly interact with the API platform.

This page describes the functions of the Proxygen CLI and how to use the tool.

You need to be set up with proxy generator before using the Proxygen CLI. Refer to Getting setup with proxy generator[930] for more details.

7.14.12.2 Getting started

7.14.12.2.1 Installation

The Proxygen CLI is a python package.

To install Proxygen CLI, enter the following command in your terminal:

Install proxygen-cli

```
pip install proxygen-cli
```

If you wish to see the version of the Proxygen CLI installed (it can be useful for debugging purposes), enter the following command:

View version

```
proxygen --version
```

7.14.12.2.2 Configuring settings and credentials

To get started you will need to specify the name of the API you are developing. This is the kebab-case api name used to set up your api originally (see Getting set up with proxy generator (see page 681)).

To set this, use the following command:

Specify the API name

```
proxygen settings set api <API-NAME>
```

There are two ways to authenticate via the Proxygen CLI:

1. Using your user login credentials such as a username and password.

930 https://nhsd-confluence.digital.nhs.uk/display/APM/Getting+set+up+with+proxy+generator?src=contextnavpagetreemode

2. Using your private key and client_id.

7.14.12.2.2.1 Setting up for user access

If you are setting up user access for the first time you will need to:

1. Request a user account. To do this contact the platforms-api-producer-support[931] channel asking for a proxygen user account to be set up and provide the following details:

- your nhs.net email address
- the proxygen-managed api/s your account will need access to

To setup the Proxygen CLI credentials with user access enter the following command:

Proxygen CLI credentials

```
proxygen credentials set
```

2. The CLI will ask you for your username, and password. These credentials will be securely stored on your local machine in the following directory:

- ~/.proxygen/credentials.yaml

The client_id and secret used for user access will be automatically added to this file after running the command.

You must have permissions to manipulate instances/secrets/specs for the API or commands will fail. If your permissions are incorrect, contact the API platform team via the platforms-api-producer-support[932] channel.

Production deployments

Proxygen CLI user access only allow deployment to non-prod environments. Machine user access is required for production environment deployments.

7.14.12.2.2.2 Setting up for machine-user access

After setting up your API using the instructions at Getting set up with proxy generator (see page 681), you should have the following:

- a private key from a key pair for machine user access
- the key id for that specific key pair
- the client_id for your api machine user access (usually `<your-api-name>-client`)

Using this information enter the following command:

931 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY
932 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY

Machine user access

```
proxygen credentials set private_key_path <PATH_TO_PRIVATE_KEY> key_id
<KEY_ID_FOR_PRIVATE_KEY> client_id <MACHINE_USER_CLIENT_ID>
```

The CLI will ask you for your username and password. For machine authentication, the username and password **must** be left blank. The credentials will be securely stored in the following directory:

- ~/.proxygen/credentials.yaml.

If you are switching between using user access and machine-user access remember:

- proxygen-cli will favour user access if username and password credentials are set
- user access uses `proxygen-cli-user-client` as a client_id. **If you switch between access modes remember to switch the client_id**
- the value for client_secret should remain untouched as it is not used for machine-user access
- if you lose the client_secret for the `proxygen-cli-user-client`, contact the platforms-api-producer-support[933] channel

7.14.12.2.2.3 Configuring credentials and settings file

It is possible to manually set up your credentials and api details without using commands. The Proxygen CLI stores credentials in:

- ~/.proxygen/credentials.yaml file.

Edit the file as shown below:

Manually set up credentials

```
client_id: example-test-client
private_key_path: /home/APIM/creds/proxygen_key/example_proxygen.pem
key_id: cli-key-1
base_url: https://identity.prod.api.platform.nhs.uk/realms/api-producers
client_secret: ***** #will be set automatically
password: '' #set for user access
username: '' #set for user access
```

Additionally, you will need to edit the ~/.proxygen/settings.yaml as shown below:

933 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY

Edit YAML file settings

```
api: #your api name
endpoint_url: https://proxygen.prod.api.platform.nhs.uk
spec_output_format: yaml
```

7.14.12.3 Managing your credentials

Once you have completed the initial setup for Proxygen CLI you can make additional changes to your credentials.

To manage additional credentials or update existing ones, use the following command:

Update credentials

```
proxygen credentials set <KEY> <VALUE>
```

To read a specific value from your credentials:

Read a specific value

```
proxygen credentials get <KEY>
```

To list all the stored credentials in the ~/.proxygen/credentials.yaml file:

Stored credentials list

```
proxygen credentials list
```

To remove a specific credential:

Remove credentials

```
proxygen credentials rm <KEY>
```

Optional flags available, as follows:

- -- help - Displays help message highlighting the commands available
- -- force - Only used in conjunction with the 'set' command to force re-entry of all credentials from scratch

7.14.12.4 Managing your settings

The main setting you will need to be concerned about is your "api" which you would have already set earlier on this page. However, you can further manage your settings, if you wish. Your settings are stored in:

- ~/.proxygen/settings.yaml.

To add additional settings or to update an existing setting, use the following command:

Add additional settings

```
proxygen settings set <KEY> <VALUE>
```

To list read a specific value from your settings file:

List a specific value

```
proxygen settings get <KEY>
```

To list all the stored settings in:

- ~/.proxygen/settings.yaml file:

List all the stored settings

```
proxygen settings list
```

To remove a specific setting:

Remove a setting

```
proxygen settings rm <KEY>
```

Optional flag available:

- -- help - Displays help messages highlighting the commands available at your disposal

7.14.12.5 Deploying an API instance

To deploy an instance of an API you will need an OAS specification file. See Configuring your connection to your API backend with Proxy Generator (see page 688) for more detail on how to create one.

The OAS specification file can be deployed using the following command:

Deploy OAS specification

```
proxygen instance deploy <env> <base-path> path/to/specification.yaml
```

You should see a response like the one below if the instance has been successfully deployed:

Spec successfully deployed

```
$ proxygen instance deploy int reasonable-adjustments ra-fhir.yaml --no-confirm
✓ Deploying https://int.api.service.nhs.uk/reasonable-adjustments
```

Multi-segment base paths

The Proxygen CLI supports multi-segment / multi-part base paths with underscores. For instance, if you intend to deploy a base path like "reasonable-adjustments/FHIR/STU3", it should be represented as "reasonable-adjustments_FHIR_STU3".

Gate your production deployments

Ideally deploying to production would be continuous. However, in reality, many programmes do need to gate/ manually approve/ or provide extra security around the action of deploying to production.
In the Proxygen model:

- We've delegated to the API Producer the ability to deploy to production - this means you need to consider:
    - how to link the call to Proxygen to your existing production release process
    - if your release pipeline script / code is in the same repo as your API Proxy spec, you might need to protect your pipeline code from malicious changes that would allow any repo user to make a production release
- Currently Proxygen does **not** support separate credentials for PTL and production. If there is a lot of demand for this then we will add it, as there are a number of further benefits.

The Electronic Prescription Service (EPS) worked through this problem, and if you have access you can read their identified attack vectors and solutions: Restricting proxygen deployments[934].

934 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/864407940/Restricting+proxygen+deployments

7.14.12.5.1 Managing your API instances

Get the spec used to deploy the instance in environment:

Deployed instances

```
proxygen instance get <env> <base-path>
```

Delete an API instance:

Delete API instance

```
proxygen instance delete <env> <base-path>
```

Get a list of all instances that have been deployed:

List all deployed instances

```
proxygen instance list
```

Optional flags available:

- --help - Displays help message highlighting the commands available at your disposal
- --api - Override the default API
- --no-confirm - Bypass the confirmation message and spec preview before deploying

7.14.12.6 Managing your API specifications

For a more in depth explanation on publishing your API documentation and what you will need before publishing, see Publishing your API documentation with proxy generator[935].

To publish your API specification, first deploy it on the UAT version of Bloomreach (see page 525), then on production. Publishing to UAT allows you to preview the appearance of the API documentation and locate any errors in the code before going live.

Deploying your specification using Proxy Generator stores it on the API platform infrastructure.

To create or update your OAS specification, use the 'publish' command, as follows:

935 https://nhsd-confluence.digital.nhs.uk/display/APM/Publishing+your+API+documentation+with+Proxy+Generator

Publish command

```
# Production
proxygen spec publish <path_to_spec>

# UAT
proxygen spec publish <path_to_spec> --uat
```

Once your specification exists, use the 'get' command to view the OAS specification:

Get spec

```
# Production
proxygen spec get

# UAT
proxygen spec get --uat
```

To delete a published spec, use the 'delete' command:

Delete spec

```
# Production
proxygen spec delete

# UAT
proxygen spec delete --uat
```

Optional flags available:

- --help - Displays help message highlighting the commands available at your disposal
- --api - Override the default API
- --no-confirm - Bypass the confirmation message and spec preview before deploying
- --uat - Target the UAT instance of the API catalogue

7.14.12.7 Managing secrets used by your API

Under the API key security model, the API platform employs authentication against your backend server using a secret API header. This header is included within the header of every request sent. To ensure seamless operation, you must configure both the name of the HTTP header and the corresponding value of the API key.

The API key, considered sensitive information, is securely stored within the API platform as a 'secret'. Before configuring any instances reliant on this key, it's necessary to deploy it using Proxy Generator.

Below is an example of how to do this via the CLI, including the response you might get back:

Example response

```
# Note that, the --apikey parameter sets the secret type so it can be referenced as
an apikey secret in the target definition
$ proxygen secret put internal-dev example-api-key --apikey --secret-file ~/tmp/
secret_file.txt
✓ Putting secret example-api-key in internal-dev
{
  "name": "example-api-key",
  "type": "secret",
  "last_modified": "2023-10-26T12:25:29.002000+00:00",
  "environment": "internal-dev",
  "apikey": true,
  "mtls": false,
  "version_id": "30f26d35-aa04-4a7e-893e-440398bcbbeb"
}
```

To describe a saved secret you can run the following command:

Saved "secret" command

```
proxygen secret describe <env> <secret_name>
```

To list all secrets currently saved:

List "secrets"

```
proxygen secret list
```

To delete a secret:

Delete a "secret"

```
proxygen secret delete <env> <secret_name>
```

Optional flags available:

- --help - Displays help message highlighting the commands available at your disposal
- --api - Override the default API
- --apikey - Tag the secret as an apikey
- --mtls-cert - Provide a path to PEM encoded certificate for mutual TLS
- --mtls-key - Provide a path to PEM encoded private key for mutual TLS
- --no-confirm - Bypass the confirmation message and secret preview before deleting a secret

7.14.12.8 Interacting with your API's docker repository

With the Proxygen CLI, you gain the capability to seamlessly interact with the docker repository linked to your API. This feature facilitates the retrieval of docker login credentials directly from an API, specifically tailored for an AWS Elastic Container Registry (ECR) repository.

The container images reside within a secure Elastic Container Registry (ECR) repository within the AWS infrastructure managed by API management. To enable the push and pull of images from this repository, you will need to acquire authentication details. Typically, these credentials comprise a username, password, and registry URL.

To generate a docker login command for AWS ECR repo with credentials from proxygen run the following command:

Generate docker login command

```
# 'docker get-login' command will return the exact command required to authenticate
Docker with ECR.
# You can copy/paste the returned command back into the shell like this:
$ proxygen docker get-login
docker login -u AWS --password <password> <Docker registry URL>
$ docker login -u AWS --password <password> <Docker registry URL>

# Or streamline the whole thing using the Bash command expansion:
$ $(proxygen docker get-login)
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
WARNING! Your password will be stored unencrypted in <homedir>/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Additionally, if you just want to retrieve the AWS registry you can run the following command:

Retrieve AWS repo

```
proxygen docker registry
```

Optional flags available:

- --help - Displays help message highlighting the commands available at your disposal
- --api - Override the default API

7.14.12.9 Retrieve a token for use with the pytest-nhsd-apim python testing package

For testing deployed proxies we recommend using the pytest-nhsd-apim python testing package[936]. This is a pytest plugin that provides a set of fixtures and utilities to help you test your API proxies with simple integration and automatic authentication.

To use the package perform the following steps:

1. Export `APIGEE_ACCESS_TOKEN` as an environment variable so that the package can interact with the API platform's underlying functionality.
2. Ensure you are authenticated to use Proxygen, the endpoint will provide you a token for testing.

You can use proxygen to retrieve this token uisng the following command:

Retrieve token

```
proxygen pytest-nhsd-apim get-token
```

This provides a response with the token and expiry in the shape below (token redacted):

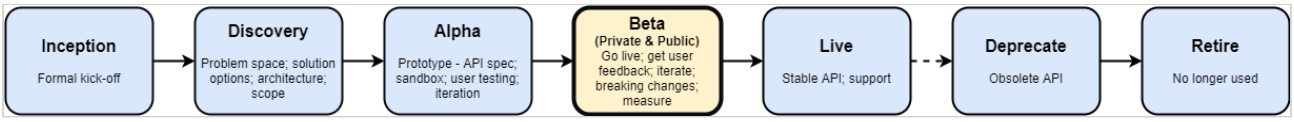
Token response

```
{
  "pytest_nhsd_apim_token": "************",
  "expires_in": 43199
}
```

For full detail on how to use the package, see the pytest-nhsd-apim documentation (see page 718).

7.15 Beta phase

7.15.1 Overview



936 https://github.com/NHSDigital/pytest-nhsd-apim

Beta is about moving your API into production and making it available to a wider audience. In private beta this is limited to a smaller audience, in public beta this is widened to a larger (invitation only) audience with increased volumes.

You will :

- design the onboarding process for API consumers
- get your API into production
- get a small number of external developers to use it
- iterate on their feedback
- transition your service to live

If you are delivering an API standard for a peer-to-peer API, beta is slightly different - you will not deliver anything into production yourself - rather you will work with third party early adopters to deliver their APIs into their production environment.

7.15.2 Process steps

For detailed process steps, see the API process checklist (see page 198).

7.15.3 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

- Preparing your API beta for production (see page 945)
- First time deploying into production review (see page 1120)
- Deploying to production (see page 1120)
- Deploying to production with Proxy Generator (see page 1135)
- Testing in production (see page 1144)
- API Consumer connection topologies (see page 1149)
- Beta exit review (see page 1154)
- Developer community guidance (see page 1154)
- Identifying Who Are Your API Consumers? (see page 1172)
- Qualtrics Survey to Power BI (see page 1176)

7.15.4 Preparing your API beta for production

7.15.4.1 Overview

This page describes the engineering and service management tasks of preparing your API beta for production as part of the Beta phase (see page 944) of the API process checklist (see page 198).

After preparing your API beta you should have:

- a first iteration / MVP of your API beta that is production-ready (i.e. stable, functional, secure, and well-monitored)

- conducted a penetration test of your API service
- gained approval to go-live from LSB/SLT
- a fully-qualified onboarding process approved by the Onboarding Governance Group

7.15.4.2 Prerequisites

These prerequisite steps may be run in parallel to the building your API process steps.

- learn about the different API deployment environments (see page 596)which are used for testing by external API Consumers.
- your API is ready to exit Alpha - see Alpha exit review (see page 933)

7.15.4.3 Process steps

7.15.4.3.1 Get sign-off to go live

To deploy your API service into production, you must have gained approval from various parties to do so. It is recommended you start the respective governance processes at the earliest opportunity, as these may take a long time to complete and are mostly out of your hands in this timeframe.

These processes may be ran in parallel and in any order.

- undergo the appropriate service transition process (see page 947)
- gain approval for your API consumer onboarding process (see page 999) from the Onboarding Governance Group
- conduct a penetration test (see page 968)and gain approval for your API beta go-live from Cyber Security

7.15.4.3.2 Engineering tasks

The below tasks relate to "productionising" your API beta - firming it up into a stable and secure service. They may be ran in parallel and in any order.

- implement Rate limiting (see page 1037) to increase service stability
- implement status monitoring (see page 1064)and Logging to Splunk (see page 1082) to improve visibility of your service
- add in automated smoke tests (see page 1145) so that failed deployments are spotted early
- add in the nhsd-git-secrets pre-commit hook (see page 1102)
- implement request tracing (see page 1104)
- add in your Spine-specific configuration (see page 408) (if your API beta interacts with Spine)
- auto-approval (see page 927) for API consumers (if this is part of your onboarding process)
- implement update locking (see page 1109) (if your API beta updates resources)
- Caching for your API (see page 1112) (if relevant)
- consider adding in Feature toggles (see page 1113) to gain more granular control over your API's behaviour between environments
- review dependabot alerts (see page 788)

7.15.4.4 Get approvals to deploy to production

These articles describe the governance processes you must go through before you can deploy your API beta to production.

It is recommended you start these processes at your earliest opportunity as they can take some time to complete.

- Service transition for APIs (see page 947)
- API platform hazard log (see page 966)
- Penetration testing APIs (see page 966)
- API consumer onboarding (see page 999)

7.15.4.4.1 Service transition for APIs

7.15.4.4.1.1 Overview

Before you can get your API into production, you need to make sure it is "production ready". We sometimes call this service transition.

The owning API producer team, not API Management, has responsibility for service transition.

The process depends on whether you are an NHS Digital API producer team or a third party API producer team.

Note: If you are handing over or taking over an API/Proxy, see API handover guidance (see page 963)

7.15.4.4.1.2 Service transition for third party API producer teams

See Service transition for third party API producers (see page 960).

Also see the guidance below for details on technical go-live, private beta and public beta.

7.15.4.4.1.3 Service transition for NHS Digital API producers

You need to work with the Service Design and Transition (SDT) team. For more details, see the Service Design and Transition Confluence space[937], If you can't access the site, you could contact them for more details at liveservices.sdt@nhs.net[938].

To engage with SDT, complete a New Work Request (NWR) form. The reasons for completing the NWR can be found within the form itself. You'll also need to fund their work, so have your portfolio code to hand!

You can access the NWR form here[939].

API service transition process

937 https://nhsd-confluence.digital.nhs.uk/display/SDT
938 mailto:liveservices.sdt@nhs.net
939 https://forms.office.com/Pages/ResponsePage.aspx?id=slTDN7CF9UeyIge0jXdO4xQv16X0-Z1Ip82TwtxhY5JUQThFS1dPN0FESDc4Sjg5RjlYSzBEQkNDSC4u

Once an NWR has been raised, it will be assessed by SDT and initially assigned to the appropriate Live Services Service Management Team, for discovery and assessment. Depending on the complexity of the Service Transition, either the Service Management Team or a member of the central SDT team will be allocated to manage the transition. A confluence page will be created which details the necessary assurances sign offs and service readiness activity. Once completed, the transition will be approved by the relevant Live Services head of service.

Depending on whether you have a API consumer onboarding approach in place, will depend on whether you enter technical go-live only or private beta (which includes technical go-live steps). In most cases you would normally also have a First of Type (FoT) API consumer as you enter private beta, but it is possible to enter private beta without an API consumer e.g. the API consumer has withdrawn at the last minute. See below for more detail on transitioning through the various phases.

7.15.4.4.1.4 Beta entry and exit process

If your API has several parts or phases or components or access modes, you might want to manage the service transition separately for each part.

For example, the PDS FHIR API has three access modes:

- Application-restricted access
- Healthcare worker access
- Patient access

We treated each access mode separately for getting through beta.

Steps

The first time through the process, all steps would be executed per component / access mode. There afterwards, if adding features which are minor only an RFC (see page 1120)may be required. If its a major change, then all steps would need to be executed again.

1. Enter technical go-live
2. Enter private beta
3. Enter public beta
4. Enter general availability
5. Transition onboarding to Service Design and Transition

Enter technical go-live

If you have no API consumer onboarding approach in place, you can enter technical go-live. In cases where there no personal or no data being passed through the API, then you will only need approval from your Information Asset Owner (IAO) or someone senior within the API producer delivery team. This is to mitigate any risk of data being exposed through testing.

It's recommended because:

- It allows you to get your service deployed to production, temporarily enabled and smoke tested. This also allows you time to fix any deployment or connectivity issues

- It allows you to get the majority of your governance out of the way (aside from OGG sign-off) - again, allowing you time to resolve any issues ahead of starting your private beta

Note : If you are in this phase for a long time without a customer you may encounter issues if new functionality is released, as you will need to go through the whole RFC (see page 1120)/release process.

It involves:

- Getting approval to go live
- Deploying your API (see page 882) into the production environment
- Smoke testing (see page 1145) it

You **do** need to Live Services Approval.

You **don't** need to have designed your onboarding approach.

Process:

- Raise NWR
- Service Mgt / SDT create Transition Confluence Page with OGG N/A
- Owning team raises an RFC (see page 1120).
- Service Mgt Approve RFC no Head of Service approval required at this stage.

Out of hours support

The service incident modelling and agreements you carry out as part of the service transition process will cover how you do out of hours support. Often private beta is just Bronze level support.

It is really important to formally announce to other teams when you start needing out of hours support, and this can happen at any point from Private Beta onwards:

- When you change / update your service support SLA - does it include out of hours support (Gold and Platinum)?
- If so, it is critical to inform the APIM on-call rota and ITOC. This step can often be forgotten in the rush to go live

Enter private beta

If you have an onboarding approach in place you can move directly into private beta (together with the tech go-live steps). Moving directly into **private Beta** needs full approval Live Services via the SD&T NWR process, whereas moving from **technical go-live,** only requires the **owning API producer** team to work with the Live Services to add in any remaining assurances such as OGG to the existing Transition Confluence Page before obtaining approval.

This involves:

- Getting a small number of developers onboarded through the onboarding process and using the API in production
- Iterating on their feedback

To enter private beta you need to define your private beta limits:

- Maximum number of connected systems (apps) - typically this is 5

- Maximum throughput - typically this is 5tps per connected system (this allows you to avoid having to do load testing before starting private beta)

You **do** need to have your onboarding approach e.g. SCAL, designed and approved by OGG.

You should **ideally** have at least one developer lined up for your private beta.

Entering private beta directly :

Process:

- Onboarding process needs to be designed and approved by OGG
- Raise NWR
- Service Mgt / SDT create and complete Transition Confluence Page
- Live Service Head of Service Approves

Entering private beta from technical go-live :

Process:

- Service Mgt / SDT uplifts the existing Transition Confluence Page to say the service is entering private beta
    - OGG section marked as "green"
- Live Services Head of Service Approves

Enter public beta

If you want to take your beta beyond the private beta limits, that's fine, you just need to declare it to the world.

You might need to make sure:

- Scalability approach / capacity planning approach finalised
- Team is resourced to deal with increased onboarding volumes

Process:

- Service management uplifts the existing Transition Confluence Page to say the service is entering public beta
- Owning team raises an RFC (see page 1120).
- During the switch over, is any "hypercare" of the API Consumers needed? If so, comms and agreements from the relevant teams must be organised in advance

Enter live (aka general release)

At some point you need to set beta exit criteria. The API status at this stage would be set as "stable".

The following is an example, you should tailor them to suit your API.

- Three **external** developers have fully completed onboarding (internal use by NHSD doesn't count as it doesn't test the onboarding approach)
- All three developers have completed the developer integration survey (see Metrics for success / KPIs for APIs - API consumer integration survey) and there are no ratings lower than 3 out of 5

    - Including technical and non-technical resources
- 3 apps using the API in prod for a period of 4 weeks
- 10,000 transactions have been put through the endpoint
- The error rate is <= 1 in 10,000
- Availability over a period of 4 weeks is > 99.99%
- No sev 1 or sev 2 incidents
- Sev 3, 4, 5 incidents are "reasonable"
- Review backlog - are there any breaking changes left? Are there public facing changes (e.g. updates to the API Catalogue)
- Service support model tested in production, if not already done prior to starting private beta.
    - Change / update Service levels for APIs (see page 288)
    - Does it include out of hours support (Gold and Platinum)? The APIM on-call rota (and ITOC) must be informed
- Scalability approach / capacity planning approach finalised
- During the switch over, is any "hypercare" of the API Consumers needed? If so, comms and agreements from the relevant teams must be organised in advance
- Team is resourced to deal with increased onboarding volumes
- No breaking changes expected (once your API has exited beta, to make breaking changes you'll need to create a new version of your API)

Process

- Service management /SDT uplifts the existing Transition Confluence Page to say the service is entering general release
- Service management / SDT obtains Head of Service Approval

7.15.4.4.1.5 Questions asked by compliance/assurance teams

Overview

Whilst the APIM Platform has gone through assurance to deliver a Platinum, high capacity service, many API Producer teams will have compliance groups looking at the service and asking similar questions. Many of the questions will be around non functional requirements (NFRs).

This page tries to make it easier to answer those types of questions. This page does not include the Digital Onboarding Service and the Developer Account identity service (Keycloak).

Service Level

The APIM service operates at a Platinum Service level, but as an API Producer you will have your **own** service level.

You are dependant on APIM to provide services that are:

- 99.9% availability 24x7
- This includes both Apigee and any APIs that use the APIM AWS Hosted Containers

- To see how APIM is assured at a technical level see: Non-functional requirements[940] (if you don't have access to this page contact APIM)
    - This includes the currently approved: IT Service Continuity Plan - Apigee Edge 20220412 v2.0.docx[941]

Privacy

Guidance and evidence (DPIA and IA) for APIM's compliance here can be found on this page: Privacy considerations in your design (see page 406)

Monitoring and alerting

The API proxy (and any APIM AWS Hosted Containers), is already integrated with ITOC and all Production APIs are automatically configured to deliver monitoring and alerting through:

- NHS Digital Splunk Cloud
- APIM managed Grafana instance
- Slack alert channels

For details of this see: Monitoring your API (see page 1064)

An API Producer will need to onboarding with ITOC[942] for their entire service. Some of the areas covered by on-boarding:

- Awareness of your product with ITOC
- Creation of KPIs for your entire service (not just APIM proxies)
    - APIM proxies have a "template" set of KPIs, but you might want to adjust the thresholds for your service
- Potential creation of “deep dives” or “glass table” for your service

Cyber Security threat monitoring

CSOC (Cyber Security Operations Centre) are in the process of onboarding the entire API Management platform, and so, automatically include all APIs on the platform. Just as with ITOC, API Producers need to onboard the backend of their service to CSOC. This should be notably reduced effort because of the existing work with the rest of the API Management platform.

APIM also carries out weekly manual checks on the internet facing APIs, and once fully onboarded this will be stopped. Whilst all parts of APIM are internet facing the attack surface is significantly reduced because:

- All traffic comes via Google Apigee, since backends are all configured to be protected with TLS-MA and only accessible from Apigee
- The nature of API traffic is very constrained compared to normal arbitrary web services
- Apigee stops any non HTTPS traffic from even passing through the Apigee layer

---

940 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/150667529/Non-functional+requirements

941 https://nhsd-confluence.digital.nhs.uk/download/attachments/150667529/IT%20Service%20Continuity%20Plan%20-%20Apigee%20Edge%2020220412%20v2.0.docx?api=v2&modificationDate=1650523030624&version=1

942 https://nhsd-confluence.digital.nhs.uk/display/ITOC/Service+Management#ServiceManagement-Onboarding

If Google notice an obvious attack, they will contact NHS Digital. When an attack occurs, NHS Digital will work with Google to keep legitimate traffic flowing.

It should be noted that during the 2 years operation so far, whilst there has been some small scale bot scanning, there have been no concerted attacks. This will change overtime, as visibility of the platform increases - but by then the CSOC threat monitoring should identify they issues quickly.

AWS and APIM AWS Hosted Containers

APIM meets all the requirements for monitoring the APIM managed AWS accounts, key headlines:

- NHS D Landing Zone standard CloudTrail (which is then monitored by CSOC): Secrets access; infrastructure changes; logins
- APIM configured CloudWatch to alarm on the following types of security events: root access; security group changes; AWS account IAM changes; SSM sessions; key changes; network ACL changes; users without MFA
- AWS Admins have now enabled the AWS Secrets Hub and AWS Config on the APIM to help improve security and traceability

Security

Denial of Service attacks

The service has been approved using the following technical implemtations:

- Google Apigee Edge Cloud
    - We are on the SaaS contact, and rely on Google as the primary DDoS protection
    - This page covers their approach https://docs.apigee.com/api-platform/faq/ddos-defense-edge, key points:
        - Block malformed network packets at the network of the Cloud (Google CDN edge)
        - Absorb a flood of properly formed but incomplete packets at the Edge platform layer
        - Drop malformed API calls at the Edge layer
        - Block properly formed but unauthorized calls within Edge
        - Block properly formed and authorized but excessive calls within Edge (through the use of rate limiting logic in each proxy)
    - This does not quite cover all possible scenarios, and (in an incident) we will work proactively with Google to filter out traffic created by a compromised API consumer - this is similar to other SaaS approaches
- APIM AWS Hosted Container service:
    - This is an optional service provided by APIM, and is used by less than 5% of the current traffic
    - Apigee forms the primary DDoS protection

- All AWS endpoints are protected with TLS-MA, and the source is always Google Apigee - all other traffic is rejected

A decision, early on, was taken not to use WAF because, of the mitigations listed above. In addition:

- Malicious layer 7 attacks on APIs (which WAFs help with in addition to DDoS) have a significantly smaller attack surface, very few of the APIs even allow free text query requests. Where there is a specific risk the API Producer can take extra precautions
- At the start of the programme WAF technology had not been tuned to API usage, and tended to cause more live service incidents than protection added. During the strategic procurement in FY 23/24 WAF will be re-considered

Other security NFRs

There are many other Security Non-Functional Requirements that were addressed and worked through prior to moving to live service, you can read the key detailed outcomes here: APIM Platform Non-functional requirements[943].

APIM carries out annual pen testing - see APIM Platform Annual Pen Tests[944], and then relies on:

- Google Apigee run annual SOC certification, and Cyber Security & APIM review these - you can see the log of this on: APIM Platform Non-functional requirements[945]
- For AWS, APIM rely on the NHS England CCoE / contractual arrangements - alongside following AWS Well Architected Framework, particularly the Shared Responsibility Model

Google Apigee Edge for Public Cloud only supports TLS 1.2 - Apigee docs - using-ssl-edge supported-versions-of-tls[946]

Incidents and Change Management

APIM manages incidents and changes via two mechanism (both are part of the core NHS Digital service processes):

- High Severity Security Incident (HSSI)
- National Service Desk (NSD) / ServiceNow

When an API Producer carries out their service transition, working with the Service Design and Transition (SDT) team, this will include:

- An incident triage and change management process
- How to had off tickets to the APIM resolver group "ITO API Platform"
- NHS Digital has the concept of a "standard change", which significantly speeds up the change process. The APIM Platform can carry out these types of changes in many situations, and we recommend the API Producer also works towards obtaining approval for standard changes as well

For P1/ P2 (Sev-1 or Sev-2) incidents the following pages are useful for an API Producer to reference:

---

943 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/150667529/Non-functional+requirements
944 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/214149668/APIM+Platform+Annual+Pen+Tests
945 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/150667529/Non-functional+requirements
946 https://docs.apigee.com/api-platform/system-administration/using-ssl-edge#supported-versions-of-tls

- How to report a security incident[947]
- On-call policy[948]

Note that high severity non security incidents are treated the same way as security incidents - it is all reported and coordinated via the National Service Desk.

Volume and performance testing

When considering what requests per second you need to support, the largest limiting factor is the API Producer backend. In terms of the API Platform the two components to consider are:

- Apigee - massive scalability, with appropriate notice to Apigee, no real limit on that scaling
- APIM AWS Hosted Containers - auto scaling built in

In terms of the Apigee service, the following examples show that Apigee has the capabilities to support massive traffic loads:

- Covid-19 Test History and Identity Service
    - Net Company tested to 5K transactions per second (tps)
    - Internal Tribe testing - up to about 3.5K, for evidence, see Identity Service NFT[949] for range of tests
- Personal Demographics Service (PDS FHIR API)
    - Spine NFT have done some explicit testing to take PDS FHIR to about 480tps (mix of Trace / Retrieve and update)
    - Evidence here: PDS FHIR App Based Rate Limiting NFT Results[950]
- Production peak load - 15th Dec 2021 hit 1,950tps, for evidence see Production peak traffic[951]

We have no plans to regularly test "the platform" as Apigee is supplied as a SaaS product, and with suitable notice they will support significantly higher loads. It should be noted that in many of the NFT tests (including the largest ones), we have **not** requested that Apigee pre-warm the environment - adding to the evidence for large scalability. :

- No other users of Apigee noticed any service degradation
- On a few of the tests we have found evidence that request duration increases, and sometimes even some evidence of (possible) increased latency due to auto scaling. However, many APIs are currently under 10ms response times, and 95th percentiles sometimes reach over 100ms (*todo - actually document evidence*)
- Apigee only request warning where we anticipate spikes of 10-100x average traffic within 2 minutes
    - Random day selected (23rd Jan 2024) Production (all APIs):

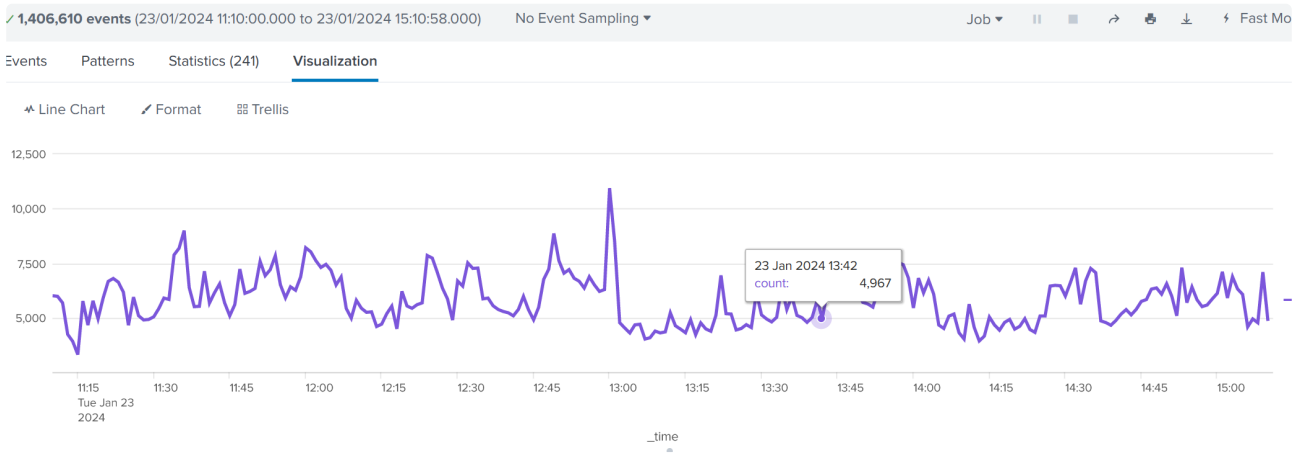
947 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110886480/How+to+report+a+security+incident
948 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117644394/On-call+policy
949 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/214147881/Identity+Service+NFT
950 https://nhsd-confluence.digital.nhs.uk/spaces/SPINE/pages/307147949/PDS+FHIR+App+Based+Rate+Limiting+NFT+Results
951 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/415613957/Production+peak+traffic

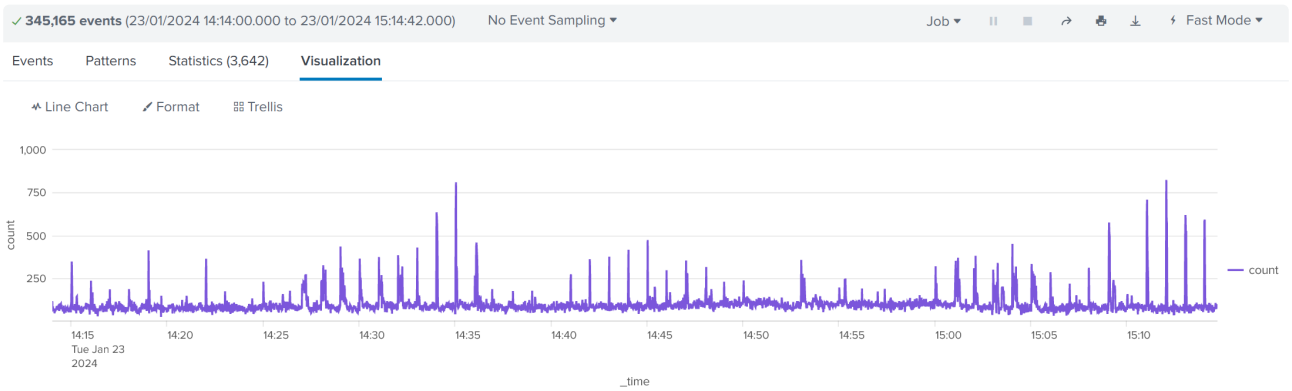
        - Over 4 hour period: 5K to 7K TPM during the day, and spikes over 10K TPM
        - Over 1 hour 70-100 TPS, with spikes to 800+ TPS
    - Busy day selected (9th Oct 2023) Production (all APIs):
        - Over 4 hour period: 4K to 6K TPM, but sustained nearly 30KTPM
        - Over 1 hour 200-700 TPS, with spikes 1,300+ TPS

As an API Producer this page sets out our guidance: Performance testing (see page 756). In summary API Producers are advised to:

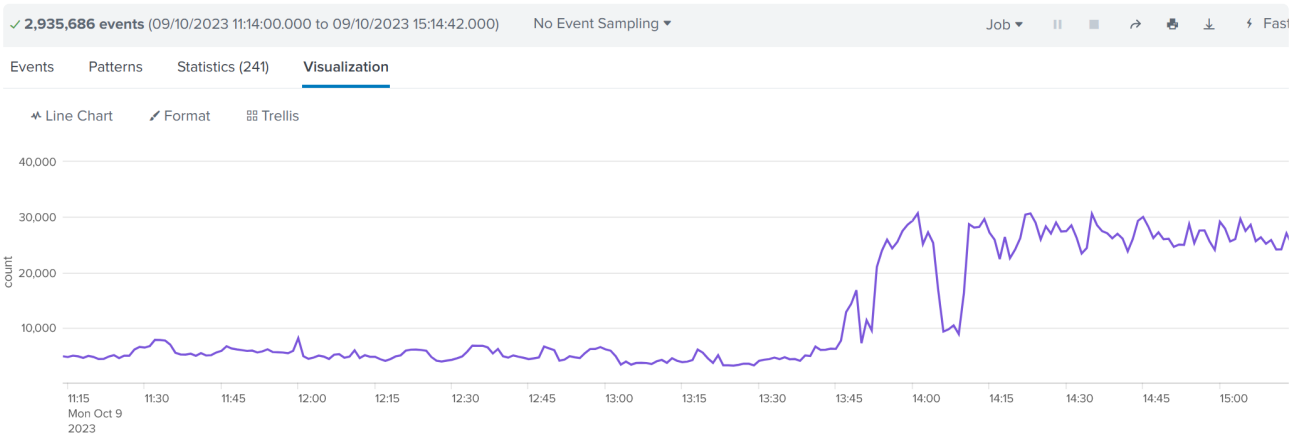
- Any NFT / performance testing you do should go via Apigee
    - We have the capacity and it improves the levels of assurance
- Where a rate limit of 10tps or less is going to be applied to the API, there is no need to performance test via Apigee
    - Depending on the client ramp up and technology stack of the solution, it might even be appropriate to enter private beta with no backend performance testing (acknowledging this as a low risk)
- The highly scalable nature of Apigee has been proven, and as a consequence we only ask you to flag your service to APIM in either situation:
    - It requires 500tps or above
    - You have APIM AWS Hosted Containers that are heavily utilised, you must performance test them to your required NFT levels

Graphs:





9th Oct 2024:



Wider Governance Approvals

The AWS Hosted Containers architecture and use case was approved by Platforms and Infrastructure Board (PIAB) on 20th Jan 2021, full submission details on this page: AWS Hosted Containers Architecture[952]

Sample request durations in production - 20 Nov 2024

All run from 7am to 6pm on 20th Nov 2024

Spine Directory Service (SDS) FHIR API

Using:

```
index="summary_apm_prod_open"  NOT (proxy.suffix IN ("/_status", "/_ping"))
proxy.name="spine-direc*" | convert num(target.sent_start) as
calc_dur.target_sent_start, num(target.received_end) as calc_dur.target_reciv_end,
num(client.received_start) as calc_dur.client_reciv_start, num(target.received_start)
as calc_dur.target_reciv_start  , num(client.sent_end) as calc_dur.client_sent_end
| eval dur_ms_target ='calc_dur.target_reciv_end'-'calc_dur.target_sent_start',
dur_ms_apigee_in='calc_dur.target_sent_start'-'calc_dur.client_reciv_start',
dur_ms_apigee_out='calc_dur.client_sent_end'-'calc_dur.target_reciv_start',
dur_ms_apigee_worst=dur_ms_apigee_in+dur_ms_apigee_out , time_bucket=strftime(_time,
"%H:%M:%S")
|fields calc_dur.*, dur_ms* , client.received_start, client.received_end,
target.sent_start, target.sent_end, target.received_start, target.received_end,
client.sent_start, client.sent_end, proxy.name, proxy.suffix, time_bucket
| fields - _raw | stats count, avg(dur_ms_apigee_worst), perc98(dur_ms_apigee_worst),
perc99(dur_ms_apigee_worst), avg(dur_ms_target), perc98(dur_ms_target),
perc99(dur_ms_target)  by proxy.suffix, time_bucket
```

Device	avg(dur_ms_apigee_worst)	perc98(dur_ms_apigee_worst)	perc99(dur_ms_apigee_worst)	avg(dur_ms_target)	perc98(dur_ms_target)	perc99(dur_ms_target)
Average	19.2402	44.97506	54.20595	67.57911	176.1948	209.9672
Max	71.05263	649.64	838.49	196.8889	1376.14	2114.93

952 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153364474/AWS+Hosted+Containers+Architecture

Endpoint	avg(dur_ms_apigee_worst)	perc98(dur_ms_apigee_worst)	perc99(dur_ms_apigee_worst)	avg(dur_ms_target)	perc98(dur_ms_target)	perc99(dur_ms_target)
Average	17.83135	44.7243	54.68365	67.65353	200.2066	238.1747
Max	88.6	963.94	1508.31	279.5217	3017.48	4119.24

- 20241120 - SDS by second durations.xlsx[953]
- 20241120 - SDS by minute durations.xlsx[954]

Personal Demographics

Using:

```
index="summary_apm_prod_open" NOT (proxy.suffix IN ("/_status", "/_ping"))
proxy.name="personal-demo*" | convert num(target.sent_start) as
calc_dur.target_sent_start, num(target.received_end) as calc_dur.target_reciv_end,
num(client.received_start) as calc_dur.client_reciv_start, num(target.received_start)
as calc_dur.target_reciv_start  , num(client.sent_end) as calc_dur.client_sent_end
| eval dur_ms_target ='calc_dur.target_reciv_end'-'calc_dur.target_sent_start',
dur_ms_apigee_in='calc_dur.target_sent_start'-'calc_dur.client_reciv_start',
dur_ms_apigee_out='calc_dur.client_sent_end'-'calc_dur.target_reciv_start',
dur_ms_apigee_worst=dur_ms_apigee_in+dur_ms_apigee_out , time_bucket=strftime(_time,
"%H:%M")
|fields calc_dur.*, dur_ms* , client.received_start, client.received_end,
target.sent_start, target.sent_end, target.received_start, target.received_end,
client.sent_start, client.sent_end, proxy.name, proxy.suffix, time_bucket
| fields - _raw | stats count, avg(dur_ms_apigee_worst), perc98(dur_ms_apigee_worst),
perc99(dur_ms_apigee_worst), avg(dur_ms_target), perc98(dur_ms_target),
perc99(dur_ms_target)  by proxy.suffix, time_bucket
```

- 20241120 - PDS by second durations.xlsx[955]
- 20241120 - PDS by minute durations.xlsx[956]

953 https://nhsd-confluence.digital.nhs.uk/download/attachments/950538326/20241120%20-%20SDS%20by%20second%20durations.xlsx?api=v2&modificationDate=1732264209763&version=1

954 https://nhsd-confluence.digital.nhs.uk/download/attachments/950538326/20241120%20-%20SDS%20by%20minute%20durations.xlsx?api=v2&modificationDate=1732264309127&version=2

955 https://nhsd-confluence.digital.nhs.uk/download/attachments/950538326/20241120%20-%20PDS%20by%20second%20durations.xlsx?api=v2&modificationDate=1732264253182&version=1

956 https://nhsd-confluence.digital.nhs.uk/download/attachments/950538326/20241120%20-%20PDS%20by%20minute%20durations.xlsx?api=v2&modificationDate=1732264253877&version=1

7.15.4.4.1.6 Service transition for third party API producers

Overview

If you are building an API on the NHS Digital API platform, but you are a third party (i.e. not NHS Digital), this page explains the process you need to follow before we'll let you put your API into production.

This is in addition to the activities on our API process checklist (see page 198) - and there might be some overlaps!

Checklist

Make a copy of this checklist and use it to make sure you do everything we require before you go into production.

Status codes

TO DO

IN PROGRESS

DONE

N/A

Activity	Status	Comments / Evidence
Information governance If your API processes personal or sensitive information, make sure you understand the legal basis (in the GDPR sense) under which your are processing the information. Also make sure you understand your role - data controller or data processer. We are still figuring out whether NHS Digital considers itself a data controller or data processer for information that passes through the API platform. Probably data processer.	TO DO	

Activity	Status	Comments / Evidence
Clinical safety Make sure your API is clinically safe - you must have a Clinical Safety Officer and your API must conform to DCB0129. For details, see https://digital.nhs.uk/developer/guides-and-documentation/introduction-to-healthcare-technology/clinical-safety. Tell us the name of your Clinical Safety Officer in the comments/evidence column. A key part of the process is for you to create a hazard log. We have created a generic platform hazard log[957] which covers hazards of hosting your API on the API platform. Make sure you consider these risks. If you want you can use this hazard log as the starting point for your own hazard log.	TO DO	
Security Ensure your API is sufficiently secure for the given use case. This is a job for your architect or technical lead. For details, see Securing your API (see page 774). As part of this, you might need to conduct a penetration test. There can be a long lead time for this. If you do need to do a pen test, you'll also need to deal with any feedback - either remediating risks or accepting them. You shouldn't be accepting any high risks. For details, see Penetration testing APIs (see page 966).	TO DO	
Service level Determine the service level for your API. For details, see Service levels for APIs (see page 288).	TO DO	

957 https://nhsd-confluence.digital.nhs.uk/download/attachments/351798812/API%20Platform%20Generic%20Hazard%20Log%20%28Revised%29.xlsx?api=v2&modificationDate=1639067255477&version=1

Activity	Status	Comments / Evidence
Live support Make sure you have a team in place to support your API to the appropriate service level (see Service levels for APIs (see page 288)) - including out-of-hours support where appropriate. Make sure you have processes in place to handle live incidents, including the escalation of API platform incidents to NHS Digital.	TO DO	
API consumer onboarding Make sure you have a process in place for onboarding API consumers. We strongly encourage you to use our digital onboarding service for this - if not, please supply a good reason why you won't be using it. It is your responsibility to make sure your onboarding process includes appropriate assurance to make sure that API consumers are also handling things like: • information governance • clinical safety • security • live support You also need a legal framework in place so that accountability is placed appropriately with API consumers. This might be a signed "connection agreement" or it might be some terms of use the API consumer agrees to.	TO DO	
Legal Sign a legal agreement with us for use of the API platform so that we both know who is responsible for what. We are still devising this agreement, but it will almost certainly state that you are responsible for: • information governance • clinical safety • security • live support • API consumer onboarding	TO DO	

7.15.4.4.1.7 API handover guidance

API/Proxy handover is solely the producer teams responsiblity.

Overview

It's common for API producer teams to hand their API/proxy over to another team. Whilst it is the producer teams responsibility to perform the handover to the new team, the Platforms team have provided some guidance on what information may be necessary for a successful handover. It also a good idea to record any handover sessions so that these can be used for future reference for the team taking over the API/proxy.

A successful handover is one where the new team has been provided with correct information, to an appropriate level of detail in order to avoid creating an increased support burden.

Below is a list of key areas/things to consider during a handover, **note this is not a comprehensive list.**

Areas to consider

Proxy logic

If the logic in your proxy differs to the standard proxy logic documented in confluence (that which comes as standard from the setup api tool) this should be stepped through during the handover and also be documented in detail for the new owning team to refer back to.

Test requirements

Any test requirements should be documented for the new owning team, this includes keycloak users, test data etc, as well any nuances.

Also the name and ID of any (Apigee) apps used for automated testing.

Secret(s) and paths

The locations and/or paths of all API/proxy specific secrets should be documented as well as their use case, purpose and maintainer.

Custom attributes

Documenting what (if any) custom attributes are required, why they are required and how they are applied.

For example ASIDs and ODS code values should be given to the new owning team, as well as defining and explaining the needs of any additional attributes.

Production apps

Name and IDs of all live (production) Apigee apps, who is consuming them and their typical traffic levels.

Onboarding

The APIs consumer onboarding process should be documented and well understood by the new onboarding lead.

Is or does the API call any Spine APIs? If so, the mechanism to setup ASIDs and Party Keys is important to handover.

Service Management

Service management should be documented for the new owning team refer back to, consider: routing, resolver groups, access to incident management tools (e.g ServiceNOW), etc.

Releases

Similar to above, release process should be documented for the new team to refer back to, consider: APIM releases or ability to self-release, release process (APIM's process or own process), RFC system access (e.g ServiceNOW), comms, etc.

7.15.4.4.1.8 Supporting your API

Overview

Before your API goes live, as part of Service transition for APIs (see page 947) you will have organised how incidents are triaged. This normally takes the form:

- 1st line: ITOC monitors your service, which includes metrics
- 2nd line: If ITOC spot an issue they raise in ServiceNow and escalate to your support team
- 3rd line: If your team spots an issue with the API Proxy they can't resolve then the call should be passed (via Service Desk) to the API management on-call team (and allocated to the resolver group "ITO API Platform")

The APIM on-call support team should have been engaged and made aware of this prior to initial private beta go live.

Ensuring APIM On-call engineers have permission to view any sensitive data

In order to diagnose live issues, on-call engineers may need to read raw logs or perform live traces on your API. This could give them access to any data that passes through the API, not just the URLs.

As an API Producer, you **must have someone** who is responsible for the safety of the data your API manages. Each programme appears to be different, and the responsibility might fall to any / all of:

- Information Asset Owner (IAO)
- Data guardian

- Data controller
- Nominated representative from Information Governance (IG - now part of PTEL Privacy, Transparency, Ethics and Legal)
- Security Officer

APIM **cannot** help you determine who that is.

They will understand if there is any Personally Identifiable Data (PID or PII), or if your API contains special category data. If you do:

- There is a **standard NHS Digital** process that requires:
    - The creation of an Information Asset
    - If there is PID / PII, then the Information Asset must be made visible via the Asset Access Request Form (AAR)
    - Details of this process are here: IAO Portal Sharepoint[958]
- The IAO (or similar) must then decide for new APIM on-call staff if they should:
    - Fill in the AAR
    - -or- that APIM should email the IAO (or similar)
    - -or- if any SC Cleared APIM staff can automatically be allowed to view your API data

The AAR forms, formerly known as Clear Data Access (CDA) forms, are on the eStore: https://estore.hscic.gov.uk/Store?tab=MyStore&id=88234b9c-d851-4682-962b-5aaac163062b.

All APIM staff (who can access the raw logs or perform live traces) go through the National Security Vetting process to obtain SC level clearance.

Informing APIM on-call team

The actual point that a team transitions to supporting live API Consumers for the first time can be subject to shifting timelines and stakeholder comms. It is particularly important to inform the APIM on-call rota, some APIs can stay in a "techincal go-live" status for months and are **not** responded to out of hours. Work with the API Producer Liaison to:

- Make an announcement on the Platforms APIM Release Co-ordination[959] Slack channel
- Inform the APIM Product Owners (currently Navin Bose)
- Ask APIM to update the ITOC runbook[960] so that out of hours teams **don't** ignore alerts
- Do you need hypercare during the first few days? If so this needs to organised in advance

958 https://hscic365.sharepoint.com/sites/IGPG/IAO%20Essentials/Forms/AllItems.aspx?id=%2Fsites%2FIGPG%2FIAO%20Essentials%2FAsset%20Access%20Requests

959 https://nhsdigital-platforms.slack.com/archives/G01519458M6

960 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/253886796/ITOC+runbook

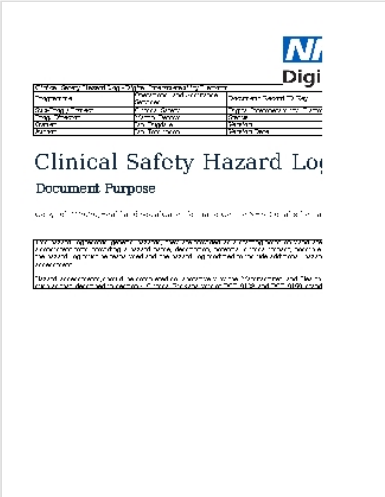
7.15.4.4.2 API platform hazard log

7.15.4.4.2.1 Overview

The api platform hazard log consists of set of generic hazards for APIM platform, for API producer teams to review in collaboration with their clinical lead. The list is not exhaustive list, but provides a starting point for consideration as part API development and assurance activities. It is expected that the clinical safety team for the API producer will develop their own hazard log and may incorporate some of these hazards and associated causes. For further information, please contact the APIM clinical team (currentlyRaman Behl)

7.15.4.4.2.2 Api plaform hazard log

Note: Clicking on the thumbnail below opens the file in preview mode, displaying a limited view. Clicking on the 'downward arrow' in preview mode will provide a full file view.


Digi

Clinical Safety Hazard Lo

Document Purpose

7.15.4.4.3 Penetration testing APIs

7.15.4.4.3.1 Overview

A penetration test (or "pen test") is a way of making sure a service is "hardened" against malicious attacks.

This page explains what penetration testing you need to do for your API.

It also explains whether you can do the penetration testing yourself or whether you need to arrange for an external company to do it.

If you need an independent penetration test doing by an external company, there is a **long lead time** (minimum of 6 weeks) on externally run Pen Tests, so these need to be planned in advance.

7.15.4.4.3.2 Do I need to do a penetration test?

For lower risk APIs you can use your standard automated testing (or extend it slightly) to continually test your API and not have a formal pen test. Your API will be included in the APIM annual pen test, so will be tested in a different context.

The API penetration testing process (see page 968) pages guides you through how to come to a decision, in conjunction with your Cyber Security contact on your pen testing needs.

7.15.4.4.3.3 When to do a penetration test

You **might** need to do a penetration test at the following points:

1. Before you launch your API into production
2. When you make a change to your API
3. On an annual basis once your API is in production

Before launch

You **always** need to do a level of security testing before you launch your API into production.

Depending on your API, you **might** need to arrange for an external company to do the testing, or you might be able to to it **internally**. See below for more details.

On a change

For APIs that are agreed as internal pen testing only, then the standard automated tests you run on every release should be sufficient. If you have a more complex API that has previously had an external pen test, then the amount of change needs to be considered.

The risks of any change should be discussed with Cyber Security, to confirm the situation for that change. The ServiceNow RFC will ask for Cyber approval of the change, so they should be aware of the change prior to the RFC reaching them.

Annually

All APIs need to be tested annually, but as part of APIM your external facing API is included in the APIM Platform Annual Pen Tests[961]. The APIM guidance on the penetration testing process (see page 968) includes that your API will get rolled up into the APIM Annual Pen test. This reduces the number of duplicated findings across the platform.

7.15.4.4.3.4 Who does the penetration test?

This depends on your API:

- If the security risk is high, you will need to arrange for an external company to do your pen test
- If the security risk is low, you can do it yourself

961 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/214149668/APIM+Platform+Annual+Pen+Tests

Some examples of APIs that may only require internal testing (defined below):

- Shows no sensitive data
- Single POST endpoint
- Interacts with a backend that only contains public information

APIM and Cyber Security have worked on a risk based assessment which will help you and Cyber agree what type of testing you need.

7.15.4.4.3.5 The penetration test process

For more details on the process, see API penetration testing process (see page 968).

7.15.4.4.3.6 API penetration testing process

Process for agreeing level of testing required through to go live:

Step 1 - Do we need to do a pen test?

Most teams ask, do we need to do a pen test? For some situations, it is possible to **not** require a full third party pen test. However, to do that you need to understand the risk your API will expose.

The Risk assessment preparation (see page 972) page helps your team & Security officers to request Cyber Security to help with a risk assessment on your API and its endpoints.

Step 2 - Risk assessment by Cyber Security

Submit this information to the Cyber Security team asking for a risk assessment to be performed.

The aim of this is to ask the question: is a third party pen test required for your API or can it be internally tested?

Step 3 - Agreement of testing

Cyber Security will assess the situation and reply back.

If the risk is sufficiently low and does not alter the attack surface significantly and the agreed internal testing has been completed then the API can be approved for release without an external pen test. Go to Step 4a.

Otherwise, if the API presents sufficient risk, the normal 3rd party penetration test process must be followed. Go to Step 4b.

Step 4 - Testing

As an API Producer, your pen tests do **not** have to include the following aspects in scope:

- Runtime:
    - Apigee perimeter

    - The OAuth mechanisms (i.e. the Identity Service) used by the platform, including CIS2 and NHS login integrations
    - AWS Hosted Container infrastructure (including: network vulnerability, config, authentication)
- Apigee Management (management UI and API)
- Apigee Developer (login, functionality, and privilege escalation)

For more information on what's covered by the annual platform pen test see APIM Platform Annual Pen Tests[962]. If this is a first time pen test, then that page and documents might help you define your pen test document more clearly.

Your areas of responsibility that need to be in scope

- The associated Apigee Proxy:
    - The endpoints in the API itself
    - Security (normally OAuth access tokens) is implemented correctly on the each endpoint, specifically different auth on different endpoints and broken object level authorization. For more info see OWASP API Top 10[963]
    - Strictly the actual authentication mechanism is not in your scope, security testing to obtain an access token is excluded. However, it can create more complications by excluding this, because the pen tester should have the ability to obtain their own access tokens
- Circumventing Apigee if you have your own backend (there must be no direct access to backend).
- Your API backend itself:
    - Can the logic inside your backend that could be compromised?
    - Any standard pen test aspect of your API backend (e.g. AWS infrastructure review, secure code review)
- If you have Docker containers using the APIM AWS Hosted Containers functionality - the pen test only has to cover:
    - Can the logic and services inside your container be exploited via the API endpoints?
    - Consider doing a secure code review, if you are following the NHS Digital Software Engineering Quality Framework (SEQF)[964] this should not be needed

If you have pen tested your API and then moved it behind Apigee speak to your Cyber Security representative to determine if you need to re-test. If you have some good automated testing of different access modes it should be fine to not re-test.

Step 4a - Internal testing (no third party test needed)

The following testing needs to be performed and documented, ready to return to Cyber Security in place of a full pen test.

- Code peer reviewed.
- Automated code, that runs on each deployment:

962 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/214149668/APIM+Platform+Annual+Pen+Tests
963 https://owasp.org/www-project-api-security/
964 https://github.com/NHSDigital/software-engineering-quality-framework

    - Security - negative testing, showing that all endpoints are protected by an access token or API key
    - Functional testing
- Code tested by testing team. Testing should include (that is not included in the automated testing):
    - Functional testing.
    - Non-Functional testing, if needed (e.g. if 5tps rate limits are in place, unlikely to need performance testing)
    - Exploratory testing.
    - Negative testing.
- Code scanned for secrets
- Code scanned using a static code and vulnerability scanning tool such as SonarCloud

Docs

- API internal testing - how to guidance (see page 974)
- Evidence template for Cyber to mitigate an external Pen Test (see page 978)

Step 4b - Third party pen test

If your API requires an external pen test, please follow the process detailed in the pages linked.

Docs:

- API third party penetration testing - how to guidance (see page 990)
- Recording a third party penetration test template (see page 996)

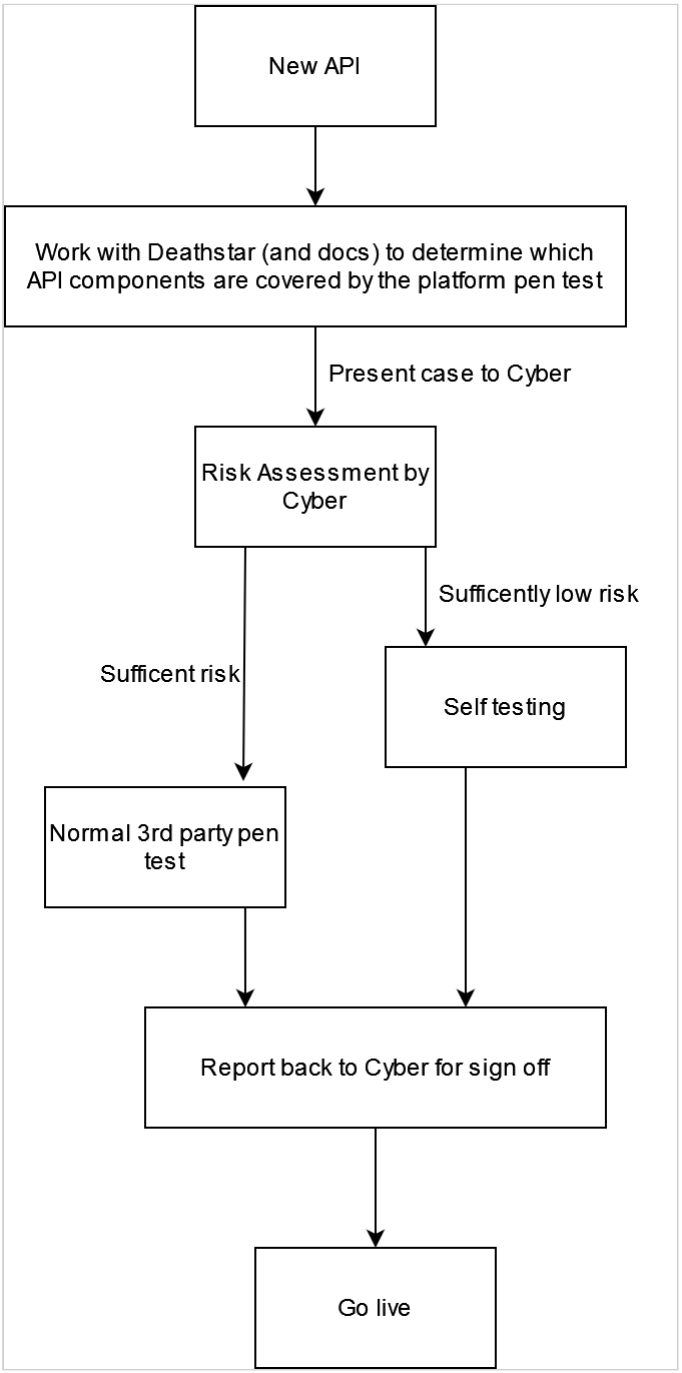
Step 5 - Return to Cyber Security

Once your testing is complete return to Cyber Security to obtain sign off for go live.

Contact details

Cyber Security	cybersecurity@nhs.net[965]	General Cyber Security Inbox Each API should have a representative from Cyber Security, who will have their own contact details

965 mailto:cybersecurity@nhs.net


New API
Work with Deathstar (and docs) to determine which API components are covered by the platform pen test
Present case to Cyber
Risk Assessment by Cyber
Sufficently low risk
Sufficent risk
Self testing
Normal 3rd party pen test
Report back to Cyber for sign off
Go live

Risk assessment preparation

Introduction

The API platform is annually pen tested. Due to this, Cyber have agreed components covered by our pen test do not need to be re-tested by each individual API team. Under certain conditions, it might be possible to delegate the pen testing work to the annual test - proceeding into Private Beta (or a change) without it.

To determine if this is possible, you should create a risk assessment for Cyber Security to assess; this page includes a sample risk assessment request and a template.

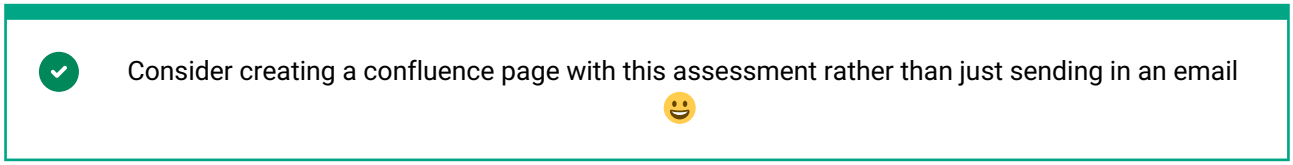
Process

To conduct a risk assessment (standard approval process) Cyber need to know the following for each endpoint:

- API function (Create, Read, Update, Delete)
- What resources does the API interact with/connect to? (what data does it have access to? is this sensitive?)
- How authentication is handled (does the API use APIM auth? etc)

Work with Deathstar and use the template below to prepare a risk assessment request. Once agreed, engage with Cyber.

Template Risk Assessment Request



Dear Cyber,

I am the <job title> working on <api name> [link to https://digital.nhs.uk/developer/api-catalogue catalogue page if possible]

On our journey to go live we are looking to determine if the API needs a 3rd party pen test or can be covered by the APIM platform pen test and some internal testing.

Provided here is the information required by Cyber (as agreed with API Platform) to conduct a standard approval process.

For the API as a whole:

1. What is the purpose of the API?
    a. One paragraph summary
    b. Insert link to API Catalogue and spec entry
2. What resources does the API connect/interact with?

    a. List resources
    b. Do these resources contain sensitive data? There should be a data classification agreed (e.g. Class III data), if so include that - keeping in mind the API might **not** expose the full dataset

1. How is authorisation handled?
    a. Does the API use APIM auth shared flows or your own custom policies?
        i. If so define which mechanism
        ii. if not please detail how auth is handled

Answer these for each endpoint of the API:

1. What is the function of the endpoint?
    a. Create
    b. Read
    c. Update
    d. Delete
2. Does it have specific authorisation rules different to the API overall?
3. Does it allow access to more or less sensitive resources to the API overall?

<If you have any other relevant resources attach them here with descriptions for example links to git repos (if public), links to confluence area, etc>

Using the table and other linked resources please can Cyber perform a risk assessment for <api name> and decide whether the API go live process can procced with internal testing, or if a full 3rd party pen test is required.

Regards

<name>

Example risk assessment request

Using the Health Research Data Catalogue[966] as an example:

For GET /datasets

- Function of the API
- Read
- Resources connected to and interacting with
- Collibra / HDR backend / Central Metastore
- NO PII/sensitive/user data
- Authentication
- Application-restricted RESTful API - API key authentication
- Apigee > API Backend secured using TLSMA

For GET /datasets/{id}

966 https://digital.nhs.uk/developer/api-catalogue/health-research-data-catalogue

- Function of the API
- Read
- Resources connected to and interacting with
- Collibra / HDR backend / Central Metastore
- NO PII/sensitive/user data
- Authentication
- Application-restricted RESTful API - API key authentication
- Apigee > API Backend secured using TLSMA

Contact details

Cyber Security	cybersecurity@nhs.net[967]	General Cyber Security Inbox Each API should have a representative from Cyber Security, who will have their own contact details

API internal testing - how to guidance

Internal testing (no third party testing needed)

API management has standard patterns for securing your API that have been penetration tested and deemed secure (see page 978) on a number of occasions.

If your API does not deviate from the established patterns, the following testing needs to be performed and documented, ready to return to Cyber in place of a full pen test.

Security Testing is a subset of broader tests that prove API is working properly (see page 1145). i.e. These are necessary but not sufficient conditions for a good suite of tests!

As an API Producer, following the NHS Digital Software Engineering Quality Framework[968], you will already be doing automated testing.

The aim of this guidance is not to vastly increase your automated testing, but help you tailor it to also prove that your API is **always** secure rather than just once a year on the required Pen Test.

967 mailto:cybersecurity@nhs.net
968 https://github.com/NHSDigital/software-engineering-quality-framework

Apigee layer security

This means testing that your API really is secured by Apigee in the manner you think.

This type of testing is easily automated and should be done by your CI/CD pipelines immediately after deployment in path-to-live (`ptl)` environments, with limited testing in production where appropriate.

Each endpoint of your API (each path and HTTP verb combination) should have:

- Security Happy Path Testing: using valid access tokens and/or API keys in requests gets 2xx responses from your API.
    - Note: Automated testing should be done in `ptl` environments, but may not be appropriate for production environments if submitting data or retrieving confidential data.
    - This should include both general functional testing, and regression testing.
    - It is important to balance speed of implementation with depth of testing - the more complex scenarios might have to be tested manually in the first phases.
- Security Negative Testing: Invalid (or not-present) access tokens and/or API keys in requests get 401 Unauthorized responses from your API.
    - This must also be done in production

Though it is configured correctly out of the box, this includes proving that the required `_status` endpoint is secured by the platform-wide "status endpoint API key".

External backend security

Prove that your backend cannot be accessed directly (bypassing the auth done by Apigee):

- HTTPS calls to various levels in the path
- At regular intervals, a port scan needs to be done (ideally at a low rate to not trigger any alarms)

This may be done automatically or manually. If automated, the pipeline that deploys your backend should run them post deployment. If manually, the results must be recorded somewhere as evidence, e.g. Confluence.

The connection between Apigee and your backend should not be weaker than the security used to access your API via Apigee.

Some APIs on the platform secure their backends with a secret apikey, stored securely in Apigee (and AWS secretsmanager).

- This is only appropriate because API consumer access to the Apigee layer is secured by their application API key.
- This is a weak form of security, only appropriate if the data transmitted by the API is not very sensitive.

For most APIs, the connection between Apigee and the backend is secured by TLS Mutual Authentication[969].

Object Level Authorization

You must prove that your API correctly enforces object level security.

This is not necessary if all users have equal access to everything exposed by your API.

This is type of requirement is common if your API exposes object identifiers (e.g. id in an URI path `/api/object/<id>)` and any given API consumer should only have access to a subset of objects.

Consider this simple illustration. Your API has a path design `/api/user/<userid>/data` . You must prove, via automated testing, that:

- an access token from **user A** **can** access `/api/user/A/data`
- an access token from **user B** **can** access `/api/user/B/data`
- an access token from **user A** **cannot** access `/api/user/B/data`
- an access token from **user B** **cannot** access `/api/user/A/data`

where **user A** and **user B** are both real users able to generate valid tokens to access your API.

The specifics of your tests will depend your object model.

This type of mis-configuration is item #1 in the OWASP Foundation Top 10 API Security Points[970]

Malicious Request Data

Ensure your API is not vulnerable to injection attacks, for example `SQL` injection.

Prove that malformed requests do not crash your application. Potential vulnerability will depend your technology stack.

This may not be necessary if your API is "read-only".

969 https://www.cloudflare.com/en-gb/learning/access-management/what-is-mutual-tls/
970 https://owasp.org/www-project-api-security/

Protection against this type of attack involves security minded design and good engineering principles.

But in particular you should:

- avoiding use of user provided strings to execute commands where possible
- preferring an allow-list of commands to a deny-list to minimize the space of possible inputs
- testing the effects of the allow-listed commands on your backend to prove they are handled safely by your application

Evidence of these design decisions and their implementation should be documented in confluence.

Transmit the minimum data necessary to do the job

As well as being an inefficient use of bandwidth, there are scenarios in which excessive data exposure presents security risks[971].

Ensure that your API does not expose anything it doesn't need to. Document this in confluence.

Non-functional testing

If your external rate limit is low (e.g. 10 requests per second), then performance and load testing is potentially not needed. You as a project or programme need to look at the risks, but if you are using the APIM Rate Limiting shared flow your backend is protected.

As your service grows, you will need to do other types of testing, but this is not needed as part of the pen-testing work. For instance, resilience testing to show that your high availability is actually working. Scalability and stress testing might be required from Live Services teams to show that your service can meet it's SLAs.

Sufficient logging and monitoring

One of the advantages of coming on the API Management platform is that you get request level logging out of the box and alerting out of the box.

If your API has sensitive data in query parameters, headers, or the URI path, you will need to prove you have configured redaction of sensitive data. See this page (see page 1082) for instructions.

Automated quality and security tools

Good engineering/DevOps practice and peer review are the primary tools, however there are also automated tools you should make use of to ensure you have not accidentally exposed your API keys or certificates.

1. Ensure each developer is Using nhsd-git-secrets tool pre-commit hook on your repo (see page 1102)
2. Ensure that Dependabot is being regularly monitored: Using Dependabot on GitHub Repos (see page 788)
3. Ensure github code scanning[972] is enabled
4. Code scanned using a static code and vulnerability scanning tool such as SonarCloud[973]

971 https://github.com/OWASP/API-Security/blob/master/2019/en/src/0xa3-excessive-data-exposure.md
972 https://docs.github.com/en/code-security/secret-scanning/about-secret-scanning
973 https://sonarcloud.io/

Previous external pen tests examples

Previous pen tests:

- Penetration Testing for PDS Update API[974]
- Private Beta - initial security position[975]
- Platform pen tests[976]

Evidence template for Cyber to mitigate an external Pen Test

This is a new form / process - Please feedback on what could make it easier to carry out and prep for this would

This page gives recommendations on how to document the tests described here (see page 974).

Apigee layer security

This process just ensures that you have sufficient test coverage.

Ensure your spec is up-to-date. Go through your API spec (and any private undocumented endpoints).

Complete the table below

URI	Verb	Happy Path	Link to test	Link to pipeline result

External backend security

We are using the TLSMA/apikey security pattern for our backend.

Here are a selection of requests that prove connecting without the credentials are not possible.

974 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117666342/Penetration+Testing+for+PDS+Update+API
975 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/105551365/Private+Beta+-+initial+security+position
976 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/144742580/Platform+pen+tests

```
curl https://my-backend.nhs.uk/

# your response goes here
```

Object Level Authorization

For the subset of endpoints that have an exposed object model.

Complete the table below for a path-to-live testing environment.

URI	Verb	Object owner	Response from object owner token	Response from other token	Link to test definition	Link to pipeline result of test

Malicious Request Data

The following endpoints run commands directly from user input (e.g. SQL).

*** Your OAS file should specify some of the definitions for API Consumers, you might be able to speed up completing this by cross referencing the details***

URI	Verb	Query Parameter	Request payload	Response	Link to test definition	Link to pipeline result of test

Transmit the minimum data necessary to do the job

Briefly describe the data model exposed by your API for each endpoint. *** You could link to your public facing OAS file on the NHS Digital site here ***

Give a short justification for each field returned. *** During the API Design phase you should have considered this, ideally a link to your API IG / Privacy / Architecture page could be inserted into here ***

Non-functional testing

If your API has an expected workload higher than tens of transactions per-second, attach log data showing your API handling the number of request specified by your Service Level Agreement (SLA).

Otherwise state that non-functional testing is not required for the low event rate specified in your SLA.

Sufficient logging and monitoring

Add a few log lines from the `requests_apm_ptl` splunk instance.

State the name of any headers/URI path parameters/query parameters that require redaction.

Attach sample log data from your API showing that any fields requiring redaction really are redacted on the redacted indexes.

Code Analysis

Attach the output of SonarCloud and Github Secret Scanning here.

APIM standard responses to pen test findings

As APIs get tested, there is a common set of findings that can appear, this page covers how they have been tackled, or are being planned to be resolved.

Issue	Approach	JIRA	Previous statements / Notes	Notes
Not using TLS1.3	(if rated higher than INFO) Accept residual risk	-	• APIM AWS Hosted Container pen test • Residual Risk Statement signed off by Stephen Koch has the following mitigation statement in: “TLS 1.2 is still valid, and we don't need to upgrade at this time. Limited client support for TLS 1.3 in some common development languages”. • Full details here: https://nhsd-confluence.digital.nhs.uk/display/APM/APIM+AWS+Hosted+Container+Pen+Test • PDS Update API Pen test – see API-02 on this page: https://nhsd-confluence.digital.nhs.uk/display/APM/Penetration+Testing+for+PDS+Update+API • AWS and apigee now support TLS1.3, APIM will work on deprecating TLS 1.2 when NHS Digital looks to move all platforms onto TLS 1.3 • As of March 2022 we have sign off from Mike Farrell that if pen tester reports not using TLS1.3 as INFO it is 'agreed not applicable', if rated higher than INFO it is 'accept residual risk' • As of April 2024 - Public Edge still does not support TLS 1.3 - https://docs.apigee.com/api-platform/fundamentals/configuring-virtual-hosts-cloud	
HTTP 401 403 error responses	Accept risk, no action		Agree an APIM wide set of error messages that align with security principles, but allow API Consumers to self-serve fixes in many cases: Pen test approach - HTTP 401 403 error responses (see page 986)	
Content Security Policy	Accept risk, no action		Suggest: • “Content Security Policies are not relevant for APIs, as there is no related content to run. If the definition is to be extended, NHS Digital can’t specific the list of domains since each API Consumer will have a different, constantly changing list”	

Issue	Approach	JIRA	Previous statements / Notes	Notes
CORS	Accept risk, no action	-	In EPS Pen test API-01 related to this and agreed CAP included this statement: "Cross origin requests are allowed because these are publicly available APIs. In order to perform any actions, the request needs to include a valid access token. An attacker would need to steal this before they could conduct an attack." APIM recommend the CORS header is set to mirror incoming domain - see API Producer Cross-origin resource sharing (CORS) (see page 786) page	
X-Frame-Options	Propose N/A		Highlighted risk (from Immunisation history API pen test[977] API-03) = "This header protects against Click-Jacking attacks, in which a user can be tricked into clicking on, or copying, website content from one service to another." Reasons for not fixing: • APIM does not serve content with iFrames data results nor are susceptible to click-jacking **Note** that OWASP cheat sheet series[978]for APIs does mention this as a header, and does recommend including... but only **incase** you serve HTML...! You might also argue inclusion allows standard security scans to pass the API	
X-XSS-Protection	Propose N/A		Highlighted risk (from Immunisation history API pen test[979] API-03) = "This header instructs web-browsers to enable additional protections against cross-site-scripting (XSS) attacks. XSS enables an attacker to embed malicious script into a web-page which is then executed by the user's web browser."	

977 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/203362233/Immunisation+history+API+pen+test
978 https://cheatsheetseries.owasp.org/cheatsheets/REST_Security_Cheat_Sheet.html#Security_headers
979 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/203362233/Immunisation+history+API+pen+test

Issue	Approach	JIRA	Previous statements / Notes	Notes
Cache-Control & Pragma	Implement - to gain some edge case benefits		Implement standard shared flow that sets `Cache-Control: no-store`, for some APIs it might be relevant to allow caching. Should we include Pragma for completeness? It is only relevant to HTTP 1.0 clients, but there is likely to be some older infrastructure in some of our clients. API security risks are focused are really only about data leakage (there is no code being sent to clients which could be found to have an exploit), but all the connections are encrypted - is this a security risk at all? API clinical risks are around the content being cached and an out of date record being used	
X-Content-Type-Options	Implement (maybe) - to gain some edge case benefits?		As APIM is not supplying the containing web page, the standard security risks don't apply. There might be a clinical risk around misinterpreting the data is MIME sniffing takes place - look at the behaviour of HTTP client software. Setting this might make it more robust?	

Issue	Approach	JIRA	Previous statements / Notes	Notes
Rate Limiting	False positive		MTI recommend "...As a standard, it is recommended to limit a user to 15 to 25 requests a minute..." (this example was taken from the 2021 APIM Annual Pen Test[980]). APIM has no plans to offer built-in "per user" rate limiting, but offer "per app". This adds some complexity to an already difficult to manage area. For instance, APIM has to consider how that per-user rate limit applies to the APIM Auth Service as well as requests across multiple APIs. At a "per app" level, the default is 10tps, which allows: • Basic automated testing in INT, but returns 429s regularly • Most backends, even without performance testing, can manage 10tps • In Production, we do have some low volume services - but some API Producers choose to raise these values As of March 2022, we have signed off from Mike Farrell that Rate limiting is a false positive no longer accept risk, no action.	
Excessive permissions configured for shadow file	No action		The permissions for this file on our Linux instances are set up to 000 as recommended meaning only the root user can read and write this file.	

980 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/311837111/2021+APIM+Annual+Pen+Test

Issue	Approach	JIRA	Previous statements / Notes	Notes
Default port number exposed in source control	Accept risk, no action.		While it's not best practice to expose the default ports, the exploits are limited given the shape of our infrastucture. It would be a lot of work to replace this method and we dont have easy solution. Note that no other port should be exposed in source control apart from the default, 9000. Agreed no action. See comments from cyber: • The hardcoded IP/port in question is an invalid/non-routable IP intended to point to the local machine (0.0.0.0). • The container this code runs in is not connected directly to the internet, instead traffic only flows on this hard coded port inside our AWS VPC and is constrained by AWS security group rules to come only from our ingress servers	

Pen test approach - Excessive permissions configured for shadow file

Introduction

In the old days of Linux, hashes were stored in `/etc/passwd` (which is why it's called `passwd`), as back when Linux was created, cracking a hash, even the weak types used back then, was practically impossible. Eventually, though, processing power advanced to cracking a hash, at least of a relatively weak password, became feasible.

There are many legitimate reasons to be able to read `/etc/passwd` as a regular user (converting UIDs to usernames, getting a user's full name, phone number, etc), so hashes were moved to `/etc/shadow`, which was given 000 permissions in our ubuntu images. An `x` in place of the password hash field for a user in `/etc/passwd` is used to indicate that the hash for that user is stored in `/etc/shadow` instead.

Our current Linux images are built using Hashicorp Packer and the permission configuration for these files is the following:

passwd file permissions

```
-rw-r--r-- 1 root root 2207 May 27 14:18 /etc/passwd
```

shadow file permissions

```
---------- 1 root shadow 1151 May 27 14:18 /etc/shadow
```

The owner of the /etc/shadow file is the user **root**. Other users are not allowed to read the file directly, to prevent them from gathering hashed passwords of others

Proposed approach

Take no action as the permissions for this file on our Linux instances are set up to 000 meaning only the root user can read and write this file.

Pen test approach - HTTP 401 403 error responses

Some pen tests pick up how an API returns a message payload in security / authorisation situations. The driving force for the concern is that detailed error messages can help attackers infer details about the the solution - which in turn will allow them to exploit weaknesses.

However, a blanket "no payload" approach does not help the API Consumers who work with the API on a day to day basis. The proposal is to:

- Allow a pre-agreed list of error messages and responses
- None of these messages make it easier for attackers (especially considering APIM Access Tokens have a 10 minute lifetime)
- None of the messages have free text, nor could provide stack dumps

This approach makes a massive difference to API Consumers, a 401 could be something trivial like the Authorisation Header is empty (due to a simple config error). If there is no payload then the API is likely to contact NHS Digital to find out why they are getting a 401. Not only does this waste time, it puts a resource loading on NHS Digital.

External examples where limited information is returned in these type of errors:

- Both the RESTful design pattern and Government Digital Service (GDS) guidelines focus on making APIs intuitive, easier to understand and easier to support
- The OAuth 2.0 standard includes some quite detailed errors for the various end points, e.g. https://datatracker.ietf.org/doc/html/rfc6749#section-5.2
    - APIM are not exposing anything more sensitive than the standard
- Spine (the older STU3) exposes some detail: https://fhir.nhs.uk/STU3/ValueSet/Spine-ErrorOrWarningCode-1

Proposed error messages

APIM uses the FHIR OperationOutcome JSON format, for a detailed breakdown of what this - see: Handling errors (see page 624). When looking at the full set of fields, it is worth noting that:

- The inclusion of the "issue.details.coding.system" value is part of the FHIR standard, and a knowledgeable attacker could look at the system value and infer that the rejection was generated by the API Management layer vs. the backend
- The standard has a cardinality of 1 to many for the "issue" object, however, in these error messages it will always be limited to 1. There will be no accidental addition of extra issues that might contain additional information
- There will be no "issue.location" nor "issue.expression" field values
- No variables returned in any of the fields - the fields are always fixed text

Scenarios with proposed text:

	Http Code	**system**	**coding.code**	**Display**	**Diagnostics**	**Notes**
No access token provided	400	APIM	API_ACCESS_TOKEN_MISSING	No Access Token sent	The Authorization Bearer header was not provided in the request	
Apigee - invalid	401	APIM	API_ACCESS_TOKEN_INVALID	Access Token invalid	The bearer token could not be retrieved from the Authorization header	
Apigee - expired	401	APIM	API_ACCESS_TOKEN_EXPIRED	Access Token expired	Access Token expired	
Apigee - needs higher auth	403	APIM	API_ACCESS_DENIED_LEVEL	Authentication was not at a high enough level	Is this possible to track? e.g. P0 presented, but needs P5?	
Apigee - unauthorized config	403	APIM	API_ACCESS_DENIED_CONFIG		Is this possible? are there scenarios where the	

	Http Code	system	coding.code	Display	Diagnostics	Notes
Apigee - unauthorized	403	APIM	API_ACCESS_DENIED	Not authorised	Access Token does not have the correct authorisation. This is the catch all code	
Backend - pass through message	xxx	Backend	From Backend	From Backend	From Backend	

Agreed

27th Sep 2021 - Cyber reviewed proposal, and didn't have an issue with the constrained approach above. Error index needs to be tidying up and circulated for formal input from existing API Producers and approval from Cyber Security.

Pen Test approach to ECS - Fargate containers and firewalls

In one of the AWS infrastructure reviews it was noted that the egress-haproxy local (linux) firewall was not enabled. This was raised as a Low severity (1.0 score). In the pen test report, the two main sentences for this recommendation were:

- "Ensure that the firewall is enabled and access to all **unneeded ports are blocked**"
- "It is recommended to ensure that the firewall is enabled on all hosts and that only the required ports are **exposed to remote users**"

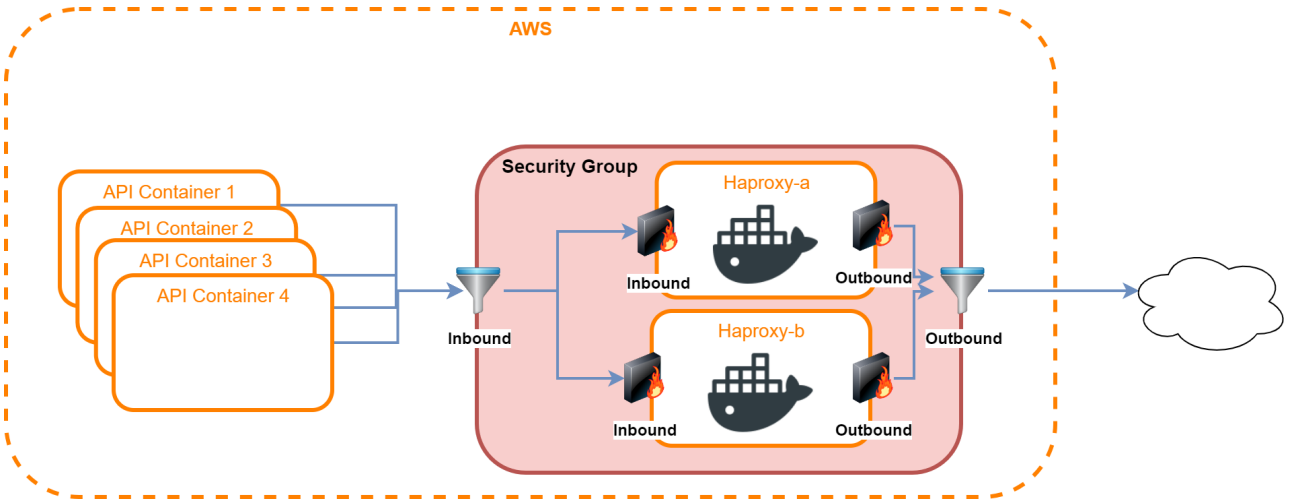
The words in bold, have been highlighted as they indicate some of the driving points.

The proposed APIM response is that they should not be enabled, as the Security Group provides the equivalent protection and suitable mitigation. The following mitigations / observations to support this are:

- By doing this, it introduces greater risks, because we are implementing something that has a more fragile architecture and more prone to failure
- Turning on the local firewall inside the Security Group then results in using having back to back firewalls. This is one of the anti-patterns identified in the NCSC guidance: https://www.ncsc.gov.uk/whitepaper/security-architecture-anti-patterns#section_5.
- Security groups allow us to manage permissions without worrying about IP addresses – but now we have to deploy, discover IPs and then update firewalls inside the containers

- Fargate does not allow this level of control inside the container, indicating that as long as Security Groups (and other similar mitigations) are in place then EC2 containers should not be treated differently
- No external requests are hitting this service directly
- These egress proxies specifically use ports to help manage complex routing - which makes this more complex than other standard implementations

The following diagrams sets context, but also shows the back to back nature of the firewalls. The security groups / firewalls are shown split between inbound and outbound (even though the are same devices) for clarity:



Proposed guidance for other projects

If a project has this issue flagged in a Pen Test, then the recommendation would be marked as "Accepted" and be closed with no action as long as:

- This was an AWS based solution
- There are no direct connections to the containers from the public internet
- The Security Groups have the same purpose, in context, as the firewalls would
- The Security Group(s) are configured to:
    - Not have any to any routing, but to always specify valid sources and destinations in the Architecture
    - Restrict inbound and outbound ports to the minimum
- Show, if needed, the mitigations to stop requests going to arbitrary outbound internet destinations
- The Security Group doesn't contain other services:
    - i.e. that would allow hops from one compromised service to another in the same Security Group
    - If there are different services, then separate Security Groups should be used to encapsulate them

Questions (open for comment!)

This article: https://degreesofzero.com/article/docker-and-firewalls.html talks about configuration of Docker, and how Docker overrides iptables behaviour - recommending that:

- Configuring iptables is complex for little gain - use firewalls outside of Docker
- Recommend binding the published ports to 127.0.0.1:

Immediate questions:

- Is this local IP port binding relevant to an EC2 hosted container?
- Do EC2 instances actually also override iptables, and so turning on the firewall has no impact?
- Is the advice on the page reasonable?
- Is there any advice APIM / NHS D take on here?

API third party penetration testing - how to guidance

Overview

Once you have determined pen testing is required, we recommend using our template[981] to keep all the information and resources together in one place.

Use the detailed guidance below for a full walk through.

In summary, the process starts by the API Producer defining the pen test using the security services framework order form. Once drafted, it must be reviewed by the API Producer Cyber Security contact and then sent on to Cyber (finance are also notified at this stage).

The work is put out for tender and awarded (this is handled by Cyber) and is then signed off by the budget holder, allowing the requester to raise a purchase order.

The environment for the pen tester should be set up around a week before the test, which may include the arrangment of any equipment to be couriered to them (eg: smartcard / reader), as well as arranging any paperwork (such as the consent form) and the preparation meeting.

One or two days before, prepare to maximise success: smoke test APIs with a smart card, ensure there are no potentially disruptive releases planned for during the test and ensure there is support for the pen tester available, particularly early on.

Once the pen test has been completed, the findings must be analysed. All issues reported must be acknowledged. We provide a list of common reportings and standard approaches (see page 980).

Reported issues must be addressed; action plans and risk assessments can capture outstanding issues all of which must be signed off by senior management before continuing the go live journey.

Below is the detailed guidance for the pen testing process.

981 https://nhsd-confluence.digital.nhs.uk/display/APM/Recording+a+Pen+Test+Template

For reference, the process page for this is here: https://hscic365.sharepoint.com/Data_Security_Centre/Pages/Specialist-Security-Services-Framework--.aspx *(which includes a full process flow).*

Pen test process

Once you have determined you need to perform a 3rd party pen test in order to go live follow the process detailed on this page.

Define pen test

It is the API Producer's responsibility to define the tests, not the role of the pen testers. This makes it significantly harder to write.

The blank template is linked to from the Cyber intranet page . It does get updated regularly so always pull down the latest from:

- Specialist Security Services Framework[982]

The scope for the testing must consider/include:

- **Backend via Proxy** - this is the majority of the testing, looking to exploit the API endpoints themselves.
    - State that **exploratory** testing is a requirement and that automated testing alone is not good enough.
    - Exploratory testing should be focused around the number of endpoints and types of parameters being sent . How can the definition of the APIs be exploited?
- **Proxy** - platform endpoints only.
- **No circumvention of Apigee** - Ensuring that Apigee can't be bypassed to reach the API backend (and / or the associated APIM AWS Hosted Container)
- List **all** endpoints that must be tested. It is wise to work out the full URLs at this point, as it can show up if any infrastructure is not ready. It's a good idea to provide a link to a published specification in the pen test spec form
- For user restricted APIs, it is sensible to use smartcards or NHS login during the testing. You need to make it clear, in the pen test form, that the APIM Authorisation (Identity) Service is out of scope. Technically, the CIS2 Authentication and NHS login integrations are **not** included in your pen test. APIM offers the mock identity services for INT, which provide valid APIM access tokens. Conversely, stakeholders and planners can then get confused about how it's going to be tested if mock services are used
- For more details about scope, see API penetration testing process (see page 968)

Once form is complete:

- Get the draft reviewed by the API Producer Cyber Security contact. For programme wide (e.g. Identity Service) contact Matthew Helm.
- Once reviewed, send to: cybersecurity@nhs.net[983]

982 https://hscic365.sharepoint.com/Data_Security_Centre/Pages/Specialist-Security-Services-Framework--.aspx
983 mailto:cybersecurity@nhs.net

- Warm up procurement by contacting finance (for contact details see below). They will create the PO later in the process.

Tender and award of external pen tester

Cyber Security handle this part of the process:

- Decide on the pen test date. This **cannot be moved**. Recent experience has suggested that the process must be kicked off a **minimum of 4 weeks** prior to the expected start of pen test execution.
- Send the completed Specialist Security Services form[984] to Cyber Security.
- Goes out for tender.
- If there is more than one response, APIM needs to choose a supplier. MTI has done a lot of Spine testing in the past.

Purchase order

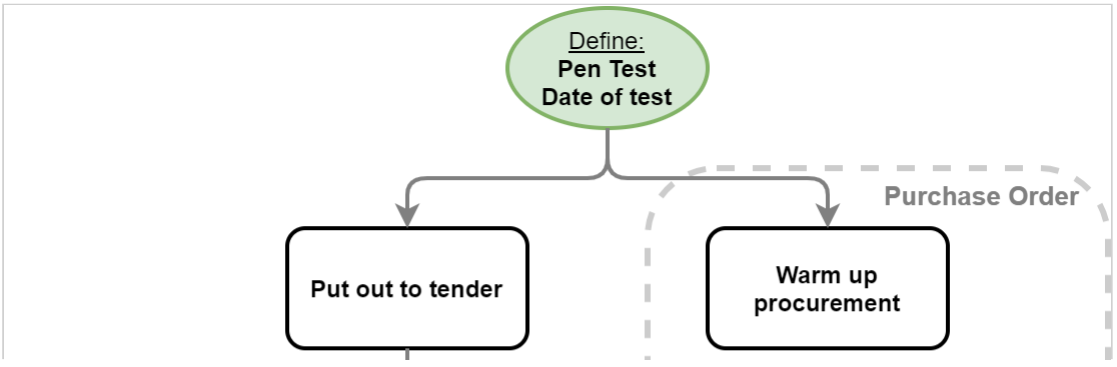
As soon as awarded to Supplier, APIM must:

1. Get the budget holder to sign and return to the supplier (Please copy in cybersecurity@nhs.net[985]).
2. Requester to raise a purchase order and provide it to the supplier and cybersecurity@nhs.net[986] to ensure the work carried out can be receipted. The following details need to be added to the PO to avoid any delays:

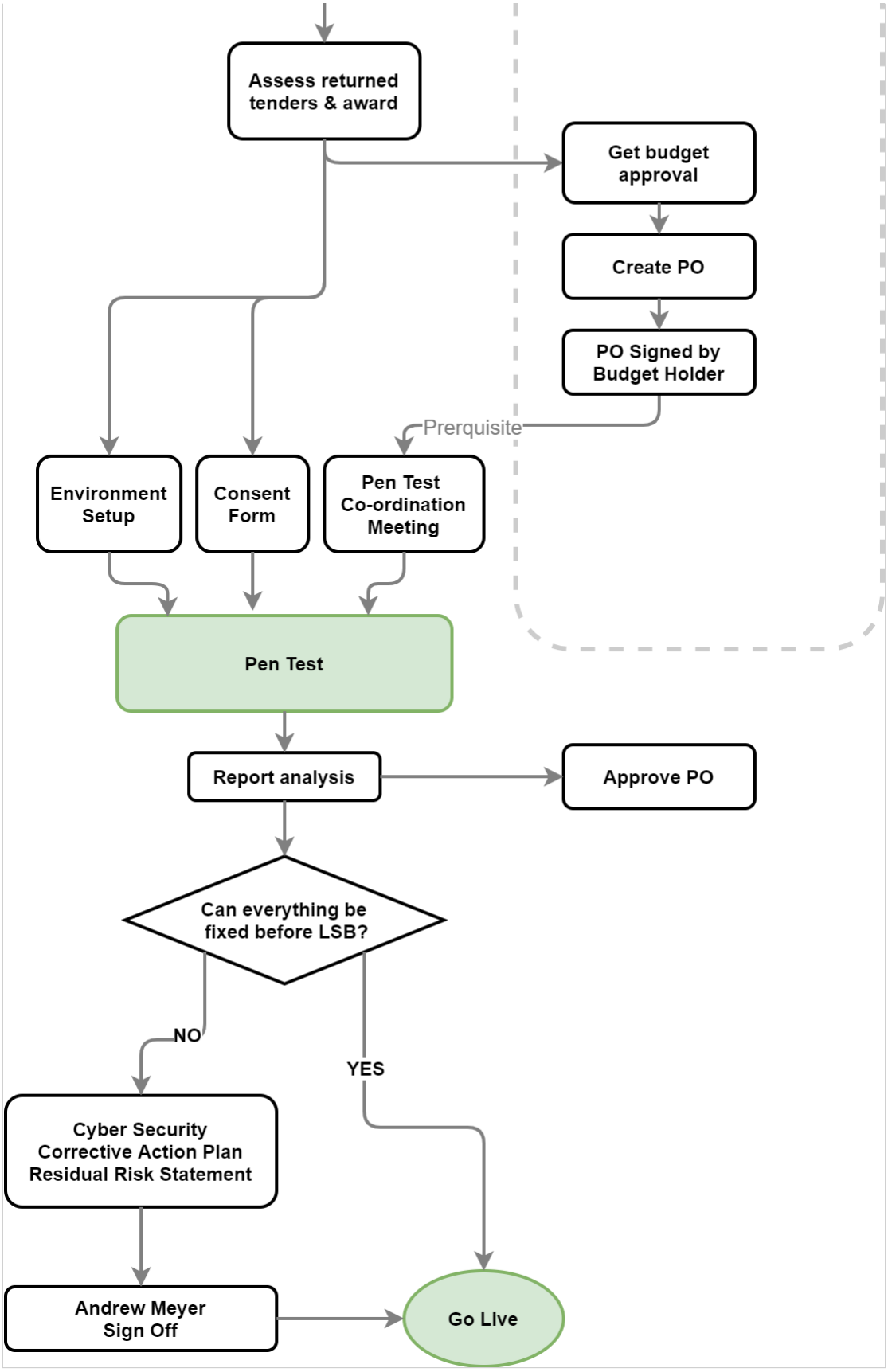
1. PRJ_1455 – Specialist Security Services.
2. Reference and title of your order (see email subject line).
3. Your portfolio number - PXXXX/XX (for squads in the **APIM Tribe** in almost every case this will be P0661/03).
4. The cost of the work.

Previous pen tests have been 3-4 days work, so budget £4K.

- PO's are raised by emailing BusOps (email address in Pen test contacts table below).
- Electronic image's of signatures were used at the key points .



984 https://hscic365.sharepoint.com/Data_Security_Centre/Pages/Specialist-Security-Services-Framework--.aspx
985 mailto:cybersecurity@nhs.net
986 mailto:cybersecurity@nhs.net


Assess returned tenders & award
Get budget approval
Create PO
PO Signed by Budget Holder
Prerquisite
Environment Setup
Consent Form
Pen Test Co-ordination Meeting
Pen Test
Report analysis
Approve PO
Can everything be fixed before LSB?
NO
YES
Cyber Security Corrective Action Plan Residual Risk Statement
Andrew Meyer Sign Off
Go Live

Environment setup

This will be different for each API, but will consist of:

- Apigee - use INT (generally - future maybe move to REF).
    - Warn Apigee about the pen test work - see https://docs.apigee.com/api-platform/faq/customer-security-testing-requests?hl=de#request. Apigee only need 1 days notice. Post on the API Producer Support channel for a case to be raised with Apigee.
- API Backend - agree which environment. Cyber Security will always ask for REF, however, for Spine testing it is now agreed that we use INT for these types of tests.
- Pen Tester - smart card:
    - They will need VPN into HSCN via AWS.
    - Pen Tester will have to have:
        - Access to the HSCN VPN via AWS.
        - Identity Agent and Credentials Manager installed.
        - Have a the following **couriered** to their home:
            - Hardware smart card reader.
            - Smart Card with appropriate access. Each smart card should be set up for **only one** environment
- Book courier:
    - Ideally 5 days before the date to keep costs down.
    - Will need pre-approval of costs. Budget a maxium of £300 for send and return, incase same day delivery is needed both ways.
    - (For PDS Update, we didn't use NHSD courier due to time constraints and tester posted the stuff and - hopefully - got reimbursed!)

Prepare for test

Early preparation:

- The pen test preparation meeting is really important for talking the pen tester through the key details of the testing.
    - This prep meeting **cannot** occur until the supplier has the PO in their inbox.
- Testing consent form will be sent over, which allows final URLs and gives consent to the pen testers to "try and break" the system.
    - A senior member of the squad, or architects can sign this off. Email approval is sufficient.
- For attended access scenarios, inform NHS Identity onboarding team of pen test and warm them up for potential need for support during execution.

Closer to the test (one or two days before):

- Smoke test all the APIs with a smart card the day before the pen test starts.
- Make sure there are no API Backend releases going on.

- Ensure Squad on standby to support on the first morning. Setting up a Slack channel to include the pen tester is very useful to ensure issues are resolved quickly. To do this:
    - Create a private channel with a suitable name.
    - A member or admin can invite the pen tester as a 'Guest' including only the single channel created and setting a custom time-limit to expire at the end of the pen testing activity.
- ** **Every hour lost simply reduces the actual amount of testing and could reduce the achieved scope, potentially resulting in invalidating the test activity.** **

Pen test

The awarded tester will perform the test, conclude their findings in a report and return it to you.

Fixes and residual risk sign off

Once the pen test has been completed the findings must be analysed. All issues reported must be acknowledged.

If all reported issues can be addressed before Live Services Board commence with the work and continue on the journey to Go Live, any outstanding issues will require a CAP (Corrective Action Plan) and Residual risk statement (RRS) to be completed by the security lead. If possible, engage with the security lead before you finalize the pen testing and submit to cybersecurity. If an RRS is required, this must be signed off by senior management.

We provide a list of common reportings and standard approaches (see page 980) use this page to identify our standard approach to the common findings.

Previous pen tests examples

Previous pen tests:

- Penetration Testing for PDS Update API[987]
- Private Beta - initial security position[988]
- Platform pen tests[989]

Contact Details

987 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117666342/Penetration+Testing+for+PDS+Update+API
988 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/105551365/Private+Beta+-+initial+security+position
989 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/144742580/Platform+pen+tests

Area	Contact	Role
Cyber Security	cybersecurity@nhs.net[990]	Find and engage with external suppliers General Cyber Security Inbox Each API should have a representative from Cyber Security, who will have their own contact details
Finance	Iain McQuillan	Budget holder for programme, primary code P0565/01
BusOps	william.blanc@nhs.net[991]	Raising PO
APIM	Tony Heap	APIM Product Owner
Business Services	Resolve IT	Courier booking

Recording a third party penetration test template

Save a copy of this page where you need it and complete it as you go through the pen testing process. Red text shows advice and where there is something you need to complete.

Complete all relevant sections or remove where not applicable.

Cyber Ref Number	<number>
Order Form	<link to form document>
Order Form Response(s)	<link to form document>
Consent Form	<link to form document>
Pen Test Result	<link to form document>
Pen Test Result Response	See below

990 mailto:cybersecurity@nhs.net
991 mailto:william.blanc@nhs.net

RRS & CAP	<link to RSS document> <link to CAP document> RRS Signoff: <link to RSS signoff document>

For full context see <link to any other relevant pages>

Notes on scope

<If there are any special notes or considerations; this could be if the scope has changed since a previous pen test, or anything that falls out of scope worth noting>

Pen tester setup

For this round of testing <insert pen tester names/company> are to carry out all the work.

Set the status bar to green and DONE once complete

Item	Status	Notes
NHS login credentials	PENDING	
Apigee App Created	PENDING	
JWKS configured	PENDING	
AWS Temporary Credentials	PENDING	
Apigee informed of pen testing dates	PENDING	

Include on if smartcard testing is required, otherwise delete

Item	Status	Notes
VPN to allow Smartcard reader to work	PENDING	
Smartcard Reader	PENDING	

Item	Status	Notes
Smartcard for INT	PENDING	<smartcard details>
Installed Identity Agent	PENDING	
Installed Credential Manager	PENDING	
Test Smartcard authentication	PENDING	

Detailed endpoint coverage

<description of required coverage>

The following are the APIs available in INT

API	Endpoint (INT)	Include/ Exclude	Action	Protection
<api name>	<link to endpoint>			

Pen test results

<summary of pen test findings, overall summary and counts>

Findings, next steps and tickets

Issue Code	Area	Severity	Description	Action (Proposed)	Ticket & status	Comments
				FIX		
				NO ACTION		
				WILL FIX		

7.15.4.4.4 API consumer onboarding

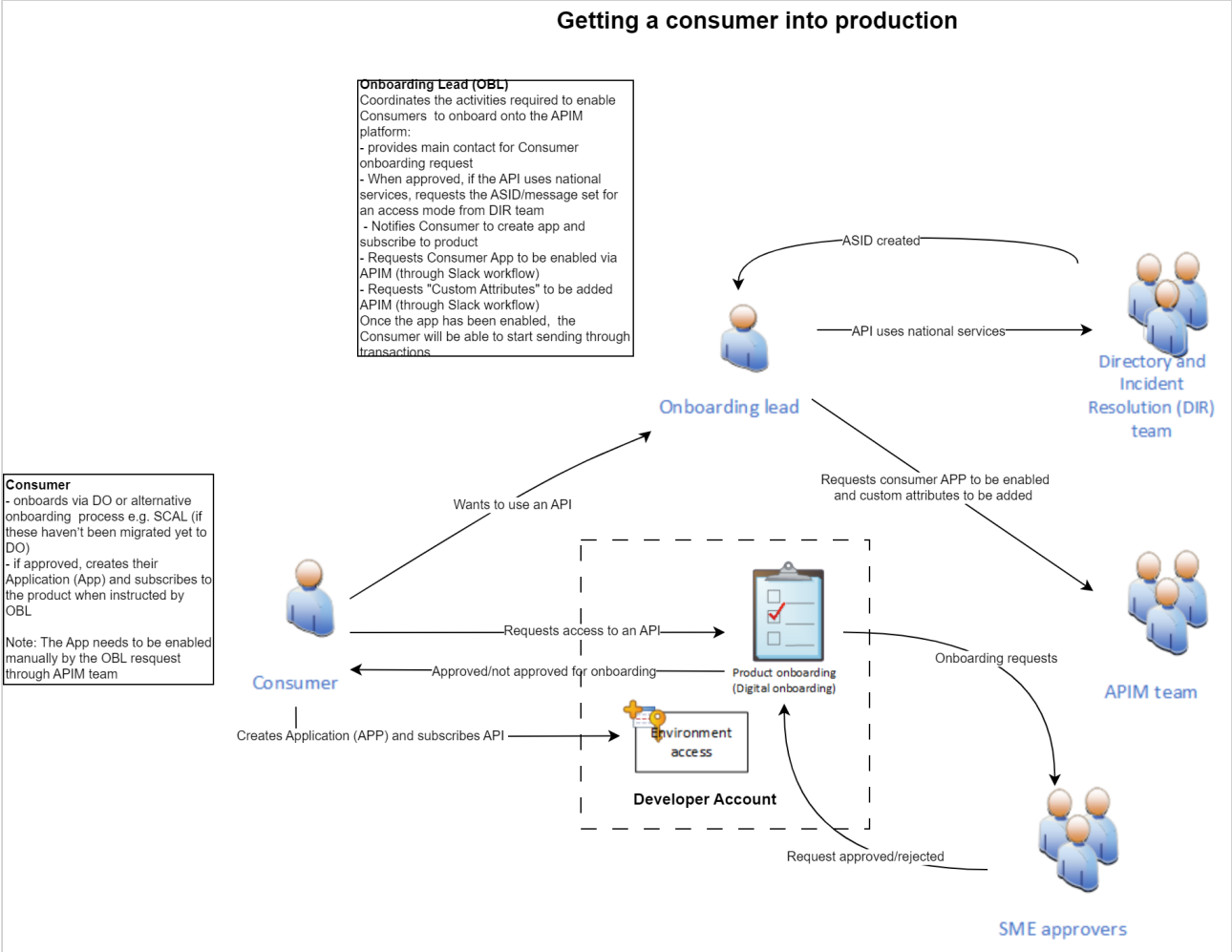
7.15.4.4.4.1 Getting a consumer into production

In the initial stages, consumers will have used the API catalogue to discover what API they want and completed testing in the integration (and possibly sandbox) environment. At this stage, they can start their onboarding journey to get their application (APP)/software service live into production (See https://digital.nhs.uk/developer/getting-started#get-your-software-onboarded for the API consumer perspective on this).

To get your consumers live, you will need to :

1. Have a nominated onboarding lead coordinate activities between the API consumers and NHS Digital, e.g. request enablement of the Application requests, requesting ASID's, custom attributes etc.
2. Have an agreed consumer onboarding process (see page 1001) in place.
3. Request that the Application be enabled in production once the consumer successfully completes the onboarding process. If your application contains more than one API you need will need to request enablement for each API separately, as enablement is per connection.

See below for an overview of the roles and responsibilities of the end-to-end process :



7.15.4.4.4.2 Onboarding lead

You'll need someone nominated as your onboarding lead - they are responsible for getting API consumers onboarded.

The onboarding lead might be someone in Service Design and Transition. However, we recommend that teams operate their onboarding process, so they can feel the pain of onboarding and iterate their service - especially during beta. Therefore we recommend the onboarding lead is someone in the API producer owning team.

For a low-usage API, this might be the delivery lead, product owner or business analyst.

Once the onboarding process is in place and approval has been given to go live, the onboard lead would coordinate the activities to:

- instruct the consumer to create or update their application and subscribe the API(s) to the application. If the consumer already has an existing application with other APIs subscribed they can simply add the new API to their existing application. At this stage the application is **not** enabled and all the following steps need to be completed before it is. Note : **consumer app must be owned by a team and not an individual**

- **for APIs that use national services,** request the ASID (see page 414)and associated message sets from the DIR (Directory and Resolution) team
- request that **each** API is enabled into production. You will need to submit a separate Slack workflow for each new API that is subscribed. For this, you will need assistance from the Deathstar team.

Where applicable, also include any custom attributes (see page 806). It is important when adding custom attributes to the form that you include the attribute name (**exactly** how the backend expects it, the case is important here too) as well as the attribute value. An examples might be x-ID, Charlie; first_name, Charlie; First_Name, Charlie etc..

- finally, instruct the consumer to run transactions through the API

7.15.4.4.4.3 The Digital Onboarding Service

The purpose of the onboarding process is to make sure the API consumer has addressed concerns around:

- privacy (IG)
- security
- clinical safety
- solution design and quality (solution assurance)

The Digital Onboarding Service (DOS) allows API consumers to submit requests for access to production via our online tool.

DOS is a new product, and at the moment, you'll need our help to get your API ready for it.

It may be helpful to read our contract with you (see page 1030) confluence page, which sets out the agreement between the API Producer team integrating their service (you) with the Digital Onboarding Service and the digital onboarding team.

To get started, contact the Digital Onboarding Service team: england.digitalonboarding@nhs.net[992]

If your API is open access and doesn't give access to personal data, see Onboarding for simple APIs (see page 1002).

In most cases, however, API consumers must complete some onboarding process. Older APIs use the Common Assurance Process (CAP) or the Target Operating Model (TOM) process. All new APIs and existing ones that previously used a Supplier Conformance Assessment List (SCAL) have migrated to the digital onboarding service.

As of 25/04/2022, you must use the Digital Onboarding Service (DOS) process.

Warning

The Onboarding Governance Group[993] meets fortnightly and requires a week after the meeting to approve your onboarding process. Plan ahead!

992 mailto:england.digitalonboarding@nhs.net
993 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?spaceKey=SM&title=Onboarding+Governance+Group

Onboarding for simple APIs

Suppose your API is straightforward; for example, it gives read-only access to open/public data (e.g. organisation data or service data). Then you can use a lightweight version of the Digital Onboarding Service:

- No technical assurance questions
- All the API consumer needs is to do is accept our online connection agreement at https://digital.nhs.uk/developer/guides-and-documentation/online-connection-agreement#top

Our onboarding team will help you set this up.

The digital assessment questions (DAQ)

The digital assessment questions, the DAQ, is a spreadsheet that manages the digital onboarding assessment content.

View the the content governance process (see page 1021) page to learn more and preview an example DAQ.

7.15.4.4.4.4 The onboarding process

The onboarding process is designed to be as self-service as possible for API consumers. It involves them taking on a lot of the risk themselves and signing a Connection Agreement - a legal agreement that states what they are accountable for.

The onboarding team manages digital onboarding, contact the Digital Onboarding Service team, england.digitalonboarding@nhs.net[994] and the assurance governance process is owned/operated by Service Operations. For more information, contact onboardinggovernancegroup@nhs.net[995]

Service Operations run the Onboarding Governance Group, a group of SMEs from IG, security, clinical safety and solution assurance - their job is to make ensure the assurance for a given service is fit for purpose.

For an API consumer-facing overview of the onboarding process, see https://digital.nhs.uk/developer/guides-and-documentation/digital-onboarding.

Preparing your assurance for digital onboarding

Here are the steps:

1. Contact Service Ops at liveservice.onboarding@nhs.net[996] to inform them of your intent to prepare your API for consumer onboarding.
    a. If your API is listed on our API backlog[997], they might already know about it, as API-Management talk regularly with them about what's coming next.
    b. Review the digital onboarding baseline template to see if it needs any amendments
        i. In particular, in the 'Demonstrate your product meets the core conformance criteria' section, you might need to update the 'Supporting information' wording within the

994 mailto:england.digitalonboarding@nhs.net
995 mailto:onboardinggovernancegroup@nhs.net
996 mailto:liveservice.onboarding@nhs.net
997 https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?projectKey=AMR&rapidView=2826

question 'Have you updated your Privacy Notice to meet the requirements of the NHS Digital services you are using?

2. Document your API-specific "requirements" in an Excel template which you will be provided
    a. Documenting your requirements can be done within the API delivery team, with support from any relevant "business SMEs" / Information Asset Owner
    b. For example, for PDS, the Demographics team
3. Work with Solution Assurance to create a risk log for your API
    a. Contacts: Unknown User (sile1), Alan Laithwaite
    b. This will help you understand what technical assurance you need to do - you do not have to require your API Consumers to fill in a technical risk log
4. Work with Clinical Safety to create a hazard log for your API
    a. Contact: Raman Behl
5. Contact the onboarding team england.digitalonboarding@nhs.net[998] to get started with the digital onboarding service. (see page 1007)
6. Present your completed digital onboarding assurance process at OGG (see below)

Presenting at OGG

> **Warning**
>
> OGG meets fortnightly and requires a week after the meeting to approve your digital onboarding process. Plan ahead!
> For further information on the OGG see Onboarding Governance Group[999] - for access please contact onboardinggovernancegroup@nhs.net[1000]

Before you can use digital onboarding with API consumers, you need to get it approved by OGG.

You'll need to:

1. Contact onboardinggovernancegroup@nhs.net[1001]
2. Provide a few slides you can talk to, for about 10 mins, then take any questions from SMEs. There's a template which OGG will provide so you can use to structure your presentation.
3. The SMEs will then receive an ‘approvals’ sheet on which they record their decision – you’ll need to allow a week for this if any SMEs cannot attend the OGG meeting. Your presentation will be circulated to support their review.
4. The SOAP (Service on a Page) is presented to the Live Services Leadership for approval rather than SRR (Service readiness Report).

998 mailto:england.digitalonboarding@nhs.net
999 https://nhsd-confluence.digital.nhs.uk/spaces/OGG/pages/605324156/Onboarding+Governance+Group
1000 mailto:onboardinggovernancegroup@nhs.net
1001 mailto:onboardinggovernancegroup@nhs.net

5. To commence engagement with an SDT Manager (previously Pipeline Manger), an NWR would need to be raised[1002].

If unclear, email liveservices.sdt@nhs.net[1003]

Preparing your overall onboarding process

Digital onboarding sits within an overall onboarding process.

You'll want to create a Confluence page that documents the process steps you need to execute each time an API consumer onboards.

The bits you'll need to focus on are:

- How the API consumer gets in touch with you
- The process for agreeing on the API consumer has a valid use case
- How do you evaluate your digital onboarding assurance process
- How you give the API consumer access to production (see page 999) - you might need the platform team to participate in this process as only they have access to production Apigee Edge to grant access.
- For signed JWT access, you'll also need to think about how to set up public keys manually. For more information, see Authentication policies and proxy flows (see page 835).

API consumer contact

You'll need a way for API consumers to communicate with you re. onboarding. The option to use the API Management email account api.management@nhs.net[1004] is **no longer available since Jan 2024**. Please set up your own email inbox to manage communications.

Help and support

As the API producer team, **it's your responsibility to design your onboarding process**, but we're here to help. The digital onboarding team are not a managed service - their remit is to convert your API requirements into digital onboarding service - work with you to iterate your content based on user feedback and NHS design principles.

Stakeholders

Key stakeholders:

- Service Ops - liveservice.onboarding@nhs.net[1005]
- Solution Assurance -Unknown User (sile1)

1002 https://hscic365.sharepoint.com/Platforms_Infrastructure_Live_Services/Pages/PILS%20NWR%20INTRANET%20PAGE/PILS-New-Work-Request.aspx
1003 mailto:liveservices.sdt@nhs.net
1004 mailto:api.management@nhs.net
1005 mailto:liveservice.onboarding@nhs.net

7.15.4.4.4.5 Example onboarding process

Contact the digital onboarding team (Daniel Miller) to get access to the digital onboarding **test environment** to see the Personal Demographics Service - FHIR API exemplar onboarding process.

Warning

The digital onboarding team **must invite API producers to the production digital onboarding environment**. This process will ensure your team are setup with the correct roles/permissions i.e. onboarding lead/assessors/approvers.
Self-serving in the production environment creates a developer organisation account, polluting the live data. You may also cause an account conflict once set up as an internal user. (This does not apply to **API consumers** who can self-serve to the digital onboarding service and create their own orgs and products).

Internal (NHS Digital) applications consuming APIs

For internal (NHS Digital) applications consuming APIs a lower level of assurance maybe appropriate for the onboarding process.

TBC

All onboarding parties must comply with the DSPT and have Clinical DCB0129 and DCB0160's in place. NHS Trusts are not considered internal parties and still require a connection agreement (CA). There are cases for internal programmes where they don't need a CA (an alternative email confirming the legal position is uploaded in place of this, as we need to put CAs in place with ourselves). Internal programmes do not need to complete DSPT as they are covered by the parent submission of X24 (NHS England). But they do still need to complete the full assurance for an API.
It works best when:

- there is a single producer to consumer connection being established, making it simple(r) to assess
- the API consumer is a project governed by Live Services Board who will be signing off the new service going live, and associated connections
    - why this helps: the Live Service Board governs the go-live safety and readiness of an NHS Digital service, often doing so via embedded SME's from the principle areas of assurance, e.g. clinical, cyber - setting up a further SCAL or onboarding process to manage this assessment, for a single connection, carries a material risk of double governance

- the API consumer is not expected to consume further NHS Digital API's, which may already have onboarding processes to follow, it may still be possible, but it is harder to justify

How to go about it:

1. Confirm the use case (just as for external applications)
2. Complete an approved NHS Digital service transition / go-live process for the application
    a. For example, approval by Live Services Approval (In the majority of cases this would just be a Head of Service approving via email.)
    b. If it is an existing app, this might be a change control process
3. Issue production credentials

For clarity:

- Completion of the full onboarding process may be required. Programmes can seek advice from Solutions Assurance to confirm this
- There is no need to sign a Connection Agreement (because the consuming app is the same legal entity as the API provider)

Privacy Notice section

Where the API is a patient-facing interface, user consent is usually implicit (for now). However, it is still important to communicate API consumers' responsibility around user consent, for example, when accessing "PDS data" or "immunisation history".

Some suggested wording which could be incorporated into a "Privacy Notice" section could be;

It is the responsibility of each API consumer to ensure that patients are aware of how their data is being used by third-party apps. User consent could be captured as part of the terms & conditions related to each service.

7.15.4.4.4.6 Digital onboarding

Get in touch

If you want to discuss digitising the journey for your onboarding, please email the Digital Onboarding Service (DOS) Product Owner Maria Timson (maria.timson2@nhs.net[1006])
Making a change to your onboarding?
Go to How to request a change to your digital onboarding content[1007]

Table of contents

1006 http://nhs.net
1007 https://nhsd-confluence.digital.nhs.uk/x/uw_TKQ

Overview

The Digital Onboarding Service (DOS) is an online portal for developer organisations that need to use NHS Digital APIs and services. Developer organisations use it to view and answer questions in four stages:

1. Organisation details
2. Product details
3. Technical conformance requirements
4. Connection agreement

Each of these stages has a number of sections.

Features

The Digital Onboarding Service includes features valuable to both API teams and onboarding users, such as:

- An **assessment tracker** that provides a single glass pane view of all onboarding activity
- **Managing any change to your requirements** has had a huge step change in capability recently, you can do it quickly and efficiently, no matter how many developers you have already onboarding
- **Metric reporting** out of the box (in progress)
- **Version control** of documents
- **Audit history** of who made changes, when and to what
- **Multiple API** onboarding
- **Access controls** for different user roles
- **Single sign-on** between digital onboarding and the developer account
- Email notification
- Simple user interface

You can:

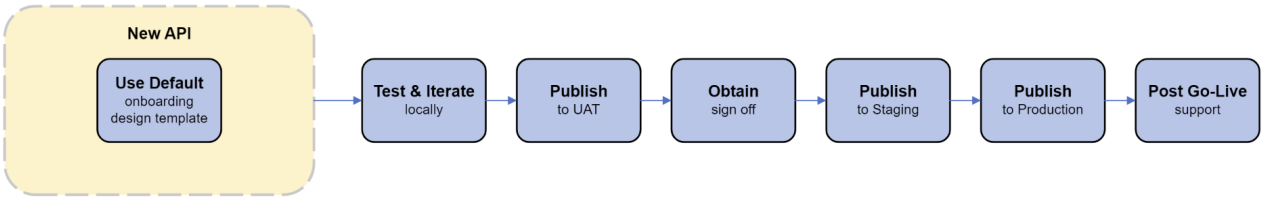
- learn more about digital onboarding by reading the documentation in the Developer hub 'Using the digital onboarding portal[1008]'.
- view the July 2021 digital onboarding roadshow[1009] .
- read our contract (see page 1030) with you.

To get started, contact the DOS product owner, Maria Timson (maria.timson2@nhs.net[1010])

Getting started with digital onboarding

This section gives a summary of our process for onboarding an API Producer team onto the digital onboarding service. At a high level, it looks like this:

1008 https://digital.nhs.uk/developer/guides-and-documentation/digital-onboarding#using-the-digital-onboarding-portal
1009 https://hscic365-my.sharepoint.com/:v:/g/personal/jasc2_hscic_gov_uk/EYmSV3-rQ6dMtva94NtO9rMB0V4Mcfz4wd1TVmzBlRzK2A
1010 http://nhs.net



Different types of API

The lifecycle can vary depending on different API factors, such as:

- If your API is open-access
- If you have a non-standard onboarding process

Delivery phases

Use the default onboarding design template

The onboarding team use a default design template (as seen in the exemplar Personal Demographics FHIR API[1011]) for new API Producer teams and access modes.

It allows the onboarding team to develop a digital onboarding experience relying on trusted design principles in a relatively short time (days).

Test & iterate locally

The onboarding team will work with nominated stakeholders within the API Producer team to iterate content in a local development environment until both parties are satisfied. Editing and publishing the onboarding content remains the onboarding team's responsibility, and we encourage API producer teams to review their requirements for duplication and complexity and question the value of the evidence they request. The onboarding team have a wealth of experience iterating content with user roles such as user researchers, interaction, service and content designers at their disposable within the team who are able to apply best practice design principles[1012] and ensure we adhere to NHS style guides[1013].

Publish to UAT

The onboarding team will publish the API Producer team's digital onboarding experience to the UAT environment to allow:

- user acceptance testing
- user training[1014]
- access to other interested stakeholders

1011 https://digital.nhs.uk/developer/guides-and-documentation/onboarding-process#digital-onboarding
1012 https://nhsd-confluence.digital.nhs.uk/display/DRM/Content+principles
1013 https://www.gov.uk/guidance/content-design/writing-for-gov-uk
1014 https://nhsd-confluence.digital.nhs.uk/x/uw_TKQ

Obtain sign-off

The onboarding team will seek sign-off from agreed stakeholders, and the Onboarding Governance Group if appropriate, that they are satisfied with the delivered onboarding experience before publishing it to Production.

Publish to staging

Once user acceptance testing is approved, and stakeholder sign-off is received, the onboarding experience is published to the staging environment as this environment mirrors Production.

Publish to production

The final step is to publish to production, where the onboarding experience will be available via the Developer hub.

Post go-live support

Once the API Producer team is live with digital onboarding, they will receive support from the onboarding team to manage their first couple of API consumers.

Working in an agile way, the onboarding team will swiftly prioritise content and process enhancements based on feedback from the API Producer team.

How we govern changes to the digital onboarding content

See our content governance page

User guides

Tell us what you think of the user guides

Complete a short survey[1015] to let us know what you think of these videos and if there's anything we can do to improve them.

Whether you are new to digital onboarding or need a refresher course, our focused video user guides have you covered. These videos will teach you how to perform certain actions or use features of the service.

We will continue to add and iterate our video content with each new feature release and user improvement, so remember to check back regularly.

1015 https://feedback.digital.nhs.uk/jfe/form/SV_ePNAnYm1hgmw2BU

ID	Description
	Navigation
1	**How to access Product onboarding** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/867202765?share=copy
2	A typical developer self-serve journey Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/870988919?share=copy
3	**How to use the assessment tracker** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/866424206?share=copy
4	**How to search** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868356701?share=copy

ID	Description
5	**How to use the breadcrumbs** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868667016?share=copy
	Users
6	**Internal user roles** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/869161430?share=copy
7	**How to add an Onboarding Lead** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/866030652?share=copy
8	**How to add assessors and approvers** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/866354924?share=copy
9	**An overview of the assessor and approver home screen** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868799894?share=copy

ID	Description
10	**How to convert an assessor or approver to an Onboarding Lead** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/867219766?share=copy
11	**How to add a user to a developer organisation** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868335765?share=copy
12	**Bulk add assessors and approvers to a developer organisation** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/908305969?share=copy
	Profile settings
13	**How to deactivate a user** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/870705294?share=copy
14	**How to enable notifications** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/866306155?share=copy

ID	Description
15	**How the Onboarding Lead email notification alerts work** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868309341?share=copy
	Products
16	**An overview of a product assessment** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868774950?share=copy
17	**How to add a product** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/867992870?share=copy
18	**How to add and remove APIs/services from a product** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/867170134?share=copy
19	**How to approve an API/service** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868699088?share=copy

ID	Description
20	**How to save information** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/893728491?share=copy
21	**How to edit and update a product name** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/893745390?share=copy
22	**How to delete a product** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/893738958?share=copy
	Organisations
23	**How to add a developer organisation** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868003476?share=copy
24	**How to manage a developer organisation** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/868685336?share=copy

ID	Description
25	**How to edit and update an organisation name** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/893747421?share=copy
26	**How to delete an organisation** Sorry, the widget is not supported in this export. But you can reach it using the following URL: https://vimeo.com/893743078?share=copy

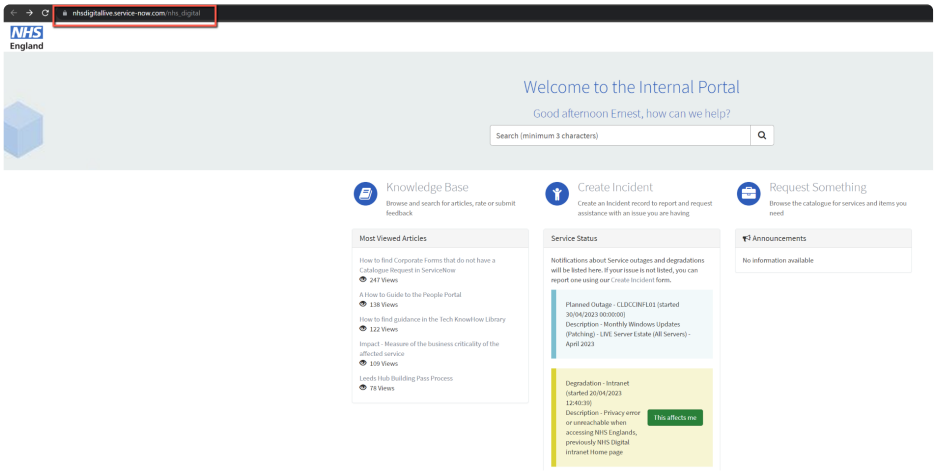
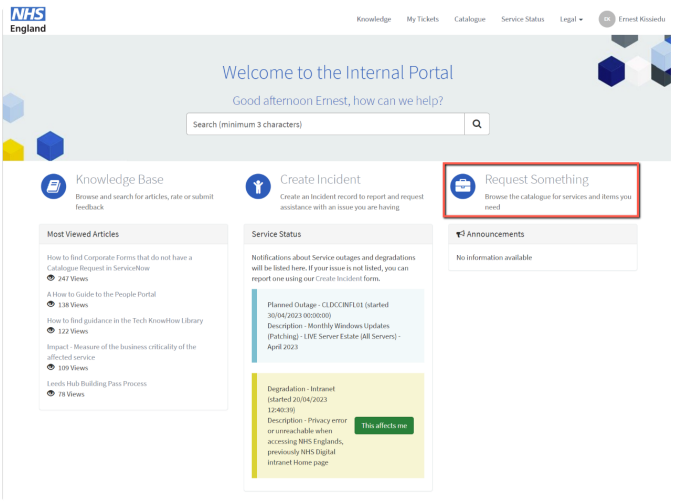
How to request a change to the digital onboarding content

This process is for content amendments only. You can raise feature requests and suggestions in the ideas portal.[1016]

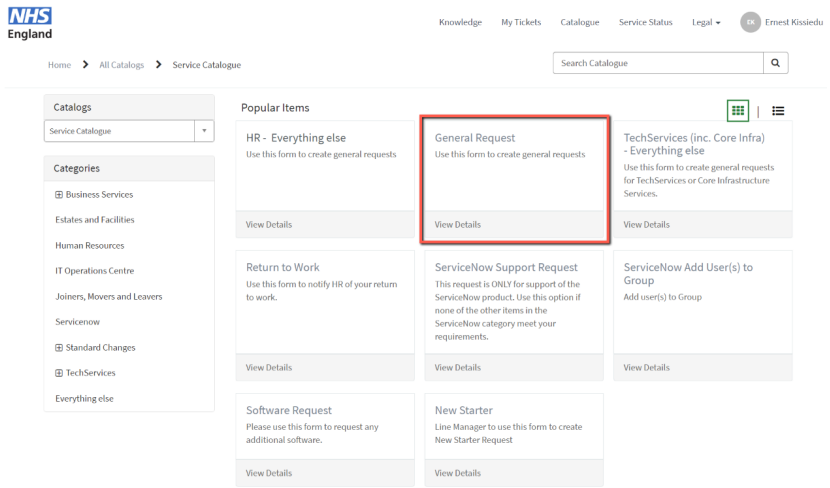
To request a content change to the digital onboarding service for API-specific sections, such as your technical conformance section, you must submit a ticket in ServiceNow.
Depending on a user's ServiceNow permissions, there are two ways to submit a ticket to the Digital Onboarding Content team:
ServiceNow **Internal Portal** (submit a new task) - available to most users.
ServiceNow **Customer Service Agent** (submit a new case) - available to a select group of users.
Also, see what is the process to request changes to the template shared sections and questions (see page 1022)

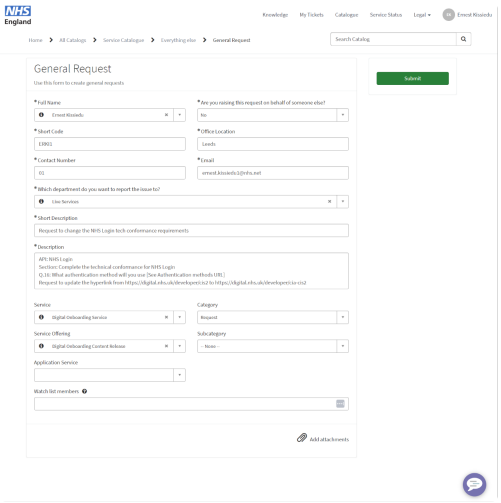
1016 https://nhs-digital-api-management.featureupvote.com/?order=popular&filter=all&tag=make-onboarding-easier#controls

Internal Portal

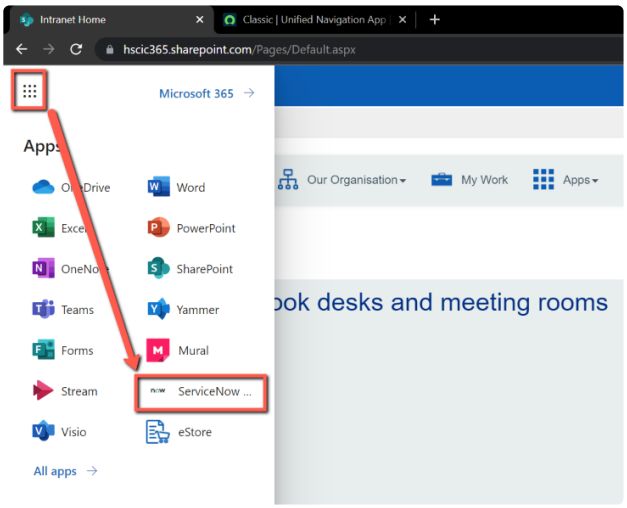
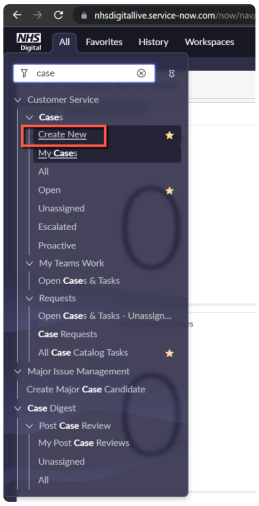
Step	Title	Summary
1.	Access the Internal Portal URL[1017]	
2.	Select 'Request Something'	

1017 https://nhsdigitallive.service-now.com/nhs_digital

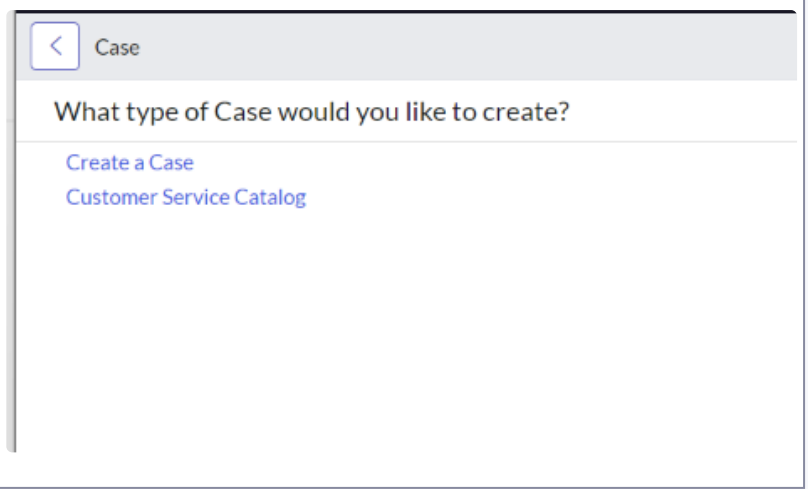
Step	Title	Summary
3.	Select 'General Request'	

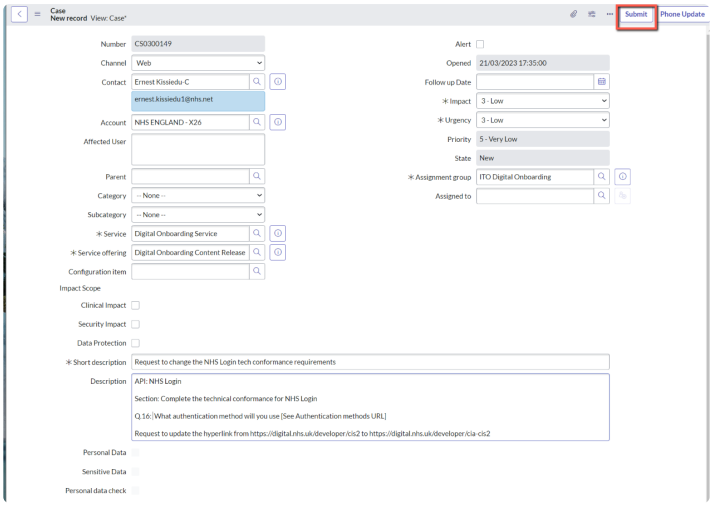
Step	Title	Summary
4.	Complete the Request form	 **Note**: The fields marked with an Asterix * are mandatory. The field 'Which department do you want to report the issue to?' will automatically populate once you select the service and service offering field. The following selections must be used to ensure your request is routed correctly to the digital onboarding team: • Service: *Digital Onboarding Service* • Service offering: *Digital Onboarding Content Release* **Guidance**: • Short description: *Keep this short and succinct* • Description: *Provide as much necessary information to allow the onboarding team to assess and action your request the first time.* Check the entered information before submitting the request.
5.	Request status	The digital onboarding team will aim to provide an update within 2 business working days.

Customer Service Agent

Step	Title	Summary
1.	Login to ServiceNow[1018]	
2.	Select 'New Case'	

1018 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/catalog_home.do

Step	Title	Summary
3.	Select 'Create Case'	

Step	Title	Summary
4.	Complete the new Case form	 **Note**: The fields marked with an Asterix * are mandatory. The following selections must be used to ensure your request is routed correctly to the digital onboarding team: • Service: *Digital Onboarding Service* • Service offering: *Digital Onboarding Content Release* • Assignment group: *ITO Digital Onboarding* **Guidance**: • Impact: [*Assign accordingly]* • Urgency: [*Assign accordingly]* • Short description: *Keep this short and succinct* • Description: *Provide as much necessary information to allow the onboarding team to assess and action your request the first time.* Check the entered information before submitting the request.
5.	Request status	The digital onboarding team will aim to provide an update within 2 business working days.

Content governance process

When do I need this process?

If you need to change the content or questions in your onboarding journey.

Those questions may be:

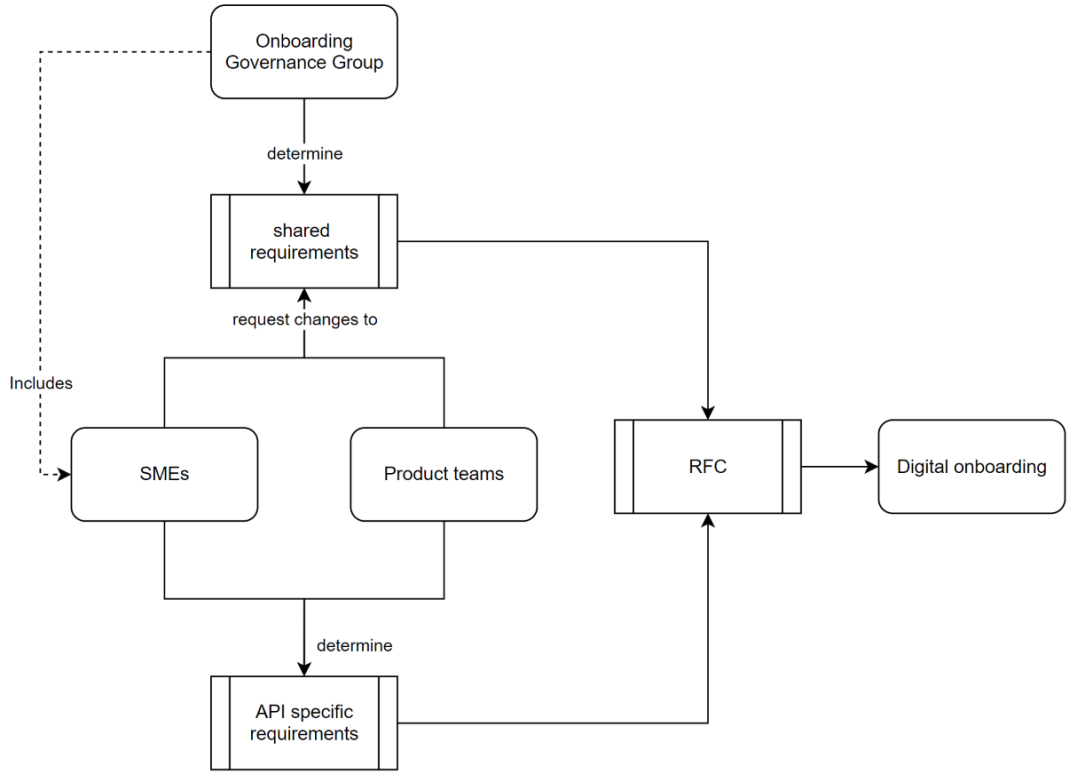
- already live and in use, and you want to get them
    - re-answered
    - removed
    - changed without being re-answered
- not fully live yet

What is the process?

Content change requests to:

- Common shared sections[1019] (such as, Provide details about your Product) must be raised by the API/service team directly with the onboarding governance group (OGG) to obtain their approval. You can contact OGG by email: onboardinggovernancegroup@nhs.net[1020]
    - Once approved, OGG shall raise the change request within ServiceNow[1021], assigned to the digital onboarding team.
- API/service specific sections (such as, Technical Conformance) do not require OGG approval but you should consult your SMEs.
    - In agreement with SMEs, a member of the API team must raise a change request within ServiceNow[1022], assigned to the digital onboarding team.

1019 https://nhsd-confluence.digital.nhs.uk/display/APM/Content+governance+process?preview=%2F576128443%2F618168922%2FDOS+sections.png#Contentgovernanceprocess-Howdoesitwork?
1020 mailto:onboardinggovernancegroup@nhs.net
1021 https://nhsd-confluence.digital.nhs.uk/x/uw_TKQ
1022 https://nhsd-confluence.digital.nhs.uk/x/uw_TKQ



How does it work?

We use a spreadsheet (see attachment below) called digital assessment questions, (DAQ), to manage the digital onboarding content.

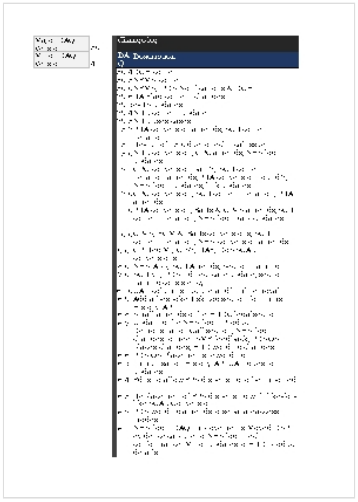
A DAQ (see DAQ attachment below) consists of many tabs which enable the display of common shared sections and API/service-specific sections (see Common vs API/service-specific sections attachment below).

Editing and publishing the onboarding content (see page 1008) remains the onboarding team's responsibility.

We use ServiceNow[1023] for the DAQ change management process.

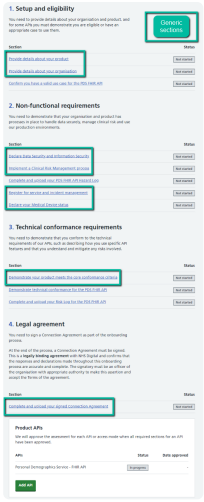
You can view historical and planned digital onboarding releases, including the Jira tickets for content changes within the APIM Release schedule. (see page 1390)

1023 https://nhsd-confluence.digital.nhs.uk/x/uw_TKQ



DAQ >>

Common vs API/service-specific sections >>



Why do we have one?

To make it easier for stakeholders and contributors to see precisely the notable changes in each content version release.

Changing content for your onboarding

Get in contact with us

Learn How to request a change to the digital onboarding content[1024]

1024 https://nhsd-confluence.digital.nhs.uk/display/APM/Training+zone#Trainingzone-Howtorequestachangetothedigitalonboardingcontent

We can help you

- iterate before you go live
- handle uplifts to requirements that are live (see page 1022),
    - including requiring users that are in-progress to re-answer the requirement, if it has changed sufficiently
- manage versions with end users

Digital Onboarding has been built from the ground up with iteration in mind.

Changing existing requirements or content

See our Content governance process (see page 1021)

Handling the requirements from Assurance areas

This may include

- Technical conformance
- Connecting Systems Risk Logs (Solution Assurance)
- Clinical Hazard Logs (Clinical)
- Connection Agreements (Legal)

Some of the requirements will still be embedded within versioned documents, e.g. MS Excel.

Changing versions of documents

Digital Onboarding has patterns for managing versioning, often delivering the documents via the API spec pages themselves for a quick, open access experience.

If you have a new version of a document, it can be quickly updated online, and developers can be required to re-upload a new version if an assurance area wishes them to do so, although this is not mandatory.

Contact us (above) if you want to discuss the design of a specific pattern for your API, or organise a new document version.

If you want to see an example, check out Immunisation History FHIR API onboarding support information[1025]

Changing your API spec onboarding section

Digital Onboarding has standard content for updating API specs onboarding section to reflect the new digital onboarding process.

If you want to see an example, check out the Immunisation History FHIR API onboarding section.[1026]

App enabled into production

1025 https://digital.nhs.uk/developer/api-catalogue/immunisation-history-fhir/onboarding-support-information
1026 https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#api-description__onboarding

Overview

The final step of onboarding a consumer is enabling their Apigee App in production.

The onboarding lead must coordinate the following process :

1. Instruct the consumer to create a new App[1027] in the Developer Hub[1028].
2. Onboarding lead requests that the App be enabled through the **enable prod app** request workflow in Slack.

APIM endeavours to complete enable prod app requests within 1 working day, however when resources are limited this may take up to 2 days.

Consumer creates a new app

The consumer will first need to create a new App[1029] in the Developer Hub[1030], subscribe to the correct API production product, and click "***request***".

Note: this does not automatically enable the App or notify the platform team they have requested it (t*his is different from PTL environments, where clicking "request" will automatically enable the App)*. The onboarding lead will still need to request the app to be enabled via workflow in Slack

Request the App to be enabled

Once the consumer has completed step 1, the onboarding lead needs to request that the App be enabled.

Only the Onboarding lead (OBL) for the API should be raising a request for App enablement in PROD. This is to ensure that the API team have confirmed the APP and the Consumer organisation.

Use this link to submit the request via Service NOW:

https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_cat_item&sys_id=fd6513da1be05210892d4046b04bcb14&sysparm_category=7496f0c81b1c9610772fa756b04bcbf7&catalog_id=-1

Please note: The Slack workflows previously used for raising this request has been migrated to Service NOW

Enable App form fields

1027 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication#step-1-create-an-application:~:text=Testing%20APIs.-,Step%201%3A%20create%20an%20application,-To%20use%20this

1028 https://digital.nhs.uk/developer

1029 https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation/application-restricted-restful-apis-signed-jwt-authentication#step-1-create-an-application:~:text=Testing%20APIs.-,Step%201%3A%20create%20an%20application,-To%20use%20this

1030 https://digital.nhs.uk/developer

Field Name	Description and examples
Requester (your) email address	someone@nhs.net
App name	MedicalCare
App ID	3ebd88d7-8cdb-79e1-96eb-14507fefffb4
API Consumer Organisation	MedicalCare Ltd
App approved by	Live Services
API/APIs products requested	PDS FHIR Application Restricted
INT test app ID	Hello World Int JWT
ASID (if spine API (optional)	Accredited System Identifiers (ASID) For example 200000000985
ODS Code connected to ASID (optional)	A8Q7N You can search for your ODS code on the ODS Portal https://odsportal.digital.nhs.uk/Organisation/Search
ODS Code of third party Supplier (if different to ASID ODS) (optional)	
Additional custom attributes (optional) (must be comma separated and include the exact key and value for each)	
JWKS resource URL (if self-hosting)(optional)	
KID for JWKS (if APIM to host)(optional)	prod-1
Please paste the public key value here: (if APIM to host)(optional)	

Field Name	Description and examples
Anything else?(optional)	Lorem ipsum dolor sit amet

What happens next

- Once the form is submitted via Service Now,
- Verify if the request was raised by the appropriate OBL
- Check the fields and APP ID etc
- Work through a checklist which is then peer reviewed.
- Endeavour to complete 'enable prod app' requests within 1 working day, however, when resources are limited this may take up to 2 days.
- Will report back to you once completed or if any further information is needed.

Setting your onboarding status

Onboarding status refers to the users' ability to select an API within digital onboarding.

The following table describes the different onboarding statues.

Onboarding status explanation

Status	Suggested uses	Visible to developers	Can be selected by developers	Visible to internal users	Can be selected by internal users
Available	APIs using digital onboarding for all new suppliers	✓	✓	✓	✓
Limited	APIs piloting digital onboarding	✓		✓	✓
Discontinued	APIs that have a newer recommended version and/or new assurance			✓	✓
Retired	Retired APIs				

We recommend setting the onboarding status to Available to allow API consumers to self-serve and enjoy a frictionless experience.

If there is a requirement to restrict access to an API, such as a closed pilot, we can set the onboarding status to Limited.

Updating the onboarding status can be completed at any time.

Common challenges in building an effective onboarding process

The purpose of this page is to capture the challenges and experience Digital Onboarding team have observed when working with peer teams in NHSD
Scope: all onboarding processes, this is not digital onboarding specific

Key themes

- Using onboarding as a *catch-all* for things we want to check, or control as a service, occasionally for purely internal reasons
    - we often do this even if it is actually a risk that is not directly applicable to the use of the service being onboarded, e.g. we sometimes assure the clinical user interface, where the NHSD service is actually an API, so we should probably stick to our lane and allow the developer to sort out interface with their users
- Missing the chance to use technology to control a risk, instead relying on onboarding to manage it
- A once-and-done approach to risk management
    - "if we check everything, then there can be no come back"
    - developer/supplier use of NHSD tech/data actually may change over time, but having onboarded, NHSD often consider it "done"
- It is hard for users to focus on what really matters:
    - Risk stratification is missing / poorly understood
        - many risks and requirements are given equivalent weight, e.g. they may all exist in one long checklist, whereas NHSD actually carries much greater risk on a specific few risks
        - the biggest risks can be "lost in the noise", because assurance is applied equally to a large list, meaning:
            - lower order risks may miss the opportunity of self-certification (e.g. think about how Terms and Conditions are used elsewhere) - reducing the legal and assurance burden on NHSD
            - the legal burden may already be on the developer/supplier, but we decide to "just check anyway"
- Risk / requirements - understanding the WHY
    - boundaries between risk/requirements can become unclear
    - silos form when related risks/requirements are assessed by different people
        - loss of efficiency
        - loss of risk control
        - loss of a common approach to assessment

        - developer senses they are being asked similar things in different ways
            - extending the process
            - increasing cost on both sides
        - developer struggles to understand what we really want
            - assurance demos do not articulate what good looks like
            - demos have to go back and try again
            - similar evidence may actually answer multiple assurance requirements, a missed opportunity
- Lack of context
    - different terms for the same thing, e,g, NHSD service spec does not align its language with SCAL requirements / risks
        - to-and-froing:-
            - developer stops developing, asks questions, waits,
            - and increases the support burden on the dev team
            - and increases lead times (sometimes dramatically)
    - technical capabilities/limitations drives NHSDs risk, so requirements are formed to manage these, however the link or explanation of this is missing
- Version control
    - >5 people working on both sides, with excel docs, rapidly breaks down on version control
    - HUGE coordination burden falls back on a few individuals
        - mistakes are easily made
- Duplicate questions
    - limited use of a minimum data set
    - duplicates arising especially where a developer is already onboarding to a peer service
        - or even the same service, just with a different security model
- Challenging inefficient processes
    - Unsure why a process exists other than "because that's the way we've always done it"

Our contract with you

do you think we should change this, or improve it?

get in touch with Gnanathanappan Ramachandran

In respect to digital onboarding we will use this page to define "our contract with you" (this is a work in progress)...

This is our agreement with an API Producer team integrating their service with the Digital Onboarding Service

It will contain things like the fact we will:

- train you
- provide you an environment to play and test your process
- provide warranty support
- focus on iterative improvement, rather than "eating the whole elephant" up front
- give you a set of NHS content style guides[1031] and digital onboardings own[1032] core service design principles to re-assess your own onboarding journey
- show you the learned-from challenges many NHSD services have experienced (see page 1029), up front
- provide templates (**tbd**) to help move your onboarding to a more standardised model, reducing burdens internally and externally
- generate process improvement over time

And we expect you to be:

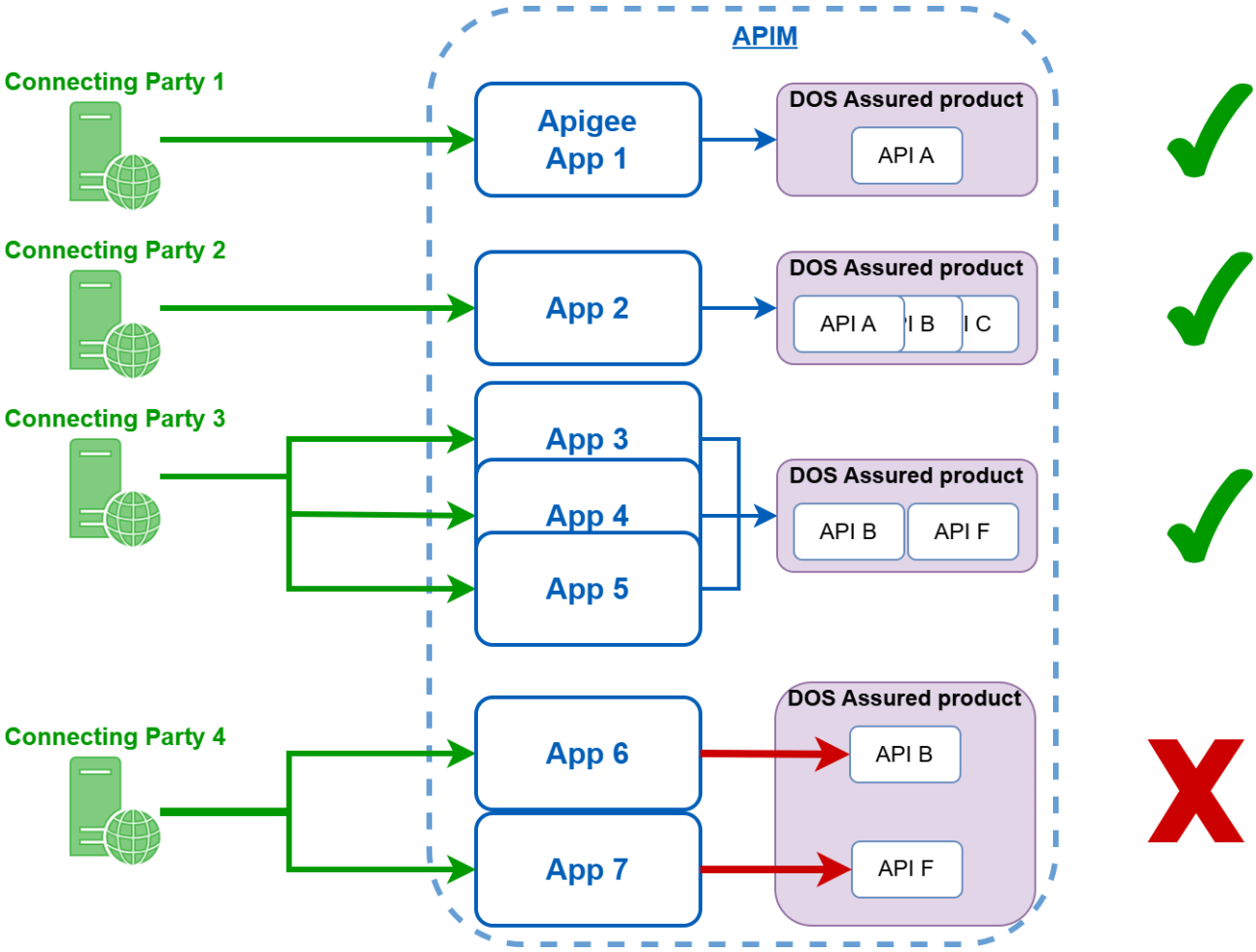
- an engaged partner, giving us an appropriate amount of time on a pre-agreed basis with named contacts
- able to iterate your onboarding for the better, when you see an opportunity to do so i.e. we'll work with you to make it less challenging to improve it frequently and in smaller ways
- ready to adapt existing assurance steps, so that they flow in a manner that is logical to the end user, as opposed to how we internally assess and manage the process (not always logical to end users!)
- able to guarantee transparency and access to current onboarding practices
- open to moving your onboarding journey over to an NHS Digital structure, as opposed to a program structure
- able to agree up front that we would move along to digitising the next point of assurance, where a change has been limited to a pure digitisation of a question - we found this has really helped teams go faster and find it easier to convert to digitisation

7.15.4.4.4.7 Onboarding and configuring Apigee applications

Standard APIs

Once a Digital Onboarding Service (DOS) Product is assured, the API Consumer creates and manages (in Environment Access) one or more Apigee Apps to manage their own deployment configuration and optimise security.

1031 https://service-manual.nhs.uk/content
1032 https://nhsd-confluence.digital.nhs.uk/display/DRM/Content+principles



"**DOS Assured Products**" - Product is a very overused use word the choice here is critical as it:

- "DOS Assured Products" is the definition of "Product" that Digital Onboarding Service (DOS), Onboarding Governance Group (OGG), ServiceNow, Partner Gateway and the Connection Agreement use
- This is **nothing** directly to do with an "Apigee Product" - to see how they are related review:
  - Access modes - compliance - security and simplifying running your API (see page 299)

You will also need to consider:

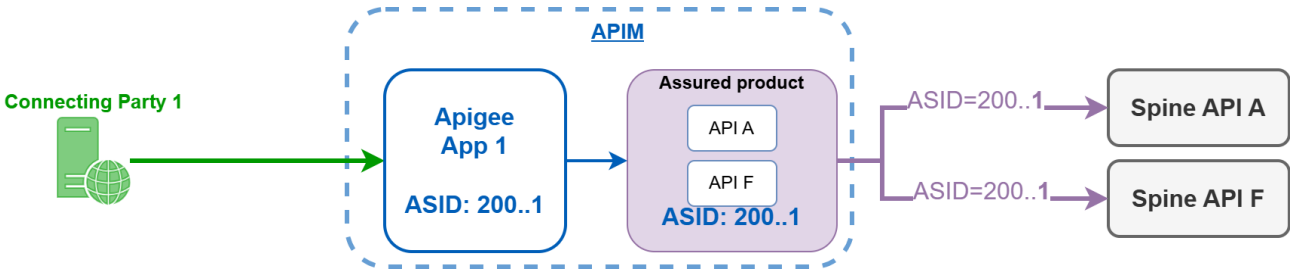
- ProductID (see page 308) (almost available) - identifies the assured product. This uniqueness will help with inter-team communication
- Access Modes:
  - The API **Consumer** might want one App per mode, or might want to have all access modes on the same app - this **must** be their choice
  - This is where the "Apigee Product" comes in, not under the "DOS Assured product"
- End User Organisations (EUOs)
- Separating duties like this means that no onboarding team has to understand the API Consumer connection topologies (see page 1149):

- This will enable API Consumer implementation teams to be fully autonomous (e.g. carrying out deployments at weekends)
- Although API Producers might want to understand from a governance perspective

Spine and the single ASID model

When one or more of the APIs requested is part of The Spine, then the single ASID model will be used:

- Issuing one ASID for each assured product
  - All API producers add their permissions (message sets / interactions) to that same ASID
  - The requirement for multiple End User Organisations to each have an ASID is replaces with a header (**NHSD-End-User-Organisation-ODS**) that accepts an ODS code
- The API Consumers creates one or more APIM Apps
- Once assured, each app is enabled and has the same ASID attached
- The process **should be tested** in INT, by issuing an INT ASID to the API Consumer and setting the configuration as it should be in Production
  - A key assurance cycle interaction proposal: Lifecycle - Organisation[1033]
  - Generic ASID available in DEV and INT for APIM Consumers: Spine based APIs (see page 410)
  - Proposed stages for managing End User Organisations (EUOs): Automation of onboarding End User Organisations[1034]



This model does not work under the following conditions:

- Where the e-RS Spine APIs are on the one ASID per EUO site model - however there is current work underway to change this
- Where both HSCN (Health and Social Care Network - older private access) APIs and APIM APIs

1033 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/678892833/Lifecycle+-+Organisation#LifecycleOrganisation-Progression

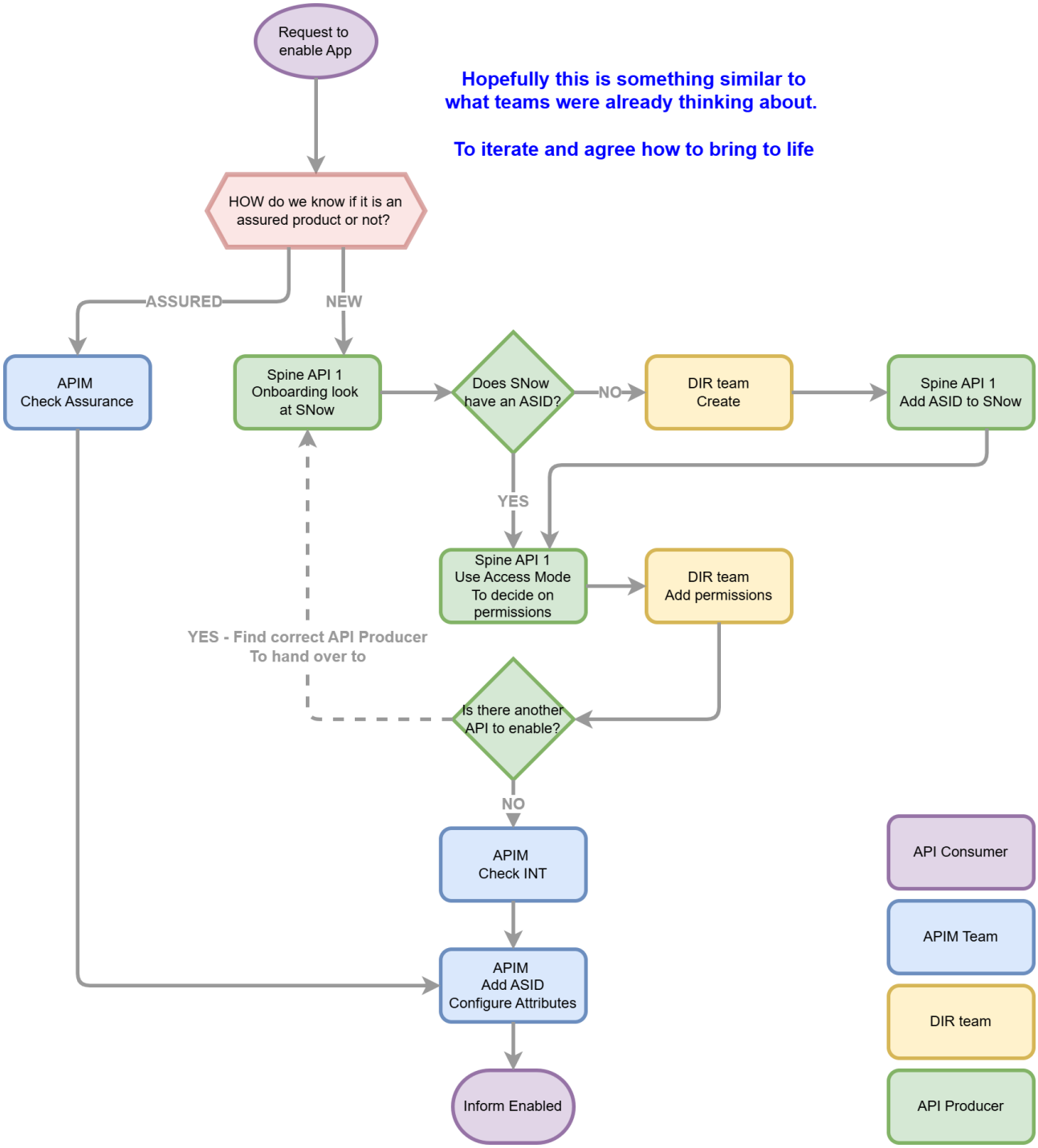
1034 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/1038783776/Automation+of+onboarding+End+User+Organisations

- The intent is that only a single ASID is used for the APIM APIs - but the HSCN based APIs will require their existing per site model to continue

Proposed business process

Service Now (SNow) will be used as the place to link the various teams together.

Overtime this could be automated with the introductions of Connecting Party Manager (CPM), changes to Environment access and links to the Digital Onboarding Service (DOS):



This diagram does not include details on:

- Access Modes
- End User Organisations (EUOs)

Background and reasoning

If you want to understand why these models came about read wider background pages:

- API Consumer connection topologies (see page 1149)
  - Public version: Configuring application connections by software topology[1035]
- Access modes - compliance - security and simplifying running your API (see page 299)
  - Managing different security requirements for API Consumers (see page 813)
- Compliance touch points[1036]
- Spine based APIs (see page 408)
- Spine based APIs and single ASID model (see page 444)
- HTTP headers for your API (see page 395)
- Products and Product IDs (see page 308)
- Lifecycle - Organisation[1037]
- D060 - Authorisation for End User Organisations[1038]
  - Automation of onboarding End User Organisations[1039]
- e-RS documents:
  - app-restricted-api-access[1040]
  - integration-process/stage-1.4---select-your-authentication-level[1041]
  - Single ASID Model / Authorisation for EUOs[1042]

7.15.4.5 Engineering tasks

Enginering tasks for proxy generator

If using proxy generator see Engineering tasks for proxy generator (see page 1115)

The following articles describe features we recommend implementing in your API beta service ahead of your first production deployment.

1035 https://digital.nhs.uk/developer/guides-and-documentation/configuring-application-connections-by-software-topology
1036 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182976643/Compliance+touch+points
1037 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/678892833/Lifecycle+-+Organisation
1038 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/540181015/D060+-+Authorisation+for+End+User+Organisations
1039 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/1038783776/Automation+of+onboarding+End+User+Organisations
1040 https://digital.nhs.uk/services/e-referral-service/api/app-restricted-api-access
1041 https://digital.nhs.uk/services/e-referral-service/api/integration-process/stage-1.4---select-your-authentication-level
1042 https://nhsd-confluence.digital.nhs.uk/spaces/ES/pages/1035403337/Single+ASID+Model+Authorisation+for+EUOs

- Rate limiting (see page 1037)
- Monitoring your API (see page 1064)
- Using nhsd-git-secrets tool pre-commit hook on your repo (see page 1102)
- Supporting API consumers with tracing unique identifiers (see page 1104)
- Update locking with ETag (see page 1109)
- Caching for your API (see page 1112)
- Feature toggles (see page 1113)

7.15.4.5.1 Rate limiting

Link to:

Rate limiting shared flow (see page 1039)

If using proxy generator see: Rate limiting with proxy generator (see page 1117)

7.15.4.5.1.1 Overview

API rate limiting is a mechanism that restricts the number of requests an API can handle within a specific time frame.

Why is API Rate Limiting Important?

- **Prevents Abuse:** Limits the number of requests a single user or application can make, preventing malicious actors from overwhelming the API.
- **Protects Resources:** Conserves system resources by ensuring that the API can handle a reasonable workload.
- **Enforces Fair Usage:** Ensures that all users have fair access to the API and prevents any single user from monopolising resources.
- **Mitigates DDoS Attacks:** Helps defend against distributed denial-of-service attacks, which aim to overwhelm an API with excessive traffic.

Potential Problems Caused by Improper Rate Limiting

- **User Frustration:** If the rate limit is too restrictive, it can lead to user frustration and a negative experience.
- **Business Impact:** Overly restrictive rate limits can negatively impact business operations, especially for APIs that support critical services.
- **Security Risks:** Insufficient rate limiting can leave the API vulnerable to abuse and attacks.
- **Inaccurate Performance Metrics:** Incorrectly configured rate limiting can skew performance metrics, making it difficult to assess the API's true capacity.

Effective rate limiting involves carefully balancing the need to protect the API with the need to provide a positive user experience.

7.15.4.5.1.2 Rate Limiting Shared Flow in Apigee

Rate limiting shared flows in Apigee provide a powerful mechanism for controlling the rate at which clients can access your APIs. By placing these flows in your API proxies, you can enforce rate limits at the API level, ensuring fair usage and preventing abuse.

There are two primary types of rate limiting: **per-application** and **per-proxy**. The recommended way to implement rate limiting in your proxy is to call the Rate limiting shared flow (see page 1039) (with proxy generator (see page 1117)), which allows you to control the rate limiting from the manifest file.

Per-Application Rate Limiting

Scope: Limits the number of requests that a specific application or client can make within a given time frame. A default limit is applied separately to each app. This limit can be overridden manually on a per-app basis (via the “Request rate limiting change” workflow in #platforms-apim-producer-support channel)

Use Cases: Prevents any single app from overwhelming the backend and ensures equal resource distribution among multiple apps, avoiding the monopolization of resources by any one[{{type}} Annotation] app. Overriding the default is suitable for scenarios where you want to differentiate treatment for different applications.

Per-Proxy Rate Limiting

Scope: Limits the total number of requests that can be handled by an API proxy within a given time frame.

Use Cases: Useful for protecting the overall API infrastructure from being overwhelmed by excessive traffic.

Key Differences:

Feature	Per-Application	Per-Proxy
Scope	Individual applications	Entire API proxy
Implementation	API keys, authentication	API gateway configuration
Use Cases	Differentiated treatment for different applications	Protecting the overall API infrastructure

7.15.4.5.1.3 Default rate limits

You, as the API Producer, control your rate limits - we set a default on the platform, just to keep usage low and safe:

- The default is applied per minute, not per individual second - so at the default limit of 5tps, you can perform up to 300 transactions in any given (rolling) minute
- The default is applied per application, per API
- Is applied to our OAuth 2.0 authorisation service
- We expect you to work with each API consumer to agree what their rate limits should be
- **NEVER** ask an API Consumer to create more Apigee Apps to increase their rate limit:
    - This means they need to manage more credentials and connections - making their integration with us harder
    - Rate limiting is **not** about reducing your consumers rate limit until they get one or two 429's, but about setting a limit that only ever triggers in **unusual** situations
    - When looking at production usage for any consumer you will see, with busier consumers, a more "consistent" level or requests - because the peaks and troughs even out. Don't be too granular with your rate limits as the peaks and troughs don't smooth out

If you you want to read more about the actual volumes you anticipate see: Performance testing (see page 756)

7.15.4.5.1.4 External developer documentation

As mentioned above, we have standard guidance for external developers on how we implement rate limits at https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits.

Within your API specification you should link through to this documentation. Specifically, do this by including details of rate limiting HTTP responses from your endpoints (HTTP status code 429). See The PDS FHIR API spec for an example of how to do this:

- Go to https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-fhir, hit ctrl-F and search for "429" to see an example
- To see how this looks in an OAS source file, go to https://github.com/NHSDigital/personal-demographics-service-api/blob/master/specification/personal-demographics.yaml, hit ctrl-F and search for "429"

7.15.4.5.1.5 Rate limiting shared flow

Content on this page

The word "product" is easily confused with Digital Onboarding Product. In this context it is the "Apigee product" (equivalent to an "access mode").

As this page is pre-proxygen the yml use of "product" is unlikely to now change.

Introduction

This page explains how to add rate limiting to your API using the ***ApplyRateLimiting*** shared flow, and to configure the rate limits using the manifest file.

At the core of the ***ApplyRateLimiting*** shared flow is a stack of four Apigee Policies.

- Per-App *Quota* policy
- Per-App *SpikeArrest* policy
- Per-Proxy *Quota* policy
- Per-Proxy *SpikeArrest* policy

The Apigee documentation on these policies (Quota[1043] and SpikeArrest[1044]) is pretty comprehensive if you want a detailed explanation. But the condensed version is that each of these policies keeps a count of requests over a specified interval:

- The *Quota* policy is designed to apply rate limiting over "long" periods of time. The shortest time interval is per-minute.
- The *SpikeArrest* policy is designed to apply rate limiting over "short" periods of time, at second or sub-second intervals.

Each time the proxy flow executes one of these policies, a counter is incremented with an Identifier (i.e. a label).

In the case of the Per-App policies, this Identifier is the `appId` . For the Per-Proxy policies this value is not specified, causing Apigee to update the counter labelled `_default.`

When the policies execute, the counts are compared to threshold values. If the count is above threshold the Shared Flow raises a fault, and returns a 429 response to the API consumer explaining which of the policies has triggered the rate limiting (see page 1051).

Warning! Avoid this common mistake

Once you've read and understood this page, you will want to pick values to enforce a rate limit for your own API. Please note the following before proceeding.

To enforce a rate limit of `N` requests per second, you must supply the value to the SpikeArrest policy in the form `Mpm` where `M=60*N` .
Setting `Nps` actually enforces the rate limit over fractions of a second, *which is almost certainly not what you want*.
Please read this section[1045] of the apigee docs for a full explanation.

It is for this reason that all examples on this page supply values to the spike arrest policy as `Mpm.`

How to do this

Rate limiting isn't complex to setup, but working through the permutations can be. This page sets out a number of things that need to be configured:

- [ ] Set your API (proxy) total and default app limits - using manifest policy variables (see page 1041)

1043 https://docs.apigee.com/api-platform/reference/policies/quota-policy
1044 https://docs.apigee.com/api-platform/reference/policies/spike-arrest-policy
1045 https://docs.apigee.com/api-platform/reference/policies/spike-arrest-policy#expandable-1

- [ ] Allow per environment variations of the limits - using the manifest as well
- [ ] Attach the ***ApplyRateLimiting*** shared flow to your proxy - see Calling the shared flow (see page 1050)
- [ ] *<Deploy your proxy>*
- [ ] Allow specific API Consumer applications to override the limits (see page 1045)
    - This is done via an Apigee Custom Attribute, API Producer Support use a microservice to generate the valid JSON and
    - Increase the auth (identity) service rate limits (see page 1044):
        - This is the **only** way to increase the auth service limits
        - If you support user-restricted access you will need to (generally) set this one to one with the app rate limit
        - **Even if you do not use** the shared flow, and you want to increase the auth service limits, you do this via the custom attributes

API proxy total and default app limits in policy variables

The rate limiting policy variables are controlled putting a `ratelimiting` attribute on your proxy's Apigee product.

You should set this in your manifest/manifest_template (see page 896) file, under the Apigee product additional attributes.

manifest.yml OR manifest_template.yml

```
---
# values are given in the 'apigee/environments/products' section of the manifest
file...
apigee:
  environments:
    - name: internal-dev
      products:
        - name: my-product-name # e.g. canary-api-internal-dev
          attributes:
            - name: ratelimiting
              value:
                my-proxy-internal-dev: # this controls the per-proxy rate limiting
policies, AND MUST MATCH YOUR PROXY NAME
                  quota:
                    enabled: true # boolean, set to false to disable the policy
entirely.
                    interval: 1 # number of timeunits to window over, non-zero
positive int
```

```
                    limit: 60000 # non-zero positive int
                    timeunit: minute # minute/hour
                  spikeArrest:
                    enabled: true # boolean, set to false to disable the policy
entirely
                    ratelimit: 60000pm # must end in pm/ps
                app: # this controls the default per-app rate limiting policies
                  quota:
                    enabled: true
                    interval: 1 # number of timeunits to window over, non-zero
positive int
                    limit: 600 # non-zero positive int
                    timeunit: minute
                  spikeArrest:
                    ratelimit: 600pm # must end in pm/ps
                    enabled: false
```

Normally product attribute values are limited to strings but the `ratelimiting` attribute has special handling in the common deployment pipelines, converting the YAML dict here to a JSON string on the product.
This allows for easier templating of different values per environment.
The pipeline will refuse to deploy a product with a `ratelimiting` attribute that does not conform to the above schema.

NOTES

- Because the the default rate-limiting values are defined on the product, changing them will require a redeploy of your proxy
- The above settings have **no impact** on the auth (identity) service

Setting different Rate Limits per-environment

It may be the case that you want your proxy to have different rate limits in different environments, e.g. a higher rate limit for ease of load testing in `int` or a lower rate limit to protect an underpowered sandbox container.

To set different values for different environments, we recommend using jinja2 templating in your `manifest_template.yml`.

You can template values anywhere, including the `ratelimiting` product attribute.

A minimal example that uses templates to set different per-environment rate limits is shown below.

```
# Templating vars
APIGEE_ENVIRONMENTS:
  - name: internal-dev
    # no custom values set so will use the | default jinja2 filter.
  - name: ref
    ratelimit: '600pm'
    quota: '600'
  - name: int
    ratelimit: '1200pm'
    quota: '1200'
  - name: prod
    ratelimit: '6000pm'
    quota: '6000'
---
apigee:
  environments:
  {% for env in APIGEE_ENVIRONMENTS %}
    products:
      - name: canary-api-{{ env.name }}
        attributes:
          - name: ratelimiting
            value:
              canary-api-{{ env.name }}:
                quota:
                  limit: {{ env.quota | default(300)}}
                  interval: 1
                  timeunit: minute
                  enabled: true
                spikeArrest:
                  ratelimit: {{ env.ratelimit | default('300pm') }}
                  enabled: true
```

The jinja2 for loop is evaluated at build time, setting different values for the limits in different environments.

The above code block is just to illustrate the per-environment templating.
This is **not a recommendation** about what values to set!

Rate Limiting and Multiple Apigee Products

Some APIs define multiple "**access modes**" to access the same API, which are "Apigee products" at the technical level.

Typically this is done when there are multiple methods of authenticating, e.g. enabling both Application-restricted and User-restricted access.

> For simplicity we recommend you set the same `ratelimiting` variables on both Apigee products

However, if you do decide it is valid to set them differently, it is important to know how this would affect the proxy behaviour.

- The proxy rate limit is the `max` of any Apigee product limits set
- Each Apigee product will be limited to the rate limit specified for that Apigee product

Example using 'Apigee Product A' and 'Apigee Product B'

- There is a counter on the proxy-level for an API that increments by per request.
- Apigee Product A has a "per-proxy" rate limit of 50 tps. Apigee Product B has a "per-proxy" rate limit of 100 tps.
- If Apigee Product A maxes out at 50 then Apigee Product B can still work up to another 50tps.
- If Apigee Product B reaches 50 tps then Apigee Product A will be throttled.

Rate Limits for the Auth Service (identity-service)

If your API uses OAuth (tokens from the auth service) for authentication - then your calls to the Auth service are also rate limited. This is based on **only two** factors:

- First limited by the per-application rate limits - see custom rate limiting (see page 1045)
- Second an overall identity-service limit - which acts on the total volume of requests from all apps

The identity-service uses this shared flow to enforce rate limits, however this mechanism of providing values is slightly different because it doesn't have control of the Apigee products that give people access to the API.

Instead it passes its `ratelimiting` var directly to the shared flow as a JSON encoded FlowCallout Parameter[1046].

> Passing the value to the shared flow this way takes precedence over anything defined on a product, however this method essentially hard codes the value in the proxy and does not allow the flexibility of per-environment settings.

1046 https://docs.apigee.com/api-platform/reference/policies/flow-callout-policy#parameterelement

For this reason it is not recommended for use by any proxy other than the identity-service. It is mentioned here only for completeness.

The identity-service currently enforces the following rate limits in all environments.

```
{
  "app": {
    "quota": {
      "limit": 300,
      "interval": 1,
      "timeunit": "minute",
      "enabled": true
    },
    "spikeArrest": {
      "enabled": false
    }
  },
  "proxy": {
    "quota": {
      "enabled": false
    },
    "spikeArrest": {
      "ratelimit": "300000pm",
      "enabled": true
    }
  }
}
```

If you need a higher total limit for the service, talk to us or raise a PR and then talk to us.

Adding custom rate limiting for an individual application

Note that only the QuotaPerApp and SpikeArrestPerApp (i.e. not the *PerProxy) policies can have their thresholds edited for an individual applications.

All applications contribute to the *PerProxy counts.

If you are applying custom rate limiting to a **prod** application you **MIGHT** need to obtain an **RFC** before doing the following steps. The risks are:

- Incorrect configuration has the potential to disrupt production traffic!
- A mistake might impact / stop traffic to another API

See Release Management (see page 2077) for guidance on how to judge the need for an RFC

Custom per-app `ratelimiting` is implemented by attaching a custom ratelimiting attribute to an app.

However, do not edit the `ratelimiting` JSON "by hand" in the Apigee UI. A malformed JSON string will cause all requests from the app to fail.

You need:

- Access to a build agent
- The name of the proxy
- The `appId` of the application you wish to give custom rate limiting
- A valid Apigee Access Token for the appropriate organization.

Instead there is a microservice to validate and edit of the JSON in a structured way.

To use it you must log in to a build agent.

The following shell commands should be run from a build agent - see KOP-021 Logging in to EC2 instances[1047]

1. Check if the app already has a `ratelimiting` attribute

```
curl https://apigee-deep-rest.ptl.api.platform.nhs.uk:9004/v1/organizations/nhsd-
nonprod/apps/<appId>/attributes/ratelimiting -H 'Authorization: bearer
<your_apigee_access_token>' | jq
```

If one is not present, you will get a 404 response with body

```
{
  "detail": "app <appId> has no attribute with name ratelimiting"
}
```

If one is present, you will get a 200 response with the custom `ratelimiting` settings, for example.

```
{
  "identity-service-internal-dev": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 27000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": true,
      "ratelimit": "450ps"
    }
  },
```

1047 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/193011226/KOP-021+Logging+in+to+EC2+instances

```
  "canary-api-internal-dev": {
    "quota": {
      "enabled": false
    }
  }
}
```

Note that apps may have custom settings for multiple proxy, like in this example.

2. Write your rate limiting JSON

The microservice will check it's properly formatted and handle inserting it into pre-existing configuration.

The schema is the same as on the proxy Apigee product, for example.

```
{
  "quota": {
    "enabled": true,
    "interval": 1,
    "limit": 60000,
    "timeunit": "minute"
  },
  "spikeArrest": {
    "enabled": false
  }
}
```

Then send the following request, note the <my-proxy-name> in the url!

```
curl -X PUT https://apigee-deep-rest.ptl.api.platform.nhs.uk:9004/v1/organizations/
nhsd-nonprod/apps/<appId>/attributes/ratelimiting/<your-proxy-name> -H
'Authorization: bearer <your_apigee_access_token>' -H 'content-type: application/
json' --data '
{
  "quota": {
    "enabled": true,
    "interval": 1,
    "limit": 60000,
    "timeunit": "minute"
  },
  "spikeArrest": {
    "enabled": false
  }
}
'
```

Note, if the `ratelimiting` attribute does not already exist, you will need to send the following instead.
However, if this is done when a ratelimiting attribute already exists, this WILL **DELETE** **all pre-existing** `ratelimiting` **configuration on the app**.
This is why you **MUST** check for existing configuration first.

```
curl -X PUT https://apigee-deep-rest.ptl.api.platform.nhs.uk:9004/v1/
organizations/nhsd-nonprod/apps/<appId>/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' -H 'content-type:
application/json' --data '
{
  "<your-proxy-name>": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 60000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": false
    }
  }
}
'
```

3. Confirm in the Apigee UI

Go to https://apigee.com/organizations/<org>/apps/details/<appId>

Look in the *Custom Attributes* section.

Once again, **DO NOT EDIT THIS BY HAND.**

4. Clear bash history

You have written your prod access tokens (typically valid for 24 hours) into a terminal that others have access to.

```
cat /dev/null > ~/.bash_history
```

```
history -c
```

will remove the command history for the next person to log into the build agent. (Pressing up in the terminal should do nothing now).

Removing custom rate limiting for an individual application

The requirements are the same as the previous section (see page 1045) (including deleting the command history)

On the build agent do:

```
curl -X DELETE https://apigee-deep-rest.ptl.api.platform.nhs.uk:9004/v1/
organizations/nhsd-nonprod/apps/<appId>/attributes/ratelimiting/<your-proxy-name> -H
'Authorization: bearer <your_apigee_access_token>'
```

Advanced

Which policies are on by default?

At the time of writing QuotaPerApp and SpikeArrestPerApp are on by default because values for them can be taken from other Apigee product values required by our pipeline.

Variable precedence of per-app values

We already have some required attributes on our manifest Apigee products to do with rate limiting.

You can use these to specify the dynamic values of the "app" section of the `ratelimiting` custom attribute.

manifest.yml OR manifest_template.yml

```
---
# values are given in the 'apigee/environments/products' section of the manifest
file...
apigee:
  environments:
  - name: internal-dev
    products:
    - name: my-product-name # e.g. canary-api-internal-dev
      quota: 600
      quotaTimeunit: minute
      quotaInterval: 1
      attributes:
        - name: ratelimit
          value: '600pm'
```

```
        - name: ratelimiting
          value:
            my-proxy-internal-dev: # MUST MATCH YOUR PROXY NAME
              quota:
                enabled: true # boolean, set to false to disable the policy entirely.
                interval: 1 # number of timeunits to window over, non-zero positive
int
                limit: 60000 # non-zero positive int
                timeunit: minute # minute/hour
              spikeArrest:
                enabled: true # boolean, set to false to disable the policy entirely
                ratelimit: 60000pm # must end in pm/ps
              # app:
              #  quota: {}
              #     enabled: true    <--- on by default, so YOU MUST SET TO FALSE to
disable
              #     interval: 1      <--- if you don't specify this here, the shared
flow uses product.quotaInterval instead
              #     limit: 600       <--- if you don't specify this here, the shared
flow uses product.quota instead
              #     timeunit: minute <--- if you don't specify this here, the shared
flow uses product.quotaTimeunit instead
              #  spikeArrest: {}
              #     enabled: false   <--- on by default, so YOU MUST SET TO FALSE to
disable
              #     ratelimit: 600pm <--- if you don't specify this here, the shared
flow uses product.ratelimit instead
```

Calling the shared flow

Inside your proxy or target XML file add a step referencing a ***FlowCallout*** immediately after an authorization step (e.g. ***OAuthV2.VerifyAccessToken***, ***VerifyAPIKey***). This is so that the API consumer's `app` has been identified and requests can be tallied against it.

proxies/default.xml OR target/default.xml

```
    <Flow name="ExampleFlowUsingRatelimit">
      <Description/>
      <Request>
        <Step>
          <Name>OAuthV2.VerifyAccessToken</Name>
        </Step>
        <!-- Must be after authorization policy! -->
        <Step>
          <Name>FlowCallout.ApplyRateLimiting</Name>
        </Step>
      </Request>
```

```
    <Response/>
  </Flow>
```

In your policies folder add an XML file defining the ***FlowCallout***.

If you are using an ***OAuthV2.VerifyAccessToken*** policy to authorize users, add the following to your ***policies*** folder.

FlowCallout.ApplyRateLimiting.xml

```
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.A
pplyRateLimiting">
    <DisplayName>Apply Rate Limiting</DisplayName>
    <SharedFlowBundle>ApplyRateLimiting</SharedFlowBundle>
</FlowCallout>
```

If you are using a ***VerifyApiKey*** policy, you will **need to provide its name as a parameter** (the name is defined as an attribute in the root element of your ***VerifyApiKey*** policy).

FlowCallout.ApplyRateLimitingApiKey.xml

```
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.A
pplyRateLimitingApiKey">
    <DisplayName>Apply Rate Limiting</DisplayName>
    <SharedFlowBundle>ApplyRateLimiting</SharedFlowBundle>
    <Parameters>
      <Parameter name="verifyApiKeyPolicyName">NameOfYourVerifyApiKeyPpolicy</
Parameter>
    </Parameters>
</FlowCallout>
```

Why do I need to provide the name of the VerifyApiKey policy?

Because Apigee assigns the app/product flow variables inside a namespace which includes the policy name for ***VerifyApiKey*** but not for ***OAuthV2.VerifyAccessToken***

Error messages

All policies trigger a 429 response from the proxy with an informative error message.

QuotaPerApp

```
{
  "message": "Your application, {developer.app.name}, has exceeded its quota of
{app_quota_limit} requests every {app_quota_interval} {app_quota_timeunit}(s) and is
being rate limited.",
  "scope": "application",
  "policy": "quota",
  "limit": {app_quota_limit},
  "interval": {app_quota_interval},
  "timeunit": "{app_quota_timeunit}",
  "ratelimiting_expiry_time_ms": {ratelimit.QuotaPerApp.expiry.time},
  "message_id": "{messageid}"
}
```

SpikeArrestPerApp

```
{
  "message": "Your application, {developer.app.name}, has created a spike in traffic
and is being rate limited. Please reduce the frequency of your requests.",
  "policy": "spikeArrest",
  "scope": "application",
  "message_id": "{messageid}"
}
```

QuotaPerProxy

```
{
  "message": "This API is currently receiving a high volume of requests and is being
rate limited.",
  "scope": "proxy",
  "policy": "quota",
  "message_id": "{messageid}"
}
```

SpikeArrestPerProxy

```
{
```

```
  "message": "This API is currently receiving a high volume of requests and is being
rate limited.",
  "scope": "proxy",
  "policy": "spikeArrest",
  "message_id": "{messageid}"
}
```

7.15.4.5.1.6 DEPRECATED Creating your own rate limiting policies

> **DEPRECATED**
>
> Until your proxy is switched over to the shared flow version, this page is still relevant.
> Ensure your backlog includes a ticket to upgrade to the more powerful Rate limiting shared flow (see page 1039)

Child pages that are important to help understanding:

- DEPRECATED Per-environment rate limits (see page 1057)
- DEPRECATED Production rate-limiting for test environments (see page 1061)

Overview

Use rate limits to protect your API from over-use and DoS attacks.

We have two types of rate limit:

1. Per-application rate limit
2. Per-API rate limit

Per-API rate limit

Use this type of rate limit to protect your back-end API from overall load.

There are two reasons you might want to do this:

1. You are launching a private beta, you haven't done load testing yet, and you want to put a low rate limit on your API
2. You are scaling up, and you have load tested your API to a certain level, but you want to be sure the overall load doesn't go over that level

Low rate limit for private beta

You can go into private beta without having made sure your API can scale by:

- Declaring that you plan to go into private beta with a low rate limit (a common choice is 5tps)
- Setting up a suitable per-API rate limit

This is normally coupled with limiting the number of participants in the private beta (e.g. maximum of 5 connecting applications).

You can then increase the rate limit (and the number of participants) once you have made your API scalable and have load tested it.

Per-application rate limits

Note: this feature is coming soon - see APM-1096[1048] - Rate limiting phase 2- strategic DONE .

To stop each calling application taking more than its fair share of the API pie, we are planning to introduce the capability to limit traffic per application. By default, each app will have a low rate limit e.g. 2tps per app per API. API consumers will be able to request a higher rate limit if they can justify it.

In fact, the developer hub implies that this is already a thing - see https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits.

Creating the rate limit policies

To achieve rate limiting on Apigee you must use policies defined within each of the API Proxies.

For rate limiting we will use the:

- Quota policy: https://docs.apigee.com/api-platform/reference/policies/quota-policy
- Spike Arrest policy: https://docs.apigee.com/api-platform/reference/policies/spike-arrest-policy

In these policies we define the limits in which the proxy can be used.

Quota

To add the quota policy you need to add a new policy in your proxy configuration.

Create a new file called 'Quota.xml' under proxies > (live or sandbox) > apiproxy > policies. This is where all the policies for the proxy are kept.

Within the file add:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Quota name="Quota" type="rollingwindow">
    <DisplayName>Quota</DisplayName>
    <Allow count="300" countRef="apiproduct.developer.quota.limit"/>
    <Interval ref="apiproduct.developer.quota.interval">1</Interval>
    <TimeUnit ref="apiproduct.developer.quota.timeunit">minute</TimeUnit>
</Quota>
```

This will set a rate limit of 300 request every 60 seconds. However, this includes some references, so if this is assigned to a product it will inherit the values that are set in the Product.

It will inherit in a way 'Product > Hard coded' in which values it decides to use.

1048 https://nhsd-jira.digital.nhs.uk/browse/APM-1096?src=confmacro

Spike Arrest

To add the spike arrest policy you need to add a new policy in your policy configuration.

Create a new file called 'SpikeArrest.xml' under proxies > (live or sandbox) > apiproxy > policies. This is where all the policies for the proxy are kept.

Within the file add:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SpikeArrest name="SpikeArrest">
  <DisplayName>SpikeArrest</DisplayName>
  <Rate ref="apiproduct.ratelimit">5ps</Rate>
  <UseEffectiveCount>true</UseEffectiveCount>
</SpikeArrest>
```

This will set a spike arrest limit of 5 requests per second. However, like the quota policy it inherits the values that are set in the product it is assigned to. So if 'ratelimit' of '7ps' is set within the product it is link with the hard coded spike limit of '5ps' will be overridden and set as '7ps'.

It will inherit in a way 'Product > Hard coded' in which values it decides to use.

Using the policies

For the quota and spike arrest policies to use the correct values from the apiproduct it is assigned to you **MUST** use a OAuth2 policy prior to using the policies. This is because the OAuth policies require variables to set the quota limits:

- `apiproduct.developer.quota.interval`
- `apiproduct.developer.quota.limit`
- `apiproduct.developer.quota.timeunit`
- `apiproduct.ratelimit`

However, different Oauth2 policies output the product variables as different variables names. In this case you then need to map those variables to the quota values using a 'AssignMessage' policy. For example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AssignMessage
name="AssignMessage.RenameQuotaVarsFromVerifyAPIKeyPolicyFromClientSecretFormParam">
    <AssignVariable>
        <Name>apiproduct.developer.quota.limit</Name>

<Ref>verifyapikey.VerifyAPIKey.FromClientSecretFormParam.apiproduct.developer.quota.l
imit</Ref>
    </AssignVariable>
    <AssignVariable>
        <Name>apiproduct.developer.quota.interval</Name>

<Ref>verifyapikey.VerifyAPIKey.FromClientSecretFormParam.apiproduct.developer.quota.i
nterval</Ref>
    </AssignVariable>
```

```
    <AssignVariable>
        <Name>apiproduct.developer.quota.timeunit</Name>

<Ref>verifyapikey.VerifyAPIKey.FromClientSecretFormParam.apiproduct.developer.quota.t
imeunit</Ref>
    </AssignVariable>
    <AssignVariable>
        <Name>apiproduct.ratelimit</Name>

<Ref>verifyapikey.VerifyAPIKey.FromClientSecretFormParam.apiproduct.ratelimit</Ref>
    </AssignVariable>
</AssignMessage>
```

Once the policies have been created you can then utilise them to use within the proxy flow.

To add the policies to the 'pre-flow' of the target endpoint we must specify what policies we want to use.

So in the target file under 'proxies' > (live or sandbox) > 'apiproxy' > 'targets' in {file_name}.xml you can specify what policy is used.

In here we can add the Quota and SpikeArrest policies to the pre-flow section of the configuration, for example:

```
<PreFlow>
    <Request>
      <Step>
        <!-- Header NHSD-Session-URID must be present and not empty -->
        <Name>RaiseFault.400BadRequest</Name>
        <Condition>(request.header.NHSD-Session-URID is null) or (not
request.header.NHSD-Session-URID ~~ ".+")</Condition>
      </Step>
      <Step>
        <Name>OauthV2.VerifyAccessToken</Name>
      </Step>
      <Step>
        <Name>Quota</Name>
      </Step>
      <Step>
        <Name>SpikeArrest</Name>
      </Step>
      <Step>
        <Name>AssignMessage.AddUserIdHeader</Name>
      </Step>
      <Step>
        <Name>AssignMessage.AddIssuerHeader</Name>
      </Step>
      <Step>
        <Name>KeyValueMapOperations.GetSecureVariables</Name>
      </Step>
      <Step>
        <Name>AssignMessage.AddAsidHeader</Name>
      </Step>
    </Request>
```

```
    </PreFlow>
```

So when a request is made to your proxy to will evaluate against these policies that we have defined.

They will check if the request has met certain condition, if not it will return a failure response.

The policies will be used in each request:

DEPRECATED Per-environment rate limits

> **DEPRECATED**
>
> Until your proxy is switched over to the shared flow version, this page is still relevant. Ensure your backlog includes a ticket to upgrade to the more powerful Rate limiting shared flow (see page 1039)

> Rate Limit changes need to be approved by your NFT representatives; and large increases should be discussed with the Deathstar Squad.

Why would you need to do this?

There are scenarios where you may want different rate limits within different environments. For example if you need to raise the rate limit in 'prod' due to external factors for a release but want to keep 'int' as it is so as not to incur too much traffic.

When we raise the **Rate Limit** we must also raise the **Quota** in equal ratio.

- The **Rate Limit** allows a specific amount of transactions per second against the proxy and prevents the API proxy's target backend against severe traffic spikes.
- The **Quota** limits the number of connections apps can make to your API proxy's target backend over a specific period of time.

How to set rate limits on separate environments

Rate limits and quotas are set on Apigee products and a default product is created automatically by the common pipelines in the 'api-management-utils' repo for each environment. The default sets the rate limit to 5 per second and the quota to 300 per minute.

It is important to note that the totals are per proxy, not per product - see how this works in the section below.

To set a different rate limit on specific environment you must define the products in your API using the following steps:

Create 'products' directory

Create a folder called 'products' in the root of your API repo.

Define product with the increased rate limit

Create a YAML file in the 'products' directory for the product and specify the follow key-value pairs (anything not specified will be applied by the default product parameters):

- **name** - Best left as an empty string as the service name will be appended.
- **environments** - a list of the environments you want the change in rate limit and quota to apply to.
- **attributes** - Specify access as public and set your rate limit here
- **quota** - Set your quota here
- **quotaInterval** - Set to 1
- **quotaTimeUnit** - Set to 'minute'

See example below

Product - Increased Rate Limit

```
name: ''
environments:
  - prod
attributes:
  - name: access
    value: public
```

```
  - name: ratelimit
    value: 130ps
quota: 7800
quotaInterval: 1
quotaTimeUnit: minute
```

Define product to use other environments

Create a YAML file in the same directory, which will list **all** other environments. This is important for the common pipelines to know when to use the new rate limit and when to use the default as it loops through the environments when creating the products.

The name parameter **must** be the same for both products to avoid duplication.

See example below

Product - Standard

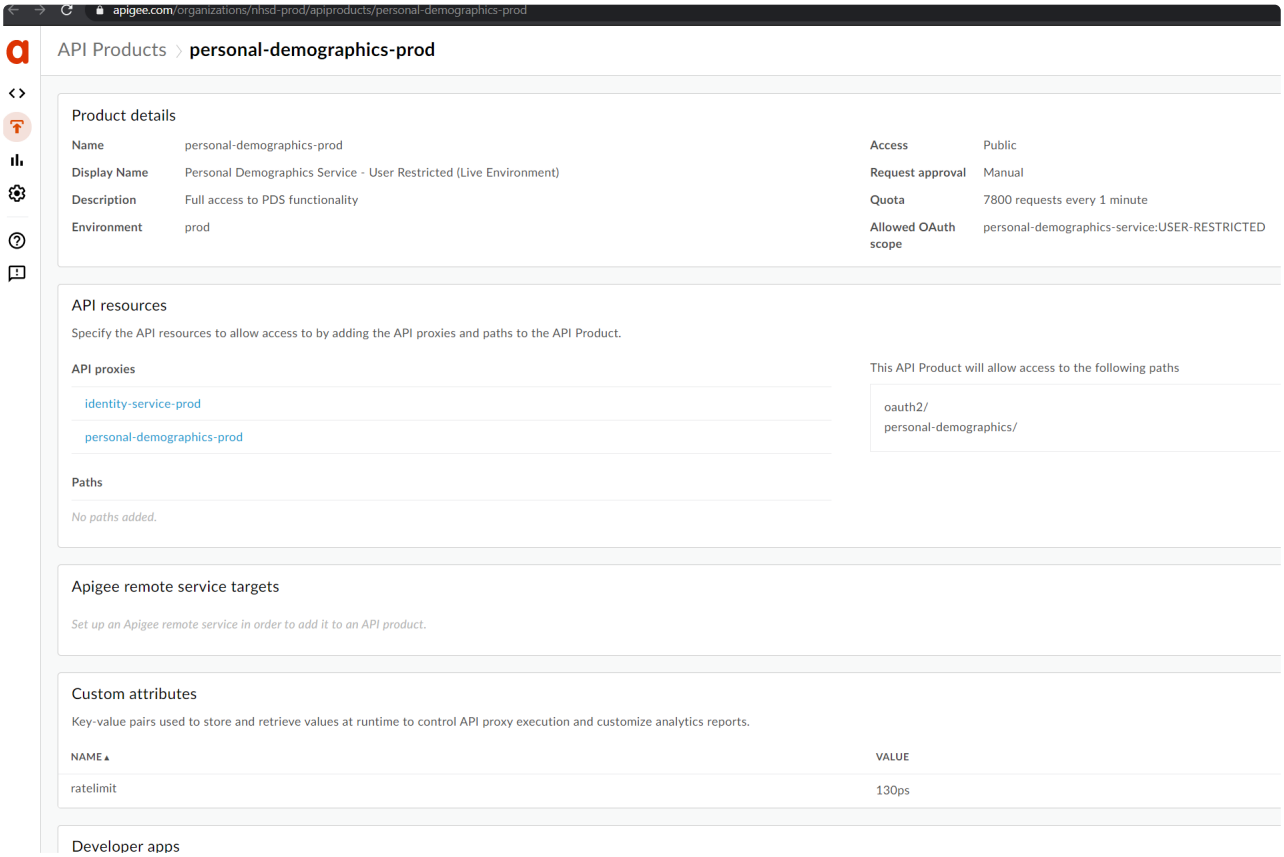
```
name: ''
environments:
  - internal-dev
  - internal-dev-sandbox
  - internal-qa
  - internal-qa-sandbox
  - sandbox
  - dev
  - ref
  - int
```

Examples on Apigee

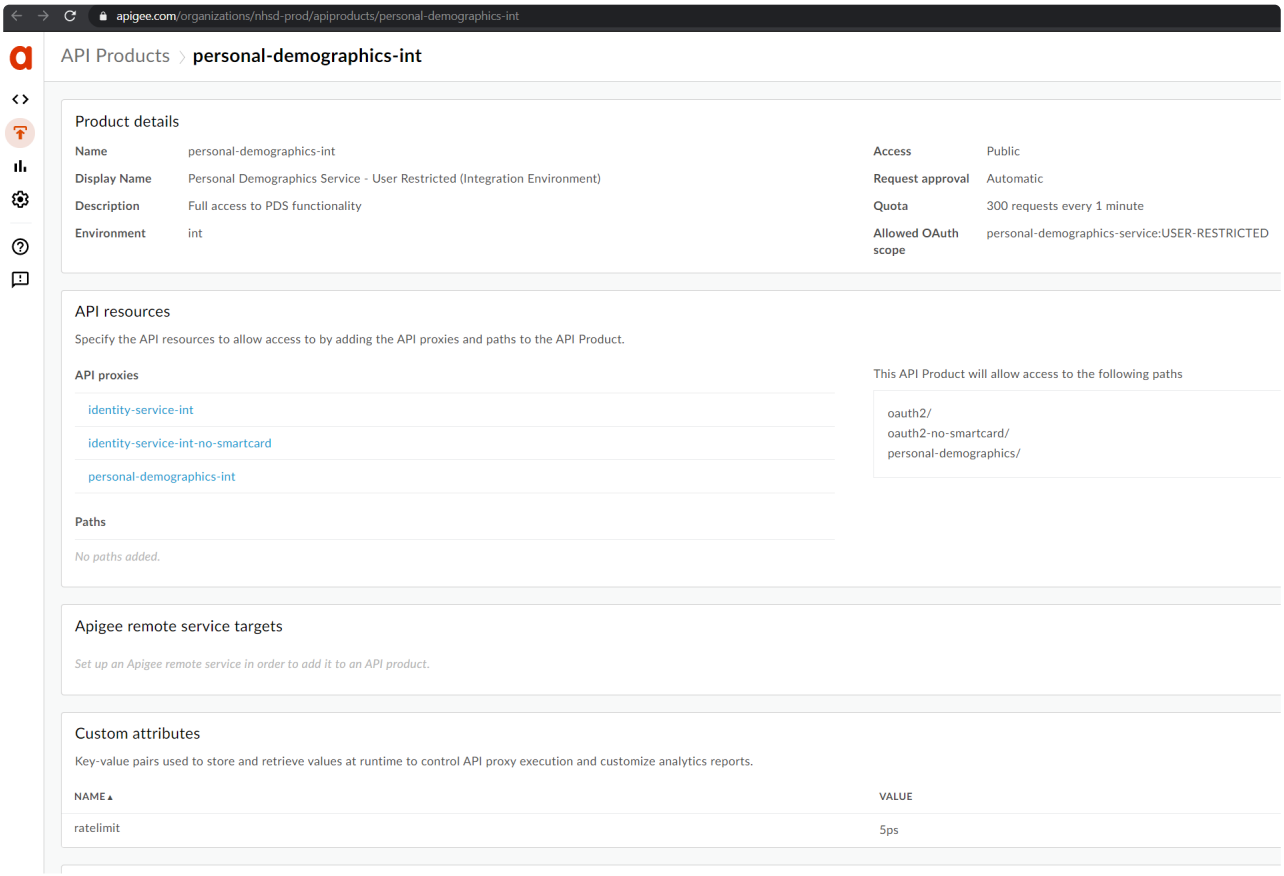
After these have been set your products will have different rate limits on Apigee.

Below are example of this from the Personal Demographics Service products.

Prod


apigee.com/organizations/nhsd-prod/apiproducts/personal-demographics-prod
API Products › personal-demographics-prod
Product details
Name personal-demographics-prod
Display Name Personal Demographics Service - User Restricted (Live Environment)
Description Full access to PDS functionality
Environment prod
Access Public
Request approval Manual
Quota 7800 requests every 1 minute
Allowed OAuth scope personal-demographics-service:USER-RESTRICTED
API resources
Specify the API resources to allow access to by adding the API proxies and paths to the API Product.
API proxies
identity-service-prod
personal-demographics-prod
Paths
No paths added.
This API Product will allow access to the following paths
oauth2/
personal-demographics/
Apigee remote service targets
Set up an Apigee remote service in order to add it to an API product.
Custom attributes
Key-value pairs used to store and retrieve values at runtime to control API proxy execution and customize analytics reports.
NAME
VALUE
ratelimit
130ps
Developer apps

Int



A note on differing rate limits per product

It is possible to create multiple products per proxy for your API. This could be done for multiple reasons but the common use case is for an API using both Application-restricted and User-restricted access. It's important to know that if setting the rate limits on the products combine to affect the behaviour of the rate limiting for the proxy:

- The proxy rate limit is the "max" of any product limits set
- Each product will be limited to the rate limit specified for that product

Example using 'Product A' and 'Product B'

- There is a counter on the proxy level for an API that increments by per request. The rate limits of the products look at the counter and check whether the limits are equal to or more than the counter.
- Product A has a rate limit of 50 tps. Product B has a rate limit of 100 tps.
- If Product A maxes out at 50 then Product B can still work up to another 50tps
- If Product B reaches 50 tps then Product A will be throttled.

DEPRECATED Production rate-limiting for test environments

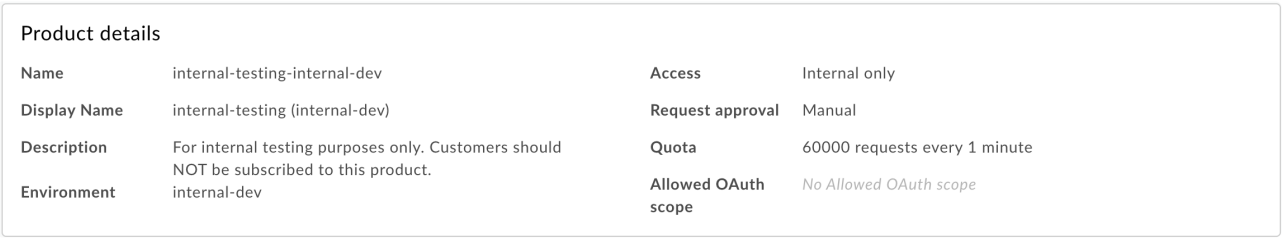
Contents

This can be more easily achieved using Custom Per-App Ratelimiting in combination with the ApplyRateLimiting shared flow.

Why would you need to do this?

This would be needed to test the api rate limiting and connection policies in a test environment before it reaches a production environment. This is because currently the API's currently in the test environments such as 'internal-dev', 'internal-qa', etc. have rate limiting set that tests don't fail while developing, so they don't conform to the production rate limits in which we set.

Currently, depending how your proxies are configured the quota for the rate limit is inherited from the API Product in which it is assigned. This means that any API proxy deploy to the internal development environments are assigned to the 'internal-testing' product for example:



However, if you would like to test the API proxy with the rate limiting used by your proxy, follow these steps.

Create a test application

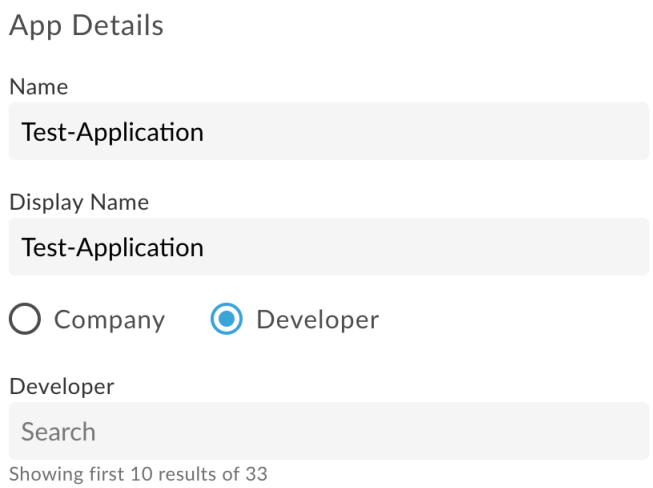
Navigate to Apigee Edge: https://apigee.com/

Once logged in navigate to:

- Publish > Apps
- Click 'Create App'

Set Application Name

To create the application, you need to fill out the form to create an application:



Assign the Developer to your Apigee account. It should appear in the drop down list.

Set Callback URL

To use the application through the simulated authentication journey, you must set the **callback url**

For the different testing environments they have their own testing application, you can use:

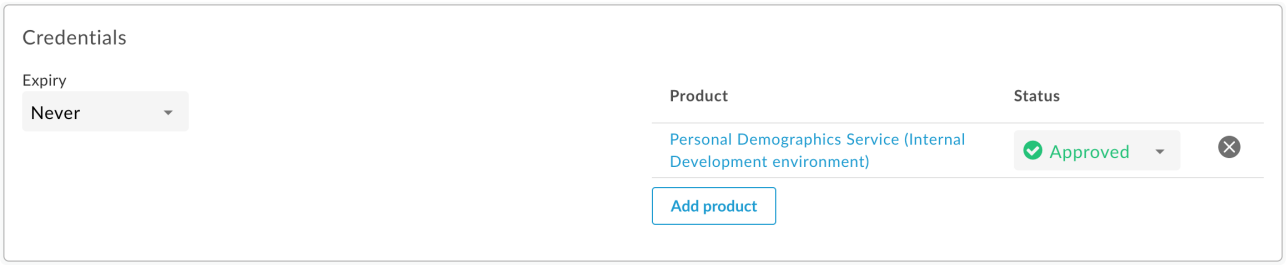
- **Internal-dev:** https://nhsd-apim-testing-internal-dev.herokuapp.com/callback
- **Internal-qa:** https://nhsd-apim-testing-internal-qa.herokuapp.com/callback
- **Internal-qa-sandbox:** https://nhsd-oauth-internal-qa-sandbox.herokuapp.com/callback

Assigning your API proxy to the application

To use your API proxy to test the rate limit with the 'correct' policies you need to assign it to the correct product.

To assign it to the product, select your 'production' product from the drop down list to select from. It should automatically approve you that product and you don't need to add the 'Identity Service' product as that is already automatically assigned.

For example:



Custom Attributes

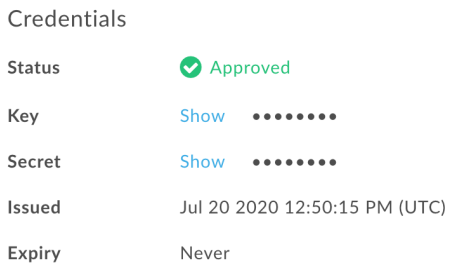
Set any custom attributes that is need for your API proxy in the creation page if any.

Then to create the application click 'create'.

Using the created application

Once you have created you can now use the application credentials to authenticate the client and make requests to your proxy.

The credentials can be found on your created application page, under 'key' and 'secret':



As the requests are using your new application it will be adhering to the assigned product policies rate limit.

7.15.4.5.2 Monitoring your API

> **Monitoring your API with proxy generator**
>
> If using proxy generator see Monitoring your API with proxy generator (see page 1115)

Page table of contents:

Subpages:

- Adding status monitoring to a service (see page 1065)
- Logging to Splunk (see page 1082)
- Troubleshooting service callouts (see page 1098)

7.15.4.5.2.1 API producer team responsibilities

API producer teams are responsible for monitoring the health of their own APIs.

This includes out-of-hours monitoring where appropriate (i.e. if your API is a silver, gold or platinum service).

The API platform is itself a platinum service, and the API platform team does monitor its health 24x7, but we only react to **API platform** issues, not to API-specific issues.

We're an altruistic bunch, so if we do notice an API-specific issue we might raise a live incident, but **don't rely on it**.

7.15.4.5.2.2 Tooling

The APIM Platform has configured three primary ways of monitoring your APIs:

- APIs & platform services up or down
    - Path to live environments: https://grafana.ptl.api.platform.nhs.uk/
    - Production: https://grafana.prod.api.platform.nhs.uk/
    - *(which send alerts to the Slack channels: #platforms-apim-alerts-ptl and #platforms-apim-alerts-prod)*
- Detailed logging (which you can also use for alerting):
    - Public API health dashboard (no personal data): https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/health
    - Restricted API health dashboard (includes personal data): https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
- Receive alerts in a separate slack channel:
    - Receive alerts for an API in a specific slack channel: [DEPRACATED] KOP-076 Creating specific alert channels for producer squads[1049]
- Alerts Prod Slack channel: platforms-apim-alerts-prod[1050]

If you want to know the details/Architecture of how it fits together see Monitoring[1051]

Pages with how to implement for API Producers:

- Adding status monitoring to a service (see page 1065)
    - _ping endpoint (see page 1073)
    - _status endpoint (see page 1075)
- Logging to Splunk (see page 1082)
    - Proxy redaction config (see page 1094)
- Troubleshooting service callouts (see page 1098)

7.15.4.5.2.3 Adding status monitoring to a service

Overview

We have a Grafana dashboard that monitors the health of all our APIs in all environments. You can access it at:

1049 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/465645917/DEPRACATED+KOP-076+Creating+specific+alert+channels+for+producer+squads
1050 https://nhsdigital-platforms.slack.com/archives/G012YEJMM5K
1051 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114033652/Monitoring

- Non-prod: https://grafana.ptl.api.platform.nhs.uk/ (currently you will need to request access to GRP-AzureNHSD-API-Management-PTL-Producers-DL (see page 161) to be able to view this dashboard)
- Prod: https://grafana.prod.api.platform.nhs.uk/ (currently you will need to request access to GRP-AzureNHSD-API-Management-PROD-DL (see page 161) to be able to view this dashboard)

It also sends slack alerts for any issues.

Each API has two monitoring endpoints:

- _ping (see page 1073) - "shallow" ping - confirms the API proxy is up and running
- _status (see page 1075) - "deep" ping - confirms that the back end service is up and running by making a call to it

For your API to appear on the Grafana dashboard you need to:

1. Make sure you have a _ping endpoint
2. Implement the _status endpoint

You can change the dashboard view by selecting the dashboard view next to the 4 squares, at the top left-hand side of the dashboard (currently this defaults to 'General/APIM'). The 'env' option below it, also enables you to change the environment view. The most useful dashboard views for producers are 'api details' and 'api status' for INT and Prod environments (access to prod environment is via a separate dashboard).

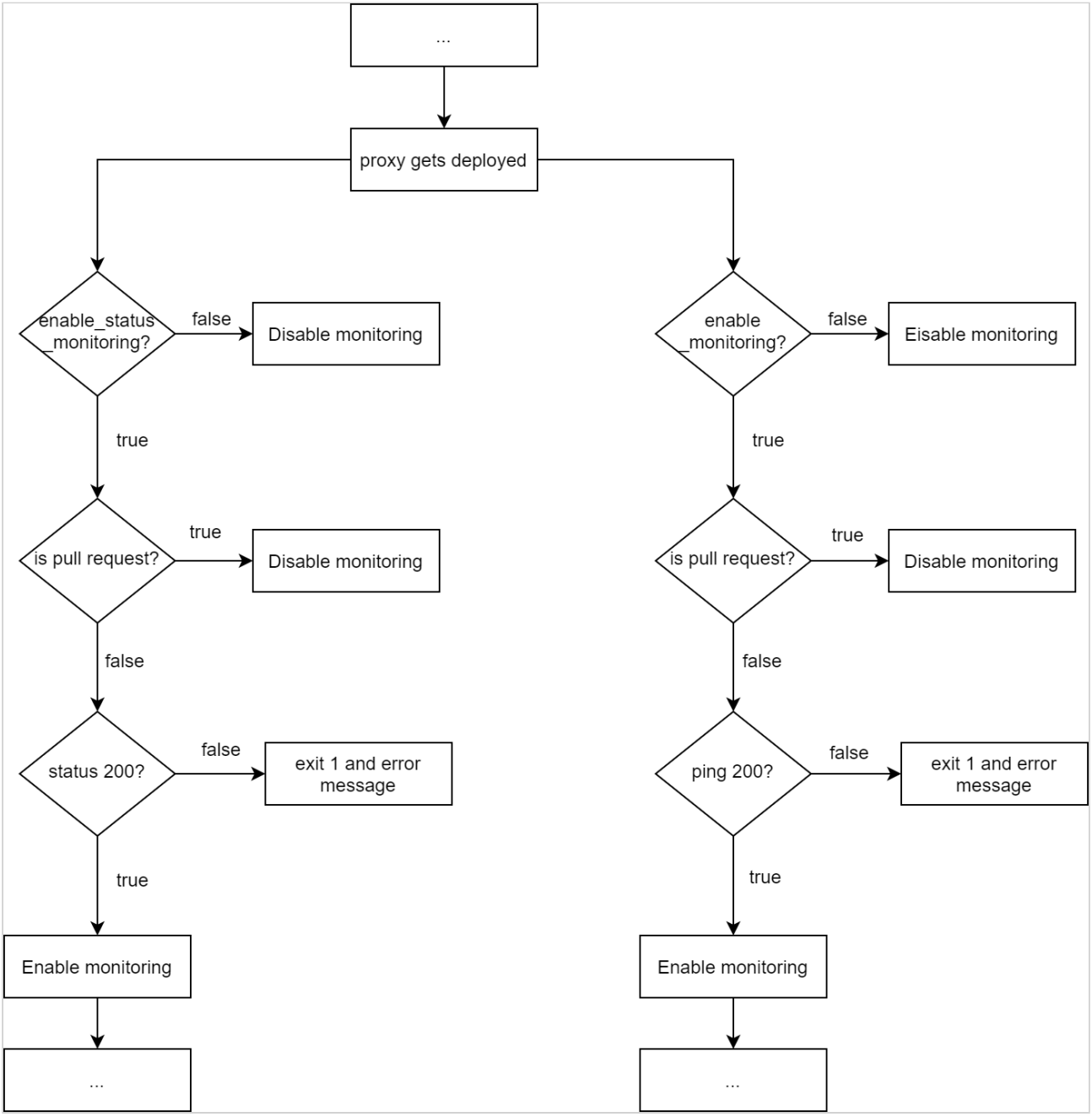
If you used our API proxy template to create your API proxy (as per Creating a GitHub repository and deployment pipelines for your API (see page 580)) then you will already have these two endpoints as they are part of the template boilerplate code. The only thing you'll need to do is amend the status endpoint to hit the corresponding health-check endpoint on your backend, otherwise, you'll get a 404 response. Refer to Monitoring[1052] for more information.

There are steps in the pipeline that are responsible for checking the status code of the ping and status endpoints and enabling and disabling monitoring endpoints (see the figure below).

Your API must have _status available prior to going into Production

There is a discovery service to store all the endpoints that our monitoring architecture has to monitor. The pipeline step works by sending a POST request with your proxy's _ping and _status endpoints to the discovery service. For example, if you're deploying ambulance-analytics on the internal-dev environment then, this step will add "ambulance-analytics@internal-dev=http_2xx https://internal-dev.api.service.nhs.uk/ambulance-analytics/_ping" to the list of monitoring endpoints.

1052 https://nhsd-confluence.digital.nhs.uk/display/APM/Monitoring



The pull-request deployments are not monitored.

You'll be responsible for monitoring your own service during runtime, however, we also keep an eye on which services are up or down. If we spot that an environment is down and we haven't been made aware of any planned downtime we'll get in touch with you to make sure you are aware. It is helpful if you let us know regards any planned downtime you intend to have by informing us in advance via Help and support (see page 2056), to avoid any false alarms.

For more background and design intent behind _ping and _status see Proxy Availability & Backend Status[1053].

Enabling monitoring for Proxygen proxies

This has two aspects:

- Your OAS file has monitoring enabled
- You have told the APIM _status endpoint where your backend health check

Proxygen OAS attributes to set

Proxy generator logging and monitoring is controlled with a single boolean flag in the proxygen OAS spec as shown below:

proxygen-oas.yaml

```
x-nhsd-apim:
  monitoring: true
```

This flag is set to **true** by default, and adds the _ping and _status endpoints to your proxy.

Telling Proxygen where your healthcheck endpoint is

When a call comes into _status, your proxy will then sends a new request to your backend healthcheck endpoint.

Follow the guidance to do this: _status endpoint (see page 1075)

Enabling monitoring for GitHub repo proxies

Monitoring is controlled by two flag variables that you can pass in your azure-release-pipeline.yml file. These variables are:

- "enable_monitoring": enable/disable ping endpoint monitoring
- "enable_status_monitoring": enable/disable status endpoint monitoring

It is important to mention that these variables are both true by default meaning that if you don't use the variables explicitly we will try to enable the monitoring when you release your proxy.

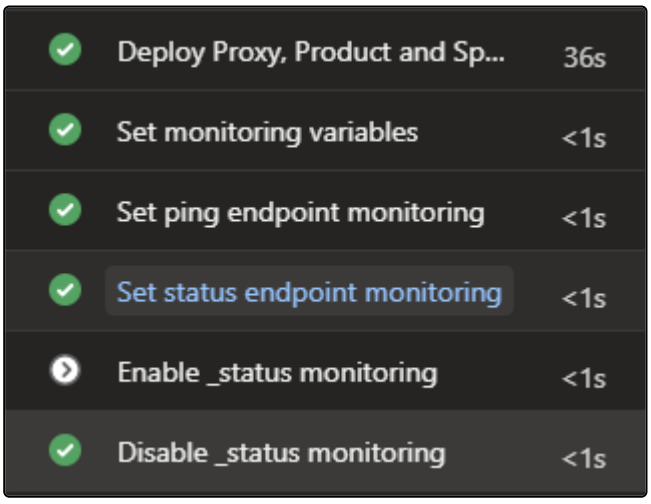
If your proxy doesn't support a status endpoint (i.e your proxy doesn't have a backend) you should explicitly define "enable_status_monitoring: false" on your release pipeline definition. Below is a code snippet showing this process:

1053 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=114902981

/azure/azure-release-pipeline.yml

```
#file: <root>/azure/azure-release-pipeline.yml

# ....

extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    product_display_name: ${{ variables.product_display_name }}
    product_description: ${{ variables.product_description }}
    spec_file: ${{ variables.spec_file }}
    enable_status_monitoring: false  # <- **** This will disable status monitoring
for all environments ****
  apigee_deployments:
        - environment: internal-dev
        - environment: internal-qa
        - environment: internal-qa-sandbox
          proxy_path: sandbox
        - environment: int
          depends_on:
              - internal_qa
              - internal_qa_sandbox
        - environment: sandbox
          proxy_path: sandbox
          depends_on:
              - internal_qa
              - internal_qa_sandbox
```

After making the above change to your pipeline configuration you need to merge it to master. After running the release pipeline you should see the step "Disable _status monitoring" in the release step list (see figure below).



There are two ways to confirm that monitoring is actually enabled/disabled. Firstly you can visit the Grafana dashboard[1054] and check if your proxy appears in the list of monitored proxies. Alternatively, you can send a GET request to the monitoring service discovery, after which you should be able to see your proxy name and the different environments in the JSON response. Here is a snippet of the response:

```
curl -X GET https://internal-dev.api.service.nhs.uk/sd-endpoints-test/service | jq .
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   847  100   847    0     0   7495      0 --:--:-- --:--:-- --:--:--  7429
[
  {
    "targets": [
      "sync-wrap@internal-dev=http_2xx https://internal-dev.api.service.nhs.uk/sync-
wrap/_status",
      "personal-demographics@prod=http_2xx https://api.service.nhs.uk/personal-
demographics/_ping",
      "monitoring-service-discovery@internal-dev=http_2xx https://internal-
dev.api.service.nhs.uk/monitoring-sd/_ping",
      "ambulance-analytics@internal-dev=http_2xx https://internal-
dev.api.service.nhs.uk/ambulance-analytics/_ping",
      "ambulance-analytics@internal-qa=http_2xx https://internal-
qa.api.service.nhs.uk/ambulance-analytics/_ping",
      "ambulance-analytics@internal-qa-sandbox=http_2xx https://internal-qa-
sandbox.api.service.nhs.uk/ambulance-analytics/_ping",
      "ambulance-analytics@int=http_2xx https://int.api.service.nhs.uk/ambulance-
analytics/_ping",
      "ambulance-analytics@sandbox=http_2xx https://sandbox.api.service.nhs.uk/
ambulance-analytics/_ping"
    ]
  }
]
```

If you disable monitoring on your proxy you should see the target disappear from the list and also the proxy disappear from the Grafana dashboard.

Adding monitoring for specific environments

If you want to disable monitoring for a particular environment then, you can override the default value parameter for each apigee_deployments separately.

1054 https://grafana.ptl.api.platform.nhs.uk/d/JYNsY_xMz/api-status?orgId=1&var-endpoint=All&var-environment=internal-dev

/azure/azure-release-pipeline.yml

```
extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
    product_display_name: ${{ variables.product_display_name }}
    product_description: ${{ variables.product_description }}
    spec_file: ${{ variables.spec_file }}
    apigee_deployments:
        - environment: internal-dev
        - environment: internal-qa
        - environment: internal-qa-sandbox
          proxy_path: sandbox
          enable_status_monitoring: false                                    # <- ****
disable status monitoring only for internal-qa-sandbox ****
        - environment: int
          depends_on:
              - internal_qa
              - internal_qa_sandbox
        - environment: sandbox
          proxy_path: sandbox
          depends_on:
              - internal_qa
              - internal_qa_sandbox
```

Add custom endpoints

There may be scenarios where you want to add or remove endpoints manually or you want to perform monitoring for endpoints other than _ping and _status. You can make a "post-hook" step in your pipeline configuration by performing a POST request to the monitoring-service-discovery[1055].

The monitoring-service-discovery proxy is deployed only to the internal-dev environment and we use this deployment for all Path-To-Live environments. There is a separate deployment for the Live environment.

GET a list of endpoints

You can get a list of monitoring endpoints by sending a GET request to the monitoring-service-discovery / service endpoint. This endpoint is open access.

1055 https://github.com/NHSDigital/monitoring-service-discovery

POST https://internal-dev.api.service.nhs.uk/monitoring-sd/service

```
curl -X GET https://internal-dev.api.service.nhs.uk/sd-endpoints-test/service
```

Add new endpoint

You can add new endpoints by sending a POST request. This endpoint is protected by the internal-dev API key. The JSON body of the request should look like this:

POST https://internal-dev.api.service.nhs.uk/monitoring-sd/service

```
{
  "<service_name>": {
    "<environment1>": [
      "<service_name>@<environment1=http_2xx <endpoint1>",
      "<service_name>@<environment1=http_2xx <endpoint2>"
    ],
    "<environment2>": [
      "<service_name>@<environment2=http_2xx <endpoint1>",
      "<service_name>@<environment2=http_2xx <endpoint2>"
    ]
  {
}
```

Here is an example of a JSON that adds _ping endpoint to the ambulance-analytics-api for the internal-dev environment:

POST https://internal-dev.api.service.nhs.uk/monitoring-sd/service

```
{
  "ambulance-analytics": {
    "internal-dev": [
      "ambulance-analytics@internal-dev=http_2xx https://internal-
dev.api.service.nhs.uk/ambulance-analytics/_ping",
    ]
  {
}
```

After POSTing the request you can confirm it by sending a GET request.

Remove an endpoint

There is no DELETE method. You can remove an endpoint by sending an empty endpoints array. Here we send a POST request and delete all the ambulance-analytics endpoints for the internal-dev environment:

POST https://internal-dev.api.service.nhs.uk/monitoring-sd/service

```
{
  "ambulance-analytics": {
    "internal-dev": []
  {
}
```

Make sure you specify the service_name and environment. Sending an empty object will remove all the endpoints.

Slack alerts channel

If an API goes down in a path to a live environment an alert is sent to a Slack channel which is monitored by the API Management platform team.

There are two slack alerts channels:

- All the Path-To-Live alerts are sent to the "platforms-apim-alerts-ptl[1056]" slack channel.
- Production is sent to "platforms-apim-alerts-prod[1057]"

The team will triage any alerts received and take action to resolve any issues, which may involve contacting you if you need to take action.

Adding statuds monitoring to your Sandbox environment

Adding monitoring to your sandbox looks much the same as adding monitoring to any of your other environments. Make sure your sandbox has a _ping endpoint (see page 1073) and a _status endpoint (see page 1075) set up and ready to go. For the healthcheck that sits behind the _status endpoint, you can create a simple endpoint that always returns a successful response.

_ping endpoint

The Ping endpoint is designed to very simply allow monitoring of the proxy and as a result, doesn't hit any back-end services; the request only goes as far as hitting a _ping endpoint on the proxy which returns a 200 status code and some additional information within the payload such as the revision version of the proxy as an example.

```
{
    "version":"v1.0.797-alpha",
```

1056 https://nhsdigital-platforms.slack.com/archives/G012K44NKUP
1057 https://nhsdigital-platforms.slack.com/archives/G012YEJMM5K

```
    "revision":"81",
    "releaseId":"23230",
    "commitId":"947eb644bcc7132a2c5561e5a201d53b49eb68c5"
}
```

Field	Value
version	This is the APIM auto generated version from GitHub. Key value during deployments
revision	Apigee revision number deployed
releaseId	Azure DevOps releaseId
commitId	GitHub Commit ID

For more background and design intent see Proxy Availability & Backend Status[1058]

Configuring your proxy for this

If you have used the proxy template[1059] to create your new proxy, you will already have a ping endpoint in your Proxy. This comprises of a flow which attaches information about the proxy to the request then returns a response. This is easily configurable.

Ping flow

Within the default.xml file, you have the below flow within your proxy which is invoked when a request hits the _ping endpoint.

```
<Flow name="AddPayloadToPing">
    <Request/>
    <Response>
       <Step>
           <Name>AssignMessage.AddPayloadToPing</Name>
       </Step>
    </Response>
    <Condition>(proxy.pathsuffix MatchesPath "/_ping") and ((request.verb
= "GET") or (request.verb = "HEAD"))</Condition>
 </Flow>
```

Breakdown of how the flow works

1058 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114902981/Proxy+Availability+Backend+Status
1059 https://nhsd-confluence.digital.nhs.uk/display/APM/Populating+an+API+repository+from+a+template

The request hits the proxy, and if the proxy is functioning properly, it will return a response created within the AssignMessage.AddPayloadToPing.xml policy. Below is an example of this policy which is included within the proxy template.

AssignMessage.AddPayloadToPing.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AssignMessage continueOnError="false" enabled="true" name="AssignMessage.AddPayloadToPing" async="false">
  <Set>
    <Payload>{"version":"{{ DEPLOYED_VERSION }}","revision":"{apiproxy.revision}","releaseId":"{{ RELEASE_RELEASEID }}","commitId":"{{ SOURCE_COMMIT_ID }}"}</Payload>
    <StatusCode>200</StatusCode>
    <Verb>GET</Verb>
    <Headers>
      <Header name="Content-Type">application/json</Header>
    </Headers>
    <Version>1.1</Version>
  </Set>
  <AssignTo createNew="false" transport="http" type="request"/>
  <IgnoreUnresolvedVariables>false</IgnoreUnresolvedVariables>
</AssignMessage>
```

After the response is created from the AssignMessage.AddPayloadToPing.xml policy, this is then returned from the proxy.

_status endpoint

The _status endpoint is designed to indicate if the API (or proxy) is actually running end to end. For instance, in PDS, Apigee might be up, but there is a problem with Spine. Overall this means that the proxy is down.

A service is made up of one or more backend services. Ideally this status should return the status for every one of those backend services - making it easy to spot the failure point.

Key design aspects:

- An incoming request to _status makes a call to check the health of one or more of your backend services
- The payload of the response will follow the draft RFC for health https://inadarei.github.io/rfc-healthcheck/
- The HTTPS Status code will be **200** for all outcomes (where Apigee works correctly), which is a deviation from the RFC
- Simple & lightweight - what existing health checks can you re-use here?

In practice this means:

- In all circumstances _status will return a JSON object
- HTTP Status code code returned by _status will be: 200
- The object will be made up of:

    - Single field "status" which will be: pass / warn / fail
    - Proxy key details (e.g. version)
    - List of target servers checked
- Endpoint is protected by an API Key, with a manually approved API Product
- We suggest that you start with the simplest health checks, and over time add more complex ones
- APIM Splunk logging actually logs the _status content body (not normally done), so that you can see the current / historic detail in Splunk

For detailed field level guidance on _status (and reasoning for each field) see: Proxy Availability & Backend Status[1060]

But my backend is serverless, there is no point having an _status polling it...
APIM **still recommends** you have _status checking:

- Is the problem APIM or my backend? This can save significant time during an incident
- _status is about end to end connectivity
    - For instance, maybe Route53 has been misconfigured and the traffic is no longer hitting your Lambdas
- Raising visibility about possible root cause - the more people that can see which or what specific part of your service is down, the faster the issue gets resolved
- Reuse an existing endpoint if possible - in many APIs there is an existing endpoint that can trigger every component in the backend chain
- Alerting before your customers report it - _ping & _status are about ensuring you are fixing an issue before your consumers contact you
- Start small - only add lots of service checks when you have time (but ideally before you have a complex to diagnose outage!)
- If you have very, very low traffic volumes you might find that _status does swamp your service. However, if it's serverless there should be a trivial cost

Configuring Proxygen

If you are using Proxygen you need to tell it where your own healthcheck endpoint is, that is then returned, via the APIM _status endpoint:

```
x-nhsd-apim:
  target:
    type: external
    healthcheck: /_mybackendhealthcheck
```

You can specify any path, but it is always related to your target server.

1060 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114902981/Proxy+Availability+Backend+Status

(API Producer support:

** can a team put in a full url if the healthcheck is on a different domain / sub-domain?*

** I take it is a 200 status that they results in a status of "pass"?*

)

Configuring your GitHub repo based proxy

If you have used the proxy template[1061] to create your new proxy, you will already have a status endpoint in your Proxy. The endpoint defaults to the `/healthcheck` endpoint on the `targetEndpoint ig3(spine)`; however, you can easily reconfigure the target to fit your use-case.

Status flow

In the file `default.xml`, you have the following `statusEndpoint` flow which is called when you hit the `_status` endpoint of your proxy.

status flow

```
<Flow name="StatusEndpoint">
      <Description/>
      <Request>
        <Step>
          <Name>KeyValueMapOperations.GetSecureVariables</Name>
        </Step>
        <Step>
          <Condition>private.apigee.status-endpoint-api-key NotEquals
request.header.apikey</Condition>
          <Name>RaiseFault.401Unauthorized</Name>
        </Step>
        <Step>
          <Name>ServiceCallout.CallHealthcheckEndpoint</Name>
        </Step>
      </Request>
      <Response>
        <Step>
          <Name>javascript.SetStatusResponse</Name>
        </Step>
      </Response>
      <Condition>(proxy.pathsuffix MatchesPath "/_status") and ((request.verb =
"GET") or (request.verb = "HEAD"))</Condition>
    </Flow>
```

There are four steps which are processed in order. Some of the steps have conditions attached to them, meaning they will only run if the condition is met.

1061 https://nhsd-confluence.digital.nhs.uk/display/APM/Populating+an+API+repository+from+a+template

KeyValueMapOperations.GetSecureVariables

There are 2 API keys for the status endpoints. One for the Production environment and another one shared across all the other environments. The policy below will load the `apiKey` from the key values map for the environment, you are in and store it in the variable `status-endpoint-API-key`. This variable is used later in the flow to verify the value passed via the "`apiKey`" header.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<KeyValueMapOperations async="false" continueOnError="false" enabled="true" name="Key
ValueMapOperations.GetSecureVariables" mapIdentifier="apim-variables-encrypted">
    <Get assignTo="private.apigee.status-endpoint-api-key" index="1">
        <Key>
            <Parameter>status-endpoint-api-key</Parameter>
        </Key>
    </Get>
</KeyValueMapOperations>
```

RaiseFault.401Unauthorized

This policy will raise a fault which will be displayed to the user as HTTP status 401. This can be captured and handle in Apigee using `faultRules`.

This policy is called if the value of the "`apiKey`" header passed in does not match the stored "`apiKey`" value.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<RaiseFault async="false" continueOnError="false" enabled="true" name="RaiseFault.401
Unauthorized">
    <DisplayName>RaiseFault.401Unauthorized</DisplayName>
    <Properties/>
    <FaultResponse>
        <Set>
            <Headers/>
            <Payload contentType="text/plain"/>
            <StatusCode>401</StatusCode>
            <ReasonPhrase>Server Error</ReasonPhrase>
        </Set>
    </FaultResponse>
    <IgnoreUnresolvedVariables>true</IgnoreUnresolvedVariables>
</RaiseFault>
```

ServiceCallout.CallHealthcheckEndpoint

if the `apikey` header is valid then it will make a call to the chosen healthcheck service using a `serviceCallout` policy. This can be configured to call either a target server as the template does or you can call out to a URL.

Target Server Callout

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ServiceCallout async="false" continueOnError="true" enabled="true" name="ServiceCall
out.CallHealthcheckEndpoint">
    <DisplayName>ServiceCallout.CallHealthcheckEndpoint</DisplayName>
    <Properties/>
    <Request clearPayload="true" variable="healthcheckRequest">
        <IgnoreUnresolvedVariables>false</IgnoreUnresolvedVariables>
    </Request>
    <Response>healthcheckResponse</Response>
    <HTTPTargetConnection>
        <SSLInfo>
            <Enabled>true</Enabled>
        </SSLInfo>
        <LoadBalancer>
            <Server name="ig3"/>
        </LoadBalancer>
        <Path>/healthcheck</Path>
    </HTTPTargetConnection>
    <Timeout>20000</Timeout>
</ServiceCallout>
```

This is an example of calling a healthcheck service using a URL.

https://docs.apigee.com/api-platform/reference/policies/service-callout-policy#url-as-a-variable

URL Service Callout

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ServiceCallout async="false" continueOnError="true" enabled="true" name="ServiceCall
out.CallHealthcheckEndpoint">
    <DisplayName>ServiceCallout.CallHealthcheckEndpoint</DisplayName>
    <Properties/>
    <Request clearPayload="true" variable="healthcheckRequest">
        <IgnoreUnresolvedVariables>false</IgnoreUnresolvedVariables>
    </Request>
    <Response>healthcheckResponse</Response>
    <HTTPTargetConnection>
        <URL>http://example.com/healthcheck</URL>
    </HTTPTargetConnection>
    <Timeout>20000</Timeout>
</ServiceCallout>
```

javascript.SetStatusResponse

This is the `SetStatusResponse` policy it just calls the javascript file `SetStatusResponse.js`

javascript.SetStatusResponse

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Javascript async="false" continueOnError="false" enabled="true" timeLimit="200" name="
javascript.SetStatusResponse">
    <DisplayName>javascript.SetStatusResponse</DisplayName>
    <Properties/>
    <ResourceURL>jsc://SetStatusResponse.js</ResourceURL>
</Javascript>
```

This JavaScript file is responsible for taking the response from the healthcheck call and converting it a JSON response returned by the proxy.

SetStatusResponse.js

```
// These values are rendered by pipeline's template step
var version = "{{ DEPLOYED_VERSION }}"
var releaseId = "{{ RELEASE_RELEASEID }}"
var commitId = "{{ SOURCE_COMMIT_ID }}"

var apiproxy_revision = context.getVariable('apiproxy.revision');

var healthcheck_service_response_code = context.getVariable('healthcheckResponse.stat
us.code');
var healthcheck_service_response = context.getVariable('healthcheckResponse.content')
;
var healthcheck_service_request_url = context.getVariable('healthcheckRequest.url');

var healthcheck_service_request_has_failed = context.getVariable("servicecallout.Serv
iceCallout.CallHealthcheckEndpoint.failed");

var healthcheck_service_status = "fail";

if(healthcheck_service_response_code/ 100 == 2){
  healthcheck_service_status = "pass";
}

timeout = "false";

if(healthcheck_service_response_code === null &&
healthcheck_service_request_has_failed){
  timeout = "true";
```

```
}

try {
    healthcheck_service_response = JSON.parse(healthcheck_service_response)
}
catch(e) {
 // Response is not a JSON, so leave it as it is.
}

var healthcheck_service = {
  "healthcheckService:status" : [
    {
      "status": healthcheck_service_status,
      "timeout" : timeout,
      "responseCode" : healthcheck_service_response_code,
      "outcome":  healthcheck_service_response,
      "links" : {"self": healthcheck_service_request_url}
    }]
};

var apigee_status = "pass";

if(healthcheck_service_status != "pass"){
  apigee_status = "fail";
}

var response = {
  "status" : apigee_status,
  "version" : version ,
  "revision" : apiproxy_revision,
  "releaseId" : releaseId,
  "commitId": commitId,
  "checks" : healthcheck_service
};

context.setVariable("status.response", JSON.stringify(response));
context.setVariable("response.content", JSON.stringify(response));
context.setVariable("response.header.Content-Type", "application/json");
```

JSON minimum response

```
{
    "status": "pass",
    "version": "",
    "revision": "",
    "releaseId": "",
    "commitId": "",
    "checks": {
        "healthcheckService:status": [
            {
```

```
                "status": "pass",
                "timeout": false,
                "responseCode": 200,
                "outcome": "<html><h1>Ok</h1></html",
                "links": {
                    "self": "http://healthcheckService.example.com/_status"
                }
            }
        ]
    }
}
```

Route Rules

You also have the following route rule. This rule means that if the request has come in on the `_status` endpoint, then the proxy won't attempt to call out to the target server.

route rule

```
    <RouteRule name="NoRouteStatus">
      <Condition>(proxy.pathsuffix MatchesPath "/_status") and ((request.verb =
"GET") or (request.verb = "HEAD"))</Condition>
    </RouteRule>
```

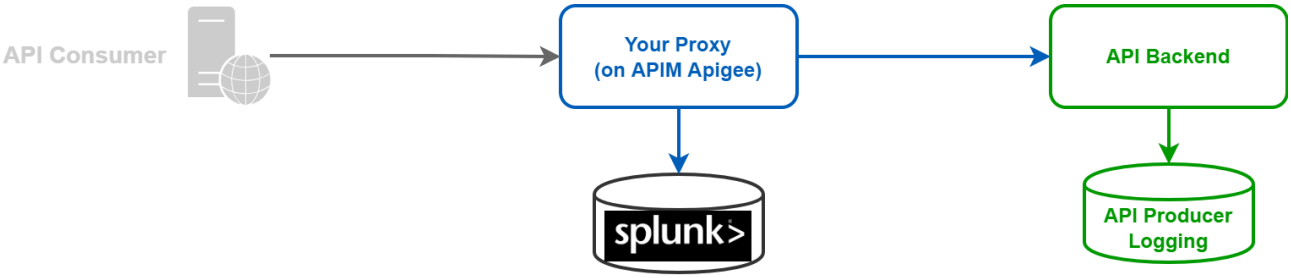
A few route rules are doing different things, and it's important you get them in the right order as they are processed in order when a request is received.

7.15.4.5.2.4 Logging to Splunk

Overview

We use **NHS Digital Splunk** for monitoring API traffic, including alerting, analytics and diagnostics. All API requests are logged to Splunk, unless you remove the LogToSplunk shared flow from your Proxy.

Thinking about your end to end service, there are multiple places where you need to log information:



In most projects the "API Producer Logging" is also the NHS Digital Splunk cloud instance, and you can then create dashboards, reports and alerts based on data from your backend API indexes and the from the APIM indexes.

The most important concept in Splunk is the "index", where the log data is stored, and APIM splits it's indexes into:

- API Requests:
    - Raw, which can include Personally Identifiable Data (PID / PII)
    - Safe (redacted) versions of the raw indexes with the PID removed or masked
- AWS logs - it you are using the APIM AWS Hosted container these might be of use
- Audit logs

We have a number of pre-built Splunk dashboards for quick viewing.

For full details of how logging works under the covers, see Logging[1062].

Dashboards

API Management Health Dashboard

Visit the dashboard here[1063]. The dashboard can be filtered by the index, environments, proxies and application to get a more focused view on a particular API, using the drop down menus under the dashboard heading.

Access

Anyone with access to the NHS Digital Splunk Cloud instance can view this dashboard.

Purpose

The API Management Health dashboard displays details of requests on the Platform across environments, proxies and applications. The dashboard **avoids showing any sensitive data** by searching against the redacted indexes: *summary_apm_prod_open* and *summary_apm_ptl_open*.

The dashboard shows results from the searches listed below for the last hour, last 24 hours and last 7 days.

- Total Requests (excluding ping & status)
- % Availability
- Requests
- Request duration
- % Successful Requests (<500)
- Response Code by Percentage Breakdown

APIM KPI Reporting Dashboard

Visit the dashboard here[1064].

1062 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114033654/Logging
1063 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/health?form.index=summary_apm_prod_open&form.proxy=*&form.app=*&form.env=prod
1064 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/service?form.report_timerange.earliest=%40mon&form.report_timerange.latest=now

Access

Anyone with access to the NHS Digital Splunk Cloud instance can view this dashboard.

Purpose

The APIM KPI Reporting Dashboard dashboard displays KPIs for APIs in production and the applications that use them. Like the API Management Health dashboard it avoids showing any sensitive data by searching against the *summary_apm_prod_open* redacted index.

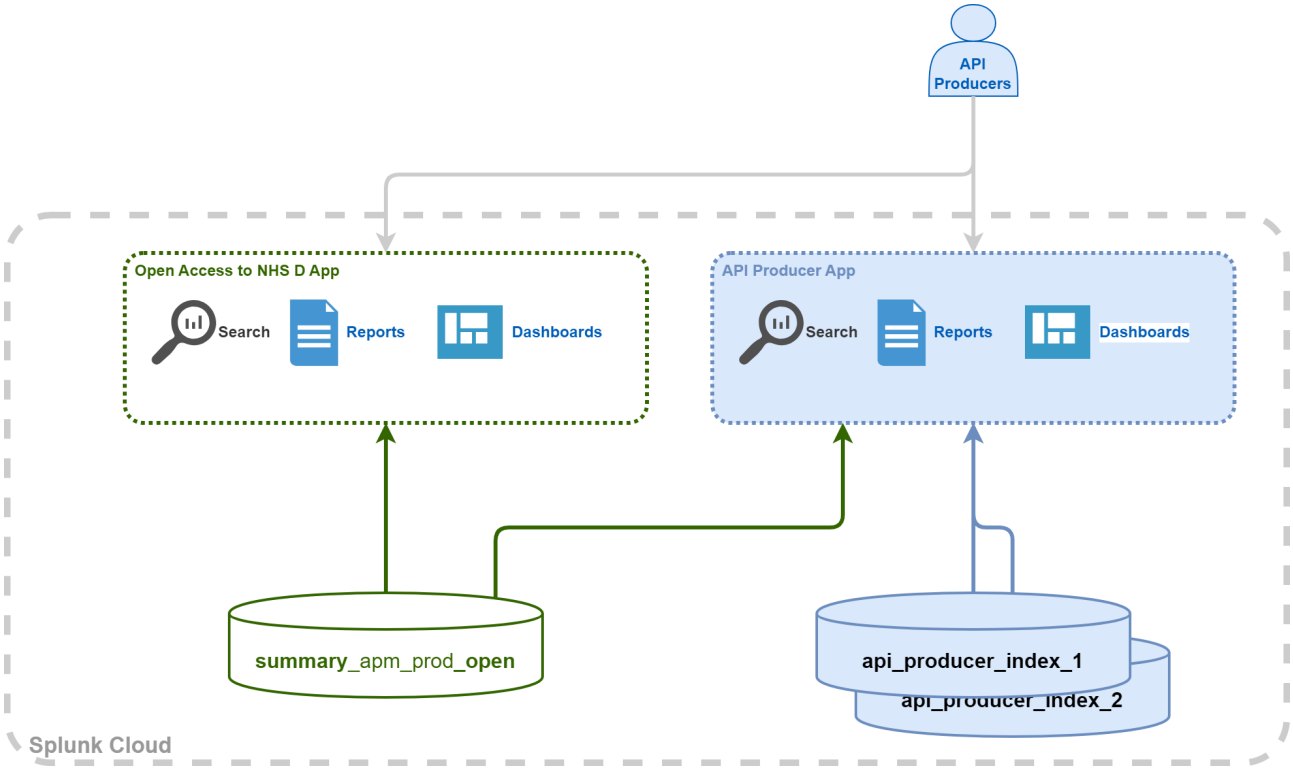
The dashboard allows the user to search against any time range and defaults to 'Month to date'. Below is a list of the searches displayed.

- Total Requests
- Requests Count Breakdown by API (Total number)
- Requests Count Breakdown by API (Graph)
- Total Applications
- Requests Count Breakdown by Application (Total number)
- Requests Count Breakdown by Application (Graph)
- % Availability
- % Successful Requests (< 500)

Building your own reports and dashboards

APIM provides some standard dashboards and reports, that you can use as an API Producer. Some APIs won't need anything else, however, if you are an API Producer that is creating your own backend - is likely to want to build out their own API Producer dashboards linking to the APIM indexes and your own.

The API Producer model agreed with the Splunk Team, ITOC Pipelines and APIM looks like:



This model allows your teams to:

- freely manage and update your reports and dashboards, without involving APIM
- Is something your team is likely have done later, the APIM situation just might bring that forward
- Allows APIM Splunk AD groups to be kept separate from your API Producer Splunk groups

When you are ready to build out your own Splunk Dashboards (even if you **don't** have indexes) you should:

- Contact the ITOC Pipelines team: itoc.pipeline@nhs.net[1065]
- Ask for a new Splunk App
- They will take you through a process to create the App, and any new indexes, and create one or more Splunk AD Groups
    - e.g. in the case of NRL this was: GRP-SPLUNK-CLOUD-RECORDLOCATOR-NONPROD-USER
- You will need to specify one or more Data Custodians, who are the ones that validate all requests for access to **your** AD Groups. You can read more about it here: Splunk User-Access-Requests[1066]
- Important - your AD groups **must not** allow access to APIM AD Groups. It will be impossible for all parties to manage
- Once done your team can edit and manage it's own reporting

Once this is done, your team should:

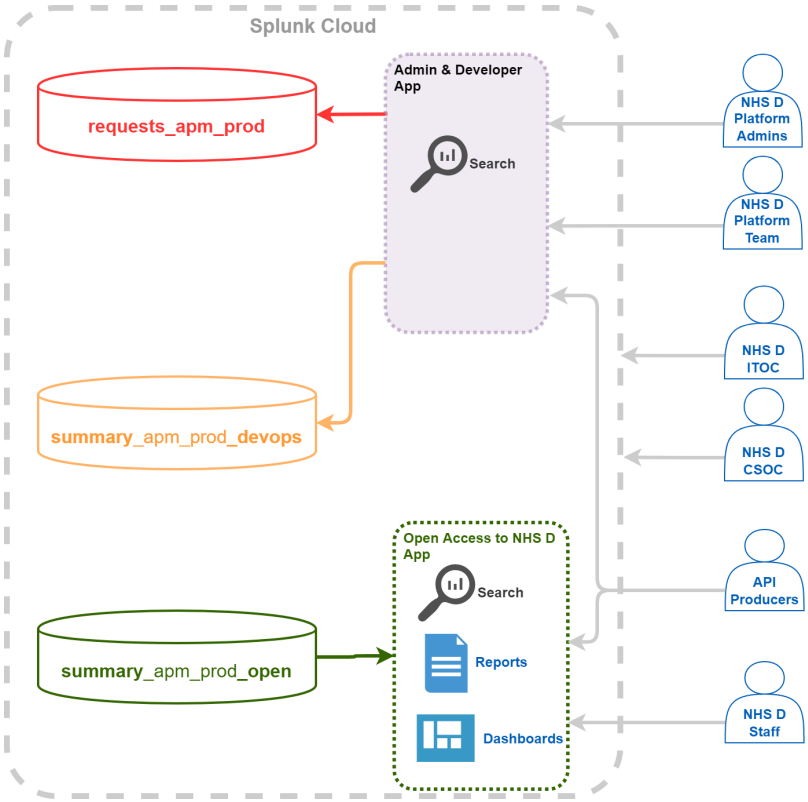
- Read more about how the Splunk team works and the setup:

1065 mailto:itoc.pipeline@nhs.net
1066 https://hscic365.sharepoint.com/sites/NMS/SitePages/User-Access-Requests.aspx

    - Getting data in (sharepoint.com)[1067]
    - Data Onboarding Process Explained[1068]
- Determine if you need more Splunk AD Groups, for instance, some indexes might have patient data (PII) in and only certain team members can view them
- Create a Key Operating Procedure (KOP, or similar) for new starters that:
    - Lists what API Producer Splunk AD Groups are needed for what roles
    - Lists what **APIM** Splunk AD Groups they should request from API Management

Splunk indexes

The LogToSplunk shared flow on your API proxy, causes your API transactions to be logged to three different Splunk indexes:



There are three "levels" of request indexes, each containing less sensitive data:

- Raw requests: requests_apm_ptl or requests_apm_prod

1067 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhs.sharepoint.com%2Fsites%2Fmsteams_2f8e71%2FSitePages%2FGetting-data-in.aspx&data=05%7C01%7Caubyn.crawford1%40nhs.net%7Ce1cf1204111e4297797408db89dd798c%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638255356531743346%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=zrNDylk5mgvlNke%2BX2CJIqjnRUmhNuDRbBJ9K3wpVR4%3D&reserved=0

1068 https://nhsd-confluence.digital.nhs.uk/spaces/SC/pages/73764904/Data+Onboarding+Process+Explained

- DevOps teams: PII redacted, but with client IPs (and a few other key fields) for incident investigation= summary_apm_ptl_devops, summary_apm_prod_devops
- Open access: Full redacted = summary_apm_ptl_open, summary_apm_prod_open

Example queries:

```
index="summary_apm_prod_open" proxy.suffix!="/_ping" proxy.suffix!="/_status"
index="summary_apm_prod_open" proxy.suffix!="/_ping" proxy.suffix!="/_status"
"proxy.name"="spine-directory-*"
```

For PTL - INT:

```
index="summary_apm_ptl_open" proxy.suffix!="/_ping" proxy.suffix!="/_status"
meta.env=int "proxy.name"="personal-demographics-*"
```

For more details, see:

- Proxy redaction config (see page 1094) for each APIs current settings
- Logging[1069]

Decommissioned indexes

The above index names are current, some teams might still be aware of the older indexes, which from Jan 2023 are going to be stop being populated:

- summary_apm_ptl
- summary_apm_ptl_redact
- summary_apm_prod_redact
- **summary_apm_prod**

All the indexes have 90 days retention so will become empty, however **summary_apm_prod** has a 3 year retention period to allow historic reporting. This **will not** be transferred to summary_apm_prod_open. If you use that index and you think this could cause a problem please contact us.

Splunk Apps for all APIs

Splunk segregates access and with Applications, and APIM has two:

1069 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114033654/Logging#Logging-Indexes

- APIM Dev Ops[1070] - primarily for DevOps teams and API Producers
- APIM Open Access[1071] - open access to any user of Splunk, and they can look at redacted indexes

How to add logging to your proxy

Add the LogToSplunk shared flow to your API Proxy

If you are creating a proxy from the template repo, this **is included by default**. Otherwise you use a FlowCallout policy[1072] in the `PostClientFlow` of a Proxy Endpoint.

`policies/FlowCallout.LogToSplunk.xml` file:

policies/FlowCallout.LogToSplunk.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FlowCallout async="false" continueOnError="false" enabled="true" name="FlowCallout.L
ogToSplunk">
    <SharedFlowBundle>LogToSplunk</SharedFlowBundle>
</FlowCallout>
```

`proxies/default.xml` file:

proxies/default.xml

```
<ProxyEndpoint name="default">
    ...
    <PostClientFlow>
        <Response>
            <Step>
                <Name>FlowCallout.LogToSplunk</Name>
            </Step>
        </Response>
    </PostClientFlow>
</ProxyEndpoint>
```

Making your logs visible and requesting sensitive data redaction

1070 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/welcome
1071 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/welcome
1072 https://docs.apigee.com/api-platform/reference/policies/flow-callout-policy

In order for your Production API traffic to be visible in Splunk (including the open-access dashboards), you need to request that we populate the summary indexes for your proxy. This allows your API transactions to flow from the "full data" indexes to the "redacted data" indexes, as explained above.

There are the following levels:

1. Non-sensitive
2. Sensitive

There are currently 3 indexes that can be considered sensitive or non-sensitive. These are parameters, headers and path.

> Every API **must** follow this process - even if it is 'non-sensitive' - the logging process removes innate sensitive data from API call logs (such as client IP) which must always be removed from open indexes regardless of proxy logic. For more info see Logging[1073].

The redaction process has the following features:

- Parameters:
  - Operates on an allow list, so you tell us the safe parameters that won't be redacted
  - The log will show **only** the parameters that have arrived with each request, and if the parameter is not on the allow list the value will be shown as "*******"
- Custom Headers
  - By default, headers are logged to the summary indexes. You can request custom headers to be logged to redacted indexes
  - Operates on an allow list similiar to parameters. This way you can sepcify any headers that won't be redacted
  - If the headers are not on the allow list the value will be shown as "*******"
  - This is on a per API basis. If you require custom headers on multiple APIs, this will require multiple requests
- Path:
  - Proxy Name and Proxy Basepath are always included
  - For (3) the rest of the path is simply not sent over - looking to add pattern matching to support redaction
- Summary indexes (summary_apm_*_devops and summary_apm_*_open) are live/real-time.

1073 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114033654/Logging

Common compliance statement: there is always a risk of the wrong values being sent to NHS Digital and PII could end up in an unexpected place and then be seen in (for instance summary_apm_prod_open). The way the log events are redacted is designed to reduce the exposure of any fields this could happen to. If an issue is spotted, report this to APIM and those records can be removed on request.

Requesting Log redaction/configuration

To request logging configuration for you API please use the APIM Request Log Redactions[1074] workflow in the Service Now.

After the Platforms team have confirmed the changes have been made and released please make sure **you** check that this is working in path-to-live environments by checking for data in the redacted indexes summary_apm_ptl_devops and summary_apm_ptl_open, some sample queries can be found below.

Viewing Logs

Permissions for accessing Splunk logs

To access Splunk, see Get access to Azure AD groups and Splunk (see page 161).

No API Producer is given access to APIM PII / PID data (i.e. requests_apm_prod), for the following reasons:

- You get all the non PID log data in the redacted indexes - no request is missed
- As an API Producer you should have access to the any PID data in your own backend logs
- In the APIM platform there should be very little logic that relies directly on PID, and if you are using ProxyGen this isn't in the current design
- The APIM logs contain data from **all** the APIs, any access would require granting you access to all API PII / PID data - this is not following minimising data access security principles (and is again why the redacted indexes were introduced)

NOTE: there is a slight complication where you use the APIM AWS Hosted Containers, we don't have a redaction process for that. If your API needs this type of access then a new pattern needs to be agreed.

Examples

1074 https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_cat_item&sys_id=350b2cd71bd4565078ac4337b04bcbb9&sysparm_category=7496f0c81b1c9610772fa756b04bcbf7

Use Splunk search https://nhsdigital.splunkcloud.com/en-GB/app/search/search

Example: Requests to PDS API, Internal Dev environment:

```
index="requests_apm_ptl" meta.env="internal-dev" proxy.name="personal-demographics-*"
```

Example: Requests to PDS API, Internal Dev environment, Specific Proxy Revision

```
index="requests_apm_ptl" host="apigee:secure" source="apigee:nhsd-nonprod:internal-
dev:personal-demographics-internal-dev:2"
```

Example: Requests to PDS API, Prod environment redacted data:

```
index="summary_apm_prod_open" proxy.suffix!="/_ping" proxy.suffix!="/_status"
 proxy.name="personal-demographics-*"
```

Implementation

The Shared Flow and associated configuration are in the **api-management-infrastructure** repo in GitLab.

- A KVM Called **splunk-cloud** holds the HTTP Event Collector[1075] token and Splunk index configuration values for a given environment
- A Shared Flow **GetSplunkSecrets** retrieves the values from the KVM
    - This is automatically included in all API Proxies using a Pre-Proxy Flow Hook[1076]. (*You do not have to include this yourself.)*
- The Shared Flow **LogToSplunk** is included in the PostClientFlow of your API Proxy, as described above

You cannot use Flow Hooks in Post-Client Flows in Apigee, and you cannot use a KeyValueMapOperations policy in a PostClient Flow. This is the reason for having separate Shared Flows, and only having to include one of them explicitly in your API Proxy.

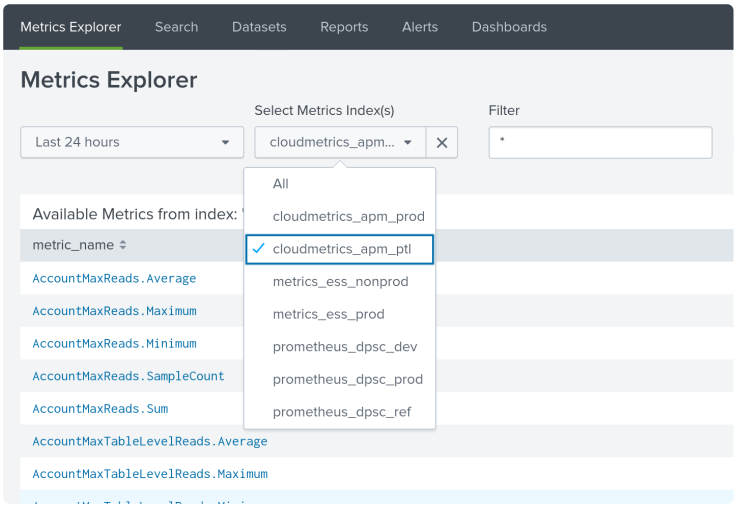
AWS Metrics

1075 https://dev.splunk.com/enterprise/docs/dataapps/httpeventcollector/
1076 https://docs.apigee.com/api-platform/fundamentals/flow-hooks

Splunk also has a metrics feature, metrics work a little differently to standard log events

Metric Discovery

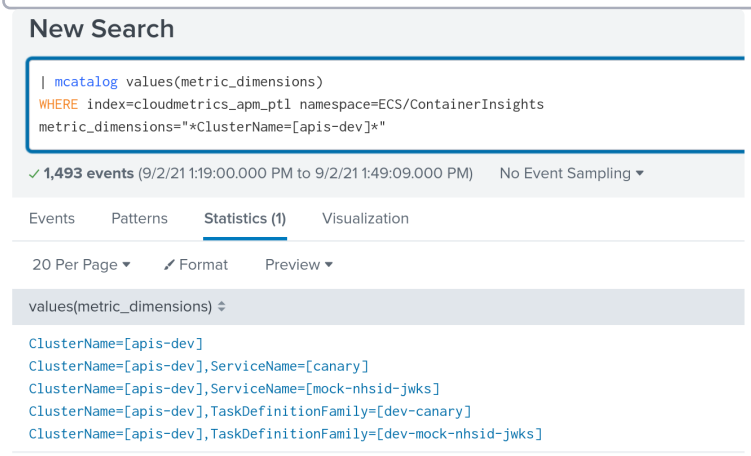
Splunk has a metrics explorer https://nhsdigital.splunkcloud.com/en-US/app/metrics_explorer/explore which is useful to get an idea of what metric names are available to query.



this is using the mcatalog[1077] command, which can be used directly for more advanced queries

Discover available metric dimensions

```
| mcatalog values(metric_dimensions)
WHERE index=cloudmetrics_apm_ptl namespace=ECS/ContainerInsights
metric_dimensions="*ClusterName=[apis-dev]*"
```

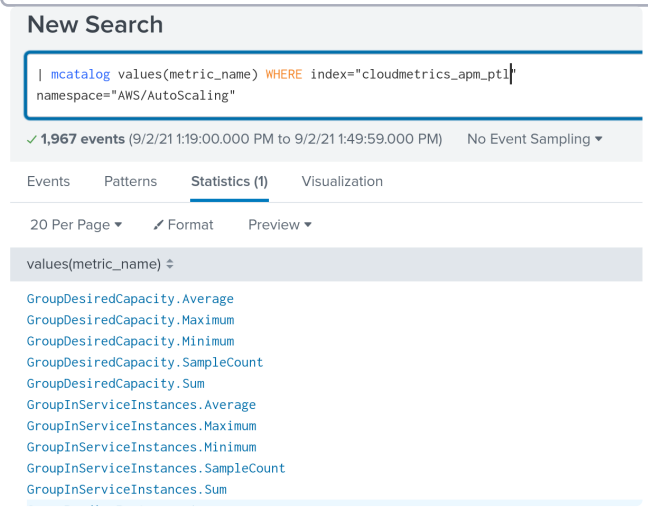


1077 https://docs.splunk.com/Documentation/Splunk/8.2.2/SearchReference/Mcatalog

Discover available metrics

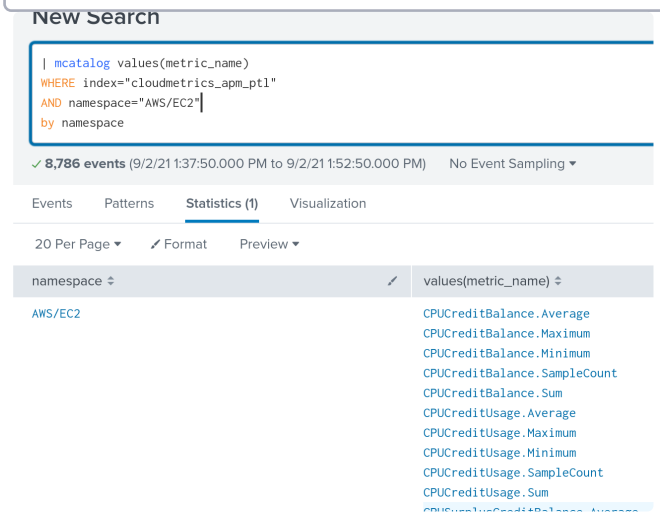
get metric names

```
| mcatalog values(metric_name) WHERE index="cloudmetrics_apm_ptl" namespace="AWS/
AutoScaling"
```



get metric names grouped by namespace

```
| mcatalog values(metric_name) WHERE index="cloudmetrics_apm_ptl" by namespace
```

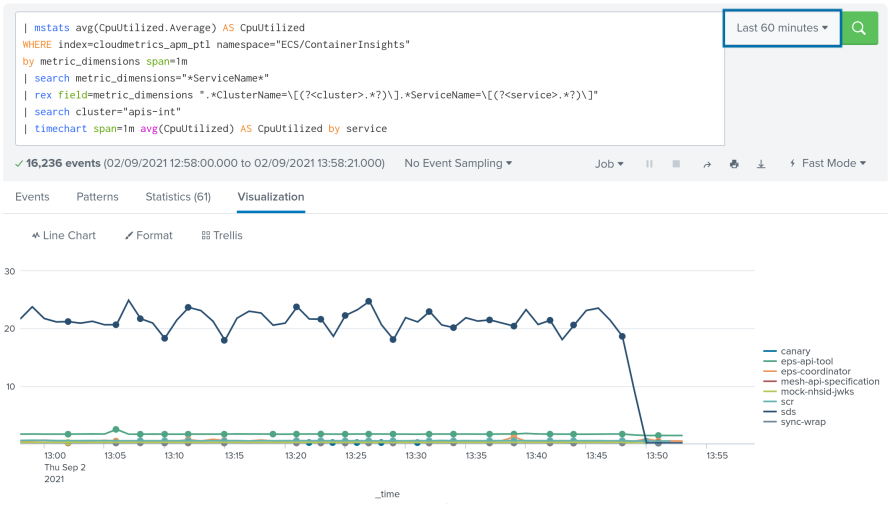


Querying Metrics

querying metrics is done via the mstats[1078] command this will create an aggr

1078 https://docs.splunk.com/Documentation/Splunk/8.2.1/Metrics/Search

```
| mstats avg(CpuUtilized.Average) AS CpuUtilized
WHERE index=cloudmetrics_apm_ptl namespace="ECS/ContainerInsights"
by metric_dimensions span=1m
| search metric_dimensions="*ServiceName*"
| rex field=metric_dimensions ".*ClusterName=\[(?<cluster>.*?)\].*ServiceName=\[(?
<service>.*?)\]"
| search cluster="apis-int"
| timechart span=1m avg(CpuUtilized) AS CpuUtilized by service
```



Proxy redaction config

This page shows the level of redaction for each API currently implemeneted on our Splunk indexes

Splunk redaction config

API	Redaction level/Type of search	Safe/allowed params
address-quality-service-api	non-sensitive	*
ambulance-analytics	non-sensitive	*
booking-and-referral	sensitive params	request.header.nhsd-target-identifier

canary-api	sensitive params	general
connecting-party-manager	non-sensitive	*
communications-manager	non-sensitive	*
covid-19-test-result	sensitive params	general
covid-medical-exemptions	sensitive params	iss questionnaire
custom-prescription-status-update	non-sensitive	*
e-referrals-service-api	sensitive path + params	schedule.actor%3AHealthcareService appointmentType _count page _include
e-referrals-service-api-fix		
e-referrals-service-api-alpha		
e-referrals-service-api-rc		
e-referrals-service-patient-care-api	sensitive params	category _revinclude
e-referrals-service-patient-care-api-alpha		
e-referrals-service-patient-care-api-rc		
electronic-prescription-service-api	sensitive params	authored-on focus%3Aidentifier owner%3Aidentifier business-status status group-identifier prescription%3Aidentifier when-handed-over
eps-api-tool	sensitive params	general

fhir-converter	sensitive params	general
fhir-dispensing	non-sensitive	*
fhir-prescribing	non-sensitive	*
gp-connect-access-record-fhir-api	sensitive params	general
gp-registrations-management-information	sensitive params	general ***Has this been reviewed? all the Endpoints are POST, so there are no parameters - this should be non-sensitive***
healthcare-worker-api	sensitive params	
health-research-data-catalogue	non-sensitive	general
identity-service	sensitive params	response_type iss redirect_uri scope nonce prompt grant_type token_type
identity-service-mock	sensitive params	response_type iss redirect_uri scope nonce prompt grant_type token_type
immunisation-history	sensitive params	general
immunisation-fhir-api	sensitive path + params	general
key-locator-api	sensitive params	general
mock-nhsid-jwks	sensitive params	general
nhs-app	sensitive params	general
nhs-website-content-api	non-sensitive	*

nrl-consumer-api	sensitive params	type, category
nrl-producer-api	sensitive params	type, category
organisation-reference-data	sensitive params	general
patient-care-aggregator-api	sensitive params	NONE
patient-care-aggregator-reporting	non-sensitive	*
patient-data-manager	sensitive path + params	general
personal-demographics	sensitive path + params	general
prescription-status-update	sensitive params	odscode
reasonable-adjustment-flag	sensitive params	general
register-with-a-gp	non-sensitive	*
service-search-api	sensitive params	general
signing-service	non-sensitive	*
spine-directory-service	non-sensitive	*
splunk-monitor	sensitive params	general
summary-care-record	sensitive params	type _sort _count composition.identifier
sync-wrap	sensitive path + params	NONE

General params list

```
"_exact-match", "_fuzzy-match", "_history", "_max-results", "general-
practitioner", "_include", "procedure-code%3Abelow", "code", "authored-on",
"focus%3Aidentifier", "owner%3Aidentifier", "business-status", "status", "group-
identifier", "prescription%3Aidentifier", "when-handed-over",
"schedule.actor%3AHealthcareService", "appointmentType", "_count", "page"
```

7.15.4.5.2.5 Troubleshooting service callouts

Overview

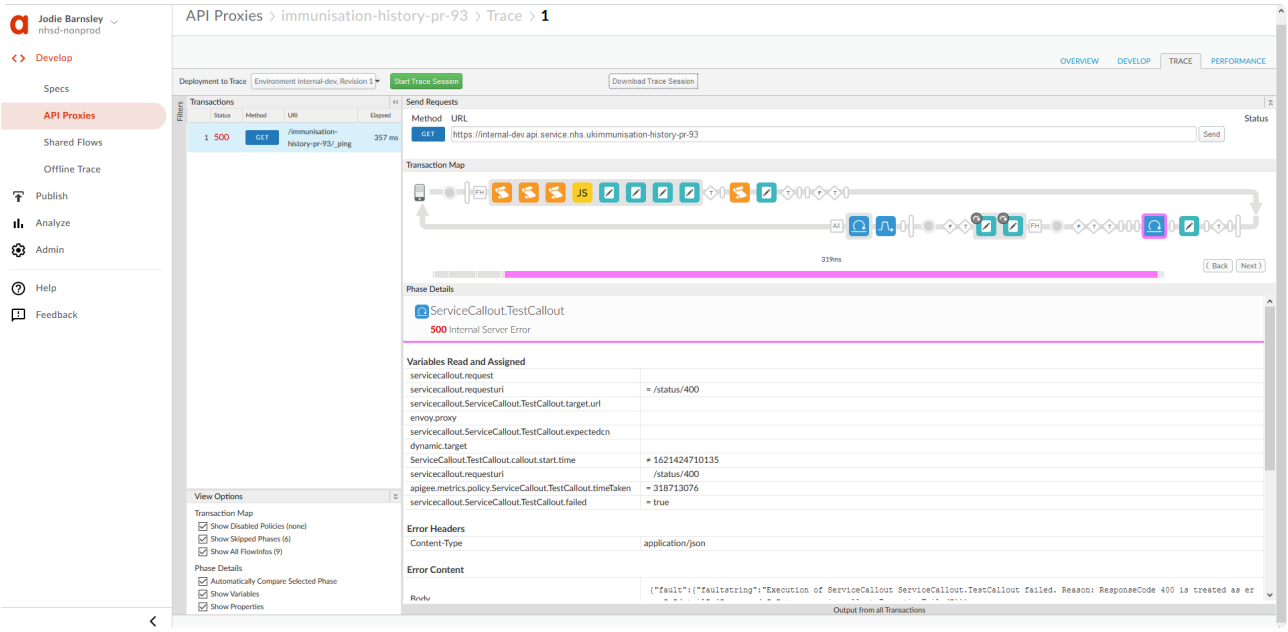
When troubleshooting 'ServiceCallout' Apigee policies in a proxy flow there can be different behaviours depending on whether the call out is happening in the main 'Flow' or 'Post Client Flow'. As a means of researching this, the examples below were created by implementing a test ServiceCallout policy, which calls out to 'https://httpbin.org/status/400' (see page 1098) to get an intentional error. The policy was added to both the 'Flow' and the 'Post Client Flow' to capture what debugging 4xxs and 5xxs in Apigee and Splunk would look like in both circumstances.

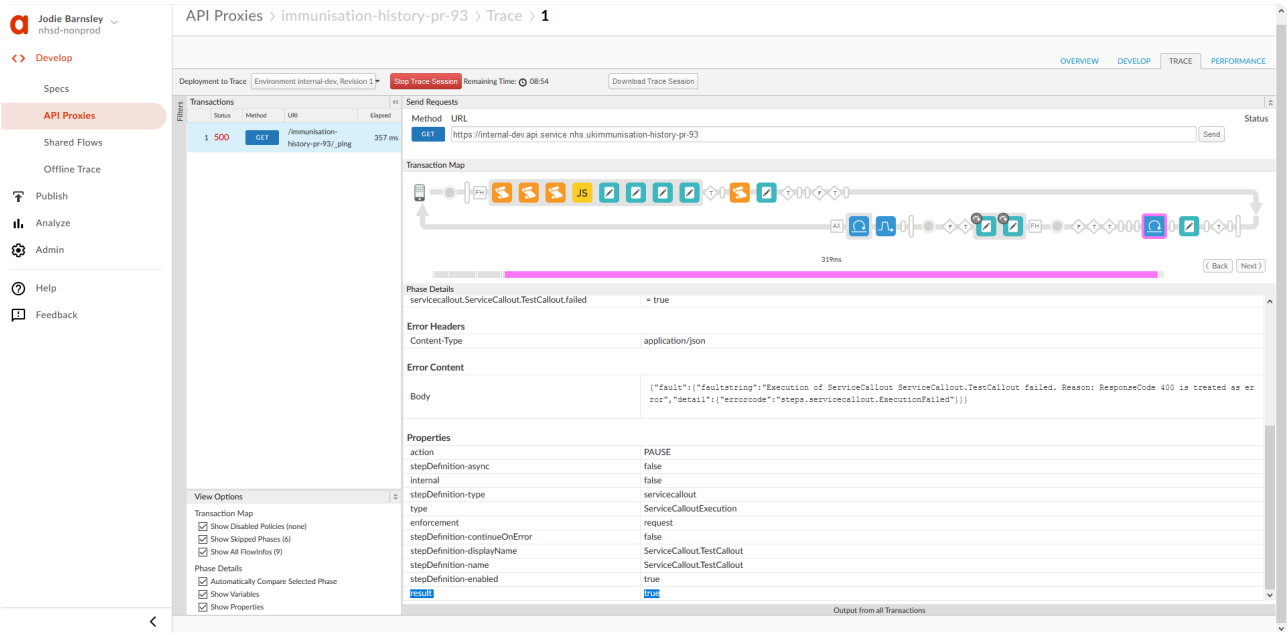
Flow

The test ServiceCallout was added to follow the 'AddPayloadToPing' policy.

Apigee

Tracing a request produced the following:





Things to note:

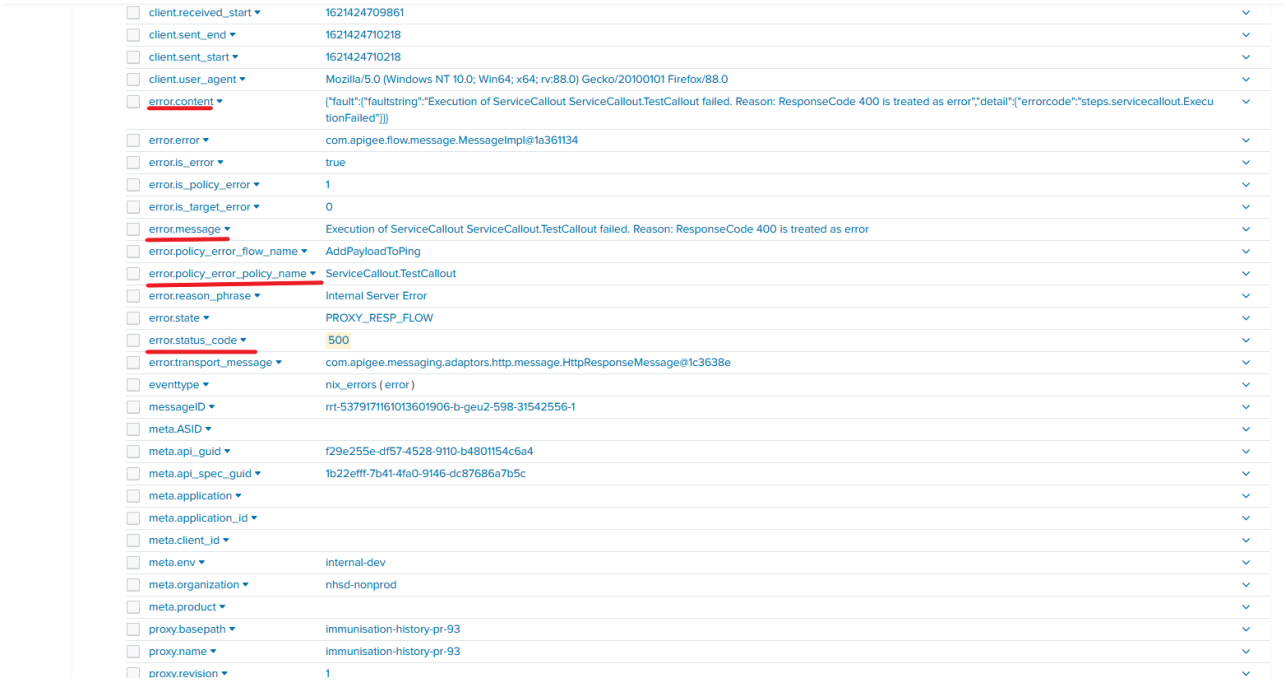
- The error in trace flagged as a status 500 as error handling wasn't implemented for this research.
- The Error Content shows the useful error message in the body, helpful for debugging.
- Under properties 'result' always equals true - so can't be used to help diagnose an issue

Click here for a download of the trace[1079]

Splunk

1079 https://nhsd-confluence.digital.nhs.uk/download/attachments/216178246/Flow-trace-1621424713269.xml?api=v2&modificationDate=1621432107846&version=1

Then located the call in Splunk:



client.received_start	1621424709861
client.sent_end	1621424710218
client.sent_start	1621424710218
client.user_agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:88.0) Gecko/20100101 Firefox/88.0
error.content	{"fault":{"faultstring":"Execution of ServiceCallout ServiceCallout.TestCallout failed. Reason: ResponseCode 400 is treated as error","detail":{"errorcode":"steps.servicecallout.ExecutionFailed"}}}
error.error	com.apigee.flow.message.MessageImpl@1a361134
error.is_error	true
error.is_policy_error	1
error.is_target_error	0
error.message	Execution of ServiceCallout ServiceCallout.TestCallout failed. Reason: ResponseCode 400 is treated as error
error.policy_error_flow_name	AddPayloadToPing
error.policy_error_policy_name	ServiceCallout.TestCallout
error.reason_phrase	Internal Server Error
error.state	PROXY_RESP_FLOW
error.status_code	500
error.transport_message	com.apigee.messaging.adaptors.http.message.HttpResponseMessage@1c3638e
eventtype	nix_errors (error)
messageID	rrt-5379171161013601906-b-geu2-598-31542556-1
meta.ASID	
meta.api_guid	f29e255e-df57-4528-9110-b4801154c6a4
meta.api_spec_guid	1b22efff-7b41-4fa0-9146-dc87686a7b5c
meta.application	
meta.application_id	
meta.client_id	
meta.env	internal-dev
meta.organization	nhsd-nonprod
meta.product	
proxy.basepath	immunisation-history-pr-93
proxy.name	immunisation-history-pr-93
proxy.revision	1

Things to note:

- Reflects Apigee tracing.
- Includes 500 error_code, policy_name and error message content.

Click here for a JSON export of the Splunk event[1080]

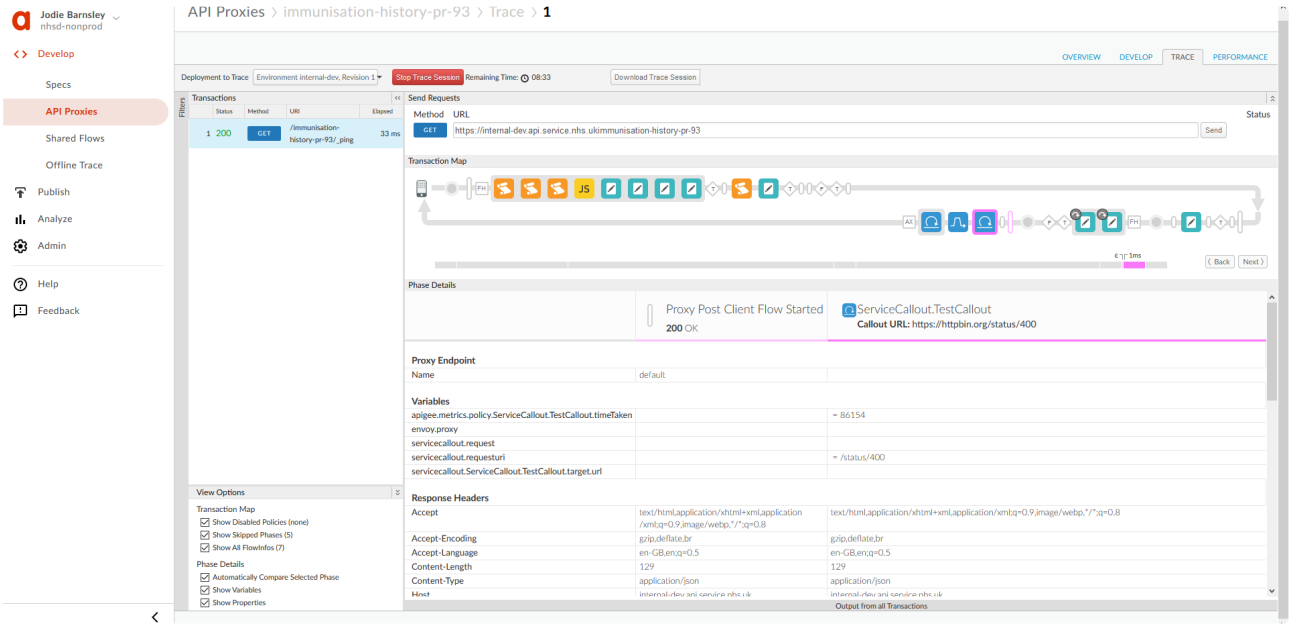
Post Client Flow

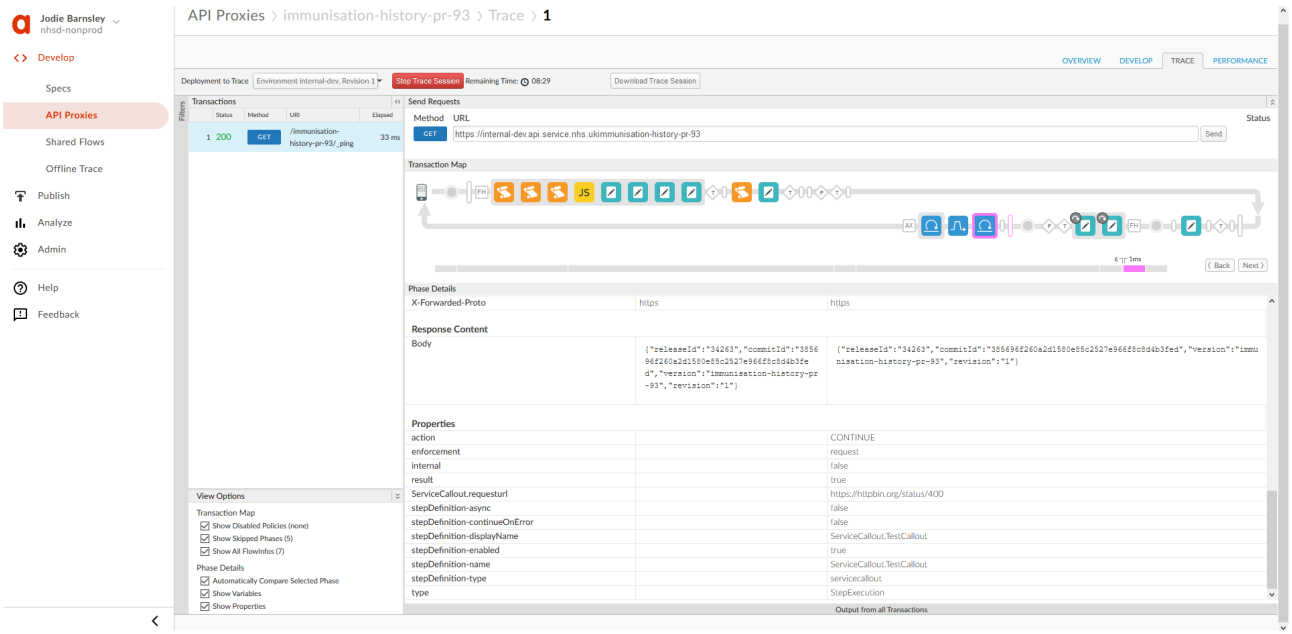
Next the policy was added to the 'Post Client Flow', just before the 'ServiceCallout', which logs to Splunk.

Apigee

The trace returns the following:

1080 https://nhsd-confluence.digital.nhs.uk/download/attachments/216178246/Splunk-flow.json?api=v2&modificationDate=1621432066839&version=1



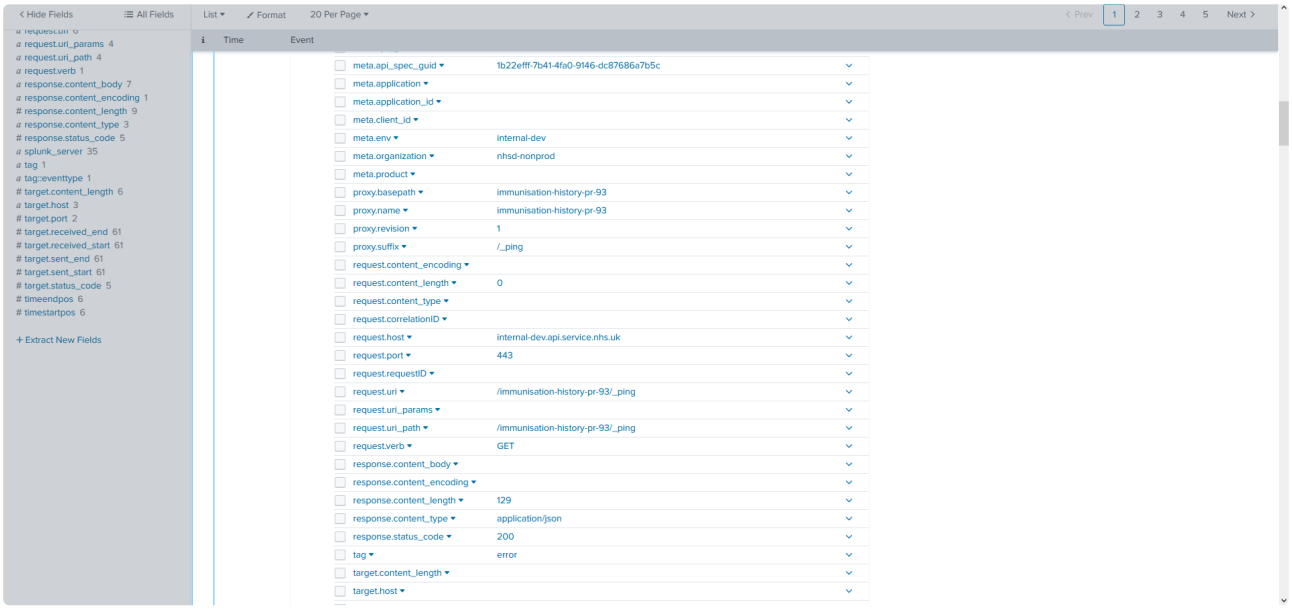


Things to note:

- This shows up as status 200 with the 'result' property again as true but with nothing useful in trace to indicate problem.
- As this is part of the Post Client Flow, the request will never appear as failed to the client application.
- Though there was an error from httpbin, there is nothing in Apigee Trace:
    - To identify that httpbin failed to connect to the target
    - httpbin did not return a 200

Click here for a download of the trace.[1081]

Splunk



Things to note:

- The Splunk instance has tag=error but nothing else to indicate a problem.

Click here for a JSON export of the Splunk event.[1082]

Conclusion

It's currently difficult to detect errors on the 'ServiceCallout' policy in the 'Post Client Flow' using Apigee's trace or in Splunk. There may be a solution and this will be discussed internally.

7.15.4.5.3 Using nhsd-git-secrets tool pre-commit hook on your repo

7.15.4.5.3.1 Overview

It is advised that you use the nhsd-git-secrets tool from https://github.com/NHSDigital/software-engineering-quality-framework/tree/main/tools/nhsd-git-secrets on your repo.

This tool allows you to:

1. Remove any secrets that may have been checked into the codebase in the past
2. Prevent any new secrets from making it into the codebase

1081 https://nhsd-confluence.digital.nhs.uk/download/attachments/216178246/PostFlow-trace-1621425165099.xml?api=v2&modificationDate=1621432096023&version=1

1082 https://nhsd-confluence.digital.nhs.uk/download/attachments/216178246/Splunk-postflow.json?api=v2&modificationDate=1621432095738&version=1

7.15.4.5.3.2 Setup

The following instructions are for Mac or Windows Subsystem for Linux (WSL).

Make sure to copy the nhsd-git-secrets folder into the root of the project repository and then navigate the terminal to the repo root:

```
cd nhsd-git-secrets
cp git-secrets ..
cp .gitallowed-base ../.gitallowed
```

Then if you don't have an existing .pre-commit-config.yaml in the root of your repo:

```
cp .pre-commit-config.example.yaml ../.pre-commit-config.yaml
```

Otherwise integrate the git-secrets example config into your existing file.

Then:

```
cd ..
pre-commit install
git add .pre-commit-config.yaml
```

Next time you do a commit, the git secrets hook should be invoked.

Using WSL you must run `dos2unix` on some of the files for this to run correctly.

7.15.4.5.3.3 Custom configuration per repository

You can add custom configuration to your own proxy with the below:

- Add individual regex expressions to nhsd-rules.txt
- Add regex rules files within wrapper.sh e.g.

```
git secrets --add-provider -- cat git-secrets/nhsd-rules-linux-mac.txt
```

- Add file/dir excludes within .gitallowed, e.g. `.*terraform.tfstate.*:*`

- Control full scan vs staged files scan within wrapper.sh by commenting/uncommenting the mode to run e.g.:

```
# Just scan the files changed in this commit
git secrets --pre_commit_hook

# Scan all files within this repo for this commit
git secrets --scan
```

7.15.4.5.3.4 Testing

- First make sure you have run `git add` if you have changed anything within git-secrets
- Run: `pre-commit run git-secrets`

7.15.4.5.4 Supporting API consumers with tracing unique identifiers

7.15.4.5.4.1 Overview

Tracing requests (including aiding in debugging situations) becomes complex very quickly. To support this APIM has settled on re-using industry wide practices and the FHIR standard for the following two headers (optional) for the API Consumer. These two support multiple use cases, reducing the total number of headers supported:

> **Note**
>
> - To support API Consumers, every API should accept these headers - but you **don't** have to fully implement them
> - If your API is Spine based these headers *may* be mandated by IG requirements, so your API will need to make them mandatory

Header	Definition	Use cases
X-Request-ID	Unique request identifier, sent by the consumer, in the format of a GUID, The subsequent times this is sent, the entire request will be either: • Rejected • Accepted as a continuation of the original request	• True unique request ID • Message de-duplication / protection against accidental resends • Request continuation • Audit / logging

Header	Definition	Use cases
X-Correlation-ID	Arbitrary string value provided by the API Consumer Tends to be unique, but doesn't have to be Returned, unchanged, in the response	• Allows API Consumers to group together multiple calls across multiple systems • Audit / logging • Equivalent to an API Consumer Transaction ID • Microsoft Observability Practices[1083]

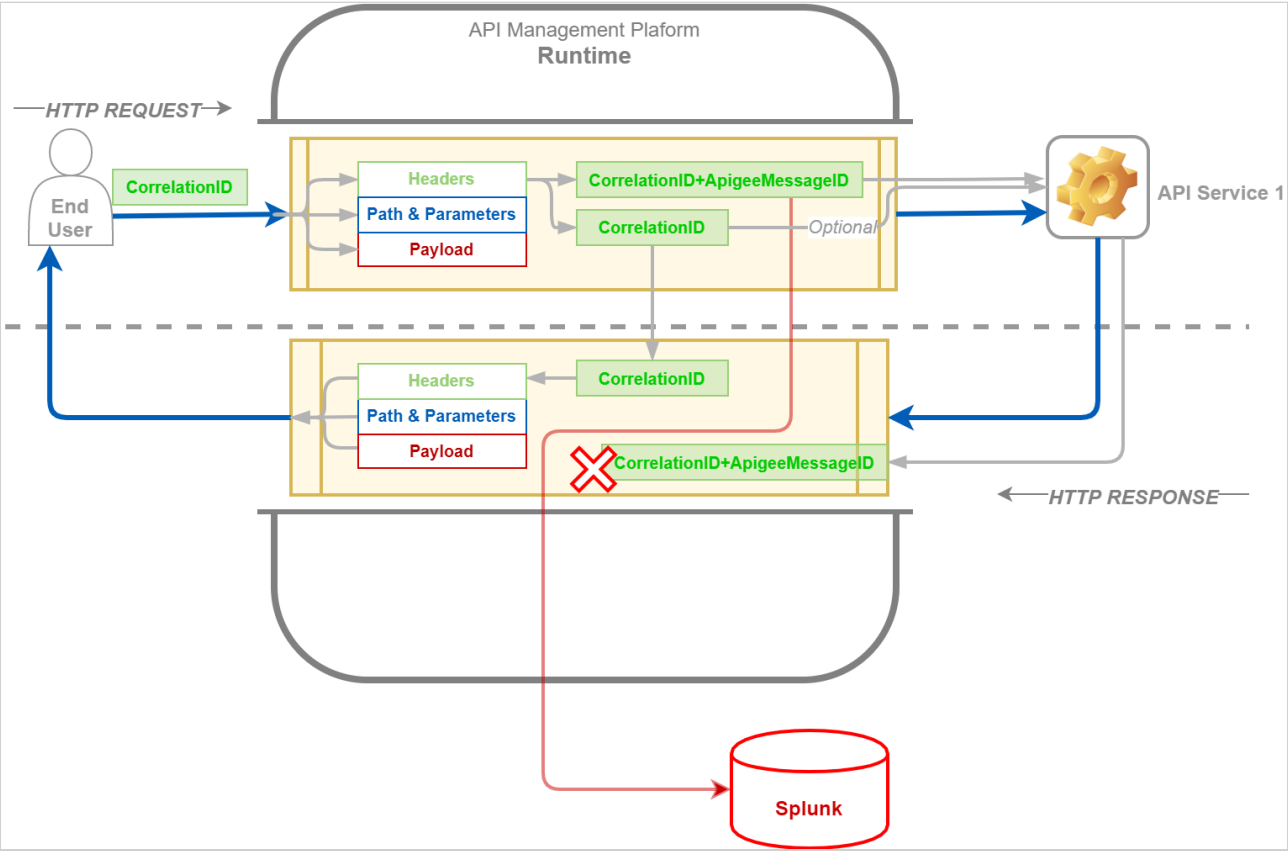
The use of these two headers in HL7 FHIR R4 APIs follows the guidance at https://www.hl7.org/fhir/http.html#custom

The design pattern here is:

- Allow the API Consumer to send either (or both) a "correlation ID" or "request ID" field:
  - It would be logged by Apigee (in Splunk) for every request
- The API Producer should provide the ability for Apigee to pass an arbitrary string that is logged against every request
- API Consumer:
  - Whilst one or both headers are optional across the platform, you as a Producer can mandate their use
  - You, as the API Producer, must not change the definition of the fields
- How to transfer from APIM to the API Backend is down to the API Producer - recommendations below

Visualisation of request and response (for **just** Correlation ID):

1083 https://microsoft.github.io/code-with-engineering-playbook/observability/correlation-id/



7.15.4.5.4.2 API Producer Proxy responsibilities

When configuring your Apigee proxy you should:

Responsibility	Mandatory	Notes
Accept both "X-Request-ID" and "X-Correlation-ID"	Yes	Whilst accepting these values is mandatory, the parameters themselves don't have to be mandatory
Splunk Logging	Yes	The Splunk request from Apigee must include any Correlation and Request IDs provided This is done automatically with the LogToSplunk shared flow, which comes with the proxy template. It is the responsibility of the producer to ensure this functionality stays implemented.

Responsibility	Mandatory	Notes
Pass to the backend for logging	Yes	• This field, at the minimum would contain the Apigee message ID (which is **not** in a GUID format) • It is recommended that the field also contain any client provided Correlation / Request IDs to speed up problem resolution. • Recommend to be placed the header "NHSD-Correlation-ID" ⚠ Caution: When concatenating in this field, you should use: "+", "." or alternatives to "-" as it can clash with GUIDs.
Return to the API Consumer the Correlation ID as is	Part	Only mandatory, if provided by the consumer
Return to the API Consumer the Request ID as is	No	Only need to return if provided by the consumer
Consider pulling values from Payload	No	For lower volume services, and where it does not conflate definitions, you can consider populating these value from the Payload. This should not replace the header(s), but be in addition.
Not enforce the incoming "X-Correlation-ID"	Yes	The incoming X-Correlation-ID is for the API Consumer to provide, for instance, it must **not** be required to be a GUID.

7.15.4.5.4.3 API Producer responsibilities

As an API Producer, we want to make integration easier, **but** having end to end request tracing for debug (and potentially audit) is a critical element to success. To achieve this:

Responsibility	Mandatory	Notes
Don't use "X-" headers	No	• The "X-" prefix was deprecated in 2012 - see RFC 6648[1084] • However, FHIR still uses "X-" prefix, and we are looking to be consistent with FHIR where possible
Accept this in a header called "NHSD-Correlation-ID"	Yes	Accept and log this value
Response has correlation id	No	Currently, Apigee does not use the correlation id in the response - however, future use cases could well arise. By implementing this now in the API backend it will make the work more future proof
"NHSD-Request-ID"	No	This contains the Request ID passed in by the API Consumer. Your backend should respond in one of the following ways: • Reject the request if this value matches a previous value • Use it as a unique identifier for a previous operation To do this you must implement a storage mechanism for this, Spine stores this value for 56 days If you don't want to add this reject / continue functionally to your API backend, only use the Correlation ID

7.15.4.5.4.4 Further reading

To see the background on the decision process see here[1085].

1084 https://tools.ietf.org/html/rfc6648
1085 https://nhsd-confluence.digital.nhs.uk/display/APM/D061+-+Supporting+API+consumers+with+tracing+unique+identifiers

7.15.4.5.5 Update locking with ETag

7.15.4.5.5.1 Overview

An ETag is an identifier commonly used to prevent simultaneous updates of a resource from overwriting each other. This can be useful for APIs with update (PUT) functionality.

7.15.4.5.5.2 Does my API need to use ETag Headers?

If your API supports updates you almost certainly need to implement some form of record locking to ensure information is updated consistently. This can be handled in different ways depending on how the backend of your service is designed. The ETag HTTP header introduced in IETF[1086] proposes an approach for combining conditional requests, which assert that the record to be updated represents the most up to date version known to the producing system.

The approach adopts an Optimistic Record Locking pattern, which helps to ensure consistency during concurrency. Essentially, Version IDs are used to control updates to records for requests coming from multiple consumers. The approach ensures updates are only made against the latest version of the record by referencing the Version ID as part of the update process between a consuming client and a producing system, where the record is mastered.

- If the Version ID included by the client on update doesn't match the Version ID for the record on the server then the update is rejected.
- The benefit of Optimistic Locking is guaranteed consistency, as all updates meet RACE conditions and are therefore applied to the correct version of the record.

Note that although the discussion and examples below relate to FHIR, the same pattern can be supported by any RESTful API that uses standard HTTP methods such as GET, PUT and PATCH.

Other reasons to use ETags

- Will any of your API Consumers need to cache the data?
- Will the payload be used as-is in another API to update data and therefore need some form of optimistic locking?
- Do you have a version number already in your payload? If so, it is trivial to add it into the ETag header

7.15.4.5.5.3 Optimistic Locking in FHIR

Within FHIR Optimistic Record Locking can be implemented using ETag HTTP Headers and weak validation. FHIR only supports weak validation (see RFC 7232[1087] for discussion including weak and strong validation).

- **Strong Validation**

Stores a hash/digest to support non repudiation of the message content in the message header.

1086 https://tools.ietf.org/html/rfc7232#page-5
1087 https://tools.ietf.org/html/rfc7232#page-5

- **Weak validation**

  Version ID stored in the ETag message header provides a reference to the representation of the record held by the producing API/service.

The specifics of how this implemented within FHIR to manage Resource Contention: see: Http - FHIR v3.0.2 (hl7.org)[1088] - ETag has been supported in some form since DSTU2.

7.15.4.5.5.4 Using ETag Headers in FHIR

When a resource is returned from a FHIR server an ETag header is added to the message. The content of the header is shown below:

Header	Value	Notes
Etag	W/"<Version ID>"	W/ indicates that the validation is using the weak approach and is therefore not a digest of the message or message payload. The Version ID is expected to be in quotes. The value inside the quotes is under the control of the FHIR server. Implementations tend to use incrementing IDs or TimeStamps - the essential requirement being that it is unique for a given resource.

When the consuming system sends an update to the FHIR server, the Version ID that was included with the FHIR resource needs to be included in the "If-Match" header as shown below:

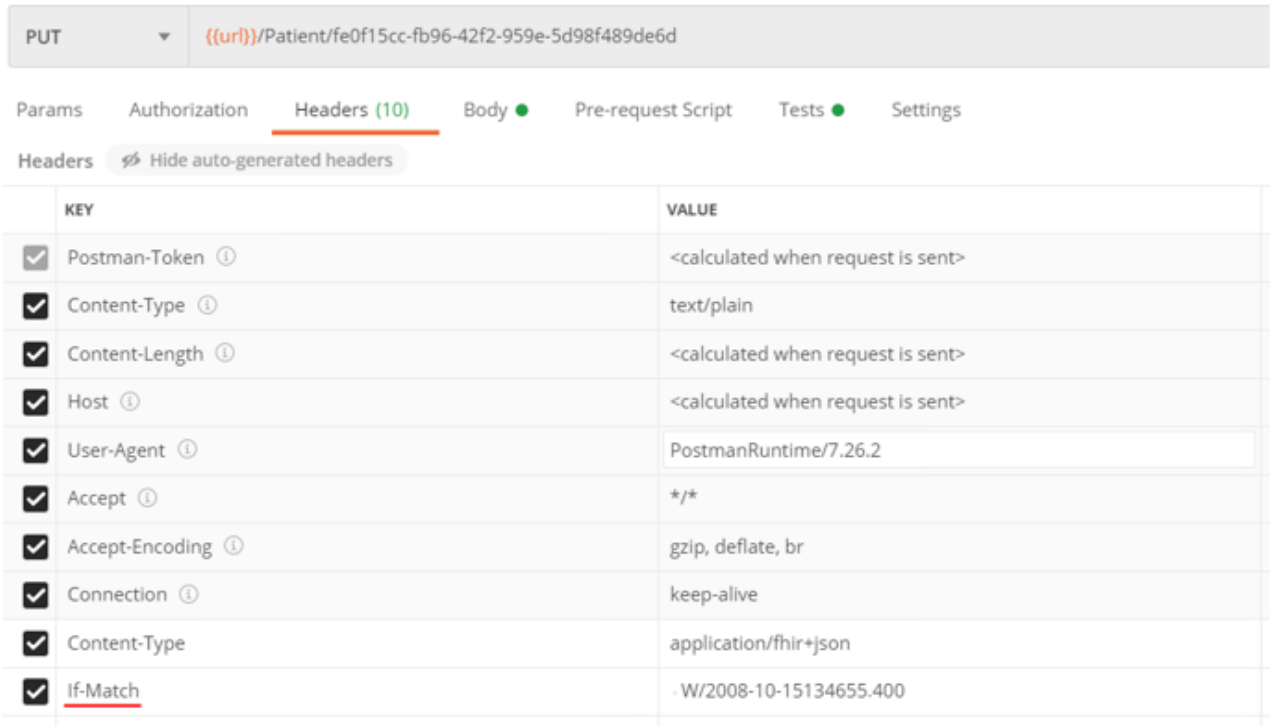
Header	Value	Notes
If-Match	W/<Version ID>	W/ indicates that the validation is using the weak approach and is therefore not a digest of the message or message payload. If the Version ID matches the Version ID of the Resource stored on the FHIR the updated is accepted.

Example

Given a Patient resource is initially returned from the FHIR server with an ETag of **W/"2008-10-15134655.400"** the record to be updated requires the If-Match header to be populated as shown in the capture from Postman below.

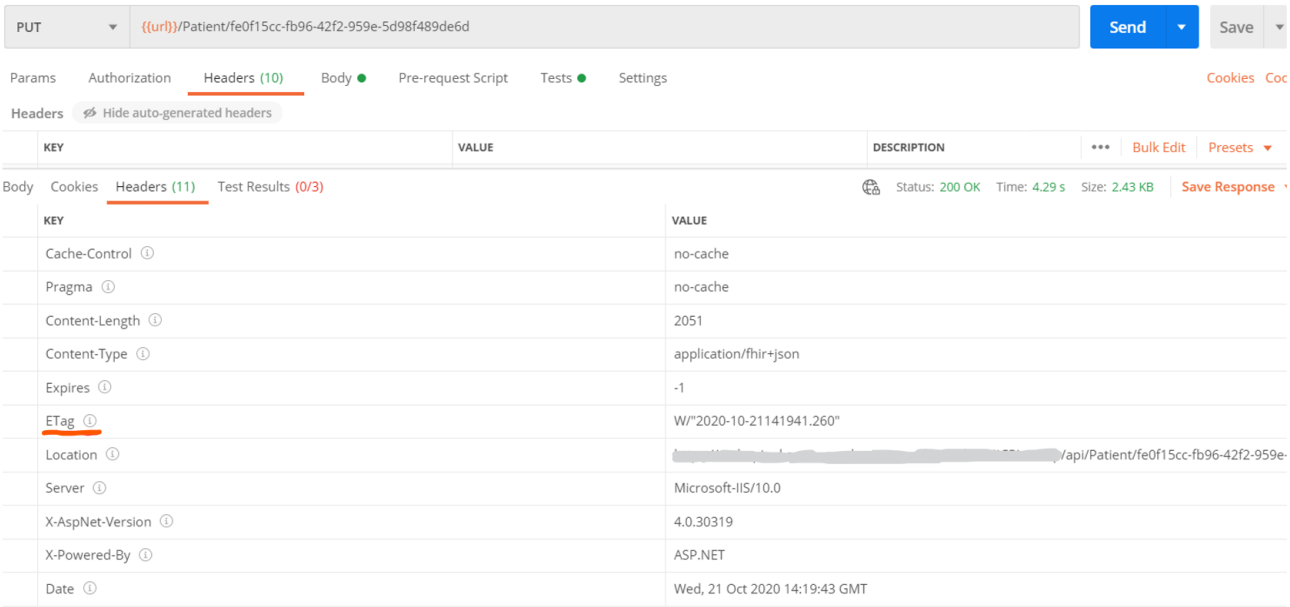
1088 https://www.hl7.org/fhir/STU3/http.html#concurrency

- Note that this implementation uses TimeStamps with millisecond precision but incrementing values or some other unique value can be used.



If the update is successful HTTP status of 200 is returned.

In the example below the FHIR server returns the FHIR resource and updates the Version ID in the ETag header to a new version (using a TimeStamp of **2020-10-21141941.260** for the Version ID).



If the version ID given in the `If-Match` header does not match, the server returns a `412 Precondition Failed` status code instead of updating the resource.

Servers can require that clients provide an `If-Match` header by returning `400 Client Error` status codes when no `If-Match` header is found.

Other response codes are discussed in RFC 7232[1089].

7.15.4.5.6 Caching for your API

7.15.4.5.6.1 Overview

For some APIs caching might improve performance and protect you from outages on your back-end or supporting services. There are two types of caching supported natively by Apigee:

1. GET request caching
2. Service call-out caching

7.15.4.5.6.2 GET request caching

If you have a GET request endpoint where the response is always the same for a given request, you could use GET request caching to reduce the number of calls to your back-end service.

This can protect against a short outage in the back-end service and a cache time-out period can be easily configured.

7.15.4.5.6.3 Service call-out caching

If your API calls out to a supporting service (not your actual back-end service), you could use service call-out caching to reduce the number of calls to the supporting service.

This protects against a short outage in the supporting service and a cache time-out period can also be easily configured.

For example, the API platform uses this in the auth service when calling out to retrieve JWKS public keys - the cache timeout period is 24 hours.

For more details on adding caching for APIs see Apigee's documentation on caching[1090] or contact us (see page 2056).

1089 https://tools.ietf.org/html/rfc7232#page-5
1090 https://docs.apigee.com/api-platform/cache/caching-edge

7.15.4.5.7 Feature toggles

7.15.4.5.7.1 Overview

A feature toggle is used to hide, enable or disable a feature during runtime. For example, during the development process, a developer can enable the feature for testing and disable it for other users allowing teams to modify system behaviour without changing code.

You can set the feature toggle per environment, so a feature might be toggled on in PTL environments but off in production.

There are some steps and considerations to take into place when introducing feature toggles in your API and this guide will tell you how to set them up and consume them.

7.15.4.5.7.2 How they work

Feature toggles are defined as key-value maps (KVMs) within Apigee. These KVM values are read in the pre-flow to your proxy (using the PreProxySharedFlow[1091]) and saved into variables with the following name structure: **apim.feature_toggles.enable-<my_feature_name>** or **apim.feature_toggles.disable-<my_feature_name>** (note the inclusion of the word enable or disable in the name as a good naming practice). Once defined and deployed to Apigee (via our platform infrastruction deployments) we can use this variables in conditional flows to trigger different behaviours in our API.

7.15.4.5.7.3 Creating feature toggles

In order to create a feature toggle, you will need to add it to the infrastructure repo **(**here[1092]**)**. To do so, create a new branch and submit your changes in a pull request. In the file 'feature_toggles.yml' you will be able to set up default values for your feature toggle for different environments (see figure below). Remember to assign only boolean values to your feature toggles, since the value type will be validated on deployment. Once merged and deployed, you will be able to consume the toggle in your API proxy.

feature_toggles.yml

```
# defaults
feature_toggles:
  enable-token-exchange: true

env_feature_toggles:
  prod:
    enable-token-exchange: false
```

1091 https://github.com/NHSDigital/api-management-infrastructure/tree/master/ansible/roles/apigee-shared-flows/templates/PreProxySharedFlow

1092 https://github.com/NHSDigital/api-management-infrastructure/blob/master/ansible/roles/apigee-key-value-maps/vars/main/feature_toggles.yml

7.15.4.5.7.4 Consuming feature toggles

Once the feature toggle is created you will be able to use it in your flow conditions through its corresponding variable name.

In the following example, we can see two different *behaviours* (conditional flows) being controlled by the feature toggle 'apim.feature_toggles.enable-token-exchange'.

feature toggle implementation

```
<Flows>
    <Flow name="isenabled">
        <Description/>
        <Request/>
        <Response>
            <Step>
                <Name>Is-Enabled</Name>
            </Step>
        </Response>
        <Condition>apim.feature_toggles.enable-token-exchange = true</Condition>
    </Flow>
    <Flow name="isnotenabled">
        <Description/>
        <Request/>
        <Response>
            <Step>
                <Name>Is-Not-Enabled</Name>
            </Step>
        </Response>
        <Condition>apim.feature_toggles.enable-token-exchange != true</Condition>
    </Flow>
</Flows>
```

Controlling feature toggles

The control of the state of the feature toggles should be done through the infrastructure repo following the instructions under 'Creating feature toggles'. Please don't use the Apigee UI to change the state of your toggles. This is because when changing the state of a toggle you are changing the release configuration of your proxy and this may need to be coordinated with other release activities.

7.15.4.5.7.5 Testing feature toggles

If you use a feature toggle you should test that it works i.e. that the toggled behaviour is correctly enabled/disabled by the toggle.

7.15.4.6 Engineering tasks for proxy generator

The following articles describe features we recommend implementing in your API beta service (for APIs created using proxy generator) ahead of your first production deployment.

- Monitoring your API with proxy generator (see page 1115)
- Rate limiting with proxy generator (see page 1117)

7.15.4.6.1 Monitoring your API with proxy generator

Duplicate page of Adding status monitoring to a service (see page 1065), which has more details

7.15.4.6.1.1 Overview

With proxy generator monitoring your API is simplified. We use Splunk for monitoring API traffic, including alerting, analytics and diagnostics. All API requests are logged to Splunk by default unless you toggle this feature off.

We log to Splunk for two reasons:

- Requests to APIs
- stdout for APIM AWS Hosted Containers

You can use Splunk to query either of these to troubleshoot your API. An example query on how to specify the index of your log can be seen below:

```
index="requests_apm_prod" proxy.suffix!="/_ping" proxy.suffix!="/_status"
"proxy.name"="spine-directory-*"
```

7.15.4.6.1.2 API producer team responsibilities

API producer teams are responsible for monitoring the health of their own APIs.

This includes out-of-hours monitoring where appropriate (i.e. if your API is a silver, gold or platinum service).

The API platform is itself a platinum service, and the API platform team does monitor its health 24x7, but we only react to **API platform** issues, not to API-specific issues.

We're an altruistic bunch, so if we do notice an API-specific issue we might raise a live incident, but **don't rely on it**.

7.15.4.6.1.3 Enabling monitoring with proxygen

With proxy generator logging and monitoring is controlled with a single boolean flag in the proxygen OAS spec as shown below:

proxygen-oas.yaml

```
x-nhsd-apim:
  monitoring: false
```

This flag is set to true by default, automatically enables logging to splunk[1093]and allows you to view your API through the Grafana dashboards. It exports some prometheus metrics, which are then used by the Grafana dashboard. It will also set up the blackbox probes and alerts which you can keep track on in the Slack platforms-apim-alerts-prod[1094] channel.

To turn monitoring off, simply change the monitoring value in the OAS specification to false.

7.15.4.6.1.4 Dashboards

Grafana

We have a Grafana dashboard that monitors the health of all our APIs. You can access it at:

- https://grafana.prod.api.platform.nhs.uk/ (currently you will need to request access to GRP-AzureNHSD-API-Management-PROD-DL (see page 161) to be able to view this dashboard)

It also sends Slack alerts for any issues.

Each API has two monitoring endpoints:

- _ping (see page 1073) - "shallow" ping - confirms the API proxy is up and running.
- _status (see page 1075) - "deep" ping - confirms that the back end service is up and running by making a call to it.

For your API to appear on the Grafana dashboard, you need ensure you have both the _ping and _status endpoint. These endpoints are enabled by default within proxy generator.

You can change the dashboard view by selecting the dashboard view next to the 4 squares, at the top left-hand side of the dashboard (currently this defaults to 'General/APIM'). The 'env' option below it, also enables you to change the environment view. The most useful dashboard views for producers are 'api details' and 'api status'.

API management health dashboard

1093 https://nhsdigital.splunkcloud.com/en-GB/app/launcher/home
1094 https://nhsdigital-platforms.slack.com/archives/G012YEJMM5K

Visit the dashboard here[1095]. The dashboard can be filtered by the index, environments, proxies and application to get a more focused view on a particular API, using the drop down menus under the dashboard heading.

Anyone with access to the NHS Digital Splunk Cloud instance can view this dashboard.

The API Management Health dashboard displays details of requests on the platform across environments, proxies and applications. The dashboard **avoids showing any sensitive data** by searching against the redacted index '*summary_apm_prod_open*'

The dashboard shows results from the searches listed below for the last hour, last 24 hours and last 7 days.

- total requests (excluding ping & status)
- % availability
- requests
- request duration
- % successful requests (<500)
- response code by percentage breakdown

7.15.4.6.2 Rate limiting with proxy generator

7.15.4.6.2.1 Overview

Use rate limits to protect your API from over-use, DoS attacks and ensure fair use between consuming applications. Proxy Generator allows rate limits to be set using a snippet in the per-instance specification.

You can use it to enforce two types of rate limiting:

1. Per-application
2. Per-proxy

We recommend you do both, especially because that's how we promote rate limiting to API consumers - see https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits.

If you want to read more about the actual volumes you anticipate see: Performance testing (see page 756)

Relationship between proxies and access modes

Before considering rate limiting strategies, it's important to understand the relationship between two important elements of the API platform: proxies and access modes. Under the hood each access mode is an "Apigee Product". Where you have a consumer who has more than one access mode connected to their application, you might want to talk to Producer Support if you need different rate limits per access mode.

Proxy

Can be looked at as direct representation of the back-end API. All requests are routed via the proxy and one is created per API instance.

1095 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/health?form.index=summary_apm_prod_open&form.proxy=*&form.app=*&form.env=prod

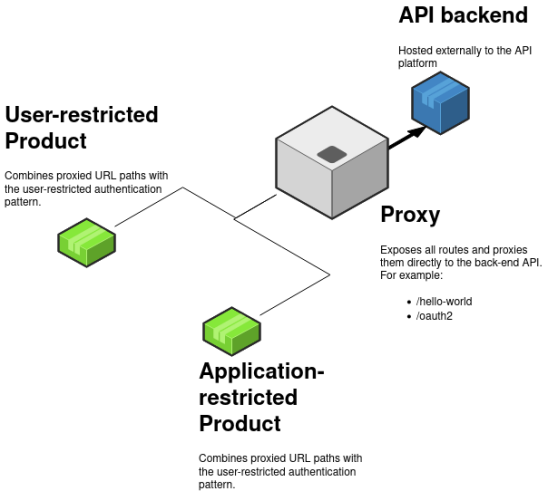
Rate limits applied at a proxy level affect all requests before they are routed to the back-end, so it applies to **all** products.

Application

Each consumer sets up an Apigee application, and a different rate limit. Is how an API is published to a consumer.

Hello World Specification

```
x-nhsd-apim:
  monitoring: false
  access:
  - title: User Restricted Access Mode
    grants:
      nhs-login-p5: []
      nhs-cis2-aal3: []
  - title: Application Restricted Access Mode
    grants:
      app-level0: []
      app-level3: []
  target:
    type: external
    healthcheck: /_status
    url: https://example.external.apikey.secured.backend.com
  ratelimiting:
    proxy:
        timeunit: "second"
        limit: 5
    app_default:
        timeunit: "day"
        limit: 1000
```



Applying rate limiting to your API

It's **strongly** suggested that API producers implement at least a per-second limit, which helps maintain their estimated maximum traffic flow. This will prevent outages during levels of exceptional organic traffic and/or potential DoS attacks.

See the example above (see page 1118)for reference.

Rate limiting can be applied in one, or both, of the following ways:

1. `proxy` - the limit is applied across the entire proxy, regardless of which product or app the requests come through.
2. `app_default` - the limit is applied separately to each app. E.g. 5 requests per second, per app. This limit can be overridden manually on a per-app basis - please contact APIM support via the #platforms-apim-producer-support channel.

Each rate limit value is a dictionary containing the following keys:

- `timeunit` - can be any of the following values:
  - "second" - works out the average number of requests allowed by dividing your defined limit by the number of milliseconds in a second, and ensures that this limit isn't breached. For example, a limit of 5 requests per second will allow one request every 200 milliseconds (1000 / 5). This is the recommended approach for protection against traffic spikes.
  - "minute", "hour", "day" - this approach works like a standard quota limit - rate limiting will apply if the number of requests exceeds the provided limit in any defined time window.
- `limit` - a positive integer which specifies the maximum number of requests over the given time unit.

In the referenced example, the entire proxy has a policy ensuring that the backend limit of 5 requests per second cannot be exceeded. In addition to that, each app is allowed to make up to 1000 requests per day.

7.15.5 First time deploying into production review

This is to:

- Run through the list in the **First time deploying into production activities** section
- Spot any general snags that might affect production
- Discuss if your proxy connects / relies on any Spine based APIs (see page 408)
- Do a security review
- Ensure the service has sufficient test coverage, including what smoke tests you intend to do to ensure the service is running
- Deploy the proxy (at at least agree when it will be if it has to be out of hours)

Attendees:

Producer Team

- Technical lead

API Management

- Engineers, including an engineer from APIM (done by posting in our Slack 'platforms-apim-producer-support' channel requesting for 1 volunteer from deathstar to attend the meeting)
- Architect (optional - unless you have Spine connections)

Pre-requisites:

- You must have completed the 'First time deploying into production activities' contained within API process checklist - Beta (private) and shared a link with the completed answers

What is covered:

- A run through of the first time deploying into production activities
- Discussion around planning the first production release, including coordination of related deployments, e.g. Infrastructure.

After the review:

- Once the review has passed, you will be green-lit and we will work with you to support the roll-out.

To request a review, contact us. - see Help and support (see page 2056).

7.15.6 Deploying to production

> **Depolying to production with proxy generator**
>
> If using proxy generator see Deploying to production with Proxy Generator (see page 1135)

7.15.6.1 Overview

This page describes how to deploy your API into production, either for the first time or subsequently.

You can either do this yourself, which is our preferred approach (because it gives you greater flexibility and control over your releases), or we can deploy on your behalf. The overall principles are the same regardless of which approach you take but there are some differences in configuration and process that we spell out below.

In general the process involves:

- configuring the API proxy for deployments (first time only)
- creating a release plan (includes raising Request For Change)
- scheduling a release

This page describes the process for creating a release plan and raising a Request For Change (RFC) for API proxy-related changes to a live environment and not changes to your API back-end code. You may however, find some of the details useful for raising RFCs in general.

RFCs are currently raised through ServiceNow. Get access to ServiceNow (see page 167). Once the RFC is raised, a release plan is created and then scheduled into a release.

7.15.6.2 Self-serve releases

Deploying to production yourself is simple to configure and allows for flexibility when releasing your API (see Using manual approval on pipelines (see page 916)). By default, production deployments depend on an approval gate with only API Platform admins able to approve and therefore release your API into production, so you will need to configure your proxy code to allow for a bespoke manual approval gate and set up your approvers.

Below are some considerations on whether to deploy to production yourself or via the API Platform team.

7.15.6.2.1 Self-serve deployments

Deploying to production yourself allows you to manage who can release your API and when you can release your API. It keeps the whole release process within your own team and removes the dependency on the API Platform team. Whether you deploy your API to production in working hours or out of hours is down to you and the approvers of your RFC. This should be assessed with the impact of the changes in mind. The API Platform team can still advise on this even if you are releasing the API yourself.

The API Platform team hold a 'release co-ordination' call (organised by Steven Walton) to finalise any details for proxy deployments. When self-releasing it is not a requirement to attend one of these calls before a deployment as it's important for you to take ownership of these deployments but if you need any help or advice for you release this is a good place to go for this. Get in touch with the platform team via the *#platforms-apim-producer-support* channel (see Get access to Slack[1096] for access) to ask for an invite to the Tuesday or Thursday call prior to your release.

1096 https://nhsd-confluence.digital.nhs.uk/display/APM/Get+access+to+Slack

7.15.6.2.2 API Platform team deployments

When it's not possible to release to production yourself, you will need the API Platform team to do the release for you. The default configuration for proxies means production deploys need to be approved by an API Platform admin. This enhanced support may be if you don't have the resources for out of hours deployments when one is needed.

API proxy changes are scheduled every Tuesday and Thursday at 11pm. On these days there is a 'co-ordination' call (organised by Steven Walton) to finalise the release. This should be attended by the person who raised the RFC (or failing that a dev/tech lead from the team). It usually takes 2 business working days for all the approvals to come through via ServiceNow, so it is advisable to create the RFC and plan a few days in advance before you intend to implement the change.

Once the RFC is approved in ServiceNow and the change has been finalised on the co-ordinations call, it will be deployed at the designated time by a member of the API Management team On-Call team.

7.15.6.3 Deploying to production for the first time

If you are releasing into production for the first time:

- You must follow Deploying to production for the first time (see page 1131)
- If you want to carry out your own releases you must be set up for this see: Using manual approval on pipelines (see page 916)

7.15.6.4 Determining what needs an RFC and timing of the release

As an API Producer, almost all changes will require an RFC. However, the timing of a release is also a key consideration - this draft page sets how the API Platform team currently approaches this and might provide useful guidance: Release Management (see page 2077).

7.15.6.5 Creating an RFC

7.15.6.5.1 Before you begin

Release dependencies

Consider: Are there any dependencies for your release?

- Have you or Producer Support created a PR against the infrastructure repo?
- Are you changing secrets at all?
- Are you changing target servers?

If so, ensure you have consulted with and scheduled any dependent releases with the Platform team.

It is important to consider the below whether you are self-releasing or the API Platform team will be releasing your API but for an API Platform team release it is worth creating a release plan at the same time as raising your RFC see the 'Scheduling a release for a Platform team deployment' section of this page for further details.

If you are 'self-releasing' you will still need to go through your own RFC process. Before you can 'self release', your team will need to be set up with a ServiceNow group (to raise an RFC and get approvals etc). Please see Get access to ServiceNow- Set up a new assignment group (see page 167).

You will need:

- release tag for the version of the release you intend to deploy, this would normally correspond to the azure pipelines build no, e.g. `#v1.2.1043-beta+36888`
- release plan, that covers dependencies, out-of-band update process, like KVM changes, and impacted services that may need redeploys
- preferred release date and time (typically 11pm on a release day for Platform team releases).

Also consider the following:

- check your pipelines to ensure the build has been successful, including all path-to-live environments
- deployments should be brought through and grouped according to how you would want them to be rolled back, i.e. if one bit fails, do you want some of the change to remain, or does *all* of it need rolling back?
    - Nesting them within the confluence instructions is an excellent approach
    - Include the "RFC codes" at every point, which are only created once a RFC is raised

These items will make it easier to complete and submit the RFC.

How to raise a release RFC change request in ServiceNow

The RFC Ticket Process

A release RFC consists of a type of ticket, a change request, raised in ServiceNow. This change request ticket passes through various states, such as review and implementation. Before the deployment of a release into production, the proposed changes detailed in the Change Request must be approved by various governing boards. Once approved, the deployment can be performed at the designated time by a member of the On-Call team.

How to raise a release RFC in ServiceNow

This guide will cover how to raise a change request ticket in ServiceNow. The official guidance for raising and completing an RFC can be found here: NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1097]

1097 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

There are guides for ServiceNow on the intranet. For more information see ServiceNow Change Management User Guide v2.01[1098]

(General NHSDigital Sharepoint ServiceNow Training and Support[1099])

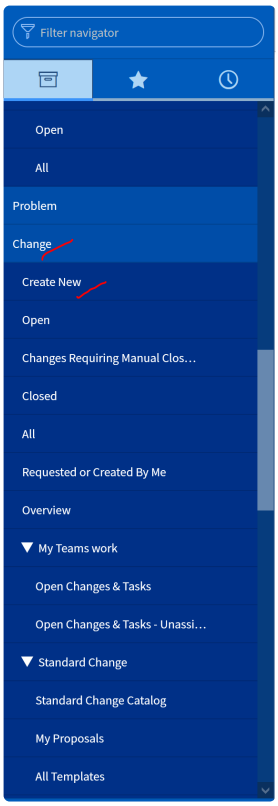
Login to ServiceNow

- Log in to ServiceNow[1100] using your shortcode

Creating a new Change Request

Using the menu bar on the left, scroll to find 'change' OR use the menu search bar to search for 'change'

Click on 'Create New'



Selecting the type of Change Request

You will be presented with 3 choices:

- **Create Normal Change:** 'Normal changes' are a general purpose change type that requires one or more approvals.

1098 https://hscic365.sharepoint.com/:w:/r/sites/ServiceNowImplementation/_layouts/15/Doc.aspx?sourcedoc=%7B10F3A1C4-3E03-4DC3-B6C1-3FC88927B9D6%7D&file=ServiceNow%20Change%20Management%20User%20Guide%20v2.01.docx&action=default&mobileredirect=true

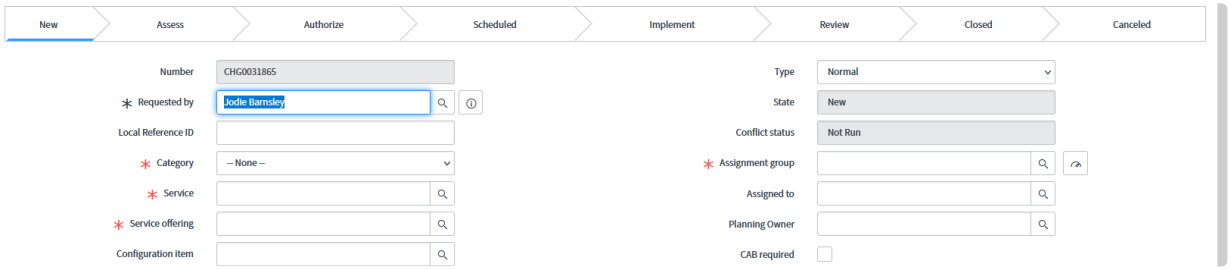
1099 https://hscic365.sharepoint.com/sites/ServiceNowImplementation/SitePages/ServiceNow-Training-and-Support-Portal.aspx

1100 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2F$pa_dashboards_overview.do

- **Standard**: Select from available preapproved change templates. These changes do not require approval.
    - See Standard Changes (see page 2093) guidance for raising a new standard change template, or for raising a change using an existing template, KOP-049 Raising an RFC using a Standard Change Template[11011102] .
- **Create Emergency Change:** Emergency changes are a change type that must be implemented as soon as possible.

Filling in the Request

In the top section of the form complete:



Select a Category

Usually 'Platforms'

Select the Service

Usually 'API Platform Service'

Select the Service offering

Usually 'API Platform'

Select an Assignment group

Usually 'ITO API Platform'

If possible, assign to the person on 2nd line support for the planned day of release. You can find this by looking at On-call rota (Out of hours - 1800 - 0800)[11031104].

Configuration Item

Use the search tool to select the more relevant item for your changes. (Typically this will be API Platform/API/DOS, generally an areas that fits your change step and has a support group of ITO ITOC Techops)

(Only required if marked as mandatory / not always required)

1101 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template
1102 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template
1103 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153365587/On-call+rota+Out+of+hours+-+1800+-+0800
1104 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153365587/On-call+rota+Out+of+hours+-+1800+-+0800

Select Risk level

Options: High, Moderate, Low

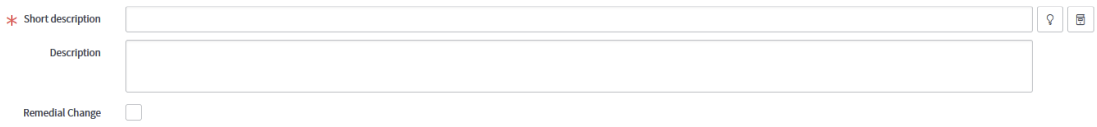
Select Impact level

Options: 1 - High, 2 - Moderate, 3 - Low

See NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1105] for the impact level criteria

Retrospective Change

Tick this box if this change is being raised after the change has been implemented / started. This will allow a retrospective date to be entered in Scheduled start / end fields



Provide a short description (title) for the Change Request

A title should be provided in the 'Short description' field

As we are trying move to templated Standard Change Request tickets, we need to demonstrate to Service Management that we can perform regular deployments consistently. We need to gather evidence of this. So ideally we want common change request tickets to follow a consistent naming pattern. The naming pattern for your Change Request should be:

> *APIM <service-name> Updates*

Examples include:

- APIM Identity Service Updates
- APIM PDS Updates
- APIM Ambulance Analytics Updates
- APIM Utils Updates

Enter a Description

Write a description of the changes being made, also should include a link to the ticket for an in-depth view of the change in 'Description'.

Example Description:

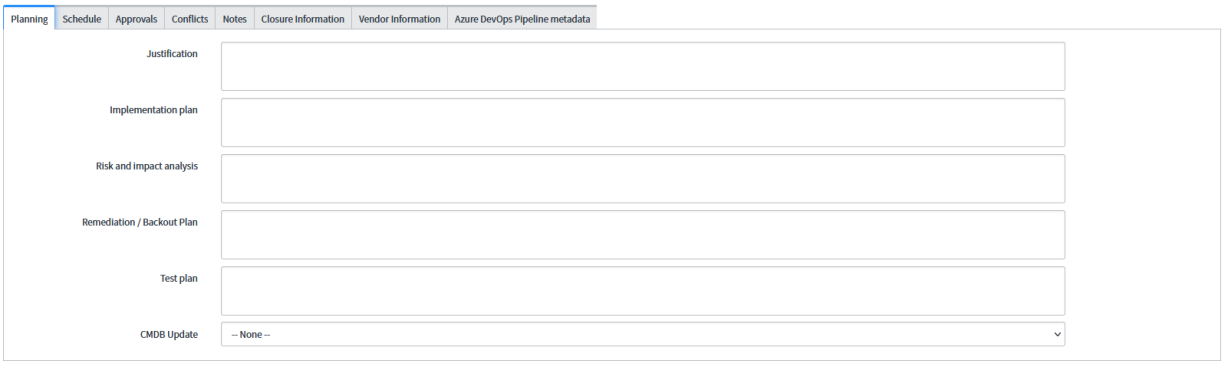
1105 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

```
This release covers the following tickets:

- JIRA-001
- JIRA-002
- JIRA-003

All production releases have a corresponding release plan: [insert link from
https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans]

All production releases follow our documented release KOPs, found here: https://
nhsd-confluence.digital.nhs.uk/display/APM/
KOP-018+Releasing+changes+to+Production
and here: https://nhsd-confluence.digital.nhs.uk/display/APM/
KOP-019+Release+process+-+Individual+API

Rollbacks in case of issues during the release process are detailed here:
https://nhsd-confluence.digital.nhs.uk/display/APM/
KOP-017+Rolling+back+to+a+stable+release
```

Remedial Change

Tick this box if change has been raised to resolve an issue in an earlier failed change

Planning Tab



Justification

Enter a justification for the release - for example: a small description, reference to epic name, details of provided fix/feature etc.

Implementation plan

The standard implementation plans can be found here, ensure you have read KOP-018[11061107] and KOP-019[11081109]. Make reference to these in this section.

Example:

> "We will follow our standard process for making changes on the night:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
>
> and use this KOP for actually making the update:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API"

Risk and impact analysis

- Enter any proposed risks that your change might bring if issues/errors were to arise.
- The impact of the risk if they were to occur.
- The impact of the change as a whole, normally based on things such as chance of service outage.
    - Example: "Impact is very low - no change to any item that's currently in use."
- What have you done to mitigate the chance of this risk happening? I.e. testing throughout development.
    - Example: "Full automated test coverage for this change and it has been manually tested in PTL environments."

Further information and templated can be found here NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1110]

Remediation/ backout plan

The standard backout plan can be found here, ensure you have read KOP-016[11111112] and **KOP-017**[11131114]. Make reference to these in this section.

Example:

> "We will follow our KOP for rolling back:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release"

Test plan

How can it be tested to check the release has correctly processed. i.e. Check Grafana dashboard to check service live? Look for your changes in live? ping and status endpoints, smoke test endpoints

Example:

1106 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1107 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1108 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1109 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1110 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true
1111 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1112 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1113 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1114 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release

"Login to Apigee portal and check to see that the 'Assign message policy - remove external scopes' exists.

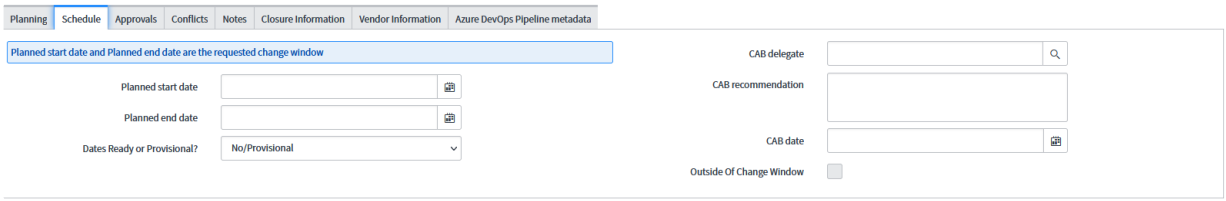
Check Grafana dashboards to make sure all services are still running as expected.

Check Splunk logs lo ensure we can still see transactions being processed successfully."

CMDB update

Usually 'Not Required'

Schedule Tab

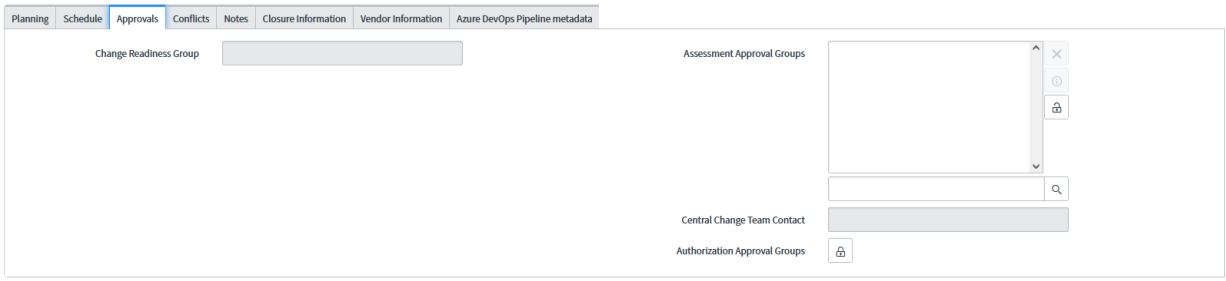


The 'planned start date' (and time) of the release is usually 23:00 on the day of release specified.

For the 'planned end date' (and time), specify 00:00 of the day after release. (An one hour window from start to end)

'Dates ready or provisional?' If you are ready to release your code (just awaiting approval) please select Yes/Ready, otherwise select No/Provisional.

Approvals Tab



Change readiness group

This field is a non-editable field. It will be auto-populated post change submission with change readiness group of the selected service offering / change readiness group of selected service.

Assessment approval group

This is an optional field. It will be auto-populated with assessment approval groups of the selected service offerings / service , if present. This field can also be used by the change team to add additional groups for assessment approvals.

Azure DevOps pipelines metadata tab

You may choose to include Azure DevOps Pipeline metadata here (as well as in the Confluence Release Plan).

This (currently) is not required.

Next steps

Submit to save and submit the change request.

Approvals

You will be emailed as approvals are actioned and these can be viewed on the change request in ServiceNow.

7.15.6.6 Scheduling a release for an API Platform team deployment

Copy the RFC NumberHere - Template[1115] page, replacing the 'NumberHere' with the number of the RFC you've created.

Update with:

- A list of tickets involved in the change. *Note: You can automate this process by using our script. See the script and our instructions on how to use it* here[1116].
- Relevant Azure tags and links for the prepared release.
- Any instructions for the person releasing the API.
- Rollback instructions.

Once the RFC is raised and a release plan has been created you will need to :

- Put an entry into the API Management release schedule (see page 1390) (in pending release status) in Confluence to schedule the release
- Once the RFC is fully signed off , mark it off as signed-off in the release schedule (see page 1390)

Additional advice

- RFC sign off can be slowed by Solutions Assurance if you have not signed off each part of the development with your squads resident SA person. Only submit RFC once you have got their agreement.
- Your confluence instructions need to be explicit for the out of hours support, so ensure all known eventualities are covered otherwise the change might could be fully rolled back.

7.15.6.7 Changing the date of release

Once an RFC is approved, APIM is unable to change the release date. Only Change Management team is able to do this. They can be contacted via the shared mailbox, liveservices.central-changemgmt@nhs.net[1117]

1115 https://nhsd-confluence.digital.nhs.uk/display/APM/RFC+NumberHere+-++Template
1116 https://github.com/NHSDigital/api-management-infrastructure/tree/master/scripts/release-notes
1117 mailto:liveservices.central-changemgmt@nhs.net

7.15.6.8 Deploying to production for the first time

When you first start deploying your service into production, you'll need to:

7.15.6.8.1 Add information about your production environment into your manifest file

7.15.6.8.1.1 Locate your manifest file

Your manifest file will be at the root of your project, and will usually be called something like `manifest_template.yml.`

Here is an example `manifest_template.yml` file. You will notice that it's in two sections; the front matter, which contains variables, and a template, which uses the variables to create a larger manifest. The templating is used because a fully unrolled manifest is quite large and contains a lot of repetition.

```
APIGEE_ENVIRONMENTS:
  - name: internal-dev
    display_name_suffix: Internal Development
  - name: internal-dev-sandbox
    display_name_suffix: Internal Development Sandbox
  - name: internal-qa
    display_name_suffix: Internal QA
  - name: internal-qa-sandbox
    display_name_suffix: Internal QA Sandbox
  - name: ref
    display_name_suffix: Reference
    ratelimit: '31500pm'
    quota: '31500'
  - name: sandbox
    display_name_suffix: Sandbox
    product_visibility: private
  - name: int
    display_name_suffix: Integration Testing

ACCESS_MODES:
  - name: user-restricted
    nameSuffix: ''
    displayName: Healthcare Worker
    scopes: ['urn:nhsd:apim:user-nhs-id:aal3:personal-demographics-service']
    requireCallbackUrl: true
    description: User restricted
  - name: application-restricted
    nameSuffix: -application-restricted
    displayName: Application-Restricted
    scopes: ['urn:nhsd:apim:app:level3:personal-demographics-service']
```

```
    requireCallbackUrl: false
    description: Application Restricted
---

meta:
  schema_version: 1
  api:
    name: personal-demographics
    guid: eef32850-52ae-46da-9dbd-d9f3df818846
    spec_guids:
      - a343a204-f2d2-4287-a2e5-b5cb367e35bb
apigee:
  environments:
{% for env in APIGEE_ENVIRONMENTS %}
{% set product_visibility = env.product_visibility | default('public') %}
    - name: {{ env.name }}
      products:
{% set approval_type = env.approval_type | default('auto') %}
{% for mode in ACCESS_MODES %}
      - name: personal-demographics-{{ env.name }}{{ mode.nameSuffix }}
        approvalType: {{ approval_type }}
        attributes:
          - name: access
            value: {{ product_visibility }}
          - name: ratelimit
            value: {{ env.ratelimit | default('300pm') }}
        description: {{ mode.description }}
        displayName: Personal Demographics Service - {{ mode.displayName }}
({{ env.display_name_suffix }})
        environments: [ {{ env.name }} ]
        proxies:
          - personal-demographics-{{ env.name }}
          - identity-service-{{ env.name }}
        scopes: {{ mode.scopes }}
        quota: {{ env.quota | default('300') }}
        quotaInterval: '1'
        quotaTimeUnit: minute
{% endfor %}
      specs:
      - name: personal-demographics-{{ env.name }}
        path: personal-demographics.json
{% endfor %}
{% endfor %}
```

7.15.6.8.1.2 Add a production section to APIGEE_ENVIRONMENTS

Under the APIGEE_ENVIRONMENTS variable, add another item for your production environment. Make sure your YAML is valid!

```
  - name: prod
    approval_type: manual
    display_name_suffix: Production
    ratelimit: '15000pm'
    quota: '7800'
```

7.15.6.8.2 Add a production step to your release pipeline

7.15.6.8.2.1 Locate your release pipeline

Your release pipeline will be under `azure/azure-release-pipeline.yml`.

Here is an example `azure-release-pipeline.yml`.

```
name: "$(SourceBranchName)+$(BuildID)"

trigger: none
pr: none

resources:
  repositories:
    - repository: common
      type: github
      name: NHSDigital/api-management-utils
      ref: refs/heads/edge
      endpoint: NHSDigital
  pipelines:
  - pipeline: build_pipeline
    source: "Personal-Demographics-Service-Build"
    trigger:
      branches:
        include:
          - refs/tags/v*

pool:
  name: 'AWS-ECS'

variables:
  - template: project.yml

extends:
  template: azure/common/apigee-deployment.yml@common
  parameters:
    service_name: ${{ variables.service_name }}
    short_service_name: ${{ variables.short_service_name }}
    service_base_path: ${{ variables.service_base_path }}
```

```
    enable_monitoring: true
    enable_status_monitoring: true
    jinja_templates:
      REQUIRE_ASID: false
      PDS_TARGET_SERVER: spine-demographics
    secret_ids:
      - ptl/app-credentials/jwt_testing/non-prod/JWT_TESTING_API_KEY
      - ptl/app-credentials/jwt_testing/non-prod/JWT_TESTING_WITH_ASID_API_KEY
    secret_file_ids:
      - ptl/app-credentials/jwt_testing/non-prod/JWT_TESTING_WITH_ASID_PRIVATE_KEY
      - ptl/app-credentials/jwt_testing/non-prod/JWT_TESTING_PRIVATE_KEY
    config_ids:
      - /ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/
CLIENT_ID
      - /ptl/azure-devops/env-internal-dev/test-app/internal-testing-internal-dev/
CLIENT_SECRET
    apigee_deployments:
      - environment: internal-dev
        post_deploy:
          - template: templates/pds-tests.yml
      - environment: internal-qa
        post_deploy:
          - template: templates/pds-tests.yml
        depends_on:
            - internal_dev
      - environment: internal-qa-sandbox
        enable_status_monitoring: false
        proxy_path: sandbox
        post_deploy:
          - template: templates/sandbox-tests.yml
      - environment: int
        depends_on:
          - internal_qa
          - internal_qa_sandbox
      - environment: ref
        depends_on:
          - internal_qa
          - internal_qa_sandbox
      - environment: sandbox
        proxy_path: sandbox
        enable_status_monitoring: false
        post_deploy:
          - template: templates/sandbox-tests.yml
        depends_on:
          - internal_qa
          - internal_qa_sandbox
```

7.15.6.8.2.2 Add a production step to apigee_deployments

To enable a production deployment, add in a new item to apigee_deployments:

```
      - environment: prod
        depends_on:              # which environments to depend on before deploying
          - internal_qa
          - internal_qa_sandbox
        jinja_templates:         # any additional values you need to template into
your proxy at deploy time
          REQUIRE_ASID: true
          PDS_TARGET_SERVER: spine-demographics
```

7.15.7 Deploying to production with Proxy Generator

7.15.7.1 Overview

When going live with an API for the first time there are a few dependencies and activities that must first be completed.

The general process involves:

- configuring your production instance backend
- asking the API platform team to enable you for production deployments
- creating a release plan (includes raising a Request For Change)

This page describes the process for creating a release plan and raising a Request For Change (RFC) for API proxy-related changes to a live environment and not changes to your API back-end code. You may however, find some of the details useful for raising RFCs in general.

RFCs are currently raised through ServiceNow. Make sure you have access to ServiceNow (see page 167) and once the RFC is raised, you will create a release plan and schedule a release.

7.15.7.2 Deploying to production for the first time

If you are deploying to production for the first time, first make sure you have completed the 'First time deploying into production activities' on the Beta private section of the API process checklist using proxy generator (see page 227).

Follow the steps in this section before deploying to production using Proxy Generator.

7.15.7.2.1 Prepare your production instance

Deploying your production instance using Proxy Generator is done in the same way as for any other environment.

When preparing your production instance first make sure you have:

- deployed and tested non production instances (see page 698)
- configured your production instance backend (including deploying necessary secrets) (see page 688)

- have your production OAS payload written and ready to deploy (see the Proxy Generator spec for example and schema[1118])

7.15.7.2.2 Enabling production deployments

APIs using Proxy Generator have API-level config that can be adjusted by the API platform team and `disallowed_environments` contains environments an API is prevented from deploying to. By default, production deployments are disallowed for APIs using Proxy Generator. This is to prevent teams going live before the API is ready.

To deploy to production you must first request this by posting a query in the #platforms-apim-producer-support Slack channel (see page 2056) asking for `prod` to be removed from your `disallowed_environments` as you are deploying to production for the first time.

Once you have followed these processes you will be ready to deploy to production. Read below about raising and RFC and then follow the instruction for Deploying your API (see page 698)

7.15.7.3 Managing production deployments with RFCs

As an API Producer, almost all changes will require an RFC. However, the timing of a release is also a key consideration. This page sets how the API Platform team currently approaches this and might provide useful guidance: Release Management (see page 2077).

7.15.7.4 Creating an RFC

Before you can release, your team will need to be set up with a ServiceNow group (to raise an RFC and get approvals etc). Please see Get access to ServiceNow- Set up a new assignment group (see page 167).

You will need:

- release plan, that covers dependencies, out-of-band update process, like impacted services that may need redeploys
- preferred release date and time.

Also consider that deployments should be brought through and grouped according to how you would want them to be rolled back, i.e. if one bit fails, do you want some of the change to remain, or does *all* of it need rolling back?

- Nesting them within the confluence instructions is an excellent approach
- Include the "RFC codes" at every point, which are only created once a RFC is raised

These items will make it easier to complete and submit the RFC.

1118 https://digital.nhs.uk/developer/api-catalogue/proxy-generator#post-/apis/-api_name-/environments/-environment-/instances

Please see the below excerpt on how to raise an RFC. Some details may vary e.g. your assignment group.

How to raise a release RFC change request in ServiceNow

The RFC Ticket Process

A release RFC consists of a type of ticket, a change request, raised in ServiceNow. This change request ticket passes through various states, such as review and implementation. Before the deployment of a release into production, the proposed changes detailed in the Change Request must be approved by various governing boards. Once approved, the deployment can be performed at the designated time by a member of the On-Call team.

How to raise a release RFC in ServiceNow

This guide will cover how to raise a change request ticket in ServiceNow. The official guidance for raising and completing an RFC can be found here: NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1119]

There are guides for ServiceNow on the intranet. For more information see ServiceNow Change Management User Guide v2.01[1120]

(General NHSDigital Sharepoint ServiceNow Training and Support[1121])

Login to ServiceNow

- Log in to ServiceNow[1122] using your shortcode

Creating a new Change Request

Using the menu bar on the left, scroll to find 'change' OR use the menu search bar to search for 'change'

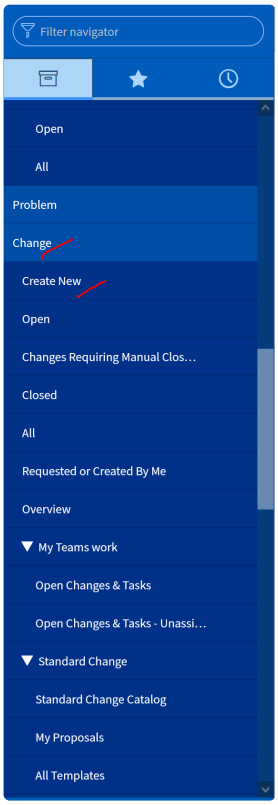
Click on 'Create New'

1119 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

1120 https://hscic365.sharepoint.com/:w:/r/sites/ServiceNowImplementation/_layouts/15/Doc.aspx?sourcedoc=%7B10F3A1C4-3E03-4DC3-B6C1-3FC88927B9D6%7D&file=ServiceNow%20Change%20Management%20User%20Guide%20v2.01.docx&action=default&mobileredirect=true

1121 https://hscic365.sharepoint.com/sites/ServiceNowImplementation/SitePages/ServiceNow-Training-and-Support-Portal.aspx

1122 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2F$pa_dashboards_overview.do



Selecting the type of Change Request

You will be presented with 3 choices:

- **Create Normal Change:** 'Normal changes' are a general purpose change type that requires one or more approvals.
- **Standard**: Select from available preapproved change templates. These changes do not require approval.
    - See Standard Changes (see page 2093) guidance for raising a new standard change template, or for raising a change using an existing template, KOP-049 Raising an RFC using a Standard Change Template[11231124] .
- **Create Emergency Change:** Emergency changes are a change type that must be implemented as soon as possible.

Filling in the Request

In the top section of the form complete:

1123 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template

1124 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template

Select a Category

Usually 'Platforms'

Select the Service

Usually 'API Platform Service'

Select the Service offering

Usually 'API Platform'

Select an Assignment group

Usually 'ITO API Platform'

If possible, assign to the person on 2nd line support for the planned day of release. You can find this by looking at On-call rota (Out of hours - 1800 - 0800)[11251126].

Configuration Item

Use the search tool to select the more relevant item for your changes. (Typically this will be API Platform/API/DOS, generally an areas that fits your change step and has a support group of ITO ITOC Techops)

(Only required if marked as mandatory / not always required)

Select Risk level

Options: High, Moderate, Low

Select Impact level

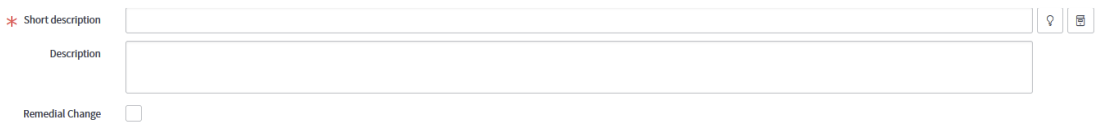
Options: 1 - High, 2 - Moderate, 3 - Low

1125 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153365587/On-call+rota+Out+of+hours+-+1800+-+0800
1126 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153365587/On-call+rota+Out+of+hours+-+1800+-+0800

See NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1127] for the impact level criteria

Retrospective Change

Tick this box if this change is being raised after the change has been implemented / started. This will allow a retrospective date to be entered in Scheduled start / end fields



Provide a short description (title) for the Change Request

A title should be provided in the 'Short description' field

As we are trying move to templated Standard Change Request tickets, we need to demonstrate to Service Management that we can perform regular deployments consistently. We need to gather evidence of this. So ideally we want common change request tickets to follow a consistent naming pattern. The naming pattern for your Change Request should be:

> *APIM <service-name> Updates*

Examples include:

- APIM Identity Service Updates
- APIM PDS Updates
- APIM Ambulance Analytics Updates
- APIM Utils Updates

Enter a Description

Write a description of the changes being made, also should include a link to the ticket for an in-depth view of the change in 'Description'.

Example Description:

```
This release covers the following tickets:

- JIRA-001
- JIRA-002
- JIRA-003

All production releases have a corresponding release plan: [insert link from
https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans]
```

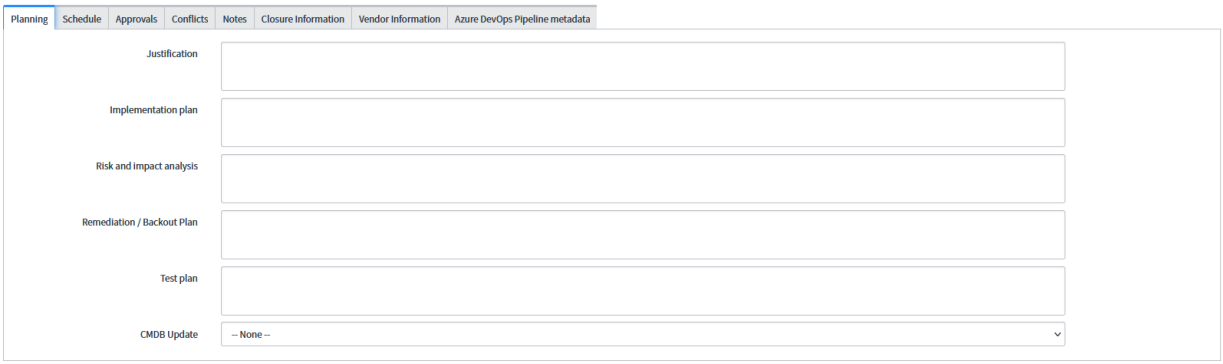
1127 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

```
All production releases follow our documented release KOPs, found here: https://
nhsd-confluence.digital.nhs.uk/display/APM/
KOP-018+Releasing+changes+to+Production
and here: https://nhsd-confluence.digital.nhs.uk/display/APM/
KOP-019+Release+process+-+Individual+API

Rollbacks in case of issues during the release process are detailed here:
https://nhsd-confluence.digital.nhs.uk/display/APM/
KOP-017+Rolling+back+to+a+stable+release
```

Remedial Change

Tick this box if change has been raised to resolve an issue in an earlier failed change

Planning Tab



Justification

Enter a justification for the release - for example: a small description, reference to epic name, details of provided fix/feature etc.

Implementation plan

The standard implementation plans can be found here, ensure you have read KOP-018[11281129] and KOP-019[11301131]. Make reference to these in this section.

Example:

> "We will follow our standard process for making changes on the night:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
>
> and use this KOP for actually making the update:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API"

Risk and impact analysis

- Enter any proposed risks that your change might bring if issues/errors were to arise.
- The impact of the risk if they were to occur.

1128 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1129 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1130 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1131 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API

- The impact of the change as a whole, normally based on things such as chance of service outage.
    - Example: "Impact is very low - no change to any item that's currently in use."
- What have you done to mitigate the chance of this risk happening? I.e. testing throughout development.
    - Example: "Full automated test coverage for this change and it has been manually tested in PTL environments."

Further information and templated can be found here NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[1132]

Remediation/ backout plan

The standard backout plan can be found here, ensure you have read KOP-016[11331134] and **KOP-017**[11351136]. Make reference to these in this section.

Example:

> "We will follow our KOP for rolling back:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release"

Test plan

How can it be tested to check the release has correctly processed. i.e. Check Grafana dashboard to check service live? Look for your changes in live? ping and status endpoints, smoke test endpoints

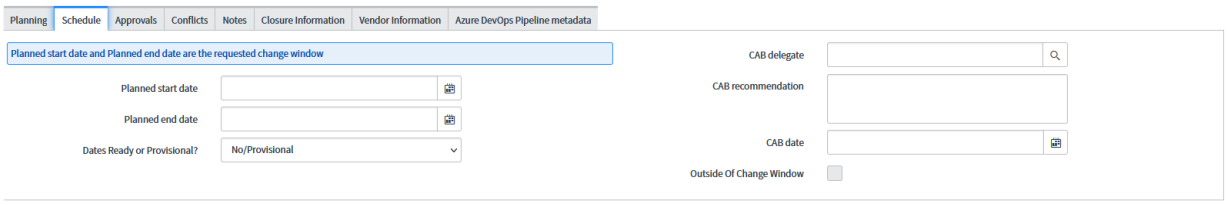
Example:

> "Login to Apigee portal and check to see that the 'Assign message policy - remove external scopes' exists.
>
> Check Grafana dashboards to make sure all services are still running as expected.
>
> Check Splunk logs lo ensure we can still see transactions being processed successfully."

CMDB update

Usually 'Not Required'

Schedule Tab



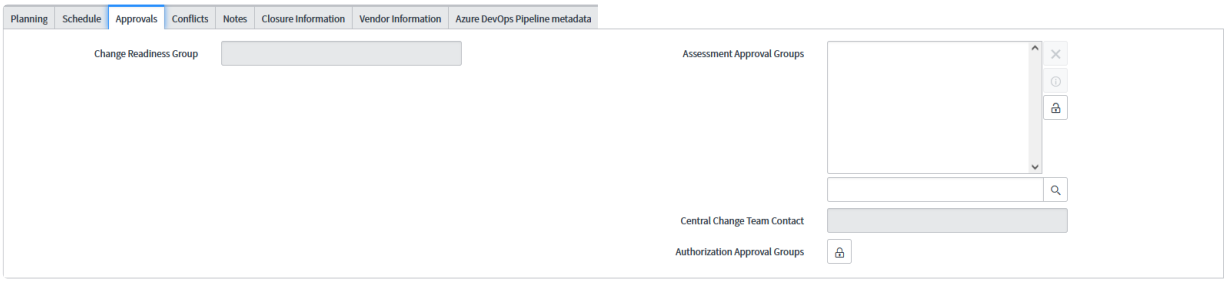
1132 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true
1133 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1134 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1135 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1136 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release

The 'planned start date' (and time) of the release is usually 23:00 on the day of release specified.

For the 'planned end date' (and time), specify 00:00 of the day after release. (An one hour window from start to end)

'Dates ready or provisional?' If you are ready to release your code (just awaiting approval) please select Yes/Ready, otherwise select No/Provisional.

Approvals Tab



Change readiness group

This field is a non-editable field. It will be auto-populated post change submission with change readiness group of the selected service offering / change readiness group of selected service.

Assessment approval group

This is an optional field. It will be auto-populated with assessment approval groups of the selected service offerings / service , if present. This field can also be used by the change team to add additional groups for assessment approvals.

Azure DevOps pipelines metadata tab

You may choose to include Azure DevOps Pipeline metadata here (as well as in the Confluence Release Plan).

This (currently) is not required.

Next steps

Submit to save and submit the change request.

Approvals

You will be emailed as approvals are actioned and these can be viewed on the change request in ServiceNow.

7.15.7.5 Scheduling a release for an API Platform team deployment

Copy the CHG NumberHere - Template (see page 1485) page, replacing the 'NumberHere' with the number of the RFC you've created.

Update with:

- A list of tickets involved in the change.
- Remove references to Azure tags.
- Any instructions for the person releasing the API.
- Rollback instructions.

Once the RFC is raised and a release plan has been created you will need to :

- Put an entry into the API Management release schedule (see page 1390) (in pending release status) in Confluence to schedule the release
- Once the RFC is fully signed off , mark it off as signed-off in the release schedule (see page 1390)

> **Additional advice**
>
> - RFC sign off can be slowed by Solutions Assurance if you have not signed off each part of the development with your squads resident SA person. Only submit RFC once you have got their agreement.
> - Your confluence instructions need to be explicit for the out of hours support, so ensure all known eventualities are covered otherwise the change might could be fully rolled back.

7.15.7.6 Changing the date of release

Once an RFC is approved, APIM is unable to change the release date. Only Change Management team is able to do this. They can be contacted via the shared mailbox, liveservices.central-changemgmt@nhs.net[1137]

7.15.8 Testing in production

7.15.8.1 Overview

For some APIs it might be OK for consumers to test their software directly against your production API.

This makes most sense for APIs that provide read-only access to fairly public data e.g. the Organisation Data Service or the content syndication APIs.

If you're doing this, make it clear in your documentation and also make it clear what you expect consumers to do and not to e.g. probably don't do performance testing!

7.15.8.2 Smoke testing

A smoke test is a non-exhaustive end-to-end test of the critical parts of a system and testing this as a consumer (see page 925).

The benefits of running smoke tests are that you can reliably ensure that the end-to-end functioning of a system is working as intended, primarily testing the happy path scenarios of basic functionality.

See Smoke testing (see page 1145) for guidance on setting up automated "smoke tests" for your API.

1137 mailto:liveservices.central-changemgmt@nhs.net

7.15.8.3 Smoke testing

7.15.8.3.1 Overview

This page provides guidance on setting up automated "smoke tests" for your API.

7.15.8.3.2 Definition of smoke test

A smoke test is a non-exhaustive end-to-end test of the critical parts of a system. Its main intention is to establish the end-to-end functioning of a system **in the production environment** - in our case that the client can authenticate and talk to the proxy, the proxy can talk to the backend and the client should be able to get useful data.

Therefore, smoke tests can be very basic – they are principally testing for basic connectivity and happy path functionality.

7.15.8.3.2.1 Happy path versus non happy path smoke tests

Running smoke tests in production is tricky for some APIs. For example, if your API retrieves personal data, how do you smoke test it without querying data for a real patient?

There is a concept of test data in production which might help. Synthetic patient data can be set up for exactly this purpose.

Alternatively, a valid smoke test might be to try to retrieve non-existent data. This will cause your API to be exercised all the way through to the data store and return, for example, a 404 response.

7.15.8.3.2.2 Manual versus automated smoke tests

Ideally, smoke tests should be automated so that they run every time your API is deployed to production.

This might not be possible if your API has user restricted auth that requires a security device (e.g. NHS CIS2 Smartcard). In this case you might need to run manual smoke tests, certainly when you first deploy your API to production and ideally when you make major changes.

There is massive value to having automated tests to ensure that all your endpoints require an Apigee access token and to demonstrate that they reject a request if no access token or an invalid access token are presented. (**tbc** *to making the results of this type of automated testing visible in the pipelines and visible to APIM Tribe devops*)

7.15.8.3.3 Where and when to run smoke tests

It is advisable to have a common automated smoke test that can run as part of a continuous deploy pipeline. It should run once after the deploy to verify that the deploy was successful.

Your API should have a full set of automated tests, which includes all API business functionality, in the PTL environments. These should **not** be run in Production unless you have a particularly complex solution (such as EPS and the Signing Service) and have full support from your Information Governance and Clinical Safety teams.

7.15.8.3.3.1 Recommendations

For a live API:

- There should be at least one smoke test
    - If your _status endpoint actually triggers every key component in your solution then that could well be a very simple and **all** you need
    - Remember that for the busier APIs that have constant traffic 24x7, there is minimal benefit to a smoke test since the traffic is going to stop immediately after deployment triggering the relevant alarms
- It should run as part of an automated deployment to production
- It should be documented such that it can be run through manually if necessary
- A smoke test should check the version(s) of the deployed API, potentially front and back end and, if appropriate, components.
    - Returning version info /_status is the preferred way of doing this, as /_status should available with a lower security bar
- Do any of the smoke tests require **live credentials** that could access PII or more sensitive data?
    - This tends to apply when moving into Production for the first time to do a technical go-live
    - The smoke tests should be **approved** by the API Producer Information Asset Owner (IAO) and potentially Cyber Security
    - It is critical there is a defined period that this testing can occur, and the credentials revoked
    - Ideally the application should also be deleted as well. Future smoke tests should create new applications
    - Spine (see below) has a similar process / approach for production testing

Levels of complexity and sensitivity:

- Application restricted - little or no sensitive data
- Application restricted - PII or more sensitive special category
- User restricted - PII or more sensitive special category
- User restricted - complex, requires an application to perform manual testing

7.15.8.3.3.2 Smoke testing process for Spine APIs

Non-standard

The following section goes against standard NHS Digital policies and procedures. However, previously Spine recognised this need, and APIM tribe is integrating platforms and services across NHS Digital. The ability to smoke test is important because:

- APIM (nor any API Producer) should not have to rely on external third parties to verify that the basic API functionality is working
- Support technical go lives (i.e. evidence to senior NHS D & NHS X stakeholders) prior to fully on-boarding API Consumers
- Massively de-risk major releases, allowing the API Producer (sometimes APIM themselves) to smoke test

To complete an end to end smoke test in production requires:

- A test app in Apigee
- Production smartcard (where user restricted APIs are involved)
- Spine ASID
- Spine test data.

A previously created Spine smoke test guidance document: TD00nnnn Synthetic Test Data Guidance Final v1.0.docx[1138]

An API Producer Team, along with their IAO and Cyber Security an agreement on the following is needed:

- Agree owner within team of this activity
- Frequency and reasons to run smoke tests
- Longevity of:
    - Test Application (would a new one be created each time?)
    - Spine ASID (disable until next needed?)
    - Smartcard (e.g. issue to one person, but have the role expire after 30 days)
- Actual testing to be carried out, which might require the preparation of specific test data

Apigee Test Application

The API Producer team creates an Apigee app via the developer portal to access their API as a Consumer (see page 926) would. Ideally it should be named something like "<API Proxy Name> - PROD - Smoke Test":

When the team has LSB approval (or similar) for a 'go-live', or a 'technical go-live' and smoke test; the "nominated" onboarding lead will need to raise a producer support request via slack workflow, to enable the App in production (as described in App enabled into production (see page 1025)).

Production smartcard

As this is something we are working through at the moment, contact Aubyn Crawford to coordinate with the RA team and Cyber Security. For this stage the following questions need to answered:

1138 https://nhsd-confluence.digital.nhs.uk/download/attachments/169716134/TD00nnnn%20Synthetic%20Test%20Data%20Guidance%20Final%20v1.0.docx?api=v2&modificationDate=1652193587748&version=1

Question	Notes
What testing have you carried out in the test environment before seeking to test in Live?	Solutions Assurance will be inputting to your development, and this question can be answered by them. It might be as simple as an email from them. Question - if a team has LSB approval, that will already have had SA approval?
Smartcards in Production cannot be shared - they must belong to named users. Who requires a card?	List the users, and specify the expiry time for the role to be allocated (e.g. 30 days) If a smoke tester already has a live smartcard, then they must be named
What SPINE data / applications will be accessed as part of your testing in Live?	List APIs / Spine services
For the actual smoke test, in Production, detail the testing to be carried out	Simpler the better for manual testing, a page on Confluence might suffice. Example for PDS is here: Production Smoke Testing[1139]
Has the IAO approved use?	The IAO might wish to consult with IG on this prior to approval

On the smart card you need to specify:

- A single generic role R8015 (you should be familiar with this from the INT instructions[1140] - for APIM smoke testing no other roles are needed)
- The owning ODS code - previous users have been grouped under the ODS code Solutions Assurance (Emma Bell) use

Spine ASID

Each application requires a dedicated ASID in Spine and it is especially important to isolate any test application system usage. To get one allocated you need the message sets (which you should have organised and tested in INT for your API) and then following the process here because the API Producer is acting as an API Consumer:

https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=114037475#SpineandAccreditedSystemIdentifier(ASID)-new-consumer-asid

1139 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/114910424/Production+Smoke+Testing
1140 https://nhsd-confluence.digital.nhs.uk/display/APM/Smartcard+reader+and+card+access

7.15.9 API Consumer connection topologies

7.15.9.1 Securing connections to NHS England

We already ask API Consumers to have one Apigee application per environment, and not to reuse it. In Production it is important to to consider keeping connections isolated from each other. You should consider have more than one Apigee application when:

- You have distinct software solutions
- The solution is deployed in multiple places, maybe on premise and cloud
    - Note that your CI / CD pipeline and call routing might require the **same** credentials across cloud availability zones, although any multi-region deployments we would strongly recommend separate credentials
- Tenanted solutions, where you make a deployment per customer (e.g. per Trust)

What benefits does this bring:

- Each application has it's own credentials:
    - If these are compromised then there is a massively reduced impact
    - Data contamination between applications is significantly less likely to occur through misconfiguration
- Aligns with Cyber Security requirements in many organisations, and builds on the "defence in depth" strategy
- It allows clearer management of activity (and risks) to both API Consumer and the various teams inside NHS Digital

Are there technical implications?

- Setting up more Apigee applications:
    - Takes more effort
    - Increases complexity
    - Increases potential confusion when working with different companies
    - However, many API Consumer implementation teams already work on a per customer deployment basis, reducing / removing any additional burdens
- Digital Onboarding Service needs to group applications per approval
- Where Private Key JWT is required, you can share a common JWKS endpoint across Production (still not across environments).
    - You still need to consider how to keep your private key minimally exposed, ideally deployed directly to an HSM (Hardware Security Module) by a CI / CD pipeline that auto rotates every month

7.15.9.1.1 Apigee applications - supported (once agreed) models

For a simple visuals only summary see: Onboarding and configuring Apigee applications (see page 1031)

7.15.9.2 Compliance and security controls to protect patient data

For lower sensitivity APIs, how the API Consumer connects and uses the data or updates the NHS E held datasets is protected by the legal and compliance responsibilities they take on. This is **still** the case for the more sensitive APIs - however, NHS Digital takes extra steps in the compliance and security controls as part of it's responsibility to look after patient data:

- Chain of responsibility - many solutions are created by a third party but then reused by solutions down stream
    - NHS England legal agreements ensure the responsibilities are passed on down the chain of contracts
    - For the more sensitive APIs, additional technical controls have been agreed to allow NHS England to follow the chain electronically
- Partition connections to NHS England:
    - Should a consumer have one app per "access mode"?
    - How many deployment sites does the API Consumer have?
- CSOC monitoring of API usage for misuse
- Audit requirements
- Data sharing agreements
- Encouraging user restricted access modes
    - National RBAC for healthcare workers
- Track End User Organisations (EUOs)

7.15.9.2.1 End User Organisations (EUOs)

Creating software is complex and expensive, and most GPs and Trusts engage a third party (software supplier) to do this, or to use an existing product in the market place. This means the person using the software (the End User) is almost never the software provider staff, but a healthcare worker. Data security and governance is carried out by the organisation that the healthcare worker is employed by - the End User Organisation (EUO). NHS England is responsible for the data and services made available, an so requires some understanding which organisations, on a daily basis, are connecting through a software provider.

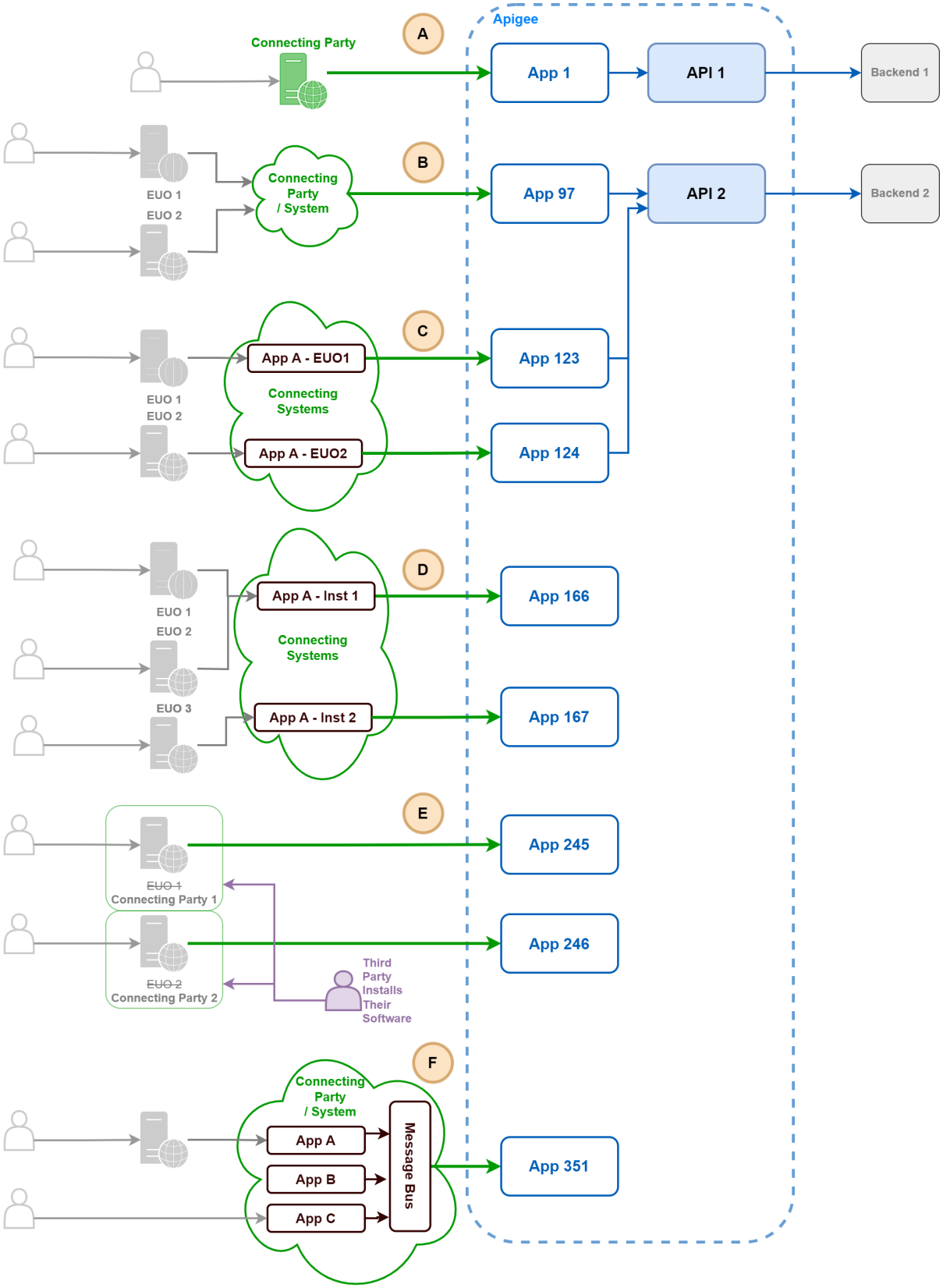
On this basis, API Management can support a number of compliance / security levels (*will offer - some features still to be built*). As an API Producer you should ask these questions:

- Does the API need to keep a record of which individual accessed the data or service?
- Does the API need to keep a record of which organisations accessed the data or service?
- Does the API need to actively reject a request based on the organisation making the request?

7.15.9.2.2 Connection topology variations

The following deployment models can exist, note that:

- App restricted is possible for all topologies
- The End User Org might be an actual server application, or a web client


Apigee
A
Connecting Party
App 1
API 1
Backend 1
B
EUO 1
EUO 2
Connecting Party / System
App 97
API 2
Backend 2
C
EUO 1
EUO 2
App A - EUO1
Connecting Systems
App A - EUO2
App 123
App 124
D
EUO 1
EUO 2
EUO 3
App A - Inst 1
Connecting Systems
App A - Inst 2
App 166
App 167
E
EUO 1
Connecting Party 1
EUO 2
Connecting Party 2
Third Party Installs Their Software
App 245
App 246
F
Connecting Party / System
App A
App B
App C
Message Bus
App 351

These topology classifications were designed to help different teams work together, and understand risk better.
If the process and (later automation) is completed, then API producers will not need to understand the connection topologies and the API Consumers can self-manage.
The topologies will then just be a measure of risk exposure.

Topology	Description / Example	Number of Apigee Apps
A	Web site / National app where users access directly	1
B	Cloud solution, that is accessed from specific EUOs	1 (tbc)
C	Tenanted solution (aggregator) - each EUO has a separate deployment	1 per EUO
D	Grouped tenanted solution (grouped aggregator)- Connecting party deploys one tenant per group of EUOs	tbc
E	**Should** appear as multiple "A" connections	1 per EUO
F	A shared authentication solution is deployed that serves multiple Products (apps)	1

TODO: Investigate / agree term to indicator that the Supplier is operating "on behalf of EUOs", independent of the deployment configuration.

7.15.9.2.2.1 Spine considerations

For Spine and the Spine Secure Proxy (SSP), the equivalent terms for "tenanted"[1141] are:

- Aggregator - API Consumer side
- Data Centre Hosted Practice System - API Producer where third parties provide the data, which SSP routes the Consumer to

1141 https://developer.nhs.uk/apis/spine-core/ssp_system_topologies.html

In the above diagram "tenanted" is about deployment isolation **not** about handling data on behalf of.

Where a proxy connects to a Spine API - an Accredited System IDentifier (ASID) is required, which holds the authorisation settings for that ASID. Read more about this here: Spine based APIs (see page 408).

- ASIDs are not shown on the above topology diagram because:
  - The ASID is, now, **always** that of the Connecting Party
  - The ODS code associated with that ASID must be for the Connecting Party
- You **must** design your onboarding, and authorisation models to focus on a single one Apigee App. If you do not do this it has a wide ranging effect on how an API Consumer connects to other APIs

7.15.9.2.2.2 Real world examples

As this contains third party supplier names - the table is moved to Compliance touch points[1142]

7.15.10 Beta exit review

The last step of completing Beta is to hold an exit review with your producer liaison (currently Jonathan Eagle). This will likely be conducted during a regular catch up review meeting.

This is to:

- do a review of the activities you've done during beta - as per Beta (public) process checklist (see page 198)
- agree we think you've done everything necessary to exit discovery
- decide whether to proceed to live

To request a review, contact us - see Help and support (see page 2056).

7.15.11 Developer community guidance

7.15.11.1 Overview

For a general introduction to the developer community, see https://digital.nhs.uk/services/api-platform/developer-community.

1142 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182976643/Compliance+touch+points

7.15.11.2 Owning and managing sub-communities

For people who own those APIs and services

We advocate for empowering product teams to nurture and expand their product categories, akin to fostering sub-communities.

By actively engaging with these communities, valuable outcomes can be achieved.

From **collecting user feedback** to **community driven development**, engaging in **product testing and announcements** and **generating innovative ideas**, tapping into this community proves instrumental.

Such involvement not only **lowers support costs** but also establishes an adaptable, efficient approach to enhance user experiences throughout various product usage phases.

7.15.11.3 Operations

Joining the community

Joining the community - What is the Developer Community? - [1143]NHS England Developer Community[1144]

You only need an e-mail address and password to create your initial account.

1. Navigate to NHS England Developer Community[1145]
2. Select

3. This will navigate you to the NHS Digital developer account[1146] page, create an account.

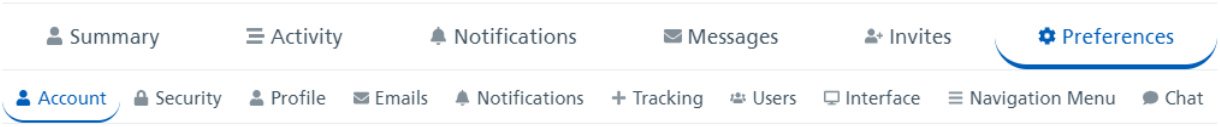
Congratulations, you have joined the community.

1143 https://developer.community.nhs.uk/t/joining-the-community/572
1144 https://developer.community.nhs.uk/
1145 https://developer.community.nhs.uk/
1146 https://identity.prod.api.platform.nhs.uk/realms/developer-identity/protocol/openid-connect/auth?client_id=discourse&nonce=b426669fe1da16ed2ee5abbf21f4c23601b054cacb202210e92b3e5e2268b16a&redirect_uri=https%3A%2F%2Fdeveloper.community.nhs.uk%2Fauth%2Foidc%2Fcallback&response_type=code&scope=openid+profile&state=014777835da12656cddbf49b27a918a7d968972f87b2bd47

Personalise your information

You can now personalise your information.

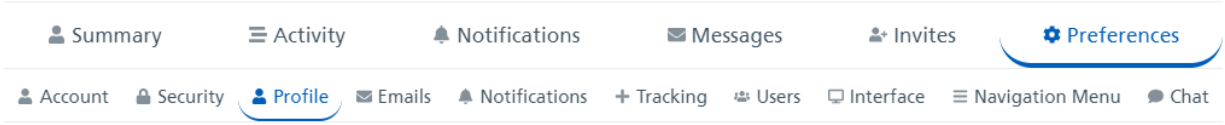
From **Profile** select **Preferences** - **Account**



From account you can update:

- username
- profile picture
- add alternative email
- name

From **Profile** select **Preferences** - **Profile**



From profile you can update:

- about me
- timezone
- location
- website
- job role
- organisation
- profile header
- user card background
- featured topic
- date of birth (month, date)

Guidelines for community discussion

Guidelines on how to engage with the community can be found in our What is the Developer Community pages.

Guidelines for community discussions - What is the Developer Community? - [1147]NHS England Developer Community[1148]

1147 https://developer.community.nhs.uk/t/guidelines-for-community-discussions/571
1148 https://developer.community.nhs.uk/

New category request

To request a new category[1149], perform the following steps:

1. Login to the Developer Community[1150].
2. Using the left navigation navigate to **Messages** and click the **+** button, see figure 1.

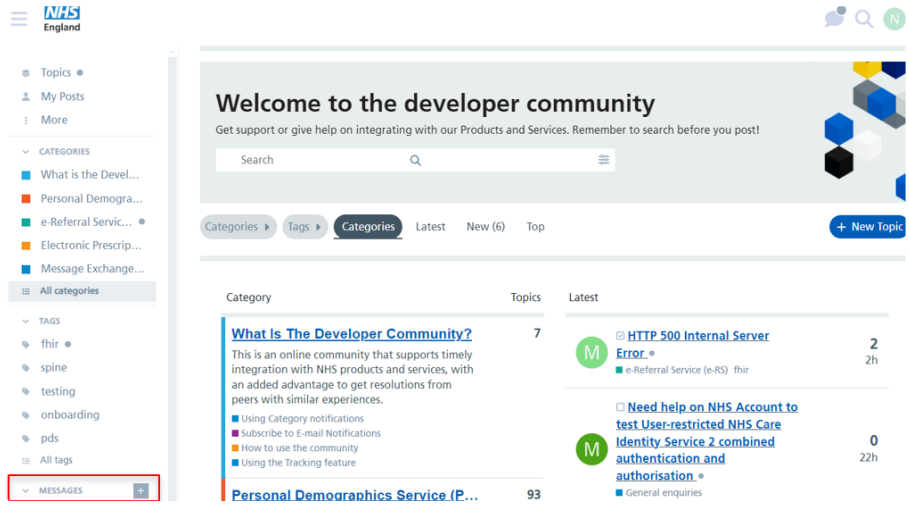


Fig 1: Developer Community - Messages

The **Start a message** form is displayed, see figure 2.

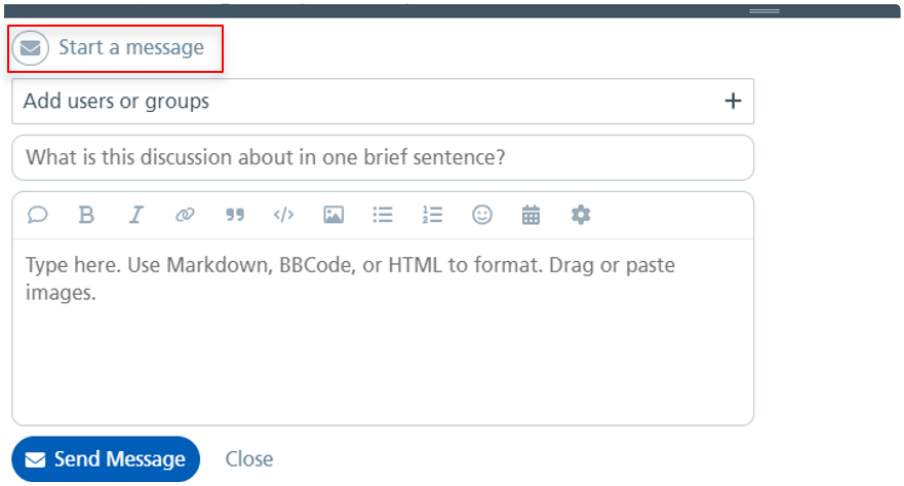


Fig 2 Developer Community - Start a message

3. Click the **+** button and search for **Moderators**, see figure 3.

1149 https://developer.community.nhs.uk/t/new-category-requests/594
1150 https://developer.community.nhs.uk/

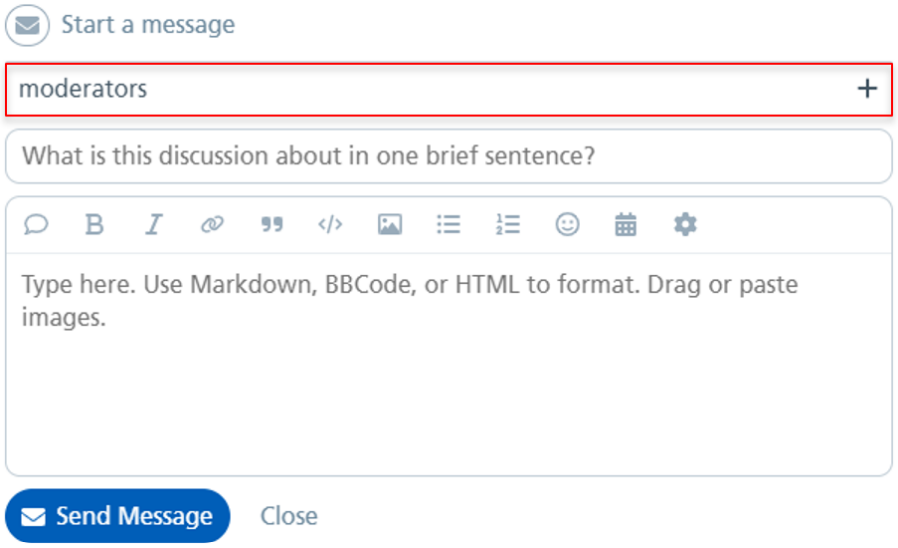


Fig 3 Developer Community - Select "Moderators"

4. Enter a "subject" and message in the text fields, see figure 4.

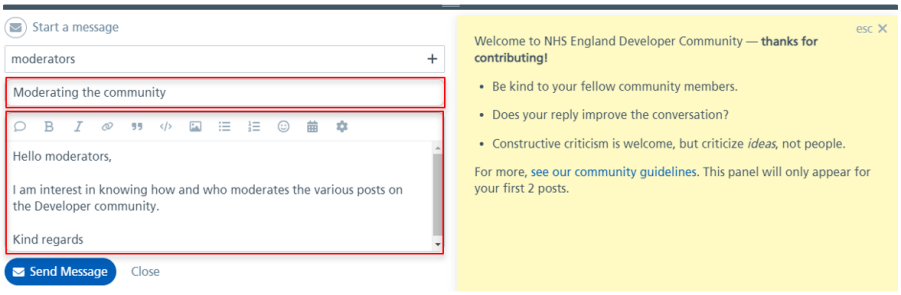


Fig 4 Developer Community - Compose message

5. Click the **Send Message** button, see figure 5.

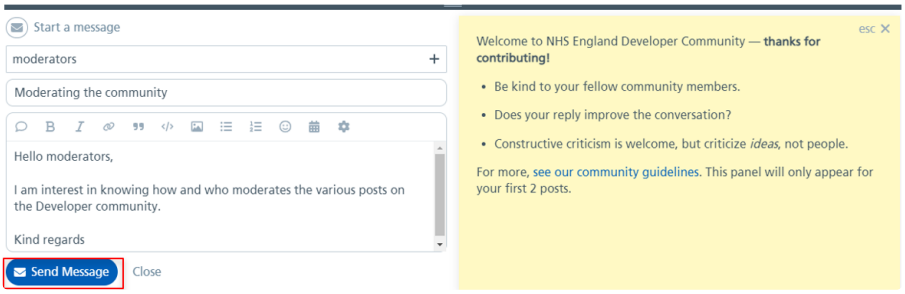


Fig 5 Developer Community - Send message

Your message has now been sent.

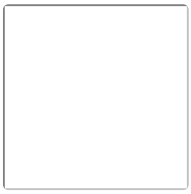
Subscribing to emails

About the Subscription to E-mail Notifications category - What is the Developer Community? / Subscription to E-mail Notifications -[1151] NHS England Developer Community[1152]

How do I receive email notifications for the community?

Ensure you are signed up to use the community[1153] *(Users who are not signed up will not receive email notifications).*

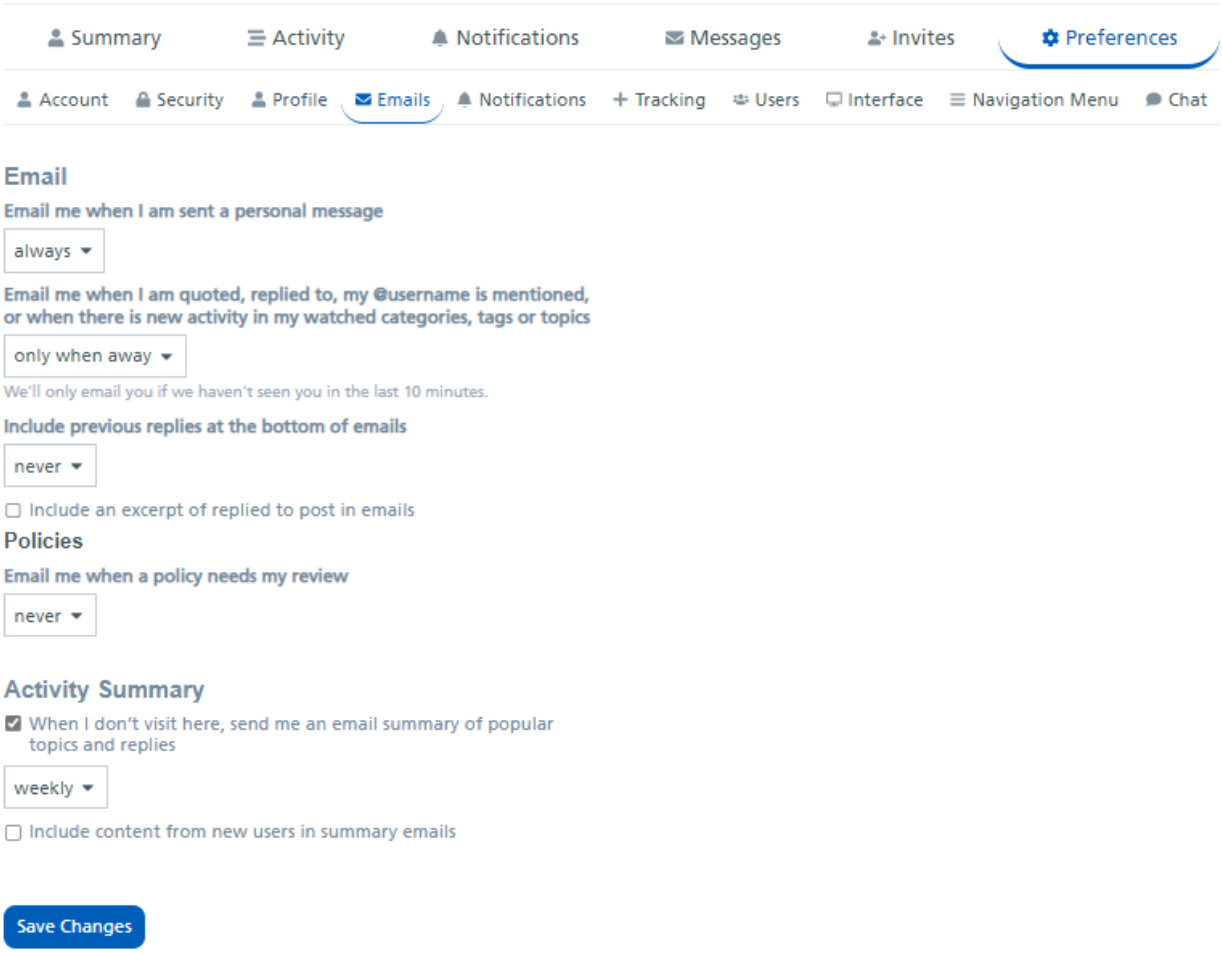
Please follow the steps below to ensure your email preferences are set correctly:

- select your **profile icon** in the top right-hand corner of the screen
- select the **role icon**
- select the **preferences icon.** from the dropdown menu 

This will bring up a range of filters, click on '**Emails**'.

- set your desired preferences.

1151 https://developer.community.nhs.uk/t/about-the-subscription-to-e-mail-notifications-category/492
1152 https://developer.community.nhs.uk/
1153 https://developer.community.nhs.uk/



Don't forget to **Save Changes**.

Note: to receive email notifications for 'watched' categories and tags, you will need to be logged out from the community.

Note: Ensure notifications are set in your preference settings.

To receive email notifications for interested posts, the following settings should be enabled:

- go back to home page and click on the category of choice
- click on the Alarm Icon on the top right and click on watching
- go back to the first three [3] steps and click on tracking - this should be your view

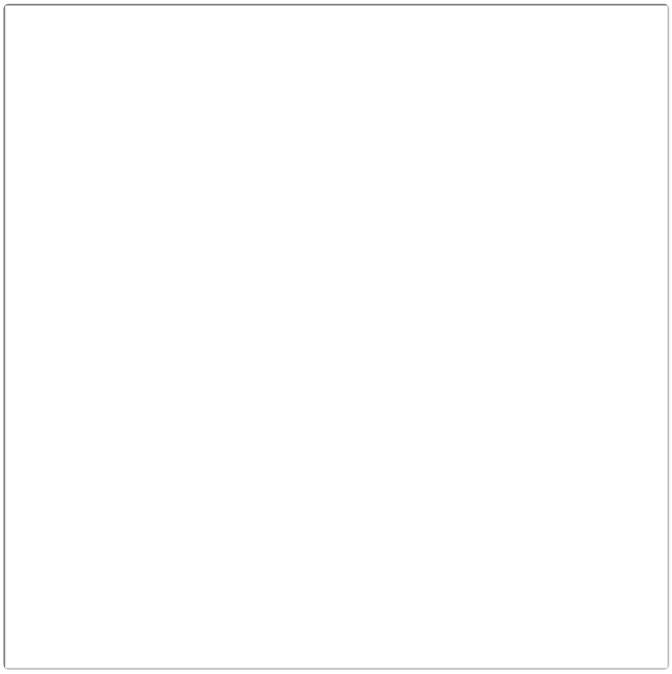
Congratulations! You are subscribed

Subscribing to category notifications

Using Category notifications - What is the Developer Community? / Using Category notifications - [1154]NHS England Developer Community[1155]

Notifications can be set from a **Category** (e.g. PDS PHIR, EPS, MESH) page.

- visit any **Category page**, and select the **notification control** icon above the topic list on the right hand side of the page
- select your chosen Notification preference (**Watching**, **Tracking**, **Watching First Post**, **Normal** or **Muted**)
- once selected the icon will automatically update to reflect your choice



1154 https://developer.community.nhs.uk/t/using-category-notifications/496
1155 https://developer.community.nhs.uk/

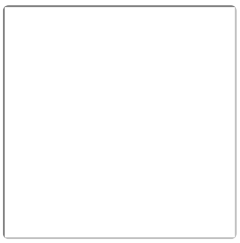
Using the tracking feature

About the Using the Tracking feature - What is the Developer Community? / Using the Tracking feature - [1156]NHS England Developer Community[1157]

How do I Track Categories or Tags?

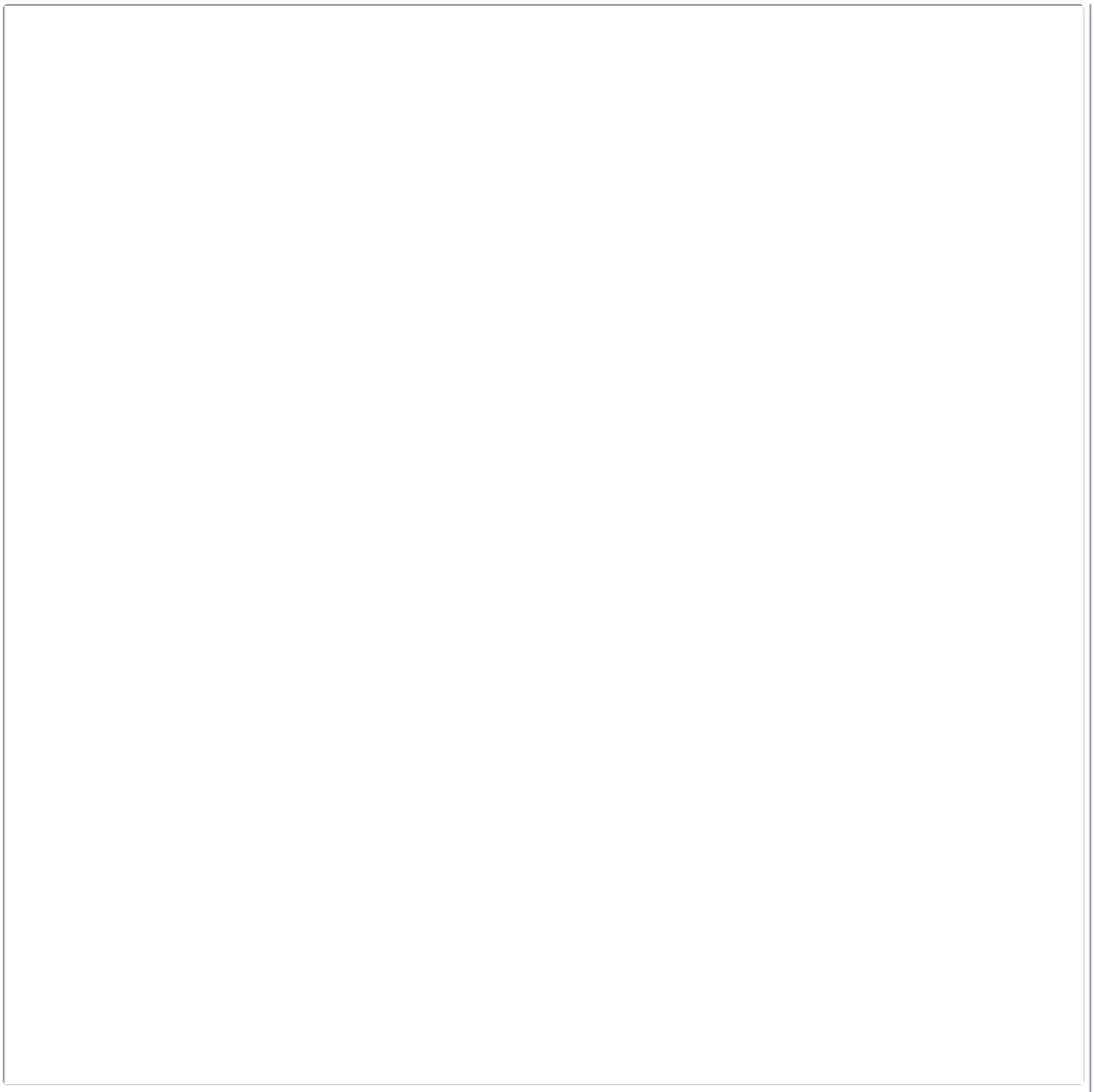
Ensure you are signed up to use the NHS developer community[1158] *(Users who are not signed up will not receive email notifications).*

Please follow the steps below to ensure your email preferences are set correctly:

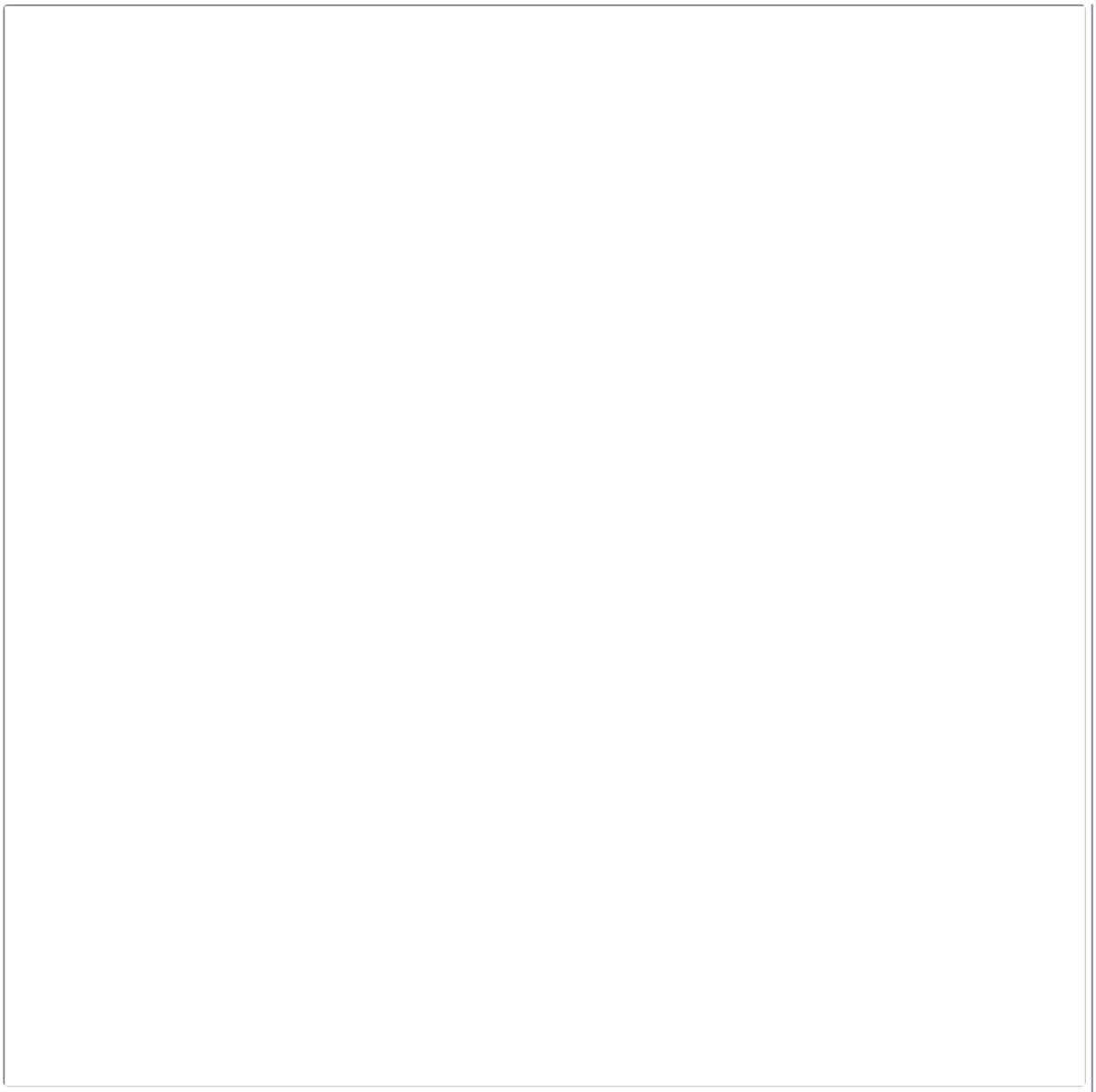
- select your **profile icon** in the top right-hand corner of the screen.
- select the **role icon**
- select the **preferences icon**  from the dropdown menu

This will bring up a range of filters, click on '**+Tracking**'.

1156 https://developer.community.nhs.uk/t/about-the-using-the-tracking-feature/499
1157 https://developer.community.nhs.uk/
1158 https://developer.community.nhs.uk/



- first set your **Topics** settings (or retain defaults)



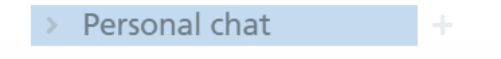
- set your desired preferences, **note** you can add more than one entry into each section (**Watched**, **Tracked**, **Watching First Post** and **Muted**) by selecting the **+** symbol.

Remember to **Save Changes**

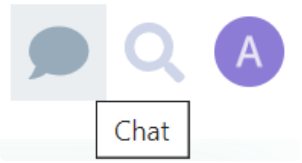
Personal chat

Using Personal chat - What is the Developer Community? - [1159]NHS England Developer Community[1160]

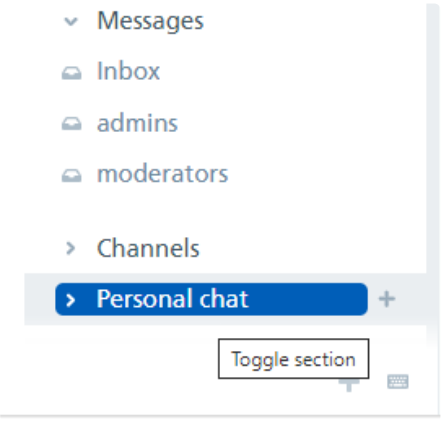
You can create Personal Chats by using the icon to the right of the Personal Chats header.



You can quickly access personal chats from the speech bubble icon located in the top right of the screen, selecting will open the chat window.

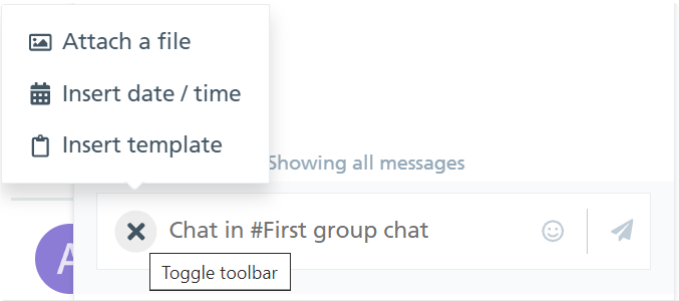


Alternatively, you can access chat from the left hand navigation



Chat toggle

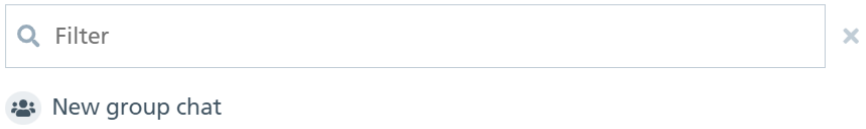
The toggle toolbar allows you to attach a file, insert date/time and insert a template.



Individual chat

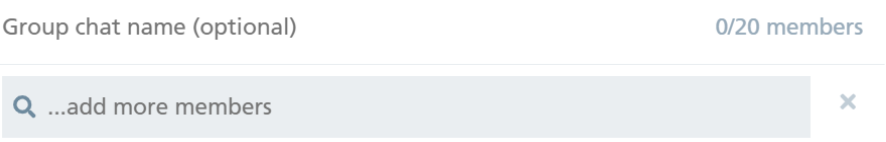
Chats can be with an individual or group based

1159 https://developer.community.nhs.uk/t/using-personal-chat/593
1160 https://developer.community.nhs.uk/

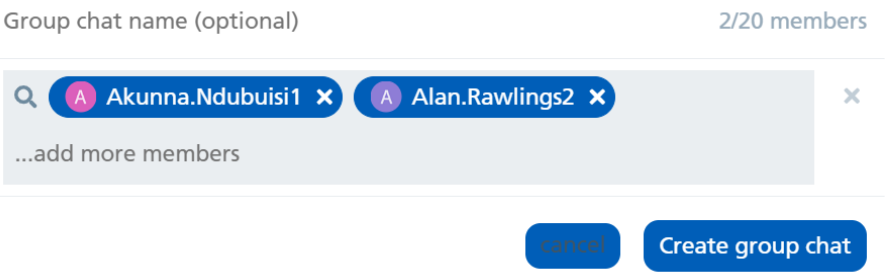


Group chat

Chats can be with groups.

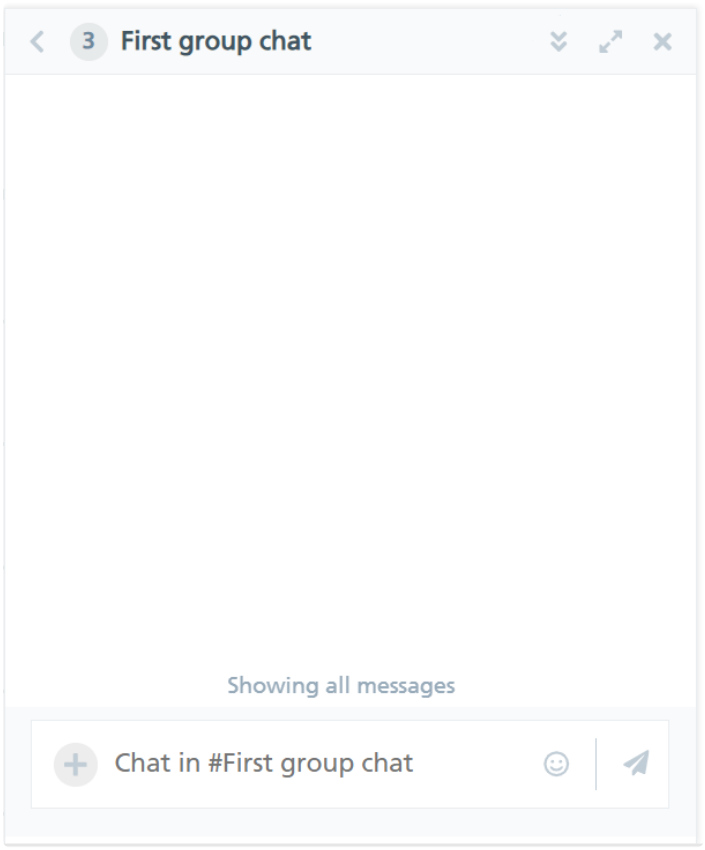


You can name the group chat, each member of the group chat will be listed in the member window.



Chat window

Opening chat will display the Chat Window.



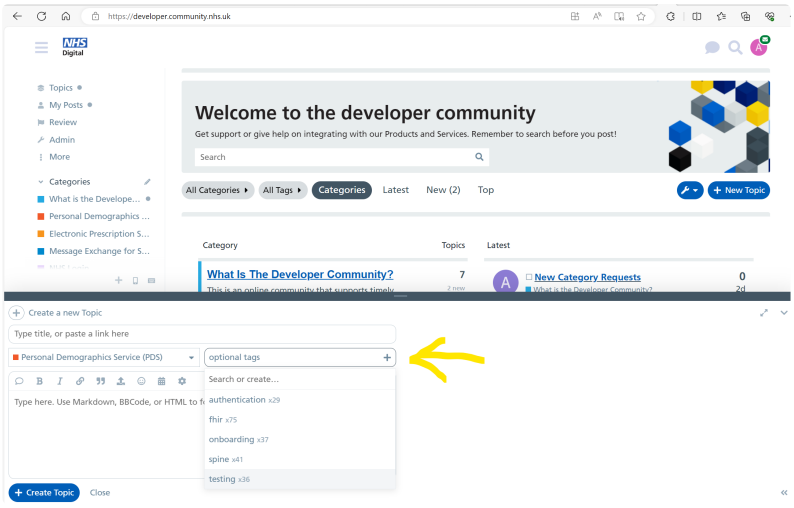
By default, messages in channels last for 90 days.

Creating tags

Tags are only managed within a post on the community.

This post will typically already be linked to a category, so there is assurance of notifications to the required resolver teams with or without tags as tags are optional.

However, we recommend product teams creating unique tags for their categories as it relates to frequency of certain user queries.



Using the like button

The heart shaped icon serves as the like button. Product teams should encourage users to use the like button so we can measure customer satisfaction

Marking as Resolved

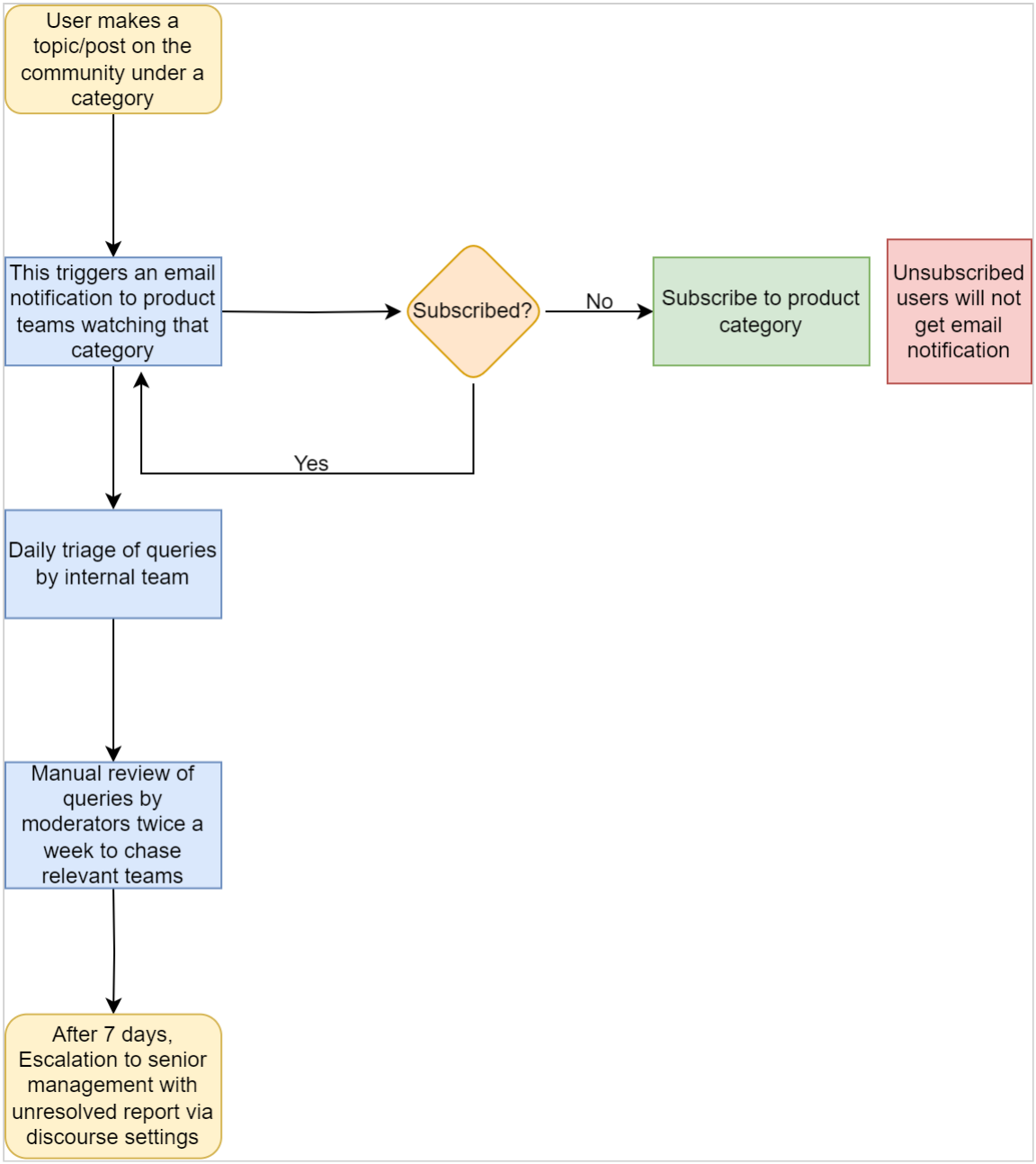
Product teams should click the 'resolved tick' for posts that are resolved for end users

This is the initial look when not ticked.

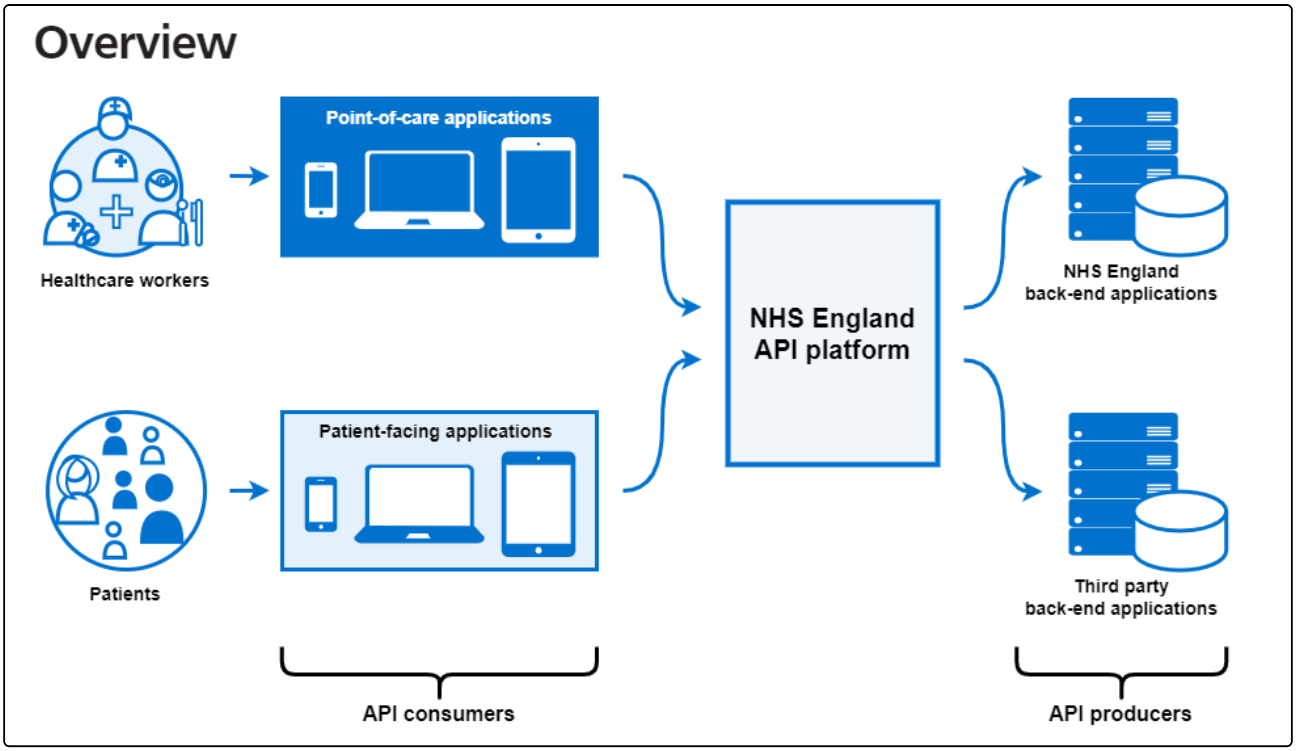
This is how it looks when ticked, showing the provided response was the solution and topic can be closed.

Unresolved Topics/Posts

Product teams should get email notifications for every topic or post under their product categories.
Teams should promptly respond to user queries to minimise delays and mitigate the impact on end users.
A delayed response to users may lead to them reverting to using emails and slack.
This could also result in a negative reputation for the NHS among it's users.



7.15.12 Identifying Who Are Your API Consumers?



Your API consumers are the developers and applications that will interact with your API. Here are some potential groups:

Consumers can be grouped into two general sectors, internal and external, information on both groups is provided below.

7.15.12.1 Internal Consumers

To identify internal API consumers, you need to focus on understanding your organisation's structure, the teams, and the systems that will interact with your API.

Understand the Purpose of Your API

Define the core functionality of your API and its purpose within the organisation.

Does it support internal applications, automate workflows, or centralise data access?

Identify the business problem it solves for internal users.

Map the Organisational Structure

Identify the departments or teams within your organisation that are likely to use the API.

Examples:

Data Teams: Analysts, data engineers, or machine learning teams who may need API data.

Operations Teams: Teams leveraging internal tools for automation efficiencies or monitoring.

Product Teams: Teams building or managing products that depend on API services.

Analyse Existing Systems

Identify Dependencies: Look at internal tools, software, or platforms that the API might integrate with or support.

Example: If you're building an API for user authentication, identify applications that rely on authentication.

Review Current APIs: Examine existing APIs within the organisation to see who uses them and why. This can give insights into likely consumers for your API.

Define Use Cases

Develop specific use cases for the API:

Which teams or applications will consume the API?

What data, features, or functionality will they require?

Examples:

Here are some internal use cases for APIs within a healthcare organisation:

Integrating Systems and Data

- **EPR Integration:** Connecting different Electronic Patient Record (EPR) systems to share patient information seamlessly.
- **Laboratory Information Systems (LIS) Integration:** Integrating LIS systems to access lab results and reports.
- **Pharmacy Information System (PIS) Integration:** Connecting to PIS to manage medication prescriptions and dispensing.

Workflow Automation

- **Appointment Scheduling:** Automating appointment scheduling and reminders for outpatient, pre-assessment and inpatient events.
- **Patient Registrations:** Streamlining patient registration.

Clinical Decision Support

- **Drug Interaction Checking:** Integrating with drug databases to identify potential drug interactions.
- **Clinical Guidelines:** Providing clinicians with evidence-based guidelines and decision support tools.
- **Predictive Analytics:** Using AI and machine learning to predict patient outcomes and identify high-risk patients.

Population Health Management

- **Patient Risk Stratification:** Identifying high-risk patients for targeted interventions.
- **Quality Improvement:** Tracking and analysing quality metrics to improve patient care.
- **Public Health Reporting:** Sharing data with public health agencies for disease surveillance and outbreak response.

Research and Development

- **Clinical Trial Data Management:** Collecting and analysing clinical trial data.
- **Biomedical Research:** Integrating with research databases and tools to accelerate research.

7.15.12.2 External Consumers

Identifying external API consumers involves researching potential users outside your organisation who will benefit from the API and aligning its functionality to their needs.

Define the Purpose of Your API

Clarify the Problem It Solves: Determine the key problem or use case your API addresses for external users. For example:

Does it provide access to specific data?

Does it enable integrations or extend functionality for third-party systems?

Identify the Value Proposition: What makes your API valuable to external consumers? (e.g., ease of integration, unique data, or powerful functionality.)

Segment the Potential Audience

Target Developers: Determine which developer groups may use your API:

Third-party developers:

Developers who want to build applications or services that leverage your API's functionality.

Building Custom Applications: Developers can build custom applications that leverage your API's functionality.

Extending Your Product's Reach: Third-party developers can create integrations and extensions for your product.

Creating Innovative Solutions: Developers can combine your API with other APIs to create innovative solutions.

Partners:

Other businesses or organisations that may integrate your API into their systems.

Strategic Partnerships: Collaborating with other businesses to offer joint solutions.

Data Sharing: Sharing data and insights with partners.

Co-Innovation: Jointly developing new products and services.

Public:

If your API is publicly available, anyone can potentially consume it.

Analyse Use Cases

Understand what external consumers might do with your API:

Application Use: What kind of applications or platforms will use your API? (e.g., mobile apps, web platforms, or IoT devices.)

Frequency of Use: Will they use your API for high-volume transactions, periodic reporting, or occasional requests?

Here are some examples of external use cases for healthcare APIs:

Consumer Health Applications

- **Fitness and Wellness Apps:** Integrating with wearable device APIs to track user health metrics.
- **Medication Tracking Apps:** Connecting with pharmacy APIs to provide medication reminders and refill notifications.
- **Symptom Checker Apps:** Utilising medical knowledge bases and AI to provide preliminary diagnoses based on symptoms.

Research and Development

- **Clinical Trial Data Sharing:** Sharing clinical trial data with researchers and pharmaceutical companies.
- **Biomedical Research:** Accessing and analysing large datasets from various sources.
- **Genomic Data Analysis:** Integrating with genomic databases to analyse genetic information.

Public Health

- **Disease Surveillance:** Monitoring disease outbreaks and trends.
- **Health Information Exchange:** Sharing health information between different healthcare organizations.
- **Emergency Response:** Coordinating emergency response efforts and resource allocation.

Health and Wellness Devices

- **Wearable Devices:** Integrating with fitness trackers and smartwatches to collect health data.
- **Medical Devices:** Connecting medical devices to healthcare systems for remote monitoring and data analysis.

Research Competitors and Industry Trends

Look at similar APIs in your domain to identify who consumes them and why.

Analyse public documentation, forums, and developer communities to see how those APIs are used.

Identify gaps or unmet needs in existing offerings that your API could address.

Identifying Your Target Consumers:

To effectively design and market your API, you should consider the following:

Needs and Requirements: What specific needs and requirements do your target consumers have?

Technical Skills: What is the technical expertise of your target audience?

Integration Preferences: What integration methods (e.g., SDKs, libraries) are preferred by your target consumers?

Security and Compliance: What security and compliance standards need to be met?

By understanding your target consumers and their needs, you can tailor your API and documentation to provide a seamless and valuable offering.

7.15.13 Qualtrics Survey to Power BI

7.15.13.1 Qualtrics Popup Survey Implementation

A Qualtrics popup survey has been enabled on **20+ Service Catalogue pages** (e.g., Message Exchange for Social Care and Health (MESH)[1161]). Users can submit feedback, and all responses are stored securely on the **Qualtrics server**.

7.15.13.2 Automated Feedback Data Processing

To analyze survey responses, we use a **Python script** (`Qualtrics-to-PowerBI`) that:

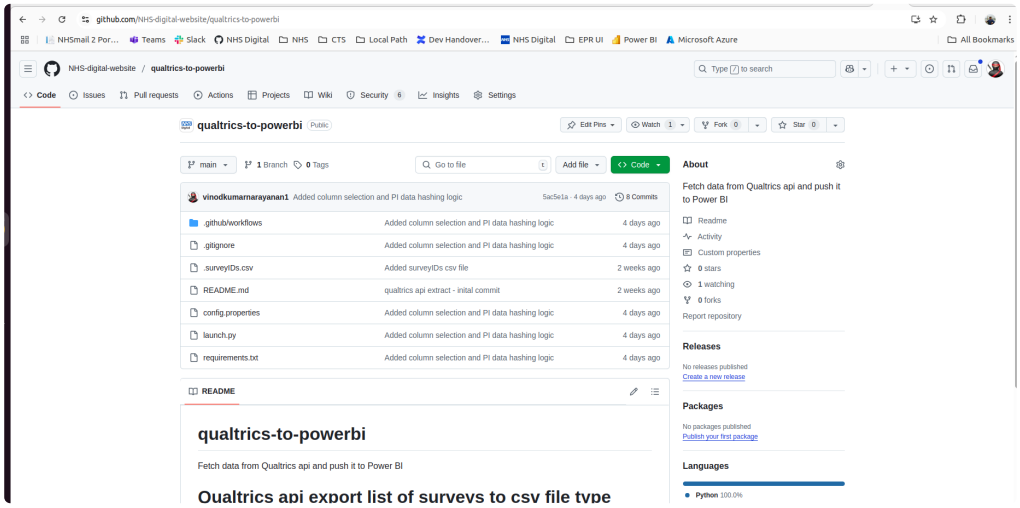
1161 https://digital.nhs.uk/services/message-exchange-for-social-care-and-health-mesh

1. Connects to Qualtrics via **REST API**.
2. Fetches survey response data.
3. Exports the results into a **CSV file** for further analysis.
4. Push the CSV data in to **PowerBI** datasets.

7.15.13.3 Scheduled Execution via GitHub Actions

The script is hosted in the **NHS Digital GitHub repository**:
qualtrics-to-powerbi[1162]

It runs **automatically every day at 7:00 AM** through a **GitHub Actions pipeline**, ensuring up-to-date feedback data is available for reporting in PowerBI.



7.15.13.4 Data Integration Requirements from Qualtrics to Power BI & SharePoint

1. Data Extraction & Transformation:

- From the 44 available columns in Qualtrics, only 12 need to be extracted:
  `Progress | RecordedDate | ResponseId | QID10 | Q1 of 4 | Q2 of 4 | Q3 of 4 | Q4 of 4_4 | Q4 of 4_5 | Q4 of 4_6 | QID7 | Referer`

2. Data Hashing Requirements:

- **Partial Hashing**: Apply hashing to email IDs and phone numbers in these columns:
  `Q3 of 4 | Q4 of 4_5`
- **Full Hashing**: Completely hash names, phone numbers, and email IDs in these columns:
  `Q4 of 4_4 | Q4 of 4_6`

3. Initial Data Load (One-Time):

1162 https://github.com/NHS-digital-website/qualtrics-to-powerbi

- Seed 2,000+ survey records from Qualtrics into Power BI datasets.

4. Daily Incremental Data Sync:

- Fetch only new/updated survey responses from Qualtrics and append them to existing Power BI datasets.

5. Daily Full Export & Grouping:

- Export all survey data daily and split it into 3 CSV files based on URL groupings.
- Save these grouped files to predefined SharePoint folder locations.

Key Workflows:

1. **Power BI Integration**: Initial bulk load + daily incremental updates.
2. **SharePoint Export**: Daily full exports with URL-based grouping for CSV distribution.

Waiting from Charlotte:

1. Grouping URL's
2. SharePoint folder creation and path.

7.15.13.5 Steps to Configure Microsoft Entra App & Power BI Integration

Follow the steps below to:

1. **Register an application in Azure AD (Entra)**
2. **Create a security group**
3. **Enable Power BI admin settings**
4. **Grant workspace access in Power BI**

This setup allows secure access to **Power BI content using a service principal and application secret**.

For detailed instructions, refer to the full Confluence guide:
Power BI Content with Service Principal and Application Secret (see page 1178)

Steps involved:

7.15.13.5.1 Step 1: Create a Microsoft Entra App

1. **Register an App** in Azure Portal → **App registrations** → **New registration**.
2. Provide:
    - **Name**

- **Supported account types**
- (Optional) **Redirect URI**

3. Note the **Application (Client) ID** from the **Overview** tab.
4. Generate a **Client Secret** under **Certificates & secrets** → Save the secret value.

7.15.13.5.2 Step 2: Create a Security Group

1. In **Microsoft Entra admin center** → **Groups** → **New group**.
2. Configure:
   - **Group type** (e.g., Microsoft 365)
   - **Name**, **Email** (if applicable), **Description**
   - Enable **Microsoft Entra roles** if needed.
3. Add **Owners/Members** manually or via bulk upload.

7.15.13.5.3 Step 3: Enable Power BI Admin Settings

1. In **Power BI Admin Portal** → **Tenant settings** → **Developer settings**.
2. Enable:
   - **Embed content in apps**
   - **Allow service principals to use Power BI APIs**
3. Restrict access to a **specific security group** if needed.

7.15.13.5.4 Step 4: Grant Access in Power BI Workspace

1. Navigate to the desired **Power BI workspace** → **Workspace access**.
2. Add:
   - **Service Principal** (using the **App Display Name**) **OR**
   - **Security Group** (created in Step 2).
3. Assign role (**Admin/Member/Contributor**) → **Add**.

7.16 Live phase

7.16.1 Overview

The live phase is about supporting the service in a sustainable way, and continuing to iterate and make improvements

You will:

- publicise your service is live
- ramp down the team, but ensure you still have a support capability
- commit to not making breaking changes; or if we do, ensure we version the API
- continue to support your API consumers through API consumer onboarding (see page 999)

7.16.2 Process steps

For detailed process steps, see the API process checklist (see page 198).

7.16.3 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

7.17 Sunsetting (deprecation and retirement) phases

7.17.1 Overview

Deprecation and retirement are for when an API is no longer needed:

- a **deprecated** API[1163] is still available in production, and still supported, but is obsolete and planned for retirement
- a **retired** API[1164] is no longer available for use

To deprecate an API you will:

- get agreement from relevant stakeholders to deprecate

1163 https://digital.nhs.uk/services/api-platform/api-delivery-process#10-deprecate-your-api
1164 https://digital.nhs.uk/services/api-platform/api-delivery-process#11-retire-your-api

- tag the deprecated API/versions as deprecated in the API catalogue
- engage with API consumers to let them know and to agree a retirement date

To retire an API you will:

- tag the retired API/version in the API catalogue
- raise an RFC in ServiceNow to request the change
- ensure that the API/version is no longer available for use in production
- let everyone know the API is retired

The process is slightly different for API standards - see Sunsetting phases for an API standard (see page 1183).

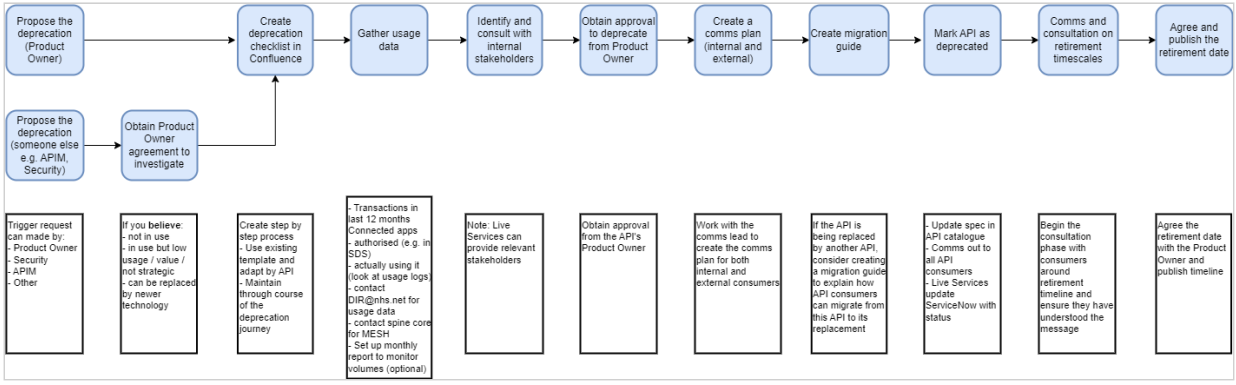
7.17.2 Deprecating an API

Deprecating an API or version means:

- the API is still available to use in production, and possibly still in use
- new integrations are either advised against or forbidden
- existing users of the API have been advised that the API is due to be retired and when, and are expected to migrate away from the API before that time
- the API is listed in the API catalogue but clearly marked as deprecated

> At this stage, no technical changes are being made, this is a communication of intention to retire the API **at some point** in the future, with a retirement date being defined as part of this process

Here's a summary of the process:

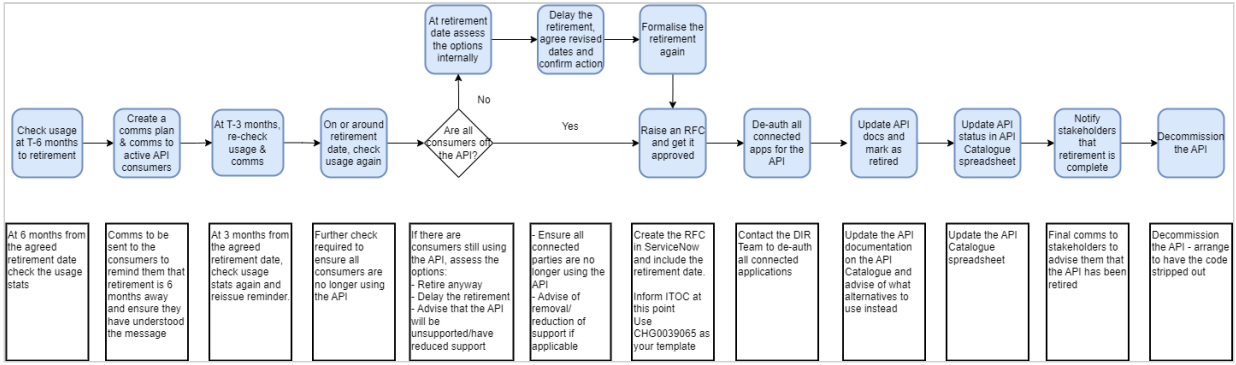


7.17.3 Retiring an API

Retiring an API or version means:

- the API is no longer available for use in production
- the API code might have been removed, or it might still be there, but is dormant and inaccessible (anyone who previously had permission to access it has been revoked)
- Making the **technical changes** to decommission the API
- the API is listed in the API catalogue, but hidden by default, and clearly marked as retired

Here's a summary of the process:



To raise an RFC:

- Access ServiceNow
- Follow this link[1165] - this should create a new change template
- Use CHG0039065 as your template if required
- Contact Live Services should you need assistance (liveservices.central-changemgmt@nhs.net)
- Work with Live Services to get your RFC submitted

7.17.4 Deprecating / retiring API endpoints

An API endpoint may be deprecated if it is:

1165https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsdigitallive.service-now.com%2Fnow%2Fnav%2Fui%2Fclassic%2Fparams%2Ftarget%2Fchange_request.do%253FWIZARD%253Aaction%253Dfollow%2526sys_action%253D%2526sys_id%253D-1%2526sys_target%253D%2526sysparm_query%253Dtype%25253dnormal%2526sysparm_target%253D%2526wiz_action%253Dsysverb_new%2526wiz_collection%253D%2526wiz_collectionID%253D%2526wiz_collection_key%253D%2526wiz_collection_related_field%253D%2526wiz_referring_url%253D%2526wiz_view%253D&data=05%7C01%7Cmatthew.firth1%40nhs.net%7Cb0b90057ae4a4e489f8b08daf7a827f4%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638094598831372490%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=8ZfdBbGt29%2BsB5G6Hdg5t7CnRtefuh749ydyjbDPZ6o%3D&reserved=0

- redundant with the introduction of a new endpoint which provides equivalent or enhanced capabilities
- not fit for purpose, for example because it doesn't include important use-cases
- insecure or a security risk
- not being used or has limited usage

When an API endpoint is deprecated:

- we will publish a deprecation notice which includes a deprecation and retirement date, for example, Deprecation notice[1166]
- the endpoint will still be available for use
- normal service levels will still apply
- further updates to the endpoint are unlikely to be made
- new integrations will not be permitted - partners already in the latter stages of onboarding will be allowed to complete integration

Retiring an API endpoint means:

- is no longer available for use

7.17.5 Resources

The following resources are relevant to this delivery phase and are mentioned as appropriate in the API process checklist (see page 198):

- Sunsetting phases for an API standard (see page 1183)
- Sunsetting checklist template (see page 1186)
- Sunsetting checklist template (API standards) (see page 1194)
- WGLL - sample comms (both internal & external) (see page 1200)

7.17.6 Sunsetting phases for an API standard

7.17.6.1 Overview

Deprecation and retirement are for when an API standard is no longer needed:

- a **deprecated** API standard is still supported and maintained, but is obsolete and planned for retirement
- a **retired** API standard is no longer supported or maintained

To deprecate an API standard you will:

1166 https://digital.nhs.uk/services/directory-of-services-dos/deprecation-notice-gethospitalscore-endpoint

- get agreement from relevant stakeholders to deprecate
- tag the deprecated standard/version of the standard as deprecated in the API catalogue
- tag the deprecated standard/version of the standard as deprecated in the Standards directory
- set a retirement date and publish a deprecation notice

To retire a standard you will:

- tag the retired standard/version of the standard in the API catalogue
- tag the retired standard/version of the standard as retired in the Standards directory
- archive[1167] the standard/version and discourage any further use
- you may need to contact[1168] internet archives and ask that they run a further task to archive your standard
- publish a retirement notice

7.17.6.2 Deprecation

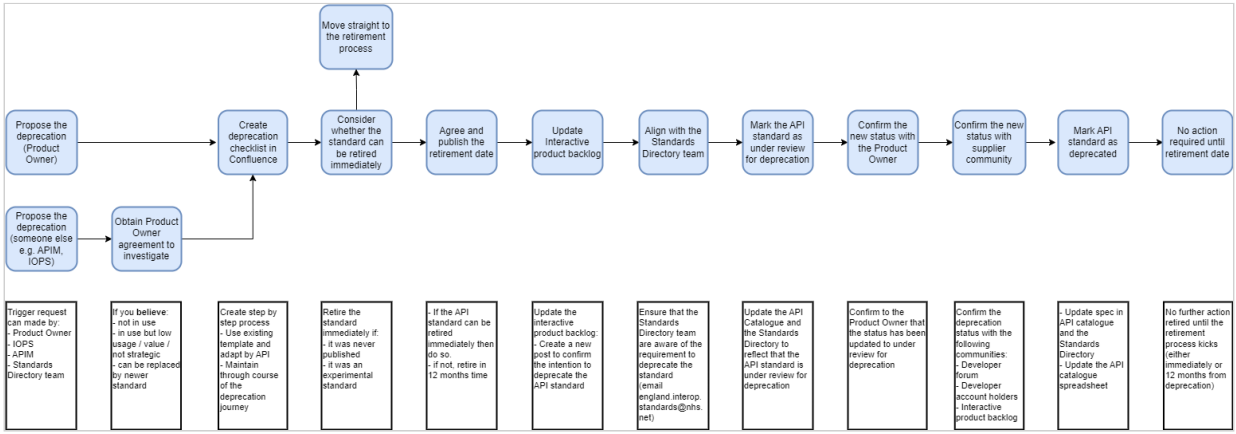
Deprecating an API standard or version means:

- the API standard is still maintained and supported
- further use of the API standard is discouraged
- a deprecation notice has been published to let people know it is deprecated and to state the retirement date
- the API standard is listed in the API catalogue but clearly marked as deprecated
- the API standard is listed in the Standards directory but clearly marked as deprecated

> At this stage, no changes are being made to the API standard, this is a communication of intention to retire the standard **at some point** in the future with a retirement date being defined as part of this process (likely 12 months from deprecation).

Here's a summary of the process:

1167 https://webarchive.nationalarchives.gov.uk/ukgwa/*/http:/digital.nhs.uk/
1168 https://www.nationalarchives.gov.uk/contact-us/



7.17.6.3 Retirement

Retiring an API standard or version means:

- the standard is no longer maintained or supported
- the standard is listed in the API catalogue, but hidden by default (i.e. archived[1169]), and clearly marked as retired
- the standard is listed in the Standards directory and clearly marked as retired

*Note - the internet archive does not update frequently, so you must check that the spec in the archive correctly reflects the status of the API catalogue spec before archiving.

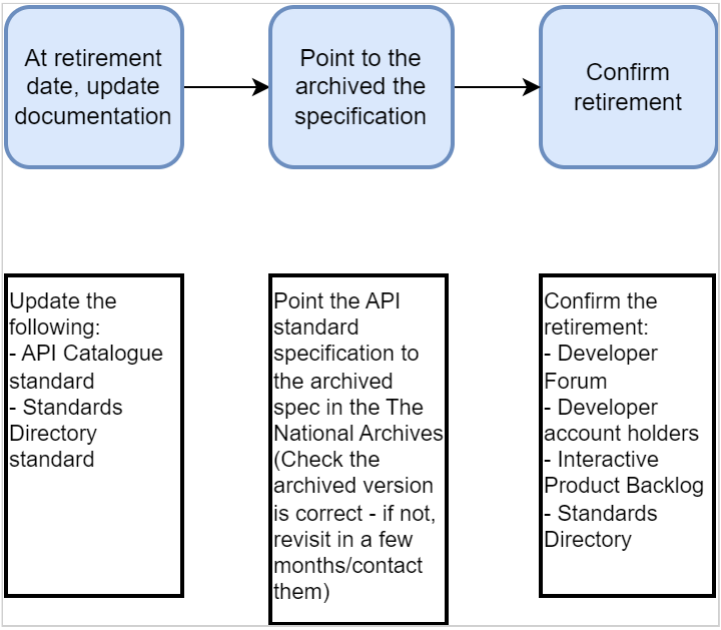
It is likely that you will have to schedule a follow up exercise to archive the spec around 3 months after the retirement date.

Internet archive service team can be contacted through a contact us[1170] page to speed up the archiving of your standard.

API Management (Polaris squad) will run a sweep of all retired API standards on a quarterly basis and archive those which have been updated in internet archives.

Here's a summary of the process:

1169 https://webarchive.nationalarchives.gov.uk/ukgwa/*/http:/digital.nhs.uk/
1170 https://www.nationalarchives.gov.uk/contact-us/



7.17.6.4 Sunsetting process checklist for API standards

To keep track of your progress with sunsetting an API standard, use Sunsetting checklist template (API standards) (see page 1194).

7.17.7 Sunsetting checklist template

Instructions for using this template

Make a copy of this page, change the name to "Sunsetting process checklist - [name of API], and save it in your API's confluence space. If you are doing this within the API Management tribe, put the page as a child of APIs | Sunsetting[1171]. Then delete this message.

7.17.7.1 Overview

This page is a record of the sunsetting (deprecation and retirement) of the API(s) or API version(s) named in the page title.

For details of the process, see Sunsetting (deprecation and retirement) phases (see page 1180).

1171 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216180507/APIs+Sunsetting

7.17.7.2 Checklists

7.17.7.2.1 Checklist statuses

Statuses (for cutting & pasting into the checklists):

TO DO

IN PROGRESS

DONE

N/A

7.17.7.2.2 Deprecation checklist:

Description	Owner	Status	Notes
Create a sunsetting process checklist page by making a copy of this page. Put it in your own Confluence space. If you are doing the deprecation process from within the API Management tribe, you can put it as a child of APIs | Sunsetting[1172].		TO DO	
Clearly identify the API(s) for deprecation, preferably by providing a link to the API specification in the API catalogue. Where appropriate, include details of endpoints and/or interaction IDs being considered.		TO DO	See the API details (see page 1192) below.
Find out whether there is a new version of API available instead of this API		TO DO	
Find out whether any connected apps are using the API(s) or are authorised to use the API(s). State who this is. • If this is a Spine API, to find out who is authorised, contact the DIR team and quote the Interaction IDs. • If this is a MESH API, to find out the transaction volumes, contact Alison Kinloch team and quote the		TO DO	See the Usage analysis (see page 1192) below.

1172 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216180507/APIs+Sunsetting

Description	Owner	Status	Notes
Interaction IDs. Alternately refer to the API_Catalogue.xlsx[1173] for volume information. • Set up a monthly report of active connected systems which can be monitored through the sunsetting period (This step is optional but would be beneficial to track progress)			
Explain why the API(s) are being considered for deprecation. Choose from the reasons below: • we have built a major new version of the API • we have built a new API that provides equivalent capabilities - such as an FHIR API • the API is not fit for purpose, for example, because it doesn't include important use cases • the API is not being used or has limited usage • the API is insecure or a security risk • Other (Give Reason)		TO DO	See the Reason for deprecation (see page 1192) below.
Add your API to the sunsetting register - API register - including a link to the sunsetting process checklist.		TO DO	
If the product owner has not requested the deprecation directly, identify them and: 1. Share the reasons for deprecation. 2. Obtain approval to deprecate the API from the product owner.		TO DO	Pre-requisite for Steps 8 and 9.
Mark the API as "under review" in the API catalogue[1174].		TO DO	
Add an entry to the Interactive product backlog[1175] to say that the API is being considered for deprecation.		TO DO	
Identify and consult with internal owners (e.g. product owner, technical owner, comms lead, live services).		TO DO	See Stakeholders (see page 1193) below.

1173 https://hscic365.sharepoint.com/:x:/r/sites/APIManagement/_layouts/15/Doc.aspx?sourcedoc=%7B9CFE46FD-001E-4296-8F3B-FDA83226841D%7D&file=API_Catalogue.xlsx&action=default&mobileredirect=true&cid=11e9c655-bf28-4c91-b2d2-f45663750d82

1174 https://digital.nhs.uk/developer/api-catalogue

1175 https://nhs-digital-api-management.featureupvote.com/

Description	Owner	Status	Notes
Create a comms plan for both internal and external audiences. For an example, see What good looks like (see page 1180).		TO DO	See Deprecation comms plan (see page 1193) below.
If the API is being/has been replaced by another API then ensure that there is a transition plan in place with the API owner's onboarding lead.		TO DO	
In particular, consider writing a migration guide for API consumers who may need to switch to the replacement API. For an example, see https://digital.nhs.uk/developer/api-catalogue/personal-demographics-service-smsp/migrating-from-the-pds-smsp-api-to-the-pds-fhir-api.		TO DO	
Mark the API as deprecated - update the API documentation. State that: • the API is deprecated • the API is still available for use • our service levels will still apply • we are unlikely to make any updates to the API • we will not permit new integrations • we will consult on when it is planned to be retired • what alternative API(s) to use instead • a link to the migration guide (if applicable)		TO DO	
Ask APIM to update the API status on the "API catalogue spreadsheet". This is important because it feeds into our overall stats on the percentage of (live, not deprecated) APIs that are using the API platform!		TO DO	
Comms out to internal stakeholders to let them know it's deprecated.		TO DO	
Check Live Services have the API listed as deprecated in Cherwell / Service Now.		TO DO	
Update the interactive product backlog to say it's deprecated. Probably with a comment (the feature won't be "done" until the API is retired).		TO DO	

Description	Owner	Status	Notes
Begin the comms and consultation phase with consumers around retirement timescales and ensure they have understood the message by acknowledging the email.		TO DO	
Agree and publish the retirement date.		TO DO	

7.17.7.2.3 Retirement checklist

Step	Owner	Status	Notes
Check the API usage 6 months prior to retirement.		TO DO	
Create a retirement comms plan.		TO DO	
Communicate the usage with API consumers to remind them of retirement in 6 months and ensure they have understood the message by acknowledging the email.		TO DO	
Communicate the usage with API consumers to remind them of retirement in 3 months.		TO DO	
If it's looking like not all API consumers will have ceased using the API by the retirement date, decide what to do about it. Options include: • Retire the API anyway (unlikely, but possible) • Delay the retirement date, in agreement with remaining API consumers. • As an incentive to API consumers to stop using the API, declare that the API will be "unsupported" or will have "reduced" support as of the retirements date		TO DO	
Ensure all connected parties are no longer using the API (and send a chaser email if so)		TO DO	

Step	Owner	Status	Notes
Formalise the retirement with Live Services by raising an RFC. Use CHG0039065 for your template if required This link[1176] should create your RFC template: Live Services can be contacted at liveservices.central-changemgmt@nhs.net[1177]		TO DO	
If delaying the retirement, check usage and communicate with stakeholders, including notice of any reduction in support levels.		TO DO	
Formalise the retirement with Live Services again (if previously delayed) by raising the RFC. *(optional)*		TO DO	
Communicate the revised support agreement		TO DO	
De-authorise all connected apps for the API(s).		TO DO	
Update the API documentation. State that: • the API is retired • what alternative API(s) to use instead		TO DO	
Update the API status in the API catalogue spreadsheet		TO DO	
Notify stakeholders that retirement is complete.		TO DO	

1176 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsdigitallive.service-now.com%2Fnow%2Fnav%2Fui%2Fclassic%2Fparams%2Ftarget%2Fchange_request.do%253FWIZARD%253Aaction%253Dfollow%2526sys_action%253D%2526sys_id%253D-1%2526sys_target%253D%2526sysparm_query%253Dtype%25253dnormal%2526sysparm_target%253D%2526wiz_action%253Dsysverb_new%2526wiz_collection%253D%2526wiz_collectionID%253D%2526wiz_collection_key%253D%2526wiz_collection_related_field%253D%2526wiz_referring_url%253D%2526wiz_view%253D&data=05%7C01%7Cmatthew.firth1%40nhs.net%7Cb0b90057ae4a4e489f8b08daf7a827f4%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638094598831372490%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=8ZfdBbGt29%2BsB5G6Hdg5t7CnRtefuh749ydyjbDPZ6o%3D&reserved=0

1177 mailto:liveservices.central-changemgmt@nhs.net

Step	Owner	Status	Notes
Decommission the API code.		TO DO	

7.17.7.3 API details

TO DO

The API(s) being considered for deprecation are:

Name	API catalogue link	Details

7.17.7.3.1 Details

TO DO

Interaction ID	Name

7.17.7.4 Reason for deprecation

TO DO

7.17.7.4.1 Reasons:

7.17.7.5 Usage analysis

TO DO

The following table gives details of usage of the API(s).

Supplier	Connecting Apps Authorised	Information Source	Connecting Apps Usage	Information Source	Use Case	Information Source	Connecting App Contact Name	Connecting App Contact Email Address

7.17.7.6 Stakeholders

> **Note**
>
> The below roles are suggestions only and should be updated with the correct stakeholders when you make a copy of this template (as should this notification)

TO DO

Role	Name	Team
Product Owner / Technical Product Owner		
Live services owner		
Technical lead		
Delivery Lead		

7.17.7.7 Deprecation comms plan

TO DO

7.17.8 Sunsetting checklist template (API standards)

Instructions for using this template

Make a copy of this page, change the name to "Sunsetting process checklist - [name of API standard], and save it in your API's confluence space. If you are doing this within the API Management tribe, put the page as a child of APIs | Sunsetting[1178]. Then delete this message.

7.17.8.1 Overview

This page is a record of the sunsetting (deprecation and retirement) of the API standard(s) or API standard version(s) named in the page title.

For details of the process, see sunsetting phases for an API standard (see page 1183).

7.17.8.2 Checklists

7.17.8.2.1 Checklist statuses

Statuses (for cutting & pasting into the checklists):

TO DO

IN PROGRESS

DONE

N/A

1178 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216180507/APIs+Sunsetting

7.17.8.2.2 Deprecation checklist:

Description	Owner	Status	Notes
Propose the deprecation. If the product owner has not requested the API standard deprecation directly, identify them and: 1. Share the reasons for deprecation. 2. Obtain approval to deprecate the API standard from the product owner.		TO DO	
Create a sunsetting process checklist page by making a copy of this page. Put it in your own Confluence space. If you are doing the deprecation process from within the API Management tribe, you can put it as a child of APIs | Sunsetting[1179].		TO DO	
Clearly identify the API standard(s) for deprecation, preferably by providing a link to the API specification in the API catalogue.		TO DO	See the API details below.
Consider whether the API standard can be retired immediately (for example, was the standard ever published? was it experimental?) - if so, skip to the Retirement section below		TO DO	Skip to the Retirement checklist (see page 1197) below
Contact the Standards Directory team and advise them of the proposal to deprecate the standard (Contacts are Bharat Sharma and Katie Wheatley or email england.interop.standards@nhs.net[1180])		TO DO	
Agree the retirement date. (Default is 12 months from deprecation but can be negotiable with good reason)		TO DO	
Find out whether there is a new version of API standard.		TO DO	

1179 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216180507/APIs+Sunsetting
1180 mailto:england.interop.standards@nhs.net

Description	Owner	Status	Notes
Document why the API standard(s) are being considered for deprecation. Choose from the reasons below: • the standard is no longer in use • the standard is in use but is no longer strategic • it has been replaced by a newer standard		TO DO	See the Reason for deprecation (see page 1199) below.
Add your API standard to the sunsetting register - API register - including a link to this API standards - sunsetting process checklist.		TO DO	
Mark the standard as "under review for deprecation" in the API catalogue[1181] and ensure that the Standards Directory[1182] is aligned.		TO DO	
Add an entry to the Interactive product backlog[1183] to say that the API is being considered for deprecation.		TO DO	
Consider creating a comms plan for both internal and external audiences.		TO DO	See Deprecation comms plan below.
Mark the API standard as deprecated - update the API documentation and the Standards directory. In the API documentation, state that: • the standard is deprecated • the standard is still supported and maintained • we are unlikely to make any updates to the standard • we will retire in 12 months from date of deprecation • what alternative standards to use instead		TO DO	
Ask APIM to update the API status on the "API catalogue spreadsheet". This is important because it feeds into our overall stats on the percentage of (live, not deprecated) APIs or standards that are using the API platform!		TO DO	
Comms out to Developer community to let them know it's deprecated.		TO DO	

1181 https://digital.nhs.uk/developer/api-catalogue
1182 https://data.standards.nhs.uk/
1183 https://nhs-digital-api-management.featureupvote.com/

Description	Owner	Status	Notes
Comms out to Developer account holders to let them know it's deprecated.			
Update the interactive product backlog to say it's deprecated. Probably with a comment (the feature won't be "done" until the API is retired).		TO DO	
If the standard cannot be retired immediately, provide 12 months notice		TO DO	
Update the deprecated API standard with the proposed retirement date.		TO DO	

7.17.8.2.3 Retirement checklist

Step	Owner	Status	Notes
Contact the Standards Directory team regarding the retirement of the standard (Contacts are Bharat Sharma and Katie Wheatley or email england.interop.standards@nhs.net[1184])			
Update the API Catalogue. State that: • the API standard is retired • what alternative API(s) to use instead (if there is an alternative)		TO DO	
Update the Standards Directory. State that: • the API standard is retired • what alternative API(s) to use instead (if there is an alternative)		TO DO	
Update the API standards status in the API catalogue spreadsheet		TO DO	

1184 mailto:england.interop.standards@nhs.net

Step	Owner	Status	Notes
Point the API Catalogue to the archived page in the national archives[1185] If the page in the archive needs updating to reflect the current, retired standard, then internet archives can be contacted here[1186]		TO DO	This action can only be completed when the national archives has been updated with the correct version of the spec. This could be as long as 3 months in advance. Schedule monthly checks to follow up/ alternatively contact[1187] them.
Notify stakeholders that retirement is complete. ▪ Developer forum ▪ Developer account holders ▪ Interactive product backlog ▪ Standards directory		TO DO	

7.17.8.3 API standard details

TO DO

The API standard(s) being considered for deprecation are:

Name	API catalogue link	Details

7.17.8.3.1 Details (if applicable to the standard)

TO DO

Interaction/workflow ID	Name

1185 https://webarchive.nationalarchives.gov.uk/ukgwa/*/http:/digital.nhs.uk/
1186 https://www.nationalarchives.gov.uk/contact-us/
1187 https://www.nationalarchives.gov.uk/contact-us/

Interaction/workflow ID	Name

7.17.8.4 Reason for deprecation

TO DO

7.17.8.5 Stakeholders

> **Note**
>
> The below roles are suggestions only and should be updated with the correct stakeholders when you make a copy of this template (as should this notification)

TO DO

Role	Name	Team
Product Owner		
Standards Directory lead		
Technical Lead		
Delivery Lead		

7.17.8.6 Deprecation comms messaging

TO DO

Comms action	Status
Announce deprecation on the developer forum	TO DO
Email to developer account holders	TO DO
Update the interactive product backlog	TO DO
Update the standards directory	TO DO

7.17.8.7 Retirement comms messaging

TO DO

Comms action	Status
Announce retirement on the developer forum	TO DO
Email to developer account holders	TO DO
Update the interactive product backlog	TO DO
Update the standards directory	TO DO

7.17.9 WGLL - sample comms (both internal & external)

7.17.9.1 What good looks like

These sample comms have been sent to stakeholders - both internally and to suppliers - to advise them of our intention to proceed with the sunsetting phase of an API. They include the decision to deprecate, the actions we require to agree a formal retirement date and conversations between the two milestones.

The sunsetting checklist[1188] should be updated using this text as a guide.

These examples are what we think good looks like.

1188 https://nhsd-confluence.digital.nhs.uk/display/APM/API+-+sunsetting+checklist+template

As each sunsetting experience is different, the text below may not be relevant for your particular deprecation, therefore please use this text as a guide.

Action	Example text
Formal deprecation notice (external)	**Formal notice of deprecation of the QASI API** This email is to advise you that we formally deprecated the <insert API name> API. The can see that the status has changed in our API catalogue at https://digital.nhs.uk/developer/api-catalogue/query-accredited-system-information-soap#api-status-and-roadmap. To re-iterate, we have **not yet set a retirement date** for the <inert API name> API – that is the purpose of this consultation with you. However, in order to keep the ball rolling, our standard process is to offer you a **four week window** to respond to the consultation – i.e. to say how long you would need to migrate.

Action	Example text
Formal deprecation notice (internal)	Use the below example for formal deprecation notice to internal colleagues, delete/ replace text as appropriate: **The Gazetteer Service - SOAP API is now formally deprecated.** Gazetteer is an API used to validate and retrieve UK-based postal addresses. For further information, see the updated API spec: Gazetteer Service - SOAP API - NHS Digital[1189]. A deprecated API is still fully available in production and still fully supported, but is not available for onboarding and will eventually be retired. We are deprecating Gazetteer as we are now recommending the Ordnance Survey Places API instead due to it being the UK government standard. For further information, see Ordnance Survey Places API - NHS Digital[1190]. We are engaged with suppliers to discuss their options and how long they would need to migrate. The retirement date has yet to be agreed. **Impact:** 1. Suppliers are now recommended to onboard with the OS Places API 2. There are limited suppliers using Gazetteer - they are Dedalus, The Access group and Kings College, (Vision were looking at using it but hadn't started onboarding) 3. As this is a deprecation event, there is no impact on clinical safety or IG 4. There may be an impact on legal for non-Public Sector suppliers who need to sign a contractor licence with us. This process needs to be agreed. **Next steps:** • Ensure the API is marked as deprecated in ServiceNow (I've given the early heads up to @BARNES, Mark (NHS DIGITAL)) • Consult with suppliers and agree a retirement date • Agree the process for contractor licence usage • Agree and publish the retirement date We are deprecating the Gazetteer Service for external use only - it may continue to be used internally - this is being looked into by the Demographics team. The process checklist for info is here: https://nhsd-confluence.digital.nhs.uk/display/APM/Sunsetting+process+checklist+-+Gazetteer+SOAP
30 day notice reminder (external)	Further to the email below, reminder that the 30-day period of consultation for the retirement of Gazetteer will be up next Weds, 18th August. I'll be in touch again following this date - however, if you have a preferred date in mind prior to that, that would be great. If you have any concerns, let me know.

1189 https://digital.nhs.uk/developer/api-catalogue/gazetteer-service-soap
1190 https://digital.nhs.uk/developer/api-catalogue/ordnance-survey-places-api

Action	Example text
30 day notice expiry (external - informal notice)	We have now passed the 4-week consultation period. Have you had the opportunity to discuss the <insert API name> retirement date? If you could confirm asap, that would be much appreciated.
Formal retirement date confirmation (internal)	
Formal retirement date confirmation (external)	**Formal notice of the retirement date of the Query Accredited System Information SOAP API** Following a period of consultation, I can confirm the retirement date for the Query Accredited System Information SOAP API is 31st October 2024. The updated status has been added to the specification in our API catalogue here[1191]. If you have any questions or concerns, please get in touch.

1191 https://digital.nhs.uk/developer/api-catalogue/query-accredited-system-information-soap

8 API register

Page table of contents:

Sub pages:

- Producer team member audit (see page 1313)
- What Good Looks Like checklist (see page 1336)

8.1 API register - new APIs

This is a list of APIs that have been built (or are in progress) on the API Platform, with links to more details.

If you want to know:

- the status of each API, see https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?rapidView=2826&view=detail&selectedIssue=AMR-7&quickFilter=11061
- which APIs have live traffic, see https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/service (and look at "Requests Count Breakdown by API")

JIRA (entries this column will be be added by APIM)	Name	Description	Link to API confluence pages	Link to API process checklist	Link to API architecture review	Link to GitHub repo	Technical author point of contact	Base path	Further path to end points	In Prod	PR Tests	Build Tests	Release Tests	Results Visible	AccessToken. Tests

AMR-31[1192] - Hello World LIVE	Hello World	Tutorial API - sandbox only, not production	N/A	N/A	N/A	https://github.com/NHSDigital/hello-world-api	Unknown User (sapa9)								
AMR-337[1193] - List Reconciliation API ONHOLD	List Reconciliation	PCRM List Reconciliation API between AWS and DSA	API Process checklist - Demographic Work Item API[1194]												

1192 https://nhsd-jira.digital.nhs.uk/browse/AMR-31?src=confmacro
1193 https://nhsd-jira.digital.nhs.uk/browse/AMR-337?src=confmacro
1194 https://nhsd-confluence.digital.nhs.uk/spaces/PCRM/pages/587374449/API+Process+checklist+-+Demographic+Work+Item+API

AMR-272[1195] - Legitimate Relationship Service (LRS) HL7 V3 RETIRE	Legitimate Relationship Service - HL7 V3 API	There are strict rules about viewing a patient's care records set out in the Care Record Guarantee (https://webarchive.nationalarchives.gov.uk/ukgwa/20130513181549/http://www.nigb.nhs.uk/guarantee) To prevent unauthorized access, a Legitimate Care Record is created every time a Healthcare worker access a record.					Unknown User (misc4)								

1195 https://nhsd-jira.digital.nhs.uk/browse/AMR-272?src=confmacro

AMR-81[1196] - Visitors and Migrants API messaging standard RETIRE	Visitor and Migrant FHIR API	This API was never built and has no endpoints													

1196 https://nhsd-jira.digital.nhs.uk/browse/AMR-81?src=confmacro

AMR-160[1197] - NHS Booking - FHIR API DEPRECATED	NHS Booking	Use this API to retrieve available slots and book an appointment for a specific health and care service, for example, from NHS 111 into an Urgent Treatment Centre (UTC). This is related to AMR-65[1198] - NHS Booking and Referral LIVE					Richard Ward								

1197 https://nhsd-jira.digital.nhs.uk/browse/AMR-160?src=confmacro
1198 https://nhsd-jira.digital.nhs.uk/browse/AMR-65?src=confmacro

AMR-89[1199] - Digital Medicines FHIR API TO SUNSET	Digital Medicine - Unused Cases	No development is started or exists for the following services and all references to these are removed from the API Catalogue[1200] and API Hub[1201] • a medication review • an appliance use review • a New Medicine Service (NMS) appointment	N/A	N/A	NA	API Hub[1202]	Unknown User (sapa 9)								

1199 https://nhsd-jira.digital.nhs.uk/browse/AMR-89?src=confmacro
1200 https://digital.nhs.uk/developer/api-catalogue/digital-medicine-fhir
1201 https://developer.nhs.uk/apis/digitalmedicines-1.2.5-private-beta/
1202 https://developer.nhs.uk/apis/digitalmedicines-1.2.5-private-beta/

AMR-115[1203] - [GP Connect] Direct Care Patient Facing Services (PFS) - Appointment Management ON HOLD	[GP Connect] Direct Care Patient Facing Services (PFS) - Appointment Management	Alternative / long term strategic API for appointment management via PFS	GPC Discovery PFS API Appointments - Direct Care APIs - CDT Confluence (digital.nhs.uk)[1204]	API M Checklist - Direct Care APIs - CDT Confluence (digital.nhs.uk)[1205]	Not started yet	Not started yet	Unknown User (jory2)								

1203 https://nhsd-jira.digital.nhs.uk/browse/AMR-115?src=confmacro
1204 https://nhsd-confluence.digital.nhs.uk/display/DCA/GPC+Discovery+PFS+API+Appointments
1205 https://nhsd-confluence.digital.nhs.uk/display/DCA/APIM+Checklist

AMR-13[1206] - Personal Demographic Service (PDS) [application restricted access] LIVE	Personal Demographics Service	Get and update patients' personal data, such as name and address.	Personal Demographics Service FHIR API[1209] - for inception to discovery Alpha onward tracked here: AMR-14 - confluence.digital.nhs.uk/display/DEMGRPH/Patient+Access+Mode+API+checklist[1210]	API process checklist - Personal Demographics Service API - API Management - CDT Confluence (digital.nhs.uk)[1211]	PDS API architecture review[1212]	https://github.com/NHSDigital/personal-demographics-service-api	Unknown User (misc4)	/personal-demographics	/FHIR/R4	Y	Y	Lint FHIR Validator	Pytest	Grouped in script All visible	N - just set to pass? Perf.

1211 https://nhsd-confluence.digital.nhs.uk/display/APM/API+process+checklist+-+Personal+Demographics+Service+API#APIprocesschecklistPersonalDemographicsServiceAPI-4.PatientAccessYellowINPROGRESS

1212 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/151425723/PDS+API+architecture+review

AMR-2[1207] - Personal Demographic Service (PDS) [healthcare worker access]

PRIVATE BETA

1207 https://nhsd-jira.digital.nhs.uk/browse/AMR-2?src=confmacro

AMR-14[1208] - Personal Demographic Service (PDS) [patient access] PRIVATE BETA															

1208 https://nhsd-jira.digital.nhs.uk/browse/AMR-14?src=confmacro

AMR-32[1213] - Electronic Prescription Service (EPS) FHIR facade [prescribe - primary care] WITHDRAWN	Electronic Prescription Service (prescribe)	Prescribe for both primary and secondary care	EPS Prescribe API[1215]	API process checklist (EPS prescribing API) (see page 1204)	Architecture Review (EPS prescribing API) (see page 1204)	https://github.com/NHSDigital/electronic-prescription-service-api	Unknown User (sapa9)	/electronic-prescriptions	/FHIR/R4	Y	Y	Lint FHIR Validator	PACT "Smoke"	Total sonly	N

1213 https://nhsd-jira.digital.nhs.uk/browse/AMR-32?src=confmacro
1215 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/117638739/EPS+Prescribe+API

AMR-12[1214] - Electronic Prescription Service (EPS) FHIR facade [prescribe] LIVE															

1214 https://nhsd-jira.digital.nhs.uk/browse/AMR-12?src=confmacro

AMR-15[1216] - Electronic Prescription Service (EPS) FHIR facade [dispense] LIVE	Electronic Prescription Service (dispense)	Download prescriptions ready to be dispensed, mark prescriptions as	EPS Dispense API[1217]	API process checklist (EPS dispensing API) (see page 1204)	TO DO	https://github.com/NHSDigital/electronic-prescription-service-api	Unknown User (sapa9)	`/electronic-prescriptions`	`/FHIR/R4`						

1216 https://nhsd-jira.digital.nhs.uk/browse/AMR-15?src=confmacro
1217 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/155877843/EPS+Dispense+API

AMR-23[1218] - Electronic Prescription Service (EPS) Tracker ONHOLD	Electronic Prescription Service (tracker)	Query spine for things like prescription status.	TO REVIEW - EPS Query API[1219]	API process checklist (EPS tracker API) #Inception (see page 1204)	TO DO	https://github.com/NHSDigital/electronic-prescription-service-api	Unknown User (sapa9)	/electronic-prescriptions	/FHIR/R4						

1218 https://nhsd-jira.digital.nhs.uk/browse/AMR-23?src=confmacro
1219 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/183273574/TO+REVIEW+-+EPS+Query+API

AMR-4[1220] - Ambulance Data Submission LIVE	Ambulance Analytics API	Designed to bridge Ambulance Data & Emergency Care Data for complex data studies to be carried out. Providing better care going forwards.	Ambulance analytics API[1221]	API process checklist (AA API)[1222]	Ambulance Analytics API architecture review[1223]	https://github.com/NHSDigital/ambulance-analytics-api	Unknown User (sapa9)	/ambulance-data-submission[1224]	-	Y	Y	Lint Sandbox	Newman "Postman Tests"	Y	N (about to be added)

1220 https://nhsd-jira.digital.nhs.uk/browse/AMR-4?src=confmacro
1221 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/105576368/Ambulance+Analytics+API
1222 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/108275017/API+process+checklist+AA+API
1223 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141202863/Ambulance+Analytics+API+architecture+review
1224 https://api.service.nhs.uk/ambulance-data-submission

AMR-8[1225] - Summary Care Record (SCR) [GPIT Futures] PRIVATE BETA	Summary Care Record API	Simplify and thus accelerate SCR integration with Spine TMS and Spine ACS. The MVP delivery is aimed at New Market Entrants, looking for integration with the service providing a middleware component which hides complexity from the GP system supplier	Summary Care Record API[1226]	API process checklist - SCR API[1227]	SCR API architecture review[1228]	https://github.com/NHSDigital/summary-care-record-api	Unknown User (misc4)	/summary-care-record	/FHIR/R4	N	N	Lint	N	-	-

1225 https://nhsd-jira.digital.nhs.uk/browse/AMR-8?src=confmacro
1226 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/117668210/Summary+Care+Record+API
1227 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/193013374/API+process+checklist+-+SCR+API
1228 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/151422861/SCR+API+architecture+review

AMR-7[1229] - Spine Directory Service (SDS) [GPIT Futures] PRIVATE BETA	Spine Directory Service (SDS) API	Search and retrieve spine endpoint information	Spine Directory Services (SDS) API[1230]	API process checklist - SDS API[1231]	SDS API architecture review[1232]	https://github.com/NHSDigital/spine-directory-service-api	Unknown User (misc4)	`/spine-directory`	/FHIR/R4	Y	Y	Lint	Pytest "smoketest", "e2e", "security"	Y	Y

1229 https://nhsd-jira.digital.nhs.uk/browse/AMR-7?src=confmacro
1230 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119736634/Spine+Directory+Services+SDS+API
1231 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125406498/API+process+checklist+-+SDS+API
1232 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141201030/SDS+API+architecture+review

AMR-5[1233] - Reasonable Adjustment API TO SUNSET	Reasonable Adjustment (RA) Flag	Search, retrieve and update a flag within Spine regarding patients with a learning disability who require reasonable adjustments to be made in order to accommodate their needs when proving care and services.	Reasonable Adjustment Flag API[1234]	API process checklist - RA Flag API - OLD[1235]	RA Flag API architecture review[1236]	https://github.com/NHSDigital/reasonable-adjustments-api	Unknown User (sapa9)	/reasonable-adjustment-flag	/FHIR/STU3	N	Y	Lint	Pytest "Integration"		In part

1233 https://nhsd-jira.digital.nhs.uk/browse/AMR-5?src=confmacro
1234 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/110879283/Reasonable+Adjustment+Flag+API
1235 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/122356681/API+process+checklist+-+RA+Flag+API+-+OLD
1236 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/141202834/RA+Flag+API+architecture+review

AMR-18[1237] - NHS App (notifications) PUBLIC BETA	NHS App	Allow 3rd parties who are already onboarded with the NHS App to send notifications and messages to registered users of the NHS App.	https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=86061642	https://nhsd-confluence.digital.nhs.uk/display/NO/API+process+checklist+-+The+NHS+App	https://nhsd-confluence.digital.nhs.uk/display/NO/API+Producer+architecture+template+-+The+NHS+App	https://github.com/NHSDigital/nhs-app-api	Unknown User (misc4)	/nhs-app	/communication/in-app/FHIR/R4	Y	.NET Core "integration"	Lint	N	Only for PR	Double Check "missing" version

1237 https://nhsd-jira.digital.nhs.uk/browse/AMR-18?src=confmacro

AMR-6[1238] - Child Protection Information Sharing (CP-IS) FHIR WITHDRAWN	Child Protection Information Sharing (CPIS)	Allow 3rd parties to check if a patient has a child protection plan and to query healthcare data related to that plan.	Child Protection - Information Sharing (CP-IS) FHIR API[1239]	API process checklist - (CP-IS API)[1240]	CP-IS API architecture review[1241]	TODO		No Repo							
AMR-16[1242] - e-Referral Service (eRS) LIVE	e-Referral Service API	Allow 3rd parties to create, retrieve and manage e-Referrals.	APIM (API Management) for e-RS[1243]	APIM process checklist (e-Referral Service)[1244]	e-RS APIM Architecture Review[1245]	https://github.com/NHSDigital/e-referrals-service-api	Unknown User (misc4)								

1238 https://nhsd-jira.digital.nhs.uk/browse/AMR-6?src=confmacro
1239 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119759050/Child+Protection+-+Information+Sharing+CP-IS+FHIR+API
1240 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/147030383/API+process+checklist+-+CP-IS+API
1241 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/144739313/CP-IS+API+architecture+review
1242 https://nhsd-jira.digital.nhs.uk/browse/AMR-16?src=confmacro
1243 https://nhsd-confluence.digital.nhs.uk/spaces/ES/pages/164529871/APIM+API+Management+Discovery
1244 https://nhsd-confluence.digital.nhs.uk/spaces/ES/pages/178948222/APIM+process+checklist+e-Referral+Service
1245 https://nhsd-confluence.digital.nhs.uk/spaces/ES/pages/199690273/e-RS+APIM+Architecture+Review

AMR-42[1246] - Signing Service (local) ONHOLD AMR-43[1247] - Signing Service (remote) BETATECHNICALLYLIVE	Signing Service API	Allows users to send data to the API to be stored whilst the user authenticates and NHS identity then signs the data for that user then returns it to them.	Signing Service API[1248] API process checklist - Signing Service (Refreshed)[1249] - New checklist	[SUPERSEDED] API Process Checklist (Signing API)[1250]	TODO Aubyn Crawford to complete (I think?)	https://github.com/NHSDigital/eps-local-signing-poc (private repo)	Unknown User (sapa9)	/signing-service	-	Y	Selenium PACT "selenium","pact"	Lint Junit Visible in DevOps	N	N	?

1246 https://nhsd-jira.digital.nhs.uk/browse/AMR-42?src=confmacro
1247 https://nhsd-jira.digital.nhs.uk/browse/AMR-43?src=confmacro
1248 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/164299846/Signing+Service+API
1249 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/456436038/API+process+checklist+-+Signing+Service+Refreshed
1250 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/164299848/SUPERSEDED+API+Process+Checklist+Signing+API

AMR-45[1251] - COVID-19 Test Results DEPRECATED	COVID-19 Test Results API	Return a list of antibody and antigen (PCR and LFT) C19 test results for a given citizen based on NHS number and verified by additional demographic data. The initial use case and therefore consumers of this API are GP Suppliers.	COVID-19 Test Results API[1252] https://blue squirrel.atlassian.net/wiki/spaces/P1/pages/326598771/ODP+45+-+Clinician+and+Citizen+Access+to+Results+API#Draft-API-Contract	https://blue squirrel.atlassian.net/wiki/spaces/P1/pages/4919802722/API+process+checklist+-+getResultsByNHSNumber+API	Architecture Review (COVID-19 Test Results API)[1253]	https://github.com/NHSDigital/covid-19-test-result	Unknown User (misc4)	/covid-19-test-result	/FHIR/R4	Y	Pytest "e2e" and "smoke"	Lint	Pytest "smoke"	Y	In part

1251 https://nhsd-jira.digital.nhs.uk/browse/AMR-45?src=confmacro
1252 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/182429378/COVID-19+Test+Results+API
1253 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/173842661/Architecture+Review+COVID-19+Test+Results+API

AMR-62[1254] - Immunisation History API LIVE	Immunisation History API	Allows citizens to get details of their own vaccination status.	Immunization History API (see page 1204)	API process checklist - Immunisation History API[1255]	Immunisation History API architecture review[1256]	https://github.com/NHSDigital/immunisation-history-api	Unknown User (misc4)	/immunisation-history	/FHIR/R4	Y	Pytest "e2e" and "smoke"	Lint Sandbox Visible in DevOps	Pytest "smoke"	Y	Almost all coverage Although automated on high envs, ONLY Basic

1254 https://nhsd-jira.digital.nhs.uk/browse/AMR-62?src=confmacro
1255 https://nhsd-confluence.digital.nhs.uk/spaces/DDAP/pages/190072175/API+process+checklist+-+Immunisation+History+API
1256 https://nhsd-confluence.digital.nhs.uk/spaces/DDAP/pages/193015595/Immunisation+History+API+architecture+review

AMR-118[1257] - COVID Medical Exemptions API RETIRE	COVID Medical Exemptions API	Allows citizens to get details of their own COVID Medical Exemptions.	COVID-19 Medical Exemptions - FHIR API[1258]	API process checklist - Exemptions API[1259]		https://github.com/NHSDigital/covid-medical-exemptions-api	Unknown User (misc4)	TBC	TBC	N	TBC	TBC	TBC	TBC	TBC

1257 https://nhsd-jira.digital.nhs.uk/browse/AMR-118?src=confmacro
1258 https://nhsd-confluence.digital.nhs.uk/spaces/DDAP/pages/324516045/COVID-19+Medical+Exemptions+-+FHIR+API
1259 https://nhsd-confluence.digital.nhs.uk/spaces/DDAP/pages/324515549/API+process+checklist+-+Exemptions+API

AMR-73[1260] - Healthcare Worker API (21/22) WITHDRAWN	Healthcare Professional	Get details for a healthcare professional, initially from SDS but maybe eventually from other back-end data sources.	Healthcare Professional API (see page 1204)	API process checklist - Healthcare Professionals API (see page 1204)	TODO	TODO									
AMR-271[1261] - *DUPLICATE* Healthcare Worker API ONHOLD															

1260 https://nhsd-jira.digital.nhs.uk/browse/AMR-73?src=confmacro
1261 https://nhsd-jira.digital.nhs.uk/browse/AMR-271?src=confmacro

AMR-230[1262] - Organisation Data Service API- (ODS) [Tactical] WITHDRAWN	ODS API (Tactical)	Provides access to the organisation reference data published by the Organisation Data Service (ODS). Note this will consist of the onboarding of an existing ODS API Suite. Currently this consists of two endpoints: one for a reduced-scope data set aimed at transactional interactions that returns results in FHIR, and the other providing access to the full ODS dataset that returns XML/	ODS API Suite - Organisation Data Service - CDT Confluence (digital.nhs.uk)[1263]	API process checklist - ODS API Suite (see page 1204)	TODO	TODO	Unknown User (mipr) Unknown User (chcr4)									

1262 https://nhsd-jira.digital.nhs.uk/browse/AMR-230?src=confmacro
1263 https://nhsd-confluence.digital.nhs.uk/display/ODS/ODS+API+Suite

		JSON. Despite there being an existing service there are currently unknowns that need resolving; these are dependent on which features of the existing service can be kept as-is and which might need to be altered prior to onboarding.													

AMR-72[1264] - FHIR Converter (medsmgmt) ONHOLD	Transformer API	Convert FHIR resources between versions, e.g. STU3 to R4	Transformation Engine[1265]	API process checklist - Transformation Engine[1266]	Architecture review template - Transformer API[1267]	TODO	Tim Coates								
	MESH API	Specification only	N/A	N/A	N/A	https://github.com/NHSDigital/mesh-api-specification									

1264 https://nhsd-jira.digital.nhs.uk/browse/AMR-72?src=confmacro
1265 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/209591715/Transformation+Engine
1266 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/227247490/API+process+checklist+-+Transformation+Engine
1267 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/209602815/Architecture+review+template+-+Transformer+API

AMR-57[1268] - GP Registrations Management Information REST API PUBLIC BETA	GP Registrations MIA PI	Capture Management Information from Foundation Suppliers in order to diagnose the health the GP2GP service, to improve the Patient Migration experience.	https://gpitbjss.atlassian.net/wiki/spaces/TW/pages/11583520873/GP+Registrations+Management+Information+API	https://gpitbjss.atlassian.net/wiki/spaces/TW/pages/11549474855/GP+Registrations+Management+Information+-+API+Process+Checklists	https://gpitbjss.atlassian.net/wiki/spaces/TW/pages/1152997806 3/GP+Registrations+Management+Information+-+API+architecture+review	https://github.com/NHSDigital/gp-registrations-management-information-api	Unknown User (iagr2) Unknown User (misc4)								

1268 https://nhsd-jira.digital.nhs.uk/browse/AMR-57?src=confmacro

AMR-109[1269] - Electronic Transmission of Prescriptions Web Services - REST API DEPRECATED	ETP API	Use this API to access dispenser information for a patient via NHS UK Web Services.	EB-1319 Electronic Transmission of Prescriptions (ETP)[1270] IN PROGRESS	API - ETP Checklist[1271] IN PROGRESS	API - ETP Architecture Review[1272] IN PROGRESS	TODO	Unknown User (kibr)								

1269 https://nhsd-jira.digital.nhs.uk/browse/AMR-109?src=confmacro
1270 https://nhsd-confluence.digital.nhs.uk/spaces/CIT2/pages/206292314/EB-1319+Electronic+Transmission+of+Prescriptions+ETP
1271 https://nhsd-confluence.digital.nhs.uk/spaces/CIT2/pages/265853127/API+-+ETP+Checklist
1272 https://nhsd-confluence.digital.nhs.uk/spaces/CIT2/pages/280953193/API+-+ETP+Architecture+Review

AMR-65[1273] - NHS Booking and Referral API LIVE	NHS Booking and Referral API	An API and services to support a universal standardised way of performing booking and referral workflows. Initially focussing on 111→ED booking and referrals	Architecture & Design[1274]	API Management - process checklist - NHS Booking and Referrals[1275]	BaRS - API architecture review[1276]	TODO	David Ruddy Unknown User (sapa9) Unknown User (misc4)								

1273 https://nhsd-jira.digital.nhs.uk/browse/AMR-65?src=confmacro
1274 https://nhsd-confluence.digital.nhs.uk/spaces/PAAB/pages/83091029/Architecture+Design
1275 https://nhsd-confluence.digital.nhs.uk/spaces/PAAB/pages/216185630/API+Management+-+process+checklist+-+NHS+Booking+and+Referrals
1276 https://nhsd-confluence.digital.nhs.uk/spaces/PAAB/pages/266994194/BaRS+-+API+architecture+review

AMR-108[1277] - EeRS (Eyecare e-Referral Service) LIVE	Eyecare e-Referral Service API	An API standard for eyecare referrals from primary to secondary care	Eyecare e-Referral Service API standard[1278]	API process checklist - Eyecare e-Referral Service API[1279]	TODO	TODO	TODO								

1277 https://nhsd-jira.digital.nhs.uk/browse/AMR-108?src=confmacro
1278 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/307600310/Eyecare+e-Referral+Service+API+standard
1279 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/307600315/API+process+checklist+-+Eyecare+e-Referral+Service+API

AMR-114[1280] - GP Connect Patient Facing Services (PFS) - Access Record API PRIVATE BETA	GP Connect Patient Facing Services (PFS) - View Record	This API is for consumers providing a patient facing view of the GP record (The links refering to this entry refer to the earlier discovery work)	PFS View Record API[1281]	API process checklist - PFS Access Record API[1282]	PFS View Record API Architecture Review[1283]	TODO	TODO								

1280 https://nhsd-jira.digital.nhs.uk/browse/AMR-114?src=confmacro
1281 https://nhsd-confluence.digital.nhs.uk/display/DCA/PFS+-+View+Record+-+All+Designs
1282 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/307146551/API+process+checklist+-+PFS+Access+Record+API
1283 https://nhsd-confluence.digital.nhs.uk/display/DCA/PFS+View+Record+API+architecture+review+template

AMR-114[1284] - GP Connect Patient Facing Services (PFS) - Access Record API PRIVATE BETA	GP Connect Patient Facing Services (PFS) - Access Record Structured API	GP Connect Patient Facing Services - Access Record Structured API (**This is the current entry**) AMB-1352[1285]	GP Connect PFS API Specs Alpha[1286]	PFS API Specs Alpha - Review Log[1287] API process checklist - PFS Access Record API[1288]		https://github.com/NHSDigital/gp-connect-access-record-fhir									

1284 https://nhsd-jira.digital.nhs.uk/browse/AMR-114?src=confmacro
1285 https://nhsd-jira.digital.nhs.uk/browse/AMB-1352
1286 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/503026107/GP+Connect+PFS+API+Specs+Alpha
1287 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/520323681/PFS+API+Specs+Alpha+-+Review+Log
1288 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/307146551/API+process+checklist+-+PFS+Access+Record+API

AMR-334[1289] - GP Connect Patient Facing Services (PFS) - User Permissions PRIVATE BETA	GP Connect Patient Facing Services (PFS) - User Permissions API	GP Connect Patient Facing Services - User Permissions API AMB-1394[1290]	User Permissions[1291]	API specification checklist - GP Connect User Permissions[1292] GP Connect PFS User Permissions API process checklist[1293]	API architecture review - GP Connect User Permissions[1294]	https://github.com/NHSDigital/gp-connect-user-permissions									

1289 https://nhsd-jira.digital.nhs.uk/browse/AMR-334?src=confmacro
1290 https://nhsd-jira.digital.nhs.uk/browse/AMB-1394
1291 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/507544757/User+Permissions
1292 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/518656050/API+specification+checklist+-+GP+Connect+User+Permissions
1293 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/583143728/GP+Connect+PFS+User+Permissions+API+process+checklist
1294 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/518657722/API+architecture+review+-+GP+Connect+User+Permissions

AMR-335[1295] - GP Connect Patient Facing Services (PFS) - Prescriptions API PRIVATE BETA	GP Connect Patient Facing Services (PFS) - Prescriptions API	GP Connect Patient Facing Services (PFS) - Prescriptions API AMB-1390[1296]	GP Connect PFS Prescription Management Alpha[1297]	GP Connect PFS Prescriptions API process checklist[1298]	GP Connect PFS Prescription Management FHIR API Architecture Review[1299]	https://github.com/NHSDigital/gp-connect-prescriptions-management-fhir									

1295 https://nhsd-jira.digital.nhs.uk/browse/AMR-335?src=confmacro
1296 https://nhsd-jira.digital.nhs.uk/browse/AMB-1390
1297 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/493112485/GP+Connect+PFS+Prescription+Management+Alpha
1298 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/571327901/GP+Connect+PFS+Prescriptions+API+process+checklist
1299 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/529503110/GP+Connect+PFS+Prescription+Management+FHIR+API+Architecture+Review

AMR-336[1300] - GP Connect Patient Facing Services (PFS) - Appointments API ONHOLD	GP Connect Patient Facing Services (PFS) - Appointments API	GP Connect Patient Facing Services (PFS) - Appointments API AMB-1392[1301]		GP Connect PFS Appointments API process checklist[1302]		https://github.com/NHSDigital/gp-connect-appointments-management-fhir									

1300 https://nhsd-jira.digital.nhs.uk/browse/AMR-336?src=confmacro
1301 https://nhsd-jira.digital.nhs.uk/browse/AMB-1392
1302 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/587379779/GP+Connect+PFS+Appointments+API+process+checklist

AMR-116[1303] - Health Research Data Catalogue (Collibra HRD UK Integration) LIVE	Collibra/ HDR Innovation Gateway	Generic API endpoint to automatically submit NHSD metadata for HDR UK Innovation Gateway publication	HDR UK - Data Architecture - CDT Confluence (digital.nhs.uk)[1304]	API process checklist Collibra/ HDR UK Integration - Data Architecture - CDT Confluence (digital.nhs.uk)[1305]	API architecture review template[1306]										

1303 https://nhsd-jira.digital.nhs.uk/browse/AMR-116?src=confmacro
1304 https://nhsd-confluence.digital.nhs.uk/display/DA/HDRUK
1305 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=320997314
1306 https://nhsd-confluence.digital.nhs.uk/spaces/DA/pages/343672579/API+architecture+review+template

AMR-53[1307] - Medical Device Information System (MDIS) ON HOLD	MDIS - Medical Devices Information Systems	As at 20/10/21 it is still to be decided which of the potential MDIS related APIs is to be created first. The current potential first APIs include: • Procedure and device data from Trusts • Device data from MHRA	API use in MDIS - Data Services Alliance - CDT Confluence (digital.nhs.uk)[1308]	API Process Checklist MDIS implantable device data landing - Data Services Alliance - CDT Confluence (digital.nhs.uk)[1309]	TO DO	TO DO -->									

1307 https://nhsd-jira.digital.nhs.uk/browse/AMR-53?src=confmacro
1308 https://nhsd-confluence.digital.nhs.uk/display/DSA/API+use+in+MDIS
1309 https://nhsd-confluence.digital.nhs.uk/display/DSA/API+Process+Checklist+MDIS+implantable+device+data+landing

		• Field Safety Notice data from MHRA This field will be updated once a decision is taken.													
AMR-130[1310] - Dose to Text API ONHOLD	Dose to Text	This API is to allow structured messages to be sent via Prescribers	Dose to text[1311]	API process checklist - dose to text API[1312]	TO DO	TO DO	Unknown User (sapa9)			N					

1310 https://nhsd-jira.digital.nhs.uk/browse/AMR-130?src=confmacro
1311 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/265859831/Dose+to+text
1312 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/328967358/API+process+checklist+-+dose+to+text+API

AMR-37[1313] - GP2GP HL7 V3 BACKLOG	GP2GP HL7 V3														

1313 https://nhsd-jira.digital.nhs.uk/browse/AMR-37?src=confmacro

AMR-227[1314] - Patient care aggregator API (Wayfinder) PRIVATE BETA	Patient Care Aggregator	This API will integrate with the NHS eReferrals service and supplier Patient Engagement Portal systems to aggregate and coherently display referrals and appointments. It's primary consumer will be the NHS app.	Wayfinder Home[1315]	API process checklist - Patient Care Aggregator[1316]	Patient Care Aggregator Architecture Review[1317]	TO DO	Unknown User (mavi6)	TO DO							

1314 https://nhsd-jira.digital.nhs.uk/browse/AMR-227?src=confmacro
1315 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/376508329/Wayfinder+Home
1316 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/376515393/API+process+checklist+-+Patient+Care+Aggregator
1317 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/376515462/Patient+Care+Aggregator+Architecture+Review

AMR-161[1318] - [GP Connect] Single Access Record Viewing (SARV) ONHOLD	SARVSAccessRecordHTML	HTML views of the GP record rendered centrally, using GP Connect Access Record Structured as a data source. Replaces the existing GP Connect Access Record HTML API	SARVS (HTML from Structured)[1319]	SARVS (HTML from Structured)[1320]			Unknown User (mast18)								

1318 https://nhsd-jira.digital.nhs.uk/browse/AMR-161?src=confmacro
1319 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=371164014
1320 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=371164014

AMR-278[1321] - e-Referrals Service Patient Care API PRIVATE BETA	e-Referral Service Patient Care API	This API will integrate with Patient Care Aggregator to surface patient data from e-Referral Service	Wayfinder[1322]	API process checklist - Patient Care e-Referral Service[1323]		https://github.com/NHSDigital/e-referrals-service-patient-care-api/		TBC							
AMR-249[1324] - GP data type 1 opt-out API ONHOLD	Type 1 Opt-Out	This API will allow authorised consumers to send and receive patient Type 1 Opt-Out statuses to and from GP System Suppliers.	Type 1 Opt-Out[1325]	Type 1 - API process checklist[1326]											

1321 https://nhsd-jira.digital.nhs.uk/browse/AMR-278?src=confmacro
1322 https://nhsd-confluence.digital.nhs.uk/display/WAYF
1323 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/394533815/API+process+checklist+-+Patient+Care+e-Referral+Service
1324 https://nhsd-jira.digital.nhs.uk/browse/AMR-249?src=confmacro
1325 https://nhsd-confluence.digital.nhs.uk/spaces/FR/pages/391978434/Type+1+Opt-Out
1326 https://nhsd-confluence.digital.nhs.uk/spaces/FR/pages/415598849/Type+1+-+API+process+checklist

AMR-250[1327] - Service Search (Directory of Health Care Services) API BUILD BETA	Service Search API	API to find information about NHS services, including lists of NHS organisation types, organisations, and coronavirus (COVID 19) walk-in sites. • search for NHS organisation types[1328], for example, GP practices, hospitals • search for NHS organisation	https://nhsd-confluence.digital.nhs.uk/display/DDAP/XXX+-+Solution+Design+-+Search+Service	API process checklist (Directory of Healthcare Services API)[1332]		https://github.com/NHSDigital/service-search-api									

1327 https://nhsd-jira.digital.nhs.uk/browse/AMR-250?src=confmacro
1328 https://developer.api.nhs.uk/nhs-api/documentation/service-search-organisation-types
1332 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/520331978/API+process+checklist+Directory+of+Healthcare+Services+API

		s[1329] that provide services listed in NHS service near you[1330], for example, Leeds General Infirmary • search for coronavirus (COVID 19) walk-in sites[1331]													

1329 https://developer.api.nhs.uk/nhs-api/documentation/service-search-organisations
1330 https://www.nhs.uk/nhs-services/services-near-you/
1331 https://developer.api.nhs.uk/nhs-api/documentation/service-search-covid-walkin

AMR-9[1333] - Organisation Data Terminology API-[ODS Strategic] LIVE	ODS (strategic)	Strategic Review of options relating to ODS API development and hosting	ODS API Strategic - Organisation Data Service - CDT Confluence (digital.nhs.uk)[1334]	API process checklist (Strategic) - Organisation Data Service - CDT Confluence (digital.nhs.uk)[1335]	TO DO	TO DO	vikarora								
AMR-321[1336] - GP Practice self serve enrolment DELIVERY BACKLOG	Register with a GP management (TBD)	API for Register with a GP with endpoints to • Add a GP practice • Update a GP Practice • Remove a GP Practice	DD130 - Detailed Design - GP Search API[1337]	GP - Registration - Onboarding on API Platform (see page 1204)	TO DO	TO DO	Jens Møller Kursch	TO DO							

1333 https://nhsd-jira.digital.nhs.uk/browse/AMR-9?src=confmacro
1334 https://nhsd-confluence.digital.nhs.uk/display/ODS/ODS+API+Strategic
1335 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=452006427
1336 https://nhsd-jira.digital.nhs.uk/browse/AMR-321?src=confmacro
1337 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/431898486/DD130+-+Detailed+Design+-+GP+Search+API

AMR-323[1338] - EPS FHIR Next Gen WITHDRAWN	EPS Next Gen APIs	EPS Futures is a restful single item FHIR API designed to change the way suppliers use EPS	Electronic Prescription Service FHIR APIs[1339] EPS Next Gen - Full FHIR[1340]	API process checklist (EPS Next Gen)[1341]	TO DO	TO DO									

1338 https://nhsd-jira.digital.nhs.uk/browse/AMR-323?src=confmacro
1339 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/117637556/Electronic+Prescription+Service+FHIR+APIs
1340 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/373752864/EPS+Next+Gen+-+Full+FHIR
1341 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/419595846/API+process+checklist+EPS+Next+Gen

AMR-322[1342] - Published Cohort Definition API ONHOLD] AMR-343[1343] - Cohort Definition Mangement API ONHOLD	Published Cohort Definitions Cohort Definition Management	Read-only API to allow clients to view condition groups and conditions which make up the anatomy of a cohort definition that has been previously approved and published.	Cohorting as a Service (CaaS) - Risk Stratification - CDT Confluence (digital.nhs.uk)[1344]	CaaS - API process checklist - Risk Stratification - CDT Confluence (digital.nhs.uk)[1345]	TO DO	TO DO	James Bancroft	TO DO		N						

1342 https://nhsd-jira.digital.nhs.uk/browse/AMR-322?src=confmacro
1343 https://nhsd-jira.digital.nhs.uk/browse/AMR-343?src=confmacro
1344 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=384401731
1345 https://nhsd-confluence.digital.nhs.uk/display/RIS/CaaS+-+API+process+checklist

		API to facilitate the design, modelling and management of cohort definitions including the ability to amend the condition groups and conditions which make up that definition. This also covers the workflow associated with managing the state of a cohort definition including approval and publication.													

AMR-326[1346] - National Record Locator Cloud BETA TECHNICALLY LIVE	NRL Futures (National Record Locator Futures	API to provide pointer data to consumers so that they may access documents stored by producers (e.g Ambulance staff "Consumer" access pointer data to Mental health crisis plan held by Mental health trust "Producer")	Archived National Record Locator Futures (Lambda)[1347]	API process checklist - NRL Futures[1348]	API architecture review checklist - NRL Futures[1349]											

1346 https://nhsd-jira.digital.nhs.uk/browse/AMR-326?src=confmacro
1347 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/461247424/Archived+National+Record+Locator+Futures+Lambda
1348 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/476979455/API+process+checklist+-+NRL+Futures
1349 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/488461594/API+architecture+review+checklist+-+NRL+Futures

Vision: To better inform clinical decision making across all health and care settings by enabling authorised Health and Care professionals **and patients** to access all relevant patient information at the point of care.

In order to meet this vision and scale to the number of documents that exist within the NHS

- Low friction onboarding

		• Horizontal scaling to meet the demand • Improve the discoverability of patient data • To be seen as a trusted **index** of patient data													

AMR-327[1350] - NHS Website Content API LIVE	NHS Website Content API	APIs/a consolidated API that surfaces content from the NHS website. This is currently accessed at developer.api.nhs.uk. Around 120 users call these APIs regularly, there are approximately 3.2 million calls to these APIs per month. Content APIs currently available are: • Health A-Z • Medicines • Mental Health • Live Well	Syndication - Syndication - CDT Confluence (digital.nhs.uk)[1351]		API process checklist - NHS.uk content - Syndication - CDT Confluence (digital.nhs.uk)[1352]		Unknown User (brmo3) Adrian Parr									

1350 https://nhsd-jira.digital.nhs.uk/browse/AMR-327?src=confmacro
1351 https://nhsd-confluence.digital.nhs.uk/display/SYN/Syndication
1352 https://nhsd-confluence.digital.nhs.uk/display/SYN/API+process+checklist+-+NHS.uk+content

		• Common Health Questions • Pregnancy																

		• Video (may be retired) These APIs enable users (developers of digital health platforms - NHS, charities, local authorities, commercial sector) to republish trusted, up-to-date textual content, images and video about various health topics so that this information is available to as many end users as possible.														

AMR-105[1353] - Immunisation FHIR API PRIVATE BETA	Immunisation FHIR API	This will be the primary immunisation API for the vaccination programme with the intention to allow CRUD (Create Read Update Delete) of immunisation records on the National Immunisation Record to be held on DPS	Immunisations FHIR API - Vaccination Programme - CDT Confluence (digital.nhs.uk)[1354] API Process Checklist[1355]	Immunisation API - Vaccination Programme - CDT Confluence (digital.nhs.uk)[1356]			Athena Arvaniti Unknown User (thbu1)								

1353 https://nhsd-jira.digital.nhs.uk/browse/AMR-105?src=confmacro
1354 https://nhsd-confluence.digital.nhs.uk/display/Vacc/Immunisation+FHIR+API
1355 https://nhsd-confluence.digital.nhs.uk/display/Vacc/Immunisation+FHIR+API+-+API+Process+Checklist
1356 https://nhsd-confluence.digital.nhs.uk/display/Vacc/Immunisation+API

AMR-332[1357] - NHS Notify (previously Communications Manager) PUBLIC BETA	Communications Manager API	The Communications Manager API will allow services to send requests into the Communications Manager solution so that messages can be sent to their intended recipients.	NHS Notify | Files transferred between MESH and NHS Notify[1358]	NHS Notify | API Process Checklist[1359]	NHS Notify | API Architecture Review Checklist[1360]	communications-manager[1361]	Mark Pullan Unknown User (phsk1)	/comms							

1357 https://nhsd-jira.digital.nhs.uk/browse/AMR-332?src=confmacro
1358 https://nhsd-confluence.digital.nhs.uk/spaces/RIS/pages/604907080/NHS+Notify+Files+transferred+between+MESH+and+NHS+Notify
1359 https://nhsd-confluence.digital.nhs.uk/spaces/RIS/pages/558504102/NHS+Notify+API+Process+Checklist
1360 https://nhsd-confluence.digital.nhs.uk/spaces/RIS/pages/631284915/NHS+Notify+API+Architecture+Review+Checklist
1361 https://github.com/NHSDigital/communications-manager

AMR-128[1362] - GP Connect - Access Record [STU3] ALPHA PUBLISHED	GP Connect Access Record: Structured FHIR	Spec-only API creating an OAS version of the current Access Record: Structured specification	Access Record: Structured[1363]												

1362 https://nhsd-jira.digital.nhs.uk/browse/AMR-128?src=confmacro
1363 https://nhsd-confluence.digital.nhs.uk/display/DCA/API+process+checklist

AMR-347[1364] - Structured Summary Care Record PRIVATE BETA	Structured Summary Care Record API	API which will orchestrate the calls to PDS, SDS and SSP to returned structured data from GP Supplier Systems (GP Connect) and transform the data into a Summary Care Record.	Solution Design[1365] API Process Checklist[1366]												

1364 https://nhsd-jira.digital.nhs.uk/browse/AMR-347?src=confmacro
1365 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/605324448/Private+Beta+-+Solution+Design
1366 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/583608668/API+process+checklist+-+Structured+GP+Summary+API

AMR-340[1367] - Multicast Notification Service PRIVATE BETA	Multicast Notification Service (was Event Management)	A lightweight cloud-based Event notification service to allow producers to publish events which are then routed to consumers that setup subscriptions around subscribable criteria using REST APIs.	Event Pattern - Working with the API (see page 1204)	API process checklist - Event Pattern (see page 1204)	Multicast Notification Service - API architecture review checklist (see page 1204)	https://github.com/NHSDigital/event-pattern-poc	Aiden Vaines	TBC	TBC	✗	✓	✓	?	?	?
AMR-341[1368] - Elective Waiting List LIVE	Waiting List API	An API that provides on-request patient waiting list data based on the NHS England's Elective Care Times data asset.	Wayfinder[1369]	API process checklist - Waiting List[1370]											

1367 https://nhsd-jira.digital.nhs.uk/browse/AMR-340?src=confmacro
1368 https://nhsd-jira.digital.nhs.uk/browse/AMR-341?src=confmacro
1369 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/376508403/Wayfinder
1370 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/614602052/API+process+checklist+-+Waiting+List

AMR-342[1371] - GP connect update record LIVE	Update Record - Structured Update	An API that enables the GP Patient record to be updated with structured data. Initially this will be for Encounter, Observation and Medicines.	API process checklist - Structured Update Record (SSP version)[1372]	API process checklist - Structured Update Record (SSP version)[1373]		https://github.com/NHSDigital/gp-connect-update-record-fhir/	Unknown User (kydo1)	**Base path** Unknown User (kydo1) **what is the base path?**	**?**						
AMR-346[1374] - Digital Genomics Test Service API DISCOVERY	Genomic Test Directory	The ability to query tests from the Test Directory based on a pre defined search criteria.	Genomics homepage[1375]	Digital Genomic Test Service API process checklist[1376]											

1371 https://nhsd-jira.digital.nhs.uk/browse/AMR-342?src=confmacro
1372 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/604916680/API+process+checklist+-+Structured+Update+Record+SSP+version
1373 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/604916680/API+process+checklist+-+Structured+Update+Record+SSP+version
1374 https://nhsd-jira.digital.nhs.uk/browse/AMR-346?src=confmacro
1375 https://nhsd-confluence.digital.nhs.uk/spaces/APCLAG/pages/626231814/Genomics+homepage
1376 https://nhsd-confluence.digital.nhs.uk/spaces/APCLAG/pages/620602734/Digital+Genomic+Test+Service+API+process+checklist

AMR-338[1377] - Genomics order communications API ALPHABUILD	Genomic Order Comms	The ability to order a new test or track an ongoing test. This would be between an Ordering Entity and a GLH with a Central Broker sat between the two.	Genomics homepage[1378]	Genomic Order Management API process checklist[1379]											
AMR-344[1380] - Prescriptions for patients PRIVATE BETA	Prescriptions for Patients API	Provides patients with access to prescriptions sent by the Electronic Prescription Service (EPS).	API process checklist - Prescriptions for Patients API[1381]												

1377 https://nhsd-jira.digital.nhs.uk/browse/AMR-338?src=confmacro
1378 https://nhsd-confluence.digital.nhs.uk/spaces/APCLAG/pages/626231814/Genomics+homepage
1379 https://nhsd-confluence.digital.nhs.uk/spaces/APCLAG/pages/620602752/Genomic+Order+Management+API+process+checklist
1380 https://nhsd-jira.digital.nhs.uk/browse/AMR-344?src=confmacro
1381 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/626234792/API+process+checklist+-+Prescriptions+for+Patients+API

AMR-17[1382] - Patient Flags (Female Genital Mutilation (FGM)) WITHDRAWN	Female Genital Mutilation (FGM) API	The FGM-IS Service supports the wider Female Genital Mutilation Prevention Programme which is in place to protect, prevent and care for children and vulnerable adults who are at risk of FGM.	FGM FHIR R4 API process checklist (see page 1204)												

1382 https://nhsd-jira.digital.nhs.uk/browse/AMR-17?src=confmacro

		There is a requirement to develop an API which will support the addition, update and removal of FGM indicators. This API will be built using FHIR UK Core R4, will be available via APIM and will replicate the functionality that is available in the current FGM API. The reason for the development of a new API is that the current API is built to DSTU2 and that is no longer supported.														

AMR-348[1383] - Validated Relationships BUILD BETA	Validated Relationships	The Validated Relationships API is a component of the Proxy services initiative being developed by the NHS, which includes the Apply for Proxy service.	Validated Relationships API Management (APIM) Integration[1384]	Validated Relationships API Process Checklist[1385]	Validated Relationships API architecture review[1386]	https://github.com/NHSDigital/validated-relationships-service-api	Simon Stead James Taylor								

1383 https://nhsd-jira.digital.nhs.uk/browse/AMR-348?src=confmacro
1384 https://nhsd-confluence.digital.nhs.uk/spaces/NPA/pages/688827123/Validated+Relationships+API+Management+APIM+Integration
1385 https://nhsd-confluence.digital.nhs.uk/spaces/NPA/pages/688827133/Validated+Relationships+API+Process+Checklist
1386 https://nhsd-confluence.digital.nhs.uk/spaces/NPA/pages/693748214/Validated+Relationships+API+architecture+review

		Our service will validate relationships between proxy users and patients to confirm a valid and appropriate proxy relationship based on a given set of rules. The first cohort we plan to enable is birthing Mothers, but there are plans to increase the scope of the user group.																

		In addition to this "verify" capability, a set of APIs will be developed for managing proxy roles for clinical and patient facing services to integrate with.													

AMR-349[1387] - NHS Website Ratings and Reviews API ON HOLD	NHS Website Ratings and Reviews API	An API that surfaces ratings and reviews about NHS services that are left on the NHS website. There are currently around 15 regular users of the API, and over 8 million API calls per month.This API enables users (developers of digital health platforms - NHS, charities, local authorities, commercial sector) to republish reviews and ratings published on the NHS	Syndication[1388]	API process checklist - NHS.uk Ratings & Reviews - Syndication - CDT Confluence (digital.nhs.uk)[1389]												

1388 https://nhsd-confluence.digital.nhs.uk/spaces/SYN/pages/163240500/Syndication
1389 https://nhsd-confluence.digital.nhs.uk/pages/resumedraft.action?draftId=653053828&draftShareId=7af83af1-7e5e-49fb-91ca-cd53da7d2881

		website so that this informati on is available to as many end users as possible.													

AMR-350[1390] - Diagnostic Imaging Report ONHOLD	Diagnostic Imaging Report API	**Part of the National Imaging Registry initiative which will deliver sharing of diagnostic imaging and associated reports across England. Pointers to diagnostic imaging study data (in DICOM format) and associated reports will be managed in the NRL V3.0 APIs. Consumers will retrieve study data directly (due to file size) and reports via APIM**	Diagnostic Imaging Report FHIR API[1391]												

1390 https://nhsd-jira.digital.nhs.uk/browse/AMR-350?src=confmacro
1391 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/673988314/Diagnostic+Imaging+Report+FHIR+API

AMR-351[1392] - Connecting Party Manager BUILD BETA	Connecting Party Manager	**For Internal and external consumers Who need to connect to NHSE products and services, in order to provide essential digital services for healthcare workers and patients Unlike today where creating and managing party connections takes a disproportionate amount of investment Our solution will look to consolidate and standardise existing capabiliti**	Connecting Party Manager (CPM - Pre Dev) (see page 1204)	CPM API process checklist for Connecting Party Manager[1393]	TO DO	TO DO	In the meantime: CPM Architecture Diagrams[1394]									

1393 https://nhsd-confluence.digital.nhs.uk/spaces/CPM/pages/697503598/CPM+API+process+checklist+for+Connecting+Party+Manager

1394 https://nhsd-confluence.digital.nhs.uk/spaces/CPM/pages/662617353/CPM+Architecture+Diagrams

		es for end point registration, certificate management and APIGEE integration We will know we have done this when users who require connections to NHSE products and services can self manage those connections via the Connecting Party Manager													

AMR-353[1395] - Address Quality Service BUILD BETA	Address Quality Service	Provide validation to address updates by adding a UPRN to the update in real time.	Address Quality Service[1396]	API process checklist - Address Quality Service[1397]	O0500 - software architecture - AQS[1398]	https://github.com/NHSDigital/address-quality-service	Afonso Cabecadas Afzal Mufti Teitur Argeseloy								

1395 https://nhsd-jira.digital.nhs.uk/browse/AMR-353?src=confmacro
1396 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/693754997/Address+Quality+Service
1397 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/734573472/API+process+checklist+-+Address+Quality+Service
1398 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/723025929/O0500+-+software+architecture+-+AQS

AMR-355[1399] - Patient Care Aggregator Reporting BETA TECHNICALLY LIVE	Patient Care Aggregator - Reporting	Produces Management Information (MI) for the Wayfinder project by allowing systems integrated to the Patient Care Aggregator to send detailed log events to a centralised data store.	Reporting Service[1400]	API Process Checklist - MIv2 Reporting Service[1401]	TO DO Enter TRG approval ID	https://github.com/NHSDigital/patient-care-aggregator-reporting	Des Kelleher Matt Cavanagh John Bates	**/patient-care-aggregator-reporting**							

1399 https://nhsd-jira.digital.nhs.uk/browse/AMR-355?src=confmacro
1400 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/662622178/Reporting+Service
1401 https://nhsd-confluence.digital.nhs.uk/spaces/WAYF/pages/758091037/API+Process+Checklist+-+MIv2+Reporting+Service

AMR-352[1402] - Clinical Tracking ON HOLD	EPS Clinical Prescription Tracking API	This is a replacement for the antiquated, HSCN-only legacy Prescription Tracking System, for which we will first build a new API which will be consumed by a new UI. The existing (legacy) Clinical Prescription Tracker UI is used by clinical staff working at prescribing and dispensing sites when there is a legitimate need to verify that an EPS prescription exists or to identify a	Clinical Prescription Tracker[1403]	API process checklist for proxy generator - Clinical Prescription Tracking Service FHIR API[1404]	TO DO	https://github.com/NHSDigital/electronic-prescription-service-clinical-prescription-tracker	Robert Gooch Matthew Popat Anthony Brown	TO DO								

1403 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/669207282/Clinical+Prescription+Tracker

1404 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/753483026/API+process+checklist+for+proxy+generator+-+Clinical+Prescription+Tracking+Service+FHIR+API

		prescription status during the prescribing and dispensing process e.g., which dispenser has downloaded it and when it was dispensed. If pharmacies have problems locating an EPS prescription in their dispensing system, checking the EPS Prescription Tracker will allow them to view the prescription status on the NHS Spine.													

AMR-356[1405] - Patient Flags (Reasonable Adjustments) BUILD BETA	Patient Flags API	This is a new API which will is being developed instead of the FGM, CP-IS and Reasonable Adjustments FHIR APIs. The API will support the following: ▪ Retrieving all Patient Flags ▪ Retrieving information related to a specific Patient Flags	Patient Flags - Design[1406]					TO DO							

1405 https://nhsd-jira.digital.nhs.uk/browse/AMR-356?src=confmacro
1406 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/753470267/Patient+Flags+-+Design

		▪ Creating a new Patient Flag and any associated resources ▪ Removing a Patient Flag													

AMR-358[1407] - Prescription Status Update ALPHA BUILD	EPS Prescription Status Update API	This is a new FHIR API which is being set up to enable Dispensing suppliers to provide status updates for the prescriptions that are being managed on their Patient Medication Record systems at Pharmacy level. The API will enable suppliers to push status updates to the endpoint. This data will be stored within the EPS National Prescription Tracking Service and used to enrich the	Mobile First Phase 2 - Patient Prescription Tracking[1408]	API process checklist using proxy generator - Prescription Status Update[1409]	Approved by TRG 06 Mar 2024 Submission[1410]	https://github.com/NHSDigital/eps-prescription-status-update-api	Robert Gooch Matthew Popat Anthony Brown	TO DO							

1410 https://nhs.sharepoint.com/:p:/r/sites/msteams_19da1d/_layouts/15/Doc.aspx?sourcedoc=%7BA7D1C185-094A-48C4-858B-81CD93229DC3%7D&file=TRG%20Mobile%20First%20Prescription%20Tracking%20Iteration%202.pptx&action=edit&mobileredirect=true

		Prescriptions for Patients FHIR API endpoint with status data.													

AMR-360[1411] - Custom Prescription Status Update - Non-FHIR standard linked to AMR-358 ALPHABUILD	EPS **Custom** Prescription Status Update API	This is a new *custom / non-FHIR* API endpoint which is being set up to enable those Dispensing suppliers whom where are fast-tracking for on-boarding to the new National Patient Prescription Status Tracking Service (NPPTS) to provide prescription status updates for the prescriptions that are being managed on their Patient Medication Record systems at Pharmacy level. The	Mobile First Phase 2 - Patient Prescription Tracking[1412]	API process checklist using proxy generator - Custom Prescription Status Update API[1413]	TRG approval provided 06 Mar 2024 Submission[1414]	https://github.com/NHSDigital/eps-prescription-status-update-api	Robert Gooch Matthew Popat Anthony Brown									

1414 https://nhs.sharepoint.com/:p:/r/sites/msteams_19da1d/_layouts/15/Doc.aspx?sourcedoc=%7BA7D1C185-094A-48C4-858B-81CD93229DC3%7D&file=TRG%20Mobile%20First%20Prescription%20Tracking%20Iteration%202.pptx&action=edit&mobileredirect=true

important caveat with this API is that it is being implemented only for those suppliers who are unable to conform to the FHIR Spec, and have an existing method in place in order to meet the ministerial commitment dealine of September 2024.

The API will enable suppliers to push status updates to the endpoint in a non-FHIR format. This API will be deprecated once have confirmation that the supplier

		(or suppliers - we only anticipate 1) are able to provide an updated FHIR-compliant payload to the PSU FHIR API. As with the PSU API, the status update data will be stored within the NPPTS and used to enrich the Prescriptions for Patients FHIR API endpoint with status data.													

AMR-359[1415] - National Document Repository ALPHA BUILD	National Document Repository	The National Document Repository stores Digitised Paper and Human Readable Patient Records.	Patient Record Migration (PRM) - Overview[1416]	API process checklist - PRM Repository[1417]	National Document Repository API architecture review checklist[1418]		Richard Byrne Danielle Anderson								

1415 https://nhsd-jira.digital.nhs.uk/browse/AMR-359?src=confmacro
1416 https://nhsd-confluence.digital.nhs.uk/spaces/GIF/pages/768063156/Patient+Record+Migration+PRM+-+Overview
1417 https://nhsd-confluence.digital.nhs.uk/spaces/GIF/pages/768063158/API+process+checklist+-+PRM+Repository
1418 https://nhsd-confluence.digital.nhs.uk/spaces/GIF/pages/807600448/National+Document+Repository+API+architecture+review+checklist

AMR-361[1419] - GP Connect – New API ONHOLD	GP Connect – New API	This is a proof of concept piece of work designed to model the behaviour of a strategic API to request and post patient data. A mock patient health system will receive the requests through the APIM platform which would check the authentication of the requestor and forward the request to the mock system.													

1419 https://nhsd-jira.digital.nhs.uk/browse/AMR-361?src=confmacro

AMR-362[1420] - Patient Audit Record Service (PARS) is a reporting service that describes how patient data has been accessed. ON HOLD	PARS - Patient Audit Record Service	The PARS API is required to fulfil 2 main purposes of the PARSservice: • Provide a cloud agnostic route for connecting systems to send data about patient access events to the PARS service	PARS API[1421]	PARS API process checklist using proxy generator[1422]	PARS API architecture review checklist[1423]	https://github.com/NHSDigital/patient-audit-record-service[1424]	Hira Farooq Matthew Potter	TO DO							

1420 https://nhsd-jira.digital.nhs.uk/browse/AMR-362?src=confmacro
1421 https://nhsd-confluence.digital.nhs.uk/spaces/PARS/pages/864404433/PARS+API
1422 https://nhsd-confluence.digital.nhs.uk/spaces/PARS/pages/864404449/PARS+API+process+checklist+using+proxy+generator
1423 https://nhsd-confluence.digital.nhs.uk/spaces/PARS/pages/864404439/PARS+API+architecture+review+checklist
1424 https://github.com/NHSDigital/patient-audit-record-service/pull/64

		• Provide a route for users and services to send patient access and user access queries to PARS and receive the requested reports.													

AMR-363[1425] - Register with a GP Surgery API BUILD BETA	Register with a GP surgery API	Use this API to access registration applications using the Register with a GP Surgery Service[1426]. This API provides: • operational parameters • a set of codeable observations		GP Reg API - Delivery Checklist[1427]												

1425 https://nhsd-jira.digital.nhs.uk/browse/AMR-363?src=confmacro

1426 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdigital.nhs.uk%2Fservices%2Fregister-with-a-gp-surgery-service&data=05%7C02%7Cjonathan.eagle1%40nhs.net%7Cd7b4addab57c406cc1e408dcb2d9cc74%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638581895699634790%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C0%7C%7C%7C&sdata=dXIWIZMPj93j5UjKrzqz5nv0RYGNzuSb6Vk9W12rcal%3D&reserved=0

1427 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/864401788/GP+Reg+API+-+Delivery+Checklist

		• questionnaire response relating to a patients registration request. You can use this API to: • retrieve a registration application You cannot use this API to: • submit a new Registration Application														

AMR-371[1428] - Genomic Data Access and Management API ALPHA BUILD	Genomic Data Access and Management API	Allows users to download read and variation data for subsections of the genome Provides a standardised set of data retrieval methods FHIR Genomics is a subset of the emerging Health Level 7 FHIR standard and targets data from increasingly available technologies such as next-generation sequencing													

1428 https://nhsd-jira.digital.nhs.uk/browse/AMR-371?src=confmacro

AMR-372[1429] - Patient Data Manager (Initially) to allow Patient Care Aggregator to enrich appointments returned from PEPS BUILD BETA	Patient Data Manager (PDM)	Patient Data Manager (PDM) will enable seamless data exchange by accepting and providing patient data in FHIR, storing data in AWS HealthLake, and ensuring standardised, interoperable, and patient-accessible data is available for direct care. PDM also supports ingestion via MESH, enabling efficient bulk transfer and transform of data	Cell PDM (Patient Data Manager) (see page 1204)	API process checklist - Patient Data Manager (see page 1204)											

1429 https://nhsd-jira.digital.nhs.uk/browse/AMR-372?src=confmacro

AMR-376[1430] - Healthcare Worker API - management and access to Healthcare Worker data BUILD BETA	Healthcare worker API	Enables other api's to search and retrieve healthcare worker data from one single API and reducing friction for developers to integrate and access healthcare workers data. Using the UUID we aim to surface data of all health care workers with a care identity profile. Users and their suppliers will have the capability to retrieve healthcare worker's data for clinical referrals	API process checklist - Healthcare Worker API (see page 1204)	Healthcare worker - API architecture review checklist[1431] Healthcare worker - API design review[1432]												

1431 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/902232957/Healthcare+worker+-+API+architecture+review+checklist

1432 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/962954834/Healthcare+worker+-+API+design+review

		or prescribing systems effortlessly.													
AMR-377[1433] - DWA Middle Service API - enable interaction between Service Now and PDS Spine. DEPRECATED	DWA Middle Service API	An API to enable interaction between Service Now and PDS Spine.		API process checklist using proxy generator - DWA Middle Service API[1434]	DWA Middle Service API architecture review checklist[1435]										

1433 https://nhsd-jira.digital.nhs.uk/browse/AMR-377?src=confmacro
1434 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/885634709/API+process+checklist+using+proxy+generator+-+DWA+Middle+Service+API
1435 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/885639892/DWA+Middle+Service+API+architecture+review+checklist

AMR-378[1436] - Central Metadata Hub API - To enable automated updates to and consumption from the Single Point of Reference BUILD BETA	Central Metadata Hub API	To enable automated updates to and consumption from the Single Point of Reference 'hub' repository of management and cataloguing information about health and care data (i.e. metadata)	Data Products - current - Data Products - CDT Confluence (digital.nhs.uk)[1437]	Central Metadata Hub API process checklist - Data Products - CDT Confluence (digital.nhs.uk)[1438]												

confluence.digital.nhs.uk%2Fdisplay%2FTPS%2FData%2BProducts%2B-%2Bcurrent&data=05%7C02%7Cjonathan.eagle1%40nhs.net%7Cec31431c192f4dce3d8508dce14ec19b%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638632975596584850%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C0%7C%7C%7C&sdata=W1tg0ot1ZoV3I3lhgsSLNJkIJbuIqdUet7cD1O4FJvQ%3D&reserved=0

1438 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsd-confluence.digital.nhs.uk%2Fpages%2Fresumedraft.action%3FdraftId%3D910427595%26draftShareId%3D9278f7ca-4aaa-4315-9afc-1100f5a997f5%26&data=05%7C02%7Cjonathan.eagle1%40nhs.net%7Cec31431c192f4dce3d8508dce14ec19b%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638632975596608912%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C0%7C%7C%7C&sdata=J%2FGWCrmsHsPsb1Xfw5QRcoDBKyvibJLpkbhn6nH7eDQ%3D&reserved=0

AMR-357[1439] - Digital Staff Passport WITHDRAWN	Digital Staff Passport API standard	This is an API standard, not an API - it's a standard for external system providers to build against and for the Digital Staff Passport service to consume.	DSP Integration[1440]	API process checklist - Digital Staff Passport API standard[1441]											

1439 https://nhsd-jira.digital.nhs.uk/browse/AMR-357?src=confmacro
1440 https://nhsd-confluence.digital.nhs.uk/spaces/DSP1/pages/889951214/DSP+Integration
1441 https://nhsd-confluence.digital.nhs.uk/spaces/DSP1/pages/889951217/API+process+checklist+-+Digital+Staff+Passport+API+standard

AMR-379[1442] - Digital Staff Passport (DSP) - Allows external workforce systems to send DSP pro-active notifications when new or amended employee info is available in their system. ON HOLD	Digital Staff Passport API	Works in conjunction with the Digital Staff Passport API standard, allowing external systems to pro-actively notify Digital Staff Passport of new or amended	DSP Integration[1443]	API process checklist - Digital Staff Passport API[1444]	Digital Staff Passport API architecture review checklist[1445]										

AMR-381[1446] - Related Person Network ALPHA BUILD	Related Person Network	Get and Update relationships between patients and their related persons, providing additional parent-child relationships based on Civil Birth Registration data that are not served on the existing HL7v3 and FHIR APIs.	Build Relationship network between people in PDS[1447]	Relationship Network - API process checklist using proxy generator[1448]	Relationship Network - API architecture review checklist[1449]	https://github.com/nhsdigital/related-person-network/	Chris Kemp Anahita Saghafi Samuel Hollow	TO DO	/FHIR/R4	N					

1446 https://nhsd-jira.digital.nhs.uk/browse/AMR-381?src=confmacro
1447 https://nhsd-confluence.digital.nhs.uk/display/DEMGRPH/Build+Relationship+network+between+people+in+PDS
1448 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fnhsd-confluence.digital.nhs.uk%2Fdisplay%2FDEMGRPH%2FRelationship%2BNetwork%2B-%2BAPI%2Bprocess%2Bchecklist%2Busing%2Bproxy%2Bgenerator&data=05%7C02%7Csamuel.hollow1%40nhs.net%7C687d4a27f3174441fe5008dd12d9729d%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638687448529002638%7CUnknown%7CTWFpbGZsb3d8eyJFbXB0eU1hcGkiOnRydWUsIlYiOiIwLjAuMDAwMCIsIlAiOiJXaW4zMiIsIkFOIjoiTWFpbCIsIldUIjoyfQ%3D%3D%7C0%7C%7C%7C&sdata=DkJS%2B%2BnMCyO1%2FWPeGmcSTxOLh4YJGMHzTFnyA5DoeP8%3D&reserved=0
1449 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/981777324/Relationship+Network+-+API+architecture+review+checklist

AMR-383[1450] - Eligibility Signposting API (Vaccinations) BUILD BETA	Eligibility Signposting API (Vaccinations)	API to return configurable (rule-based) eligibility status and next actions information for a given person (and optionally a given vaccination/ condition type)	Eligibility Signposting API High Level Architecture[1451]	API process checklist - Eligibility API[1452]			John Hadfield								

1450 https://nhsd-jira.digital.nhs.uk/browse/AMR-383?src=confmacro
1451 https://nhsd-confluence.digital.nhs.uk/display/Vacc/Eligibility+Signposting+API+High+Level+Architecture
1452 https://nhsd-confluence.digital.nhs.uk/display/Vacc/API+process+checklist+-+Eligibility+API

AMR-385[1453] - UEC EPS-DoS API ALPHA BUILD	UEC EPS DoS API	UEC Prescribers use this service to look for pharmacies that have an 'Electronic Prescription Service' available within a given location and timeframe.	EPS-DoS API[1454]	API process checklist - UEC EPS-DoS API[1455]	UEC EPS-DoS API architecture review checklist[1456]	https://gitlab.mgmt.texasplatform.uk/eps/eps-dispenser-api	Jonathan Pearce	**/epsdispenser**	**/byLocationAndTime/byNameAndPostcode**	Y (but not via APIM)					

1453 https://nhsd-jira.digital.nhs.uk/browse/AMR-385?src=confmacro
1454 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/885637360/EPS-DoS+API
1455 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/1012551871/API+process+checklist+-+UEC+EPS-DoS+API
1456 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/1012552227/UEC+EPS-DoS+API+architecture+review+checklist

AMR-380[1457] - Find the Right Service Ingress API DISCOVERY	Find the Right Service Ingress API	API to provided a single interoperable entry point for automatically ingesting UEC service data	Future Sourcing API - APIM Checklist[1458]	Future Sourcing API - APIM Checklist[1459]	Directory of Services Ingestion API - Architecture Review[1460]	TO DO	Aileen Tracey Matthew Begley Sindhu Priya Nataraj an	TO DO	None						
AMR-386[1461] - NHS Notify Supplier API DISCOVERY	NHS Notify Supplier API	API to allow print suppliers to interact with NHS Notify for letters fulfillment	NHS Notify Supplier API[1462]												

1457 https://nhsd-jira.digital.nhs.uk/browse/AMR-380?src=confmacro
1458 https://nhsd-confluence.digital.nhs.uk/spaces/FRS/pages/937829557/Future+Sourcing+API+-+APIM+Checklist
1459 https://nhsd-confluence.digital.nhs.uk/spaces/FRS/pages/937829557/Future+Sourcing+API+-+APIM+Checklist
1460 https://nhsd-confluence.digital.nhs.uk/spaces/FRS/pages/966207158/Directory+of+Services+Ingestion+API+-+Architecture+Review
1461 https://nhsd-jira.digital.nhs.uk/browse/AMR-386?src=confmacro
1462 https://nhsd-confluence.digital.nhs.uk/display/RIS/NHS+Notify+%7C+Supplier+API+process+checklist+using+proxy+generator

AMR-387[1463] - National Elective Waiting List Information and Contact Service API (NEWLICS) ALPHA BUILD	National Elective Waiting List Information and Contact Service API (NEWLICS)	The purpose of the API is to enable the digital PIFU initiative													

1463 https://nhsd-jira.digital.nhs.uk/browse/AMR-387?src=confmacro

AMR-389[1464] - Find the Right Service Search – Integrated Search API DISCOVERY	Find the Right Service Search API	API to allow connecting parties to search the DoS for services	Find the Right Service Search API[1465]	API process checklist - GP Endpoint[1466]	API architecture review checklist - GP Endpoint[1467]										

1464 https://nhsd-jira.digital.nhs.uk/browse/AMR-389?src=confmacro
1465 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/1051860909/Find+the+Right+Service+Search+API
1466 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/1051860919/API+process+checklist+-+GP+Endpoint
1467 https://nhsd-confluence.digital.nhs.uk/spaces/DOSIS/pages/1051861116/API+architecture+review+checklist+-+GP+Endpoint

	GPConnect Access Record Structured API - accessible via APIM	Migration of current GP Connect Access Record Structured API (accessible through SSP) onto APIM. It also includes the orchestration of patient and endpoint lookup.	GPC API Gateway[1468]	API process checklist - GPC API Gateway - Access Record Structured[1469]	GPC API Gateway API architecture review checklist - WIP[1470]	TO DO	Steven Hunt	TO DO							

8.2 API sunsetting register

This table is a list of APIs that have been considered for sunsetting, with links to the details.

1468 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/1071916640/GPC+API+Gateway
1469 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/1073052722/API+process+checklist+-+GPC+API+Gateway+-+Access+Record+Structured
1470 https://nhsd-confluence.digital.nhs.uk/spaces/DCA/pages/1088528922/GPC+API+Gateway+API+architecture+review+checklist+-+WIP

	API	Link to sunsetting process checklist	Notes
AMR-45[1471] - COVID-19 Test Results DEPRECATED	COVID-19 Test Results API	Sunsetting process checklist - COVID-19 Test Results - FHIR API[1472]	
	Legacy HL7 V3 Intermediary APIs	Sunsetting process checklist - Legacy HL7 V3 intermediary APIs[1473]	
	GPES HL7 V3 API	Sunsetting process checklist - GPES HL7 V3 API[1474]	
AMR-76[1475] - Query Accredited System Information (QASI) DEPRECATED	QASI API	Sunsetting process checklist - QASI[1476]	
AMR-97[1477] - Personal Demographic Service (PDS) [SMSP API] (21/22) WITHDRAWN	PDS SMSP	Sunsetting process checklist - PDS SMSP[1478]	
AMR-120[1479] - Gazetteer SOAP DEPRECATED	Gazetteer SOAP	Sunsetting process checklist - Gazetteer SOAP[1480]	

1471 https://nhsd-jira.digital.nhs.uk/browse/AMR-45?src=confmacro
1472 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/740960625/Sunsetting+process+checklist+-+COVID-19+Test+Results+-+FHIR+API
1473 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216180509/Sunsetting+process+checklist+-+Legacy+HL7+V3+intermediary+APIs
1474 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/216181485/Sunsetting+process+checklist+-+GPES+HL7+V3+API
1475 https://nhsd-jira.digital.nhs.uk/browse/AMR-76?src=confmacro
1476 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/227250791/Sunsetting+process+checklist+-+QASI
1477 https://nhsd-jira.digital.nhs.uk/browse/AMR-97?src=confmacro
1478 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/307891415/Sunsetting+process+checklist+-+PDS+SMSP
1479 https://nhsd-jira.digital.nhs.uk/browse/AMR-120?src=confmacro
1480 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/328977037/Sunsetting+process+checklist+-+Gazetteer+SOAP

	API	Link to sunsetting process checklist	Notes
AMR-244[1481] - Personal Demographic Service (PDS) [SMSP standards] TO SUNSET	PDS SMSP API Standards	Sunsetting process checklist - PDS SMSP API standards[1482]	
AMP-9	ETP Web services REST API	Sunsetting process checklist - ETP Web services APIs - REST[1483]	
AMP-17	ETP Web Services SOAP API	Sunsetting process checklist - ETP Web Services APIs - SOAP[1484]	
AMP-386	Pathology Messaging FHIR API	API deprecation and retirement checklist - Pathology Messaging FHIR API [1485]	**This FHIR (STU3) API is planned for deprecation as it would be superseded by FHIR (R4) API.**
AMP-92	Data Registers Service - REST API	Sunsetting process checklist - Data Registers Service API [1486]	

1481 https://nhsd-jira.digital.nhs.uk/browse/AMR-244?src=confmacro
1482 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/391971683/Sunsetting+process+checklist+-+PDS+SMSP+API+standards
1483 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/427899094/Sunsetting+process+checklist+-+ETP+Web+services+APIs+-+REST
1484 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/457606648/Sunsetting+process+checklist+-+ETP+Web+Services+APIs+-+SOAP
1485 https://nhsd-confluence.digital.nhs.uk/display/APM/API+deprecation+and+retirement+checklist+-+Pathology+Messaging+FHIR+API?src=contextnavpagetreemode
1486 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/461250368/Sunsetting+process+checklist+-+Data+Registers+Service+API

	API	Link to sunsetting process checklist	Notes
AMP-28	Assessment Discharge and Withdrawal - FHIR	API deprecation and retirement checklist - Assessment Discharge and Withdrawal - FHIR API [1487]	
AMP-47	Spine Security Broker API	Sunsetting process checklist - Spine Security Broker API [1488]	**Currently blocked and unable to progress as there is no software deprecation policy and we are reliant on around 300k users having the credential management client component installed on their systems - no incentive to do so.**
AMP-574	Pathology Messaging - HL7 V3	Sunsetting process checklist - Pathology Messaging - HL7 V3 [1489]	
AMP-778	Out of hours message integration	Sunsetting process checklist - Out of hours API integration [1490]	**This was a missing API which we deliberated over adding to the catalogue - MESH workflow ID traffic dictated that we should, and when we presented at Show & Tell it was recommended by key stakeholders that if we present it, we need to mark as deprecated as there is an alternative (https://digital.nhs.uk/developer/api-catalogue/gp-connect-send-document-fhir) and we should not be promoting/ recommending OOH.**

1487 https://nhsd-confluence.digital.nhs.uk/display/APM/API+deprecation+and+retirement+checklist+-+Assessment+Discharge+and+Withdrawal+-+FHIR+API?src=contextnavpagetreemode

1488 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/410126288/Sunsetting+process+checklist+-+Spine+Security+Broker+API

1489 https://nhsd-confluence.digital.nhs.uk/display/APM/Sunsetting+process+checklist+-+Pathology+Messaging+-+HL7+V3?src=contextnavpagetreemode

1490 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/497795279/Sunsetting+process+checklist+-+Out+of+hours+API+integration

	API	Link to sunsetting process checklist	Notes
AMP-815	Access Control Service - HL7 v3	Sunsetting process checklist - Access control service HL7 v3[1491]	**The Access Control Service HL7 v3 API has been replaced by Summary Care Record - FHIR, however, SCR is only available in primary care, so may impact on the decision to begin the sunsetting journey and to deprecate the API.**
AMP-149	Digital Maternity API standards	Sunsetting process checklist - Digital Maternity API standards[1492]	**Deprecated as per advice from Kevin Sprague and Katie Wheatley - to be consistent with status of standard within the Standards Directory.**
AMP-1107	COVID-19 Medical Exemptions - FHIR	COVID-19 Medical Exemptions - FHIR API - NHS Digital[1493]	**To be retired before the end of March 2023 - expediting the deprecation and retirement process.**
AMP-766	EPS DOS	Sunsetting process checklist - EPS DOS[1494]	
AMP-1035	FHIR Converter	Sunsetting process checklist - FHIR Converter API[1495]	**Moving straight to retired as per conversation with Tony, Will, Shan and Jonathan Telfer**

1491 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/512755180/Sunsetting+process+checklist+-+Access+control+service+HL7+v3

1492 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/524889238/Sunsetting+process+checklist+-+Digital+Maternity+API+standards

1493 https://digital.nhs.uk/developer/api-catalogue/covid-19-medical-exemptions-fhir

1494 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/503027595/Sunsetting+process+checklist+-+EPS+DOS

1495 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/554510061/Sunsetting+process+checklist+-+FHIR+Converter+API

	API	Link to sunsetting process checklist	Notes
AMP-1684 AMR-104[1496] - Vaccination Adverse Reactions COVID-19 API RETIRE	Vaccinations Adverse Reactions	Sunsetting process checklist - Vaccinations adverse reactions[1497]	**Conversation between Mick and Shail - move straight to retired (by word only - do not de-auth) - updated checklist with Shail and Daniela - Daniela will contact Shail in early 2024 to discuss whether RFC is required and whether we need to de-auth. 20/3/2024 have contacted Shail no further action required as API was never on APIM platform**
AMR-89[1498] - Digital Medicines FHIR API TO SUNSET	Digital Medicine - FHIR	Sunsetting process checklist - Digital Medicine - FHIR[1499]	**Request from Fleur Bradley/ Jacqueline Harris to deprecate 2 MESH workflow IDs and update the spec to announce the deprecation.**
	FGM DSTU2 - FHIR	FGM DSTU2 FHIR API - Sunsetting checklist (API standards)[1500]	**To be replaced by Patient Flags FHIR R4 API. API is only used by two consumers and is read-only. API will continue to be supported but should not be used by any new consumers.**
	FGM SMSP	FGM SMSP - Sunsetting checklist (API standards)[1501]	**To be replaced by Patient Flags FHIR R4 API. Service is only used by two suppliers and transaction volumes are very low.**

1496 https://nhsd-jira.digital.nhs.uk/browse/AMR-104?src=confmacro
1497 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/626252769/Sunsetting+process+checklist+-+Vaccinations+adverse+reactions
1498 https://nhsd-jira.digital.nhs.uk/browse/AMR-89?src=confmacro
1499 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/631294960/Sunsetting+process+checklist+-+Digital+Medicine+-+FHIR
1500 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/783879192/FGM+DSTU2+FHIR+API+-+Sunsetting+checklist+API+standards
1501 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/783879178/FGM+SMSP+-+Sunsetting+checklist+API+standards

	API	Link to sunsetting process checklist	Notes
AMR-5[1502] - Reasonable Adjustment API TO SUNSET	Reasonable Adjustment FHIR DSTU3 API	RA DSTU3 API Sunsetting checklist template (API standards)[1503]	**To be immediately retired. API never went into Production and it's functionality has been incorporated into the Patient Flags FHIR R4 API.**
AMR-377[1504] - DWA Middle Service API - enable interaction between Service Now and PDS Spine. DEPRECATED	DWA Middle Service API - HL7 V3	APIM side - Sunsetting process checklist - DWA Adaptor API[1505] Demographics - Sunsetting process checklist - DWA Adaptor API[1506]	**Following Pilot Phase 1, the DWA is comming to an immediate close - API to be retired for now.**

8.3 Producer team member audit

As part of APM-2367[1507] the need for a producer team member audit was identified.

Producer team member access should be audited quarterly, especially GitHub access.

8.3.1 Audit tasks

1. Using the AMR board, confirm the delivery lead for each API in the table below:
    a. If the delivery lead has been updated, update the table.
2. Using the AMR board, identify if any of the API producer teams from the table below have been disbanded:
    a. If a team is no longer active, identify the members as producer team leavers and add their names to a list in the audit ticket.
    b. Remove the API from the table below.
3. Using the AMR board, identify any missing APIs from the below table:
    a. Identify any APIs in Alpha Build or beyond that are being worked on by API Producers that do not already exist in the table.

1502 https://nhsd-jira.digital.nhs.uk/browse/AMR-5?src=confmacro
1503 https://nhsd-confluence.digital.nhs.uk/spaces/CLP/pages/783879125/RA+DSTU3+API+Sunsetting+checklist+template+API+standards
1504 https://nhsd-jira.digital.nhs.uk/browse/AMR-377?src=confmacro
1505 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/1009912509/Sunsetting+process+checklist+-+DWA+Adaptor+API
1506 https://nhsd-confluence.digital.nhs.uk/spaces/DEMGRPH/pages/1002786462/Sunsetting+process+checklist+-+DWA+Adaptor+API
1507 https://nhsd-jira.digital.nhs.uk/browse/APM-2367

   b. Add them to the table with the appropriate information.

4. For each API contact the delivery lead (where a delivery lead is unavailable contact the technical author) to confirm current team members, and names of any leavers since the audit.
   a. In the table below make sure the primary delivery lead is correct and that there is at least one other key team member in the table (to make it easier to contact the team)
   b. For the rest of the team, the APIM User Security Log should be checked and updated:
      i. If the user does not exist add them to the log - and enter any known permissions
      ii. If there is a user who is not in the current team - mark them as a leaver and potentially trigger a leavers process KOP-036[1508]
      iii. Update any details for individual users
   c. For each person in each team confirm the following information:
      i. Name
      ii. Shortcode
      iii. Email
      iv. GitHub username
      v. Job description
5. Fill in the Date column with today's date.
6. Create a ticket for the next audit using the template below.
7. Add the next audit to the APIM calendar
   a. Navigate to your NHS mail calendar[1509]
   b. Add a new event titled 'Producer Team Member Audit'
   c. When creating the event, add the apimanagement4@gmail.com[1510] to the attendees list
   d. Set an appropriate date for the next audit
   e. Save the event
8. Move ticket to done

8.3.1.1 Link to AMR board

AMR board[1511]

8.3.2 Audit information table

Here is a list of APIs with external teams, their primary contact and a NHS point of contact.

1508 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=233085284
1509 https://outlook.office.com/calendar/view/month
1510 mailto:apimanagement4@gmail.com
1511 https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?rapidView=2826

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
COVID-19 Test Results ✅	Abhishek Ranjan (abhishek.ranjan2@nhs.net[1512]) / Ashish Dharpure (ashish.dharpure@kainos.com[1513]) (Team Lead)	Unknown User (misc4)	Tom Smith (t.smith@kainos.com[1514]) - Product Owner (Kainos)	-	-	-	-	15 Dec 2022
			Michal Fiedoruk (Tech lead)	✅	-	-	-	
			Ryszard Madejski	✅	-	-	ryszardm-kainos	
			Alicja Natarajan	✅	-	-	Alicja-Kainos	
			Malgorzata Rogowska	-	-	-	mrogowska	
			Ali Jaffer	✅	-	-	-	
EeRS (Eyecare e-Referral Service) (Spec Only) ✅	Unknown User (kagi1)		Unknown User (kagi1) - Delivery Lead	✅	-	-	kategill	21 Oct 2022

1512 mailto:abhishek.ranjan2@nhs.net
1513 mailto:ashish.dharpure@kainos.com
1514 mailto:t.smith@kainos.com

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
e-Referral Service (eRS) (Medicus) ✅	Chris Rodgers	Alex Langley (Technical Architect) / Adam Oldfield	Leon Finnerty - Product Owner	See below				30 Nov 2022
e-Referral Service Patient Care ✅	Chris Rodgers	Adam Oldfield (Technical Architect) / Alex Langley	Leon Finnerty - Product Owner	See below				30 Nov 2022

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
e-Reffera l Service + e-Reffera l Service Patient Care (treate d as one team) ✅			Mike Weeks	-	✅	-	-	30 Nov 2022
			David Wass	✅	✅	-	nhsd-david-wass	
			Steve Williams	✅	✅	-	-	
			David Mays	✅	✅	-	nhsd-dm	
			Adam Oldfield	✅	✅	-	adamoldfield-nhs	
			Jack Wainwright	✅	✅	-	nhsd-jack-wainwright	
			Nigel Mellor	✅	✅	-	-	
			Michael Roberts	✅	✅	-	-	
			Stuart Gray	✅	✅	-	-	
			Raja Govindharaj	✅	✅	-	RajaGovindharaj16	
			Francisco Videira	✅	✅	-	francisco-videira-nhs	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Angel Pastor	✅	✅	-	nhsd-angel-pastor	
			Jamie Green	-	✅	-	jamiegreen4	
			Andrew Tuley	-	-	-	-	
			Unknown User (nasa5)	✅	✅	-	Nayira-Sanchez	
			Nigel Spowage	✅	✅	-	-	
			Sarah Rigg	-	✅	-	-	
			Steve Buxton	-	-	-	-	
			Martin Hagger	-	✅	-	-	
			Andy Hepburn	-	✅	-	-	
			Jordan Woodhead	-	✅	-	-	
			Alex Langley	✅	✅	-	langleyalex-arch	
			Chris Rodgers	✅	-	-	-	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
GP2GP Management Information REST API ✅	Sarah Walden - sarah.walden1@nhs.net[1515]	Unknown User (rojo8) TBC	Unknown User (jehu2) - Product Owner	✅	-	-	-	15 Nov 2022
			Unknown User (rojo8) - Technical Lead	✅	✅	-	rob-joscelyne-nhs	
			Unknown User (jane3)	✅	✅	-	jnewman-nhs	
			Caius Eugene (CAEU)	✅ ❓	✅ ❓	-	caiuseugene1	

1515 mailto:sarah.walden1@nhs.net

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
NHS App (notifications) ✅	Unknown User (pasa7)	Unknown User (iane1) (Tech Lead)	Unknown User (chwo7) - Product Owner	✅	-	-	-	03 Nov 2022
	Unknown User (emgo1) taking over from Pavandeep in 2023		Unknown User (iane1) - Tech Lead	✅	✅	-	ianfnelson	
	Unknown User (mimi5) is DL		Lewis Bird	✅	✅	-	lewis-bird	
			Unknown User (mabo15)	✅	✅	-	markbonnici01	
			Unknown User (roco15)	✅	✅	-	rcopettinhs	
			Unknown User (clco9)	✅	-	-	-	
			Unknown User (made10)	✅	-	-	-	
			Richard Etherington	✅	✅	-	richardetherington	
			Felix Igbinehi	✅	-	-	-	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (dajo14)	✅	-	-	-	
			Unknown User (yuma1)	✅	✅	-	✅⚠️ - missing username	
			Unknown User (eomc1)	✅	✅	-	eoghanmcl	
			Unknown User (mimi5)	✅	-	-	-	
			Unknown User (bemo2)	✅	-	-	-	
			Thomas Pedley	✅	✅	-	thomaspedleynhs	
			Unknown User (muqa1)	✅	✅	-	qasimshk	
			Unknown User (sara18)	✅	✅	-	sandyrastogi	
			Unknown User (pasa7)	✅	-	-	-	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (gash1)	✅	-	-	-	
			Unknown User (tesh2)	✅	✅	-	tsheppard01	
			Christopher Speers	✅	-	-	-	
			Unknown User (gawi2)	✅	-	-	-	
			Unknown User (cowi1)	✅	✅	-	ieatsandvich	
			Unknown User (jamc10)	✅	✅	-	jack-kainos	
NHS Booking and Referral System ✅	Unknown User (wija1)	Unknown User (misc4)	David Ruddy - Product Owner	✅	-	-	-	03 Nov 2022
			Richard Ward - Tech Lead	✅	✅	-	RichardTheWard	
			Carl De'ath	✅	✅	-	cda69	
			Leigh Goslin	✅	✅	-	Goslinl	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (wija1)	❓⚠️	-	-	-	
National Record Locator (NRL) Futures ✅	Unknown User (jowa17)	Chris Kemp	Unknown User (brev2) - Product Owner	✅	✅	-	✅⚠️ - missing username	31 Oct 2022
			Chris Kemp - Technical Lead	✅	✅	-	nomad3k	
			Unknown User (jali1)	✅	✅	-	jameslinnell	
			Tim Wang	✅	✅	-	xLasercut	
			Rowan Gill	✅	✅	-	Rohoolio	
			Unknown User (brba1)	✅	✅	-	brianbayes	
			Unknown User (shah5)	✅	✅	-	shahmed1	
			Deepa Attiguppe	✅	✅	-	deepaattiguppe	
			Joel Klinger	✅	✅	-	Jaklinger	
			Laura Haley	✅	✅	-	laurahaley1	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (jowa17)	✅	-	-	-	
Patient care aggregator API (Wayfinder) ✅	John Bates	Des Kelleher (Tech Lead)	Unknown User (sowa6) - Product Owner	✅	-	-	-	01 Nov 2022
			John Bates	-	-	-	-	
			Des Kelleher - Technical Lead	✅	✅	-	deskelleher	
Signing Service (remote) ✅	Sean Wallace (BJSS)	Matthew Popat (Tech Lead)	Allison Loose - Product Owner	✅	-	-	-	02 Nov 2022
			Sean Wallace	✅	-	-	-	
			Unknown User (opos1)	✅	-	-	-	
			Unknown User (beri2)	✅	✅	-	bernard.richardson1	
			Matthew Popat	✅	✅	-	MatthewPopat-NHS	
			Neil Sproston	✅	✅	-	neil-sproston	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (edbr2)	✅	✅	-	ed-breslin1	
			Jason Edwards	✅	✅	-	jasonandtheagronauts	
			Ashley Smith (joining soon)					
Spine Directory Service (SDS) [GP IT Futures] ✅	Richard Cooke	Jason Edwards (BJSS)	Alison Loose - Project Manager	-	-	-	-	21 Nov 2022
			Jason Edwards - Technical Lead + Service Owner	✅	✅	-	jasonandtheagronauts	
			Unknown User (sogu1)	✅	✅	-	soringumeni1	
			Robert Kerry	✅	✅	-	robkerrybjss	
Summary Care Record (SCR) [GP IT Futures] ✅	Jack Carter-Pearce	Scott Alexander	Scott Alexander - Tech Lead	✅	✅	-	SRAlexander	15 Nov 2022
			Unknown User (abma6)	✅	✅	-	abidmajid1	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Benjamin Hession	✅		-	benhession	
			Danny Khan (DAKH3)	✅		-	dfk97	
			Ellen Wootten	✅	✅	-	ellenwootten	
			Harvey Sembhy (HASE2)	✅		-	HarveyAnswer	
			Martin Wheeler (MWSH12)	✅		-	MartinWheelerMT	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
Ambulance Data Submission ✅	paul.rees-walker@hscic.gov.uk[1516] (Active on teams)	Unknown User (beda7)	Lester Drake (Dev)	-	-	-	-	14 Dec 2022

1516 mailto:paul.rees-walker@hscic.gov.uk

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
Covid-19 medical exemptions ✅	Unknown User (anre8) (service delivery manager)	Haitham Abou-Houly (Tech Lead)	Unknown User (iomc1) (service manager)	-	-	-	-	04 Jan 2023
			Unknown User (anre8)	-	-	-	-	
			Haitham Abou-Houly	✅	✅	-	abouhouly	
			Unknown User (jali1)	✅	✅	-	jameslinnell	
			Unknown User (dasm13)	✅	✅	-	hmbpos10	
			Unknown User (jami14)	✅	✅	-	jamie-mills-bjss	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
Electronic Prescription Service	Ezequiel Gomez / John Kitson	Charlotte Bunker	Jennifer Redman (PO)	✅		-		Slacked and acknowledged (slack group with Ezquiel, Jennifer and Stefano) 03 Jan 2023 Full info provided 19 Jan 2023
			John Kitson	✅				
			Ezequiel Gomez	✅				
			Matthew Popat (Tech Lead)	✅				
			Michael Walters (Tech Lead)	✅			yorkshiremoose	
			Anthony Brown	✅	noprod		anthony-nhs	
			Unknown User (stfr5)	✅	-	✅	StefanoFrazzetto	
			Unknown User (anma31)	✅			AndyMather24	
			Chris Matus	✅			c-matus	
			Matt Arnold	✅			MathewArnoldIW	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (muab2)	✅	nonprod		mukhtar-abdul-ghani	
			Taiye Ogunnigbo	✅			taiyeoguns	
			Tom Smith	✅			TomSmithCoding	
			Philip Gee	✅	nonprod	requested		
			Kris Szlapa	✅	✅	requested	kris-szlapa	
			Jack Spagnoli	✅			JackSpagnoliNHS	
			Unknown User (seak2)	✅			ayoseg	
			Omar Khan	✅			Omarisgreat	
			Unknown User (rogo7)	✅				
			Tim Coates	✅				

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Ameya Krishnamoorthy	✅				
			Jessica Mancoo	✅				
			Unknown User (gema7)	✅				
			Tamara Farrar	✅				
			Charlotte Bunker	✅			c-matus	
			Brad Gallagher					
			Elliott Seaton					
			Unknown User (kamo11)					
			Keith Davison	✅				

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Mince Chang	✅				
FHIR converter ✅	Unknown User (beda7) / Unknown User (joro21)	Unknown User (beda7)	Bebop					14 Dec 2022
Health Research Data Catalogue ✅	Unknown User (rabr7)	Unknown User (srsh1)	Laura Sato (PO)	-	-	-	-	04 Jan 2023
			Ketankumar Italiya (tech lead)	✅	✅	-	-	
			Unknown User (rabr7)			-		
			Unknown User (srsh1)	✅	✅	-	srsh1	
Immunisation History ✅	Unknown User (anre8) (service delivery manager)	Haitham Abou-Houly (Tech Lead)	Unknown User (iomc1) (service manager)	-	-	-	-	04 Jan 2023
			Unknown User (anre8)	-	-	-	-	
			Haitham Abou-Houly	✅	✅	-	abouhouly	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (jali1)	✅	✅	-	jameslinnell	
			Unknown User (dasm13)	✅	✅	-	hmbpos10	
			Unknown User (jami14)	✅	✅	-	jamie-mills-bjss	
Personal Demographics Service ✅	Unknown User (siro6)	Unknown User (misc4)	Christina Tan (PO)	✅		-	-	Slacked Grant and Simon folow up 03 Jan 2023 Received partial information from Simon - awaiting Apigee access info 17 Jan 2023 Received information from Simon 04 Feb 2023
	Unknown User (telo4)		Chris Clarke (Tech Lead)	✅	✅	-	ChrisClarke-NHSD	

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Unknown User (guju1)	✅		-	-	
			Lindsay Ismay	✅		-	Lindsay-Ismay	
			Richard Seacome	✅	✅	-	richardseacome1-nhs	
			David Hamill	✅		-	davidhamill1-nhs	
			Claudiu Dumitrascu	✅		-	claudiudumitrascu1-nhs	
			Toni Whiteford	✅		-	ToniWhiteford-NHSD	
Communications Manager	Romy Bos	Unknown User (phsk1)	Romy Bos Delivery Lead					
			Laura Mountain Product Owner					
			Unknown User (phsk1) Tech Lead					

API	Delivery Lead - primary contact	Technical author (point of contact)	Team Members	Slack	Apigee	AWS	Github (username)	Date of last audit / last updated
			Tom Hattersley Business Analyst					
			Unknown User (sema2) Developer					
			Ian Hodges Developer					
			Unknown User (beoh1) Developer					
			Ada Smaldon Developer					
			Rachel Kennedy Quality Assurance					
			Siddharth Nair Platform					

8.3.3 Comms Template

Email/Message Template

Hello there,

I am reaching out as a security office working within API Management / Platforms. We are currently conducting a security audit, specifically focusing on producer team members and their access to the systems we manage.

I am hoping that you can provide the following info for [INSERT API NAME HERE]

Firslty please confirm:

- Delivery lead
- Tech authour
- Product owner
- Tech lead

And additionally, the names and shortcodes of everyone who works on [INSERT API NAME HERE] that has access to:

- NHS Digital Platforms Slack instance[1517]
- NHSD GitHub Org (https://github.com/orgs/NHSDigital/people)
    - GitHub usernames for people who have this access
- NHSD prod[1518]/non-prod[1519] Apigee instances
- Access to an AWS role we manage (this will be any role within the accounts details on this page: AWS[1520])

8.3.4 Audit ticket template

Use the ticket template below and create in the backlog under the APM-3193[1521] epic.

1517 https://nhsdigital-platforms.slack.com/archives/C016JRWN6AY
1518 https://nhs-digital-prod.login.apigee.com/login
1519 https://login.apigee.com/login
1520 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/190067578/AWS
1521 https://nhsd-jira.digital.nhs.uk/browse/APM-3193

8.3.4.1 Title

Producer Team Member Audit <month/year>

Audits should occur quarterly, please set the ticket title <month/year> for when the next audit should commence. There should also be an entry on the APIM calendar for the next audit to increase visibility.

8.3.4.2 Description

To complete the audit, please follow the process detailed on the Producer Team Member Audit (see page 1313) page.

Acceptance Criteria:

- Confirm table is up-to-date (Steps 1-3)
- Contact delivery leads and gather information (Step 4 + 5)
- Refer to KOP-036[1522]for any leavers and update the page (Step 6 + 7)
- Create a ticket for the next audit and add to calendar (Step 8 + 9)

8.4 What Good Looks Like checklist

8.4.1 Overview

The API Management What Good Looks Like (WGLL) checklist is a list of API features that make integration easier[1523]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

This checklist is all about building an API that is easy for API consumers to integrate with. There is a separate checklist for making sure you are building good quality, maintainable software - the NHS Digital Software Engineering Quality Framework[1524].

8.4.2 Why does the checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

1522 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=233085284
1523 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission
1524 https://github.com/NHSDigital/software-engineering-quality-framework

8.4.3 Who should use the checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.4 How this relates to our API policies and "quality measures"

We have a set of higher level "quality measures" that we use to track our overall quality of APIs, and that we report up to the NHS Digital board - see API quality measures[1525].

We have also published these externally - see API policies and best practice[1526].

8.4.5 How to use the checklist

See the instructions on What Good Looks Like (WGLL) checklist - template (see page 1383).

8.4.6 Checklist summaries

The following table provides a summary of the checklists completed so far.

Directorate	Sub-directorate	API	Owner	Long term plan	Link to checklist	Key gaps
Platforms	Core Services	Message Exchange for Social Care and Health (MESH)	Jody Dunstan	Invest	MESH API - What Good Looks Like (WGLL) checklist (see page 1347)	Doesn't use API platform Testing not self-service No adoption targets
	Demographics	Personal Demographics Service FHIR	Grant Bishop	Invest	TBC	TBC

1525 https://nhsd-confluence.digital.nhs.uk/display/APM/API+quality+measures
1526 https://digital.nhs.uk/developer/guides-and-documentation/api-policies-and-best-practice

Directorate	Sub-directorate	API	Owner	Long term plan	Link to checklist	Key gaps
		Personal Demographics Service HL7 V3	TBC	Divest - replace with FHIR API	N/A	N/A
		TBC others				
	Clinicals	TBC				
	Identity Access Management	CIS2 CIA	TBC	Invest	NHS CIS2 CIA - What Good Looks Like (WGLL) checklist (see page 1355)	TBC
		NHS login	TBC	Invest	NHS login - What Good Looks Like (WGLL) checklist (see page 1363)	TBC
		TBC others				
Product Delivery	TBC	TBC				
Data Services	TBC	Terminology Server	TBC	Invest	Terminology Server - What Good Looks Like (WGLL) checklist (see page 1372)	Doesn't use API platform TBC what else?
	TBC others	TBC others				

Directorate	Sub-directorate	API	Owner	Long term plan	Link to checklist	Key gaps
IT Operations	Cyber	Cyber Alerts	TBC	Invest	Cyber Alerts API - What Good Looks Like (WGLL) checklist (see page 1339)	Doesn't use API platform No empowered product owner
	TBC others	TBC others				

8.4.7 Cyber Alerts API - What Good Looks Like (WGLL) checklist

8.4.7.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist.

The WGLL checklist is a list of features that make integration easier[1527]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

This checklist is all about building an API that is easy for API consumers to integrate with. There is a separate checklist for making sure you are building good quality, maintainable software - the NHS Digital Software Engineering Quality Framework[1528].

8.4.7.2 Why does this checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

8.4.7.3 Who should use this checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

1527 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission
1528 https://github.com/NHSDigital/software-engineering-quality-framework

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.7.4 How this relates to our "quality measures"

We have a set of seven higher level "quality measures" that we use to track our overall quality of APIs, and that we report up to the NHS Digital board - see API quality measures (see page 1339).

In the table below, the "WGLL ref." column identifies which of the features roll up into the higher level quality measures.

8.4.7.5 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist".
2. Arrange a call with relevant members of your team and relevant people from API Management. See the "participants" section for recommended attendees.
3. Complete the "Participants" section.
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate.
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap.
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply.
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

8.4.7.6 Participants

We recommend you involve:

1. Your product owner
2. Your tech lead
3. Your delivery lead
4. One of our product owners (ideally Tony Heap)
5. One of our architects (Paul Fisher or Aubyn Crawford)

The following people have been involved in completing this checklist:

Name	Role
Tony Heap	API Management - product owner
Paul Fisher TBC	API Management - technical architect
Adam Boon	Respond to a cyber alert service - product owner
Matt Cooper (in lieu of Lee Alexander)	Respond to a cyber alert service - developer
Emma Adams	NHSD website - delivery lead
Geoff Heywood	NHSD website - developer

8.4.7.7 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

Status code	Meaning
TBC	This feature has yet to be understood and identified for this API.
SUPPORTED	The API supports this feature and includes supporting comments / evidence.
PLANNED	The API does not currently support the feature but work is planned to support it.
UNSUPPORTED	The API does not currently support the feature and there are currently no plans to support it.
N/A	This feature is not applicable for this API. Please include supporting reasons.

8.4.7.8 Checklist

Category	Feature	WGLLRef.	Status	Comments / evidence	Suggested action
Make learning easier	API is included in the API catalogue[1529].		SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/cyber-alerts-rest	
	API specification page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1530].	W1	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/cyber-alerts-rest	
	API specification includes a full description of all endpoints / interactions, including request and response examples and schemas.	W2	SUPPORTED	https://digital.nhs.uk/services/data-security-centre/cyber-alerts-api/cyber-alerts-api	
	API specification is rendered in our standard format (which is based on an underlying OAS file).	W2	UNSUPPORTED	Spec is rendered as static pages in Bloomreach	Opportunity to create OAS file for the API spec.
	API is included in contextual "building healthcare software" guides, so people know when to use it.		UNSUPPORTED	Not sure what guide this would sit in, would need to think it through.	Possible opportunity to add to an appropriate guide.

1529 https://digital.nhs.uk/developer/api-catalogue
1530 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page

	There are code snippets in multiple popular languages showing how to use the API.		UNSUPPORTED		Opportunity to add, but only if value is perceived.
	There is a Postman collection showing how to use the API, if applicable.		UNSUPPORTED		Opportunity to add, but only if value is perceived.
	There is a full reference implementation showing how to use the API.		UNSUPPORTED	"Respond to a cyber alert service" could be considered a reference implementation but it's in a private repo.	Open up the repo so it's coded in the open.
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1531].	W5	UNSUPPORTED	It's kind of restful but the get alert function has the ID as a request parameter, not in the URL. So, not perfect, not bad.	Not sure it's worth making a breaking change to improve this.
	API uses FHIR, if appropriate.	W5	N/A	I asked Brian Diggle, he says FHIR not a good fit for this API.	
	API uses FHIR R4, if appropriate.	W5	N/A	I asked Brian Diggle, he says FHIR not a good fit for this API.	
	API follows APIM design best practice.	W5	UNSUPPORTED	We don't advise open access.	Proxy through API platform to provide app-restricted access with an API key.

1531 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	W 5	UNSUPPORTED	Open access, should ideally be app-restricted with API key so we can track who's using it and let them know of any breaking changes.	Proxy through API platform to provide app-restricted access with an API key.
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.		UNSUPPORTED	Not on API platform	Proxy through API platform to provide app-restricted access with an API key.
	API uses the NHSD API platform, or an approved equivalent.	W 6	UNSUPPORTED		Proxy through the API platform.
	API is internet-facing, If appropriate.	W 4	SUPPORTED		
Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	W 3	UNSUPPORTED		Provide a sandbox via the API platform.
	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	W 3	UNSUPPORTED		Provide an integration text environment through the API platform.
	Integration test environment is self-service.	W 3	N/A	No test env't	
	Integration test environment has standing test data for read-only use, if appropriate. (and the API documentation includes guidance on how to use the test data)		N/A	No test env't	

	Testing tools are available, if appropriate. (and the API documentation includes guidance on how to use the tools)		N/A		
Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.		UNSUPPORTED	API is open access, so the process is known, but this means there is no way of us forcing API consumers to accept our terms and conditions.	
	API uses digital onboarding.	W7	UNSUPPORTED		
Make running the API easier	API has health status monitoring with pro-active alerting.		TBC	Geoff not sure whether the API endpoints have monitoring.	
	API has diagnostic logging.		TBC	Not sure - didn't set up anything specific so anything would be standard Bloomreach logging, whatever that is.	
	API has the ability to implement rate limiting[1532], specifically rate limiting per calling application (so that one application can't hog the whole bandwidth).		UNSUPPORTED		
	API is regularly pen tested.		SUPPORTED	Covered by NHSD website (Bloomreach) pen test	

1532 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

	API has security threat monitoring (generally provided by CSOC).		UNSUPPORTED		
	API has a dashboard / report showing volumes and error rates.		UNSUPPORTED		
Product mindset	API has a named product owner / manager.		UNSUPPORTED	Emma believes Adam is the product owner but Adam doesn't think he is - he thinks he's only the product owner for the "Respond to a cyber alert" service.	
	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.		UNSUPPORTED	No	
	The team pro-actively gets feedback from API consumers and acts on it.		UNSUPPORTED	The team does use the API themselves and have iterated in the past - "dogfooding", but there is no attempt to get feedback from other API consumers.	
	API consumers are asked to complete a developer integration survey rating their integration experience.		UNSUPPORTED		
	API consumers rate the API at 4 out of 5 on all measures.		UNSUPPORTED		
Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.		UNSUPPORTED	No	

	The API has achieved its adoption targets.		UNSUPPORTED	No	
	The API is out of beta (or is under 6 months old)		UNSUPPORTED	Still in beta after a couple of years of being live.	

8.4.7.9 Summary of recommendations

The Cyber Alerts API was built with an internal consumer in mind - the "Respond to a cyber alert" service. Adam, Lee and Matt own that service but don't claim strong ownership over the API itself - they see that as being "managed" by the web team (Emma and Geoff).

They built the API as an open API (API first - excellent) and published the spec (also excellent) but have never done any active promotion of the API. Adam has asked within Cyber whether it can be pushed further but has never been seen as a priority.

So the API is available but not actively managed as a product.

They have no idea who is using it or how much. We'd need some stats for that.

There is a possible opportunity to capitalise on the existence of the API, maybe like this:

- Assign a product owner
- Find out whether it is already being used by anyone other than the "Respond to a cyber alert" service - get transaction logs and use source IP to figure out if anyone else is using it.
- Brainstorm use cases - who else might want to use it?
- Contact registered developers - does anyone have a use case for it
- If there is interest in using it, consider
    - Moving it out of beta
    - Proxying it through the API platform
    - Developing a comms / engagement / adoption plan

8.4.8 MESH API - What Good Looks Like (WGLL) checklist

8.4.8.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist.

The WGLL checklist is a list of features that make integration easier[1533]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

8.4.8.2 Why does this checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

8.4.8.3 Who should use this checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.8.4 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist"
2. Arrange a call with relevant members of your team and relevant people from API Management. We recommend:
    a. Your product owner
    b. Your tech lead
    c. Your delivery lead
    d. One of our product owners (ideally Tony Heap)
    e. One of our architects (Paul Fisher or Aubyn Crawford)
3. Complete the "Participants" section
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

1533 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission

8.4.8.5 Participants

The following people have been involved in completing this checklist:

Name	Role
Philip Wilson	MESH Product Owner
Matt Mercer	MESH Tech Lead
Paul Fisher	APIM Architect
Tony Heap	APIM Product Owner

8.4.8.6 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

TBC

YES

NO

N/A

8.4.8.7 Checklist

Category	Feature	API fully supports the feature	Comments / evidence	Suggested action
Make learning easier	API is included in the API catalogue.	YES	https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api	

	API spec page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1534].	YES	As above	
	API spec includes a full description of all endpoints / interactions, including request and response examples and schemas.	YES	As above	
	API spec is rendered in our standard format (which is based on an underlying OAS file).	YES	https://github.com/NHSDigital/mesh-api-specification/blob/master/specification/mesh-api.yaml	
	API is included in contextual "building healthcare software" guides, so people know when to use it.	NO	The guides are in progress with APIM.	Check there is a ticket on the APIM backlog for this. There is - AMP-626.
	There are code snippets in multiple popular languages showing how to use the API.	YES	CLI plus Python examples	
	There is a Postman collection showing how to use the API.	N/A	No open access sandbox, so not possible to have an easily usable postman collection. In any case we already have a ref impl in python, not sure this would be a value add.	

1534 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page

	There is a full reference implementation showing how to use the API.	YES	TBC evidence	
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1535].	YES	Verbs are used in guide	
	API uses FHIR, if appropriate.	N/A	FHIR not appropriate for the MESH API. FHIR is used in some MESH payloads but this checklist is about the MESH API.	
	API uses FHIR R4.	N/A	As above.	
	API follows APIM design best practice.	YES	TBC how do we assess this?	
	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	NO	Token based Authorization TBC is this something you plan to change / improve? If not, why not?	
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.	NO	TLS-MA Client Certs TBC is this something you plan to change / improve? If not, why not?	
	API is internet-facing.	YES	Internet and Spine	

1535 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	YES	public access locally runnable sandbox https://github.com/NHSDigital/mesh-sandbox	Consider adding an open access sandbox for early developer testing. Add item to product backlog for this.
	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	YES		
	Integration test environment is self-service.	NO	Not fully self serve. Account sign up is not self serve, you have to request an account, manual process. Want to improve it.	Consider making integration testing fully self serve. That includes setting up MESH mailboxes, getting access to the MESH API (how would you self serve a TLS/MA cert?) Add item to product backlog.
	Integration test environment has standing test data for read-only testing, if appropriate. The API documentation includes guidance on how to use the tools.	N/A	NA	
	Integration test environment has tools / test harnesses for running tests and/or seeding/inspecting test data, if appropriate. The API documentation includes guidance on how to use the tools.	NO	Testing tools exist but the guidance on how to use them isn't in the API spec.	Improve guidance. Add item to product backlog.

Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.	YES	Page Here[1536]	
	API uses the SCAL for onboarding, which isn't the best, but is better than the CAP process.	N/A	API uses DOS, which is better.	
	API uses digital onboarding.	YES	Page Here[1537]	
Make running the API easier	API has health status monitoring with pro-active alerting.	YES	Yes, Splunk Grafana	
	API has diagnostic logging.	YES	Splunk, extensive logging	
	API has rate limiting.	NO	Behind WAF but not rate limited. Do want to consider, difficult to do because of bursty traffic. Agree it might be useful to reduce assurance burden.	Consider adding rate limiting. Add item to product backlog.
	API is regularly pen tested.	YES	Yes	
	API has security threat monitoring (generally provided by CSOC).	YES	Yes	
	API has a dashboard / report showing volumes and error rates.	YES	Yes grafana	
Product mindset	API has a named product owner / manager.	YES		

1536 https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#api-description__onboarding
1537 https://digital.nhs.uk/developer/api-catalogue/message-exchange-for-social-care-and-health-api#api-description__onboarding

	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.	YES		
	The team pro-actively gets feedback from API consumers and acts on it.	YES		
	API consumers are asked to complete a developer integration survey rating their integration experience.	NO		
	API consumers rate the API at 4 out of 5 on all measures.	NO	But, agree, getting user feedback is a good idea. Dependent on setting up the survey.	Keep gathering survey results and iterating your service until your users say it's great!
Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.	NO	MESH carries 60m messages Per month But we don't have any targets for improving adoption, and we're not sure whether we should. If it's agreed we're not interested in driving adoption, this becomes N/A.	Consider whether the product team has a remit to encourage adoption of the service. If so, set targets and consider how to drive adoption (it's hard). Maybe add a backlog item for this?
	The API has achieved its adoption targets.	NO	Need to set some targets first. If it's agreed we're not interested in driving adoption, this becomes N/A.	As above.

8.4.9 NHS CIS2 CIA - What Good Looks Like (WGLL) checklist

8.4.9.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist.

The WGLL checklist is a list of features that make integration easier[1538]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

This checklist is all about building an API that is easy for API consumers to integrate with. There is a separate checklist for making sure you are building good quality, maintainable software - the NHS Digital Software Engineering Quality Framework[1539].

8.4.9.2 Why does this checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

8.4.9.3 Who should use this checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.9.4 How this relates to our "quality measures"

We have a set of seven higher level "quality measures" that we use to track our overall quality of APIs, and that we report up to the NHS Digital board - see API quality measures (see page 1355).

In the table below, the "WGLL ref." column identifies which of the features roll up into the higher level quality measures.

1538 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission
1539 https://github.com/NHSDigital/software-engineering-quality-framework

8.4.9.5 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist".
2. Arrange a call with relevant members of your team and relevant people from API Management. See the "participants" section for recommended attendees.
3. Complete the "Participants" section.
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate.
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap.
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply.
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

8.4.9.6 Participants

We recommend you involve:

1. Your product owner
2. Your tech lead
3. Your delivery lead
4. One of our product owners (ideally Tony Heap)
5. One of our architects (Paul Fisher or Aubyn Crawford)

The following people have been involved in completing this checklist:

Name	Role
Oliver Guildford	NHS CIS2 - product owner
~~Martin Fell~~	~~TBC~~
Richard Grace	NHS CIS2 - tech lead

Name	Role
Adrian Hall	NHS CIS2 - tech lead
Tony Heap	API Management - product owner
~~Paul Fisher~~	~~API Management - architect~~

8.4.9.7 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

Status code	Meaning
TBC	This feature has yet to be understood and identified for this API.
SUPPORTED	The API supports this feature and includes supporting comments / evidence.
PLANNED	The API does not currently support the feature but work is planned to support it.
UNSUPPORTED	The API does not currently support the feature and there are currently no plans to support it.
N/A	This feature is not applicable for this API. Please include supporting reasons.

8.4.9.8 Checklist

Category	Feature	WGLL ref.	Status	Comments / evidence	Suggested action

Make learning easier	API is included in the API catalogue[1540].		SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/nhs-cis2-care-identity-authentication	
	API specification page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1541].	W1	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/nhs-cis2-care-identity-authentication	
	API specification includes a full description of all endpoints / interactions, including request and response examples and schemas.	W2	SUPPORTED	ttps://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance[1542]	
	API specification is rendered in our standard format (which is based on an underlying OAS file).	W2	N/A	Not of value for this OIDC API.	
	API is included in contextual "building healthcare software" guides, so people know when to use it.		UNSUPPORTED	We do need to write a guide on how security works in the NHS, covering CIS2 and NHS login and RBAC and more - this would probably go in the overall intro to healthcare	Probably already have a ticket for it on AMP backlog.

1540 https://digital.nhs.uk/developer/api-catalogue
1541 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page
1542 https://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance

	There are code snippets in multiple popular languages showing how to use the API.		UNSUPPORTED	Could be done but not seen a lot of pain in this area so not a high priority.	
	There is a Postman collection showing how to use the API, if applicable.		UNSUPPORTED	Guidance has examples CURLs (https://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance) which could be easily converted into Post	
	There is a full reference implementation showing how to use the API.		UNSUPPORTED	An exemplar client has been discussed but not seen as a priority.	
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1543].	W5	N/A	API uses OIDC, an open standard - not strictly RESTful but not a problem.	
	API uses FHIR, if appropriate.	W5	N/A	FHIR not appropriate.	
	API uses FHIR R4, if appropriate.	W5	N/A	FHIR not appropriate.	
	API follows APIM design best practice.	W5	SUPPORTED	API uses OIDC, which is in line with best practice.	

1543 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	W 5	N/A	It's a security API not a business API. It uses OIDC, which is built on OAuth 2.0, but not in the way this requirement means.	
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.		N/A	This API is **part of** the API platform security, rather than using it.	
	API uses the NHSD API platform, or an approved equivalent.	W 6	SUPPORTED	It's been agreed that NHS CIS2 will **not** be proxied through the API platform, although there is talk of using the developer account as a method for getting access to test environments.	
	API is internet-facing, If appropriate.	W 4	SUPPORTED		
Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	W 3	UNSUPPORTED	Feasible. Useful? Good question.	
	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	W 3	SUPPORTED	There are multiple environments, including DEV and INT. There is potentially an open issue re. mis-alignment of environments between CIS2 and APIM.	

	Integration test environment is self-service.	W 3	PLANNED	(OG - We are interested in continuing discussions about how to best enable some sort of self service for our PTL DEV environment for now)	
	Integration test environment has standing test data for read-only use, if appropriate. (and the API documentation includes guidance on how to use the test data)		N/A	It would be tricky to have shared users. (OG - There is a standard request / set up process for users to be set up in DEV and INT)	
	Testing tools are available, if appropriate. (and the API documentation includes guidance on how to use the tools)		UNSUPPORTED	This would be tricky - would require tools to set up RBAC.	
Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.		SUPPORTED		
	API uses digital onboarding.	W 7	PLANNED	CIS2 is on the roadmap for digital onboarding.	
Make running the API easier	API has health status monitoring with pro-active alerting.		SUPPORTED		
	API has diagnostic logging.		SUPPORTED		

	API has the ability to implement rate limiting[1544], specifically rate limiting per calling application (so that one application can't hog the whole bandwidth).		N/A	Don't think applicable since it's an authentication service. Does have DoS protection.	
	API is regularly pen tested.		SUPPORTED	(OG - Yes - last pen tested when AAL2 when live June 2022)	
	API has security threat monitoring (generally provided by CSOC).		SUPPORTED	Integrated with CSOC. Also do monthly analysis, which finds more than CSOC.	
	API has a dashboard / report showing volumes and error rates.		SUPPORTED	https://nhsdigital.splunkcloud.com/en-GB/app/NHSD_nhsid_prod_sh_all_viz/service_overview?form.Index_Suffix=prod&form.Environment=live	
Product mindset	API has a named product owner / manager.		SUPPORTED	Oliver Guildford	
	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.		SUPPORTED	Could always be better.	
	The team pro-actively gets feedback from API consumers and acts on it.		PLANNED	User researcher recruited.	
	API consumers are asked to complete a developer integration survey rating their integration experience.		UNSUPPORTED		

1544 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

	API consumers rate the API at 4 out of 5 on all measures.		UNSUPPORTED		
Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.		UNSUPPORTED	(OG - CIS1 Auth stack to be decommissioned in Sept '23 - ie. everyone onto CIS2 by then!) Opportunity to market the product beyond existing known potential users.	
	The API has achieved its adoption targets.		UNSUPPORTED		
	The API is out of beta (or is under 6 months old)		SUPPORTED		

8.4.10 NHS login - What Good Looks Like (WGLL) checklist

8.4.10.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist.

The WGLL checklist is a list of features that make integration easier[1545]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

This checklist is all about building an API that is easy for API consumers to integrate with. There is a separate checklist for making sure you are building good quality, maintainable software - the NHS Digital Software Engineering Quality Framework[1546].

8.4.10.2 Why does this checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

1545 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission
1546 https://github.com/NHSDigital/software-engineering-quality-framework

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

8.4.10.3 Who should use this checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.10.4 How this relates to our "quality measures"

We have a set of seven higher level "quality measures" that we use to track our overall quality of APIs, and that we report up to the NHS Digital board - see API quality measures (see page 1363).

In the table below, the "WGLL ref." column identifies which of the features roll up into the higher level quality measures.

8.4.10.5 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist".
2. Arrange a call with relevant members of your team and relevant people from API Management. See the "participants" section for recommended attendees.
3. Complete the "Participants" section.
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate.
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap.
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply.
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

8.4.10.6 Participants

We recommend you involve:

1. Your product owner
2. Your tech lead
3. Your delivery lead
4. One of our product owners (ideally Tony Heap)
5. One of our architects (Paul Fisher or Aubyn Crawford)

The following people have been involved in completing this checklist:

Name	Role
Darren Hutton	Product owner - NHS login
Tony Heap	Product owner - API Management
Priyanka	TA
Ian	TL

8.4.10.7 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

Status code	Meaning
TBC	This feature has yet to be understood and identified for this API.
SUPPORTED	The API supports this feature and includes supporting comments / evidence.
PLANNED	The API does not currently support the feature but work is planned to support it.
UNSUPPORTED	The API does not currently support the feature and there are currently no plans to support it.

Status code	Meaning
N/A	This feature is not applicable for this API. Please include supporting reasons.

8.4.10.8 Checklist

Category	Feature	WGLLRef.	Status	Comments / evidence	Suggested action
Make learning easier	API is included in the API catalogue[1547].		SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/nhs-login	
	API specification page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1548].	W1	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/nhs-login	
	API specification includes a full description of all endpoints / interactions, including request and response examples and schemas.	W2	SUPPORTED		
	API specification is rendered in our standard format (which is based on an underlying OAS file).	W2	N/A	Use of OAS is possible for OIDC APIs, but it's probably not worth it in the way it would be for "business" APIs.	

1547 https://digital.nhs.uk/developer/api-catalogue
1548 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page

	API is included in contextual "building healthcare software" guides, so people know when to use it.		PLANNED	This is a job for API Management.	APIM to deliver security over view content. AMP-113
	There are code snippets in multiple popular languages showing how to use the API.		SUPPORTED	https://nhsconnect.github.io/nhslogin/example-oidc/	
	There is a Postman collection showing how to use the API, if applicable.		UNSUPPORTED	Possible opportunity for improvement. Ian says we do have some postman tests we hand to suppliers	Consider whether making postman collections available would benefit API consumers.
	There is a full reference implementation showing how to use the API.		SUPPORTED	The "code snippets" are actually full client apps so, yes. Biometric code is open source.	
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1549].	W5	N/A	This API uses OIDC, so REST isn't appropriate	
	API uses FHIR, if appropriate.	W5	N/A	This is not a clinical API, so FHIR isn't appropriate.	
	API uses FHIR R4, if appropriate.	W5	N/A	As above	

1549 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

	API follows APIM design best practice.	W 5	PLANNED	API uses OIDC and is integrated with the API platform using token exchange. There is an open issue re. whether combined auth pattern is appropriate. Action on APIM.	APIM to complete investigation into the appropriateness of combined auth. APM-3075[1550] - NHS login - combined auth revisit and consent FY23-24 DONE
	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	W 5	N/A	This API is OIDC which is, of course, built on top of OAuth 2.0. But it doesn't, and shouldn't **use** OAuth 2.0 in the way that this feature means.	
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.		N/A	As above.	
	API uses the NHSD API platform, or an approved equivalent.	W 6	SUPPORTED	It's been agreed that NHS login will not be proxied via the API platform.	
	API is internet-facing, If appropriate.	W 4	SUPPORTED		

1550 https://nhsd-jira.digital.nhs.uk/browse/APM-3075?src=confmacro

Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	W3	SUPPORTED	The "mock" is totally open access but isn't actually that popular. https://nhsconnect.github.io/nhslogin/compare-environments/ But maybe it would benefit from some more work - could it be less hidden?	Consider whether there is an opportunity to improve the open access "mock".
	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	W3	SUPPORTED	NHS login has multiple path to live environments. https://nhsconnect.github.io/nhslogin/compare-environments/	
	Integration test environment is self-service.	W3	UNSUPPORTED	Possible opportunity to reduce support burden with some self serve aspects but concern over curating the developer journey.	Consider whether there is an opportunity to integrate NHS login with the developer hub to allow API consumers / partners to self serve access to early PTL environments e.g. sandbox.
	Integration test environment has standing test data for read-only use, if appropriate. (and the API documentation includes guidance on how to use the test data)		SUPPORTED	Standard test pack with standard test users. They can also create their own test users.	

	Testing tools are available, if appropriate. (and the API documentation includes guidance on how to use the tools)		SUPPORTED	Some niche changes are required e.g. IM1 integration.	
Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.		SUPPORTED		
	API uses digital onboarding.	W7	SUPPORTED		
Make running the API easier	API has health status monitoring with pro-active alerting.		SUPPORTED		
	API has diagnostic logging.		SUPPORTED		
	API has the ability to implement rate limiting[1551], specifically rate limiting per calling application (so that one application can't hog the whole bandwidth).		SUPPORTED	Not rate limited by API consumer but not causing pain.	
	API is regularly pen tested.		SUPPORTED	Annual	
	API has security threat monitoring (generally provided by CSOC).		SUPPORTED		
	API has a dashboard / report showing volumes and error rates.		SUPPORTED		

1551 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

Product mindset	API has a named product owner / manager.		SUPPORTED	Darren Hutton	
	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.		SUPPORTED		
	The team pro-actively gets feedback from API consumers and acts on it.		SUPPORTED	Slack workspace, user researcher(s). Partner forum.	
	API consumers are asked to complete a developer integration survey rating their integration experience.		UNSUPPORTED		Consider issuing a developer integration survey to API consumers / partners once they have completed onboarding (you could make it part of your onboarding process like we have for the PDS FHIR API).
	API consumers rate the API at 4 out of 5 on all measures.		UNSUPPORTED		As above.

Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.		UNSUPPORTED	There are pro-active adoption activities e.g. hackathons	Consider whether setting adoption targets might be a useful wat to drive team behaviour in a positive way - would it cause you to think harder about how to encourage adoption?
	The API has achieved its adoption targets.		UNSUPPORTED		As above.
	The API is out of beta (or is under 6 months old)		SUPPORTED		

8.4.11 Terminology Server - What Good Looks Like (WGLL) checklist

8.4.11.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist.

The WGLL checklist is a list of features that make integration easier[1552]. The features provide solutions to pain points that we have seen across multiple APIs over the years, based on hundreds of person-hours of user research.

8.4.11.2 Why does this checklist exist?

The main purpose of this page is to help API producer teams identify specific improvements they could make to their API.

It also helps APIM report on the overall quality of NHS Digital's API estate - which supports management action and funding for API improvements.

1552 https://digital.nhs.uk/developer/guides-and-documentation/api-management-vision#our-mission

8.4.11.3 Who should use this checklist

If you are building an API on our API platform and following our API delivery process, you probably don't need to use this checklist - the API platform will give you some of the features "for free" and the process will encourage you to deliver the other features as you progress.

So, the primary audience is owners of APIs that are **not** on the API platform. Specifically, we may have agreed, for whatever reason, that it is not appropriate to build a given API on the platform, or move an existing API to the platform. This checklist allows you to find out what you're missing out on, and consider building those useful features yourself.

8.4.11.4 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist".
2. Arrange a call with relevant members of your team and relevant people from API Management. See the "participants" section for recommended attendees.
3. Complete the "Participants" section.
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate.
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap.
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply.
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

8.4.11.5 Participants

We recommend you involve:

1. Your product owner
2. Your tech lead
3. Your delivery lead
4. One of our product owners (ideally Tony Heap)
5. One of our architects (Paul Fisher or Aubyn Crawford)

The following people have been involved in completing this checklist:

Name	Role
Tony Heap	APIM - product owner
Paul Fisher	APIM - architect
Tracy Simmonite	NHSD - terminology team lead
Roger Jane	NHSD - terminology development
Danny Cowley	Dedalus (runs the API on behalf of NHSD) - product / content
Dougal Fleming	Dedalus - tech

8.4.11.6 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

Status code	Meaning
TBC	This feature has yet to be understood and identified for this API.
SUPPORTED	The API supports this feature and includes supporting comments / evidence.
PLANNED	The API does not currently support the feature but work is planned to support it.
UNSUPPORTED	The API does not currently support the feature and there are currently no plans to support it.
N/A	This feature is not applicable for this API. Please include supporting reasons.

8.4.11.7 Checklist

Category	Feature	Status	Comments / evidence	Suggested action
Make learning easier	API is included in the API catalogue[1553].	SUPPORTED		
	API specification page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1554].	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/terminology-server-fhir We did this in collaboration with Nigel Southwell in April 2022. Looking at the spec now, I would say it should be "API" not "APIs" (singular not plural).	APIM - update API spec - change "APIs" to "API". AMP-665
	API specification includes a full description of all endpoints / interactions, including request and response examples and schemas.	SUPPORTED	Supported, but it's bit tricky to get to the details. You have to navigate through a lot of pages until you eventually get to a page that explains which endpoints the Ontoserver product supports, and then you have to dig into the standard FHIR docs to see how exactly it works. This makes it tricky for a developer to figure out how to integrate.	Produce an OAS-based spec

1553 https://digital.nhs.uk/developer/api-catalogue
1554 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page

	API specification is rendered in our standard format (which is based on an underlying OAS file).	UNSUPPORTED		Produce an OAS-based spec
	API is included in contextual "building healthcare software" guides, so people know when to use it.	PLANNED	This is in progress with APIM - see various stories in AMP-125.	APIM to complete
	There are code snippets in multiple popular languages showing how to use the API.	SUPPORTED	See https://aehrc.github.io/fhir-ts-exemplars/.	
	There is a Postman collection showing how to use the API, if applicable.	SUPPORTED	See https://ontoserver.csiro.au/docs/6/api-fhir.html.	
	There is a full reference implementation showing how to use the API.	SUPPORTED	See https://digital.nhs.uk/services/terminology-servers/training-guides/integrate-my-system-with-the-terminology-server/developer-launchpad#example-applications.	
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1555].	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/terminology-server-fhir#technology	
	API uses FHIR, if appropriate.	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/terminology-server-fhir#technology	

1555 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

	API uses FHIR R4, if appropriate.	SUPPORTED	https://digital.nhs.uk/developer/api-catalogue/terminology-server-fhir#technology	
	API follows APIM design best practice.	N/A	It's a third party product that has been meet to conform to the FHIR global standard, so not much point assessing it against NHSD design best practice.	
	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	SUPPORTED	From Dougal: integrated to several external IdPs including nhs mail How are the API endpoints secured? Simple API key or something else? Where is this documented - I can't find it! I can see that I need to apply for a system-to-system account - I presume this will give me an API key but I'm not sure.	Improve documentation to explain the security options.
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.	UNSUPPORTED	This means API consumers using the Terminology Server API will not be able to take advantage of having a single set of access creds across all their integrations.	Make API available through NHSD API platform

	API is internet-facing, If appropriate.	SUPPORTED		
Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	UNSUPPORTED	To get access to the API, even for test purposes, API consumers need to request access via https://digital.nhs.uk/services/terminology-servers/training-guides/new-to-the-terminology-server/account-management#system-to-system-account. There is no auth-free instance for early developer testing. Could be a barrier to entry, but also, if provided, people might use it instead of prod (unless you made sure it was not fully populated). From Dougal: Open access option is not part of the service – but there is an open access sandbox provided external to the service. See: https://documenter.getpostman.com/view/634774/TVsuBmc9	Investigate options for an open access sandbox. One option would be to use the API platform.

	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	UNSUPPORTED	Looks like there is no test environment - would have to test in production. Current approach is to test on production analytics server and then use production clinical server once tested.	Consider providing a self-service formal integration test environment. You could get this pretty much for free by using the API platform - a separate logical environment could connect through to your prod server under the covers, but with a different access process and rate limiting. Feedback from your users would tell you whether this would be a benefit.
	Integration test environment is self-service.	UNSUPPORTED	Dougal: People can have a play using the user based accounts – we provide a postman collection that provides lots of examples that uses the self registration user accounts (https://github.com/NHSDigital/TerminologyServer) But I guess that this isn't really formal integration testing.	As above
	Integration test environment has standing test data for read-only use, if appropriate. (and the API documentation includes guidance on how to use the test data)	N/A	No separate test environment	
	Testing tools are available, if appropriate. (and the API documentation includes guidance on how to use the tools)	N/A	The API is read only	

Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.	SUPPORTED	https://digital.nhs.uk/services/terminology-servers/training-guides/new-to-the-terminology-server/account-management#system-to-system-account	
	API uses digital onboarding.	UNSUPPORTED	Using digital onboarding would give API consumers a more consistent experience when integrating with multiple APIs.	Use digital onboarding for granting production access.
Make running the API easier	API has health status monitoring with pro-active alerting.	SUPPORTED	Dougal: yes Dedalus provides OOH support.	
	API has diagnostic logging.	SUPPORTED	Dougal: yes	
	API has the ability to implement rate limiting[1556], specifically rate limiting per calling application (so that one application can't hog the whole bandwidth).	SUPPORTED	Dougal: You can – and there is ddos protection … but if it is a legitimate use then we wouldn't, we would ensure there was enough capacity in the service (and the cache) to maintain service to other API consumers.	
	API is regularly pen tested.	SUPPORTED	Dougal: yes	

1556 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

	API has security threat monitoring (generally provided by CSOC).	TBC	Dougal: yes, but unable to supply details in a sharable form, so marking as unsupported.	Verify that capability matches NHSD's own capability.
	API has a dashboard / report showing volumes and error rates.	SUPPORTED	Dougal: yes	
Product mindset	API has a named product owner / manager.	SUPPORTED	Tracy Simmonite is the overall lead Karim Nashar Might get a formal product owner	
	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.	TBC	This is evidenced to some extent by ongoing discussions and improvements happening between NHSD and Dedalus. Tracy says they have some funding but not sure if it's enough.	
	The team pro-actively gets feedback from API consumers and acts on it.	PLANNED	Not doing but recognise need and plan to do.	
	API consumers are asked to complete a developer integration survey rating their integration experience.	PLANNED	Might be worth us showing the team what this is.	
	API consumers rate the API at 4 out of 5 on all measures.	PLANNED	As above.	
Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.	UNSUPPORTED	Might be worth discussing how adoption targets have helped APIM.	

	The API has achieved its adoption targets.	UNSUPPORTED	As above.	

8.4.11.8 Summary of recommendations

8.4.11.8.1 1. Introduce good product management disciplines

We recommend you:

- Identify someone (possibly recruit someone) as a named product owner, who has responsibility for owning and developing the Terminology Server, including its API
- Make sure they have the resources and budget they need to do that
- Gather feedback from users of the API - specifically, people from organisations who have integrated with the API - look for coders, architects, product owners, delivery leads
- Specifically, create a developer integration survey based on our template and get them to complete it
- Define adoption targets
- Make a plan to meet your adoption targets - consider how you can reach your potential audience. We can help with that.

8.4.11.8.2 2. Provide an OAS spec

An OAS spec makes integration easier, because:

- all the technical and non-technical integration details are in one place, not spread across multiple pages and sites
- your spec is in a format that's consistent with our other APIs - which is good for API consumers using multiple APIs
- it allows API consumers to download a machine-readable version of your spec as an "OAS" file - some coders find this makes integration easier as they can auto-generate code and stubs

8.4.11.8.3 3. Use the API platform

Advantages:

- Make design and build easier:
    - Provides a consistent approach for API consumers to connect to all NHS Digital APIs - one account, one set of security details. This seems especially important for the the Terminology Server API because it seems like it would be used in conjunction with other APIs e.g. EPS, SCR, GP Connect, Immunisation History.
- Make testing easier:
    - Provides a logically separate test environment for API consumers to formally test their integration with before "going live". Under the covers, you could possibly connect this to your

prod environment, but it would seem separate to API consumers, and you could log and record the traffic separately, and also set different rate limits.

- The test environment is self service.

Disadvantages and risks:

- It needs building, which will cost money, and require skills. But APIM is here to help with two engineers full time on API producer support, for skills transfer and helping share the load.
- It introduces an additional layer in the architecture. If this breaks, the API (a platinum service) would be unavailable. However, the API platform is itself a platinum service, and is already used for other platinum services, such as PDS, and has been 100% available for at least the last 12 months. It runs in the cloud, is owned by Google, who are pretty good at high availability, and has multiple failover sites.
- It introduces an additional component to run and maintain. However, experience shows that once built, the API proxy layer rarely needs touching, so the burden is light. And, again, APIM is here as a back-up support, 24x7.

8.4.11.8.4 4. Use digital onboarding

Digital onboarding makes integration easier because:

- It's online, not by email
- it's consistent with onboarding for other NHSD APIs, using the same "developer account"
- it reduces the burden on API consumers because we don't have to ask them the same questions again and again for each API
- Your onboarding lead can grant access quickly and easily, and also keep track of who has access, and get reports and metrics

8.4.12 What Good Looks Like (WGLL) checklist - template

How to use this checklist

To use this checklist:

- Make a copy of the page
- Name it "<Your API name> - What Good Looks Like (WGLL) checklist"
- Delete this info panel

8.4.12.1 Overview

This page shows how the above API performs against the API Management What Good Looks Like (WGLL) checklist (see page 1336).

8.4.12.2 How to use this checklist

1. If you are looking at the template page right now, make a copy of it and rename it as "<Your API name> - What Good Looks Like (WGLL) checklist".
2. Arrange a call with relevant members of your team and relevant people from API Management. See the "participants" section for recommended attendees.
3. Complete the "Participants" section.
4. Complete the "Checklist" section:
    a. If your API **fully** supports a feature, put "Yes" and add a comment or a link to some evidence as appropriate.
    b. If your API doesn't **fully** support a feature, put "No" and add a comment to explain the gap.
    c. If the feature isn't applicable to your API, put "N/A" and explain why it doesn't apply.
5. Identify actions - this is the whole point, to identify opportunities to improve your API.
    a. A common type of action is to raise an item on your backlog to address the gap. In this case, once done, add a link to the Jira story or epic you've raised.

8.4.12.3 Participants

We recommend you involve:

1. Your product manager / owner
2. Your tech lead
3. Your delivery lead
4. One of our product owners (ideally Tony Heap)
5. One of our architects (Paul Fisher or Aubyn Crawford)

The following people have been involved in completing this checklist:

Name	Role

Name	Role

8.4.12.4 Status codes

Cut and paste these status codes into the "Status" column as appropriate:

Status code	Meaning
TBC	This feature has yet to be understood and identified for this API.
SUPPORTED	The API supports this feature and includes supporting comments / evidence.
PLANNED	The API does not currently support the feature but work is planned to support it.
UNSUPPORTED	The API does not currently support the feature and there are currently no plans to support it.
N/A	This feature is not applicable for this API. Please include supporting reasons.

8.4.12.5 Checklist

Category	Feature	WGLL ref.	Status	Comments / evidence	Suggested action
Make learning easier	API is included in the API catalogue[1557].		TBC		

1557 https://digital.nhs.uk/developer/api-catalogue

	API specification page provides general information about the API - functional overview, legal use, related APIs, status, service level, technology, network access, security pattern, testing, onboarding - as per Writing a bronze API cover page[1558].	W 1	TBC		
	API specification includes a full description of all endpoints / interactions, including request and response examples and schemas.	W 2	TBC		
	API specification is rendered in our standard format (which is based on an underlying OAS file).	W 2	TBC		
	API is included in contextual "building healthcare software" guides, so people know when to use it.		TBC		
	There are code snippets in multiple popular languages showing how to use the API.		TBC		
	There is a Postman collection showing how to use the API, if applicable.		TBC		
	There is a full open source reference implementation showing how to use the API.		TBC		
Make design and build easier	API is RESTful, if appropriate. Specifically to level 2 on the Richardson Maturity Model[1559].	W 5	TBC		

1558 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/125405648/Writing+a+bronze+API+cover+page
1559 https://martinfowler.com/articles/richardsonMaturityModel.html?repost=true

	API uses FHIR, if appropriate.	W 5	**TBC**		
	API uses FHIR R4, if appropriate.	W 5	**TBC**		
	API follows APIM design best practice. Designing your API (see page 322)	W 5	**TBC**		
	API uses OAuth2 for security, if appropriate (whether that be app-restricted, user-restricted with NHS CIS2 or user-restricted with NHS login).	W 5	**TBC**		
	API uses API platform security, if appropriate, as per https://digital.nhs.uk/developer/guides-and-documentation/security-and-authorisation.		**TBC**		
	API uses the NHSD API platform, or an approved equivalent.	W 6	**TBC**		
	API is internet-facing, If appropriate.	W 4	**TBC**		
Make testing easier	API has a self-service "sandbox" for early developer testing. The sandbox is open access.	W 3	**TBC**		
	API has an integration test environment for formal integration testing. The integration test environment closely matches production, including the same security.	W 3	**TBC**		
	Integration test environment is self-service.	W 3	**TBC**		

	Integration test environment has standing test data for read-only use, if appropriate. (and the API documentation includes guidance on how to use the test data)		TBC		
	Testing tools are available, if appropriate. (and the API documentation includes guidance on how to use the tools)		TBC		
Make onboarding easier	API has a known onboarding process, meaning it is possible to onboard to the API.		TBC		
	API uses digital onboarding.	W 7	TBC		
Make running the API easier	API has health status monitoring with pro-active alerting.		TBC		
	API has diagnostic logging.		TBC		
	API has the ability to implement rate limiting[1560], specifically rate limiting per calling application (so that one application can't hog the whole bandwidth).		TBC		
	API is regularly pen tested.		TBC		
	API has security threat monitoring (generally provided by CSOC).		TBC		
	API has a dashboard / report showing volumes and error rates.		TBC		

1560 https://digital.nhs.uk/developer/guides-and-documentation/reference-guide#rate-limits

	The API, and any underlying service code, are coded in the open[1561], allowing others to inspect, contribute to and re-use the source code.		TBC		
Product mindset	API has a named product owner / manager.	W 9	TBC		
	The product owner feels like they have the resources and budget to pro-actively manage (improve) the product.	W 9	TBC		
	The team pro-actively gets feedback from API consumers and acts on it.	W 9	TBC		
	API consumers are asked to complete a developer integration survey rating their integration experience.	W 9	TBC		
	API consumers rate the API at 4 out of 5 on all measures.		TBC		
	API Producer teams have a well laid out plan for Run & Maintain of the API - Post Go Live; with adequate handover and supporting documentation for any new R&M team.	T B C	TBC	**To be reviewed!**	
Adoption	The API has adoption targets - target number of API consumers; target number of transactions per month.	W 9	TBC		
	The API has achieved its adoption targets.		TBC		
	The API is out of beta (or is under 6 months old)	W 8	TBC		

1561 https://www.gov.uk/service-manual/service-standard/point-12-make-new-source-code-open

9 Release Schedule

Search Release Plans:

Search

Infra Service

We must release the infra service at least once per week. We'll do this on a Tuesday evening, if there is at least one other release going out that night. Otherwise we'll do it on the Thursday, regardless of whether there are other releases going out or not.

- Copy the [Template] IAM Release - AfCID x.x.x[1562] page and update (which includes a list of tickets that saves people opening ServiceNow)
- In the Release Plans column, put a link to each release note page
- Some smaller changes don't require an RFC - or the timing of a release might result in an in hours deployment. This draft page sets how Deathstar currently approaches this and might provide useful guidance: Release Management (see page 2077)
- Ensure that any work items being pushed through as part of a 'standard changes' have been properly triaged as per instructions here - Standard Changes (see page 2093)
- Attend the "release co-ordination session" on the day of the release, where:
    - APIM will do a quick review of dependencies and changes (with a git diff), and to spot any potentially disruptive edge cases
    - In an in-hours deploy is needed, then attending an "release co-ordination session" to do the quick review will be needed to confirm suitability for in-hours
- Update RFC Approved and Release Status as they progress

Please use the template - we need to have an quick reference to record: who originated the RFC; links to JIRA tickets; and a simple description.

For guidance see:

1. When do I need an RFC, and when can I release see - Release Management (see page 2077)
2. What are the steps when making a release

1562 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/529500791/Template+IAM+Release+-+AfCID+x.x.x

- Deploying to production with Proxy Generator (see page 1135) if using Proxy Generator; *OR*
- Deploying to production (see page 1120)

9.1 Planned Releases (with RFC)

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
17 Jun 2025	1. CHG0076052[1563] NHS e-RS APIM Referral Release R1.45 Deployment (Prod) 2. CHG0076051[1564] NHS e-RS APIM Patient Care Release R1.37 Deployment (Prod)	1. . 2. .	1. . 2. .	
17 Jun 2025	1. CHG0076652 - APIM Infrastructure Updates (see page 2053)	1. ✓ (Normal)	1.	
05 Jun 2025	1. CHG0076937 - APIM Infrastructure Updates (see page 2054)	1. ✓ (Normal)	1. ✓	
27 May 2025	1. CHG0076569 - Digital Onboarding Service (see page 2051)	1. ✓ (standard)	1. ✓	
19 May 2025	1. CHG0076180 - Identity Custom Rate Limit Update for IPlato - API Management (see page 2049) 2. CHG0076190 - Multicast Notifications Service (see page 2050)	1. ✓ 2. ✓	1. ✓ 2. ✓	

1563 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df64376e0c32dee540317da3c05013188%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1564 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df5e232ecc3e9ee540317da3c05013133%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
13 May 2025	1. CHG0074938[1565] NHS e-RS APIM Referral Release R1.44 Deployment (Prod) 2. CHG0074937[1566] NHS e-RS APIM Patient Care Release R1.36 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
12 May 2025	1. CHG0075918 (E) - APIM Infrastructure Updates (see page 2047)	1. ✅ Emergency	1. ✅	
09 May 2025	1. CHG0075754 (N) & CHG0075915 (Er) - APIM Infrastructure Updates (see page 2046)	1. ✅ (Normal, Emergency)	1. ✅	
07 May 2025	1. CHG0075648 - Mock JWKS Updates (see page 2045)	1. ✅ (Normal)	1. ✅	
29 Apr 2025	1. CHG0075379 - Digital Onboarding Service (see page 2043)	1. ✅ (standard)	1. ✅	
23 Apr 2025	1. CHG0075022 - PDS Custom Rate Limit Update for IPlato - API Management - CDT Confluence (see page 2041)	1. ✅	1. ✅	

1565 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7c3426eb3bf8665076462a4a85e45a76
1566 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dc1e3a66b3bf8665076462a4a85e45aa0

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
15 Apr 2025	1. CHG0073590[1567] NHS e-RS APIM Referral Release R1.43 Deployment (Prod) 2. CHG0073589[1568] NHS e-RS APIM Patient Care Release R1.35 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
14 Apr 2025	1. CHG0074860 - APIM Infrastructure Updates (see page 2040)	1. ✅ (standard)	1. ✅	
10 Apr 2025	1. CHG0074642 - APIM Sync Wrap Updates (see page 2039)	1. ✅ (Normal)	1. ✅	
27 Mar 2025	1. CHG0073656 - Digital Onboarding Service (see page 2038)	1. ✅ (standard)	1. ✅	
27 Mar 2025	1. CHG0073996 - Identity Service Updates[1569]	1. ✅ (Normal)		
26 Mar 2025	1. CHG0073957 - APIM Splunk Monitor Updates (see page 2034)	1. ✅ (Normal)	1. ✅	
26 Mar 2025	1. CHG0073902 - APIM monitoring-service-discovery Updates (see page 2033)	1. ✅ (Normal)	1. ✅	

1567 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3b862e26c3d422d80317da3c050131a9%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1568 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5356ea22c3d422d80317da3c0501310e%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1569 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0073996+-+Identity+Service+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
20 Mar 2025	1. CHG0073656 - Digital Onboarding Service (see page 2032)	1. ✅	1. ✅	
18 Mar 2025	1. CHG0072193[1570] NHS e-RS APIM Referral Release R1.42 Deployment (Prod) 2. CHG0072194[1571] NHS e-RS APIM Patient Care Release R1.34 Deployment (Prod)	1. ✅ 2. ✅.	1. ✅ 2. ✅	
14 Mar 2025	**1.** CHG0073383 - APIM Infrastructure Updates[1572]	1. ✅ (Normal)	1. ✅	
12 Mar 2025	1. CHG0073262 - Digital Onboarding Service (see page 2030)	1. ✅ (standard)	1. ✅	
10 Mar 2025	1. CHG0073067 - Digital Onboarding Service (see page 2029)	1. ✅ (standard)	1. ✅	
06 Mar 2025	1. CHG0073010 - Digital Onboarding Content release for DAQ 22.0 Production (see page 2028)	1. ✅ (standard)	1. ✅	

1570 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D17b16ea1c37b5e100317da3c05013110%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull%26sysparm_record_row%3D2%26sysparm_record_rows%3D13%26sysparm_record_list%3Dactive%253dtrue%255ebusiness_service%253db3f5f2381bd6c590452bdce7b04bcb80%255eORDERBYDESCstart_date

1571 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D74026665c37b5e100317da3c0501318a%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1572 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0073383+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
26 Feb 2025	1. CHG0072682 - APIM Infrastructure Updates[1573]	1. ✅	1. ✅	
24 Feb 2025	1. CHG0072472 - APIM Infrastructure Updates (see page 2025)	1. ✅ (standard)	1. ✅	
24 Feb 2025	1. CHG0072360 - PDS Custom Rate Limit Update for Anima (see page 2011)	1. ✅ (normal)	1. ✅	
18 Feb 2025	1. CHG0070958[1574] NHS e-RS APIM Referral Release R1.41 Deployment (Prod) 2. CHG0070957[1575] NHS e-RS APIM Patient Care Release R1.33 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
17 Feb 2025	1. CHG0071920 - APIM Infrastructure Updates (see page 2021)	1. ✅ Normal	1. ✅	
12 Feb 2025	1. CHG0071979 - Digital Onboarding Content release for DAQ 21.0 Production (see page 2023)	1. ✅ Standard	1. ✅	
11 Feb 2025	1. CHG0071923 - Digital Onboarding Content release for DAQ 20.0 Production (see page 2022)	1. ✅ Standard	1. ✅	

1573 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0072682+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

1574 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dceea4a6cfb9b96d0cfc6feceaeefdc4e%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1575 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dbbda0a6cfb9b96d0cfc6feceaeefdc79%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D12%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
05 Feb 2025	1. CHG0071606 - APIM Infrastructure Updates (see page 2020)	1. ✅ Standard	1. ✅	
27 Jan 2025	1. CHG0071101 - Digital Onboarding Content release for DAQ 19.1 Production (see page 2019) 2. CHG0071016 - NHS App Custom Rate Limit Update (see page 2017)	1. ✅ 2. ✅ Normal	1. ✅ 1. ✅	
14 Jan 2025	1. CHG0070601 - Digital Onboarding Service (see page 2013)	1. ✅ Standard	1. ✅	
21 Jan 2025	1. CHG0069737[1576] NHS e-RS APIM Referral Release R1.40 Deployment (Prod) 2. CHG0069738[1577] NHS e-RS APIM Patient Care Release R1.32 Deployment (Prod)	1. .✅ 2. ✅.	1. .✅ 2. ✅.	
16 Jan 2025	1. CHG0070675 - Digital Onboarding Content release for DAQ 19.0 Production - API Management - CDT Confluence (see page 2015)	1. ✅	1. ✅	
16 Jan 2025	1. CHG0070584 - Multicast Notifications Service (see page 2010)	1. ✅	1. ✅	

1576 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D05a67d8bfb5ede50ecbef3baaeefdc29%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1577 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D9ab63dcbfb5ede50ecbef3baaeefdca7%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
15 Jan 2025	1. CHG0070705 - APIM Infrastructure Updates (see page 2016)	1. ✅	1. ✅	
13 Jan 2025	1. CHG0072360 - PDS Custom Rate Limit Update for Anima (see page 2011)	1. ✅ Standard	1. ✅	
07 Jan 2025	1. CHG0070342 - NHS App Custom Rate Limit Update (see page 2009)	1. ✅ Normal	1. ✅	
19 Dec 2024	1. CHG0070107 - APIM Infrastructure Updates (see page 2008)	1. ✅ Standard	1. ✅	
19 Dec 2024	1. CHG0070092 - Digital Onboarding Service (see page 2006)	1. ✅ Standard		
12 Dec 2024	1. CHG0069773 - Digital Onboarding Content release for DAQ 18.2 Production - API Management - CDT Confluence[1578]	1. ✅ Standard	1. ✅	
10 Dec 2024	1. CHG0068205[1579] NHS e-RS APIM Referral Release R1.39 Deployment (Prod) 2. CHG0068204[1580] NHS e-RS APIM Patient Care Release R1.31 Deployment (Prod)	1. . ✅ 2. . ✅	1. . ✅ 2. . ✅	

1578 https://nhsd-confluence.digital.nhs.uk/pages/resumedraft.action?draftId=964529437&draftShareId=f878cd6e-017b-4a2e-b4bc-9d03e9b986c4

1579 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Ded7865fb3b651ad067201da985e45a43%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1580 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db72829b73b651ad067201da985e45aa6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
02 Dec 2024	1. CHG0069390 - APIM Infrastructure Updates (see page 2004)	1. Standard		
27 Nov 2024	1. CHG0069202 - APIM Infrastructure Updates (see page 2003)	1. Standard	1.	
26 Nov 2024	1. CHG0069094 - APIM Infrastructure Updates [1581]	1. Standard	1.	
21 Nov 2024	1. CHG0068626 - Multicast Notifications Service (see page 2000)	1. Normal	1.	
21 Nov 2024	1. CHG0068898 - Digital Onboarding Service (see page 2001)	1. Standard	1.	
18 Nov 2024	1. CHG0068545 - APIM Infrastructure Updates (see page 1998)	1. Normal		
14 Nov 2024	1. CHG0068584 - Digital Onboarding Content release for DAQ 18.1 Production - API Management - CDT Confluence (see page 1999)	1. Standard	1.	
12 Nov 2024	1.CHG0068438- APIM Infrastructure Updates (see page 1997)	1. Standard	1.	

1581 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0069094+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
11 Nov 2024	1. CHG0068402 - Digital Onboarding Content release for DAQ 18.0 Production - API Management - CDT Confluence (see page 1996)	1. ✅ Standard		
05 Nov 2024	1. CHG0066998[1582] NHS e-RS APIM Referral Release R1.38 Deployment (Prod) 2. CHG0066999[1583] NHS e-RS APIM Patient Care Release R1.30 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
04 Nov 2024	1. CHG0068157- APIM Infrastructure Updates (see page 1994)	1. ✅ Standard	1. ✅ 1. ✅	
31 Oct 2024	1. CHG0068037 - Digital Onboarding Content release for DAQ 17.0 Production (see page 1993)	1. ✅ Standard	1. ✅	
31 Oct 2024	1. CHG0068017 - Digital Onboarding Service (see page 1992)	1. ✅ Standard	1. ✅	
29 Oct 2024	1. CHG0067945 - Digital Onboarding Content release for DAQ 16.1 Production (see page 1991)	1. ✅ Standard	1. ✅	

1582 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D431199b2c34dde546c39935c050131d8%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D16%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1583 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D2431d536c34dde546c39935c05013101%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
23 Oct 2024	1.CHG0067720- APIM Infrastructure Updates (see page 1990)	1. ✅ Standard	1. Failed (Reverted to previous successful Release)	
22 Oct 2024	1. CHG0067640- APIM Infrastructure Updates (see page 1989)	1. ✅ Standard	1. ✅	
21 Oct 2024	1. CHG0067588 - APIM Infrastructure Updates (see page 1988)	1. ✅ Standard	1. Failed (Reverted to previous successful Release)	
21 Oct 2024	1. CHG0067527 - APIM Infrastructure Updates[1584]	1. ✅ Normal	1. ✅	
21 Oct 2024	1. CHG0067490- Digital Onboarding Content release for DAQ 16.0 Production (see page 1986)	1. ✅ Standard	1. ✅	DOS2 release
21 Oct 2024	1. CHG0067443 - Digital Onboarding Service (see page 1984)	1. ✅ Standard		DOS2 release
10 Oct 2024	1. CHG0067046 - APIM Infrastructure Updates[1585]	1. ✅ Standard	1. ✅	
10 Oct 2024	1. CHG0066905 - APIM Infrastructure Updates (see page 1981)	1. ✅ Normal	1. ✅	

1584 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0067527+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

1585 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0067046+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
08 Oct 2024	1. CHG0065919[1586] NHS e-RS APIM Referral Release R1.37 Deployment (Prod) 2. CHG0065918[1587] NHS e-RS APIM Patient Care Release R1.29 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
03 Oct 2024	1. CHG0066743 - Digital Onboarding Service (see page 1979)	1. ✅ Standard	1. ✅	
01 Oct 2024	1. CHG0066582 - Digital Onboarding Service (see page 1978)	1. ✅ Standard	1. ✅	
26 Sep 2024	1. CHG0066347 - Digital Onboarding Service (see page 1977)	1. ✅ Standard	1. ❌ Cancelled (delayed until the 01/10/24)	
25 Sep 2024	1. CHG0066319 - APIM Infrastructure Updates (see page 1976)	1. ✅ Normal	1. ✅	
24 Sep 2024	1. CHG0066257 - APIM Infrastructure Updates (see page 1975)	1. ✅ Standard	1. ✅	
19 Sep 2024	1. CHG0066136 - APIM Infrastructure Updates (see page 1974)	1. ✅ Standard	1. ✅	

1586 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db0efcf2bc36cda580317da3c05013106%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1587 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df92fcbefc32cda580317da3c05013112%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
18 Sep 2024	1. CHG0065961 - NHS App Custom Rate Limit Update[1588]	1. ✅ Normal	1. ✅	
16 Sep 2024	1. CHG0065769 - Identity Service Updates (see page 1972)	1. ✅ Normal	1. ✅	
12 Sep 2024	1. CHG0065727 - APIM - Add rate limiting custom attribute to app[1589]	1. ✅ Normal	1. ✅	
13 Sep 2024	1. CHG0063574[1590] NHS e-RS APIM Referral Release R1.36 Deployment (Prod) 2. CHG0063575[1591] NHS e-RS APIM Patient Care Release R1.28 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	
10 Sep 2024	1. CHG0065725 - APIM Infrastructure Updates (see page 1969)	1. ✅ Standard	1. ✅	
05 Sep 2024	1. CHG0065516 - APIM Infrastructure Updates (see page 1968)	1. ✅ Standard	1. ✅	

1588 https://nhsd-confluence.digital.nhs.uk/display/APM/CHGCHG0065961+-+NHS+App+Custom+Rate+Limit+Update

1589 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0065727+-+APIM+-+Add+rate+limiting+custom+attribute+to+app?src=contextnavpagetreemode

1590 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df539684cc3674ed00317da3c05013124%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D17%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1591 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7889a000c3a74ed00317da3c05013172%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D17%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
12 Aug 2024	1. CHG0064624 - APIM Infrastructure Updates[1592]	1. ✅ Standard	1. ✅	
06 Aug 2024	1. CHG0064418 - APIM Infrastructure Updates (see page 1966)	1. ✅ Standard	1. ✅	
30 Jul 2024	1. CHG0064048 - Digital Onboarding Content release for DAQ 15.1 Production (see page 1962) 2. CHG0064121 - APIM Infrastructure Updates (see page 1963) 3. CHG0064142 - APIM Infrastructure Updates (see page 1965)	1. ✅ Standard 2. ✅ Standard 3. ✅ Normal	1. 2. ✅ 3. ✅	3. Update certs for int, internal-dev, dev, prod.
17 Jul 2024	1. CHG0063443 - APIM Infrastructure Updates (see page 1961)	1. ✅ Normal	1. ✅	
26 Jun 2024	1. CHG0062472 - APIM Infrastructure Updates[1593]	1. ✅ Normal	1. ✅	

1592 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0064624+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

1593 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0062472+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
16 Jul 2024	1. CHG0062288[1594] NHS e-RS APIM Referral Release R1.34 Deployment (Prod) 2. CHG0062289[1595] NHS e-RS APIM Patient Care Release R1.26 Deployment (Prod)	1.. 2..	1.. 2..	
19 Jun 2024	1. CHG0062331 - APIM Infrastructure Updates (see page 1959)	1. ✅ Standard	1. ✅	

1594 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D6cb974bac33e0e100317da3c050131dc%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1595 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dcaf9743ec33e0e100317da3c0501310f%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
18 Jun 2024	1. CHG0062110 - APIM Infrastructure Updates (see page 1953) 2. CHG0062247 - Digital Onboarding Service (see page 1957) 3. CHG0059554[1596] NHS e-RS APIM Referral Release R1.33 Deployment (Prod) 4. CHG0059555[1597] NHS e-RS APIM Patient Care Release R1.25 Deployment (Prod) 5. CHG0062306 - Identity Service Updates (see page 1958)	1. ✓ Standard 2. ✓ Standard 3. ✓ 4. ✓ 5. ✓ Standard	1. ✓ 2. ✓ 3. ✓ 4. ✓ 5. ✓	
12 Jun 2024	1. CHG0062021 - Identity Service Updates (see page 1956)	1. ✓ Standard		

1596 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D608a9946c3da42906c39935c05013189%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D16%26sysparm_record_list%3Dbusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255EstateNOT%2BIN3%252C4%255EORDERBYDESCstart_date

1597 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D1d3665c6c35682906c39935c05013196%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D16%26sysparm_record_list%3Dbusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255Ebusiness_serviceINb3f5f2381bd6c590452bdce7b04bcb80%252C070a7abc1bd6c590452bdce7b04bcbbf%252C530ababc1bd6c590452bdce7b04bcb4b%255EstateNOT%2BIN3%252C4%255EORDERBYDESCstart_date

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
10 Jun 2024	1. CHG0062110 - APIM Infrastructure Updates (see page 1953)	1. ✅ Standard	1. ✅	
06 Jun 2024	1. CHG0061730 - APIM Infrastructure Updates (see page 1951) 2. CHG0061790 - APIM Infrastructure Updates (see page 1952)	1. ✅ Standard 2. ✅ Standard	1. ✅ 2. ✅	
22 May 2024	1. CHG0060881 - APIM Infrastructure Updates (see page 1950)	1. ✅	1. ✅	1. Non-standard change
21 May 2024	1. CHG0060857 - APIM Infrastructure Updates (see page 1949)	1. ✅	1. ✅	1. Non-standard change
15 May 2024	1. CHG0060761 - APIM Infrastructure Updates (see page 1948)	1. ✅	1. ✅	
14 May 2024	1. CHG0059554[1598] NHS e-RS APIM Referral Release R1.32 Deployment (Prod) 2. CHG0059555[1599] NHS e-RS APIM Patient Care Release R1.24 Deployment (Prod) 3. CHG0060675 - APIM Infrastructure Updates (see page 1946)	1. ✅ 2. ✅ 3. ✅ Standard	1. . 2. . 3. ✅	

1598 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D256f0ceec325c6106c39935c05013110%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1599 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D88df8026c365c6106c39935c050131e9%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
13 May 2024	1. CHG0060650 - APIM Infrastructure Updates (see page 1945)	1. ✅	1. ✅	
08 May 2024	1. CHG0060407 - APIM Infrastructure Updates[1600]	1. ✅	1. ✅	1. Non-standard change
07 May 2024	1. CHG0060198 - APIM Infrastructure Updates (see page 1942) 2. CHG0060298 - APIM Infrastructure Updates (see page 1943)	1. ✅ 2. ✅	1. ✅ 2. ✅	1. Non-standard change - release window 9-12 2. Non-standard change - release window 2pm-5pm
26 Apr 2024	1. CHG0060017 - APIM Infrastructure Updates[1601]	1. ✅ Standard	1. ✅	
24 Apr 2024	1. CHG0059853- Digital Onboarding Service (see page 1939) 2. CHG0059908 - APIM Infrastructure Updates (see page 1940)	1. ✅ Standard 2. ✅ Standard	1. ✅ 2. ✅	
17 Apr 2024	1. CHG0059608 - APIM Infrastructure Updates (see page 1938)	1. ✅	1. ✅	

1600 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0060407+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

1601 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0060017+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
16 Apr 2024	1. CHG0058488[1602] NHS e-RS APIM Referral Release R1.31 Deployment (Prod) 2. CHG0058489[1603] NHS e-RS APIM Patient Care Release R1.23 Deployment (Prod) 3. CHG0059331[1604] APIM EPS FHIR Façade Updates v1.0.1891[1605] CHG0059333[1606] APIM PfP 4. s FHIR API Apigee Update v1.0.202-beta[1607]	1. ✓ 2. ✓ 3. ✓ 4. ✓	1. ✓ 2. ✓ 3. ✓ 4. ✓	
15 Apr 2024	1. CHG0059463 - APIM Infrastructure Updates (see page 1937)	1. ✓ Standard	1. ✓	
11 Apr 2024	1. CHG0059319 - Digital Onboarding Content release for DAQ 14.4 Production (see page 1936)	1. ✓ Standard		

1602 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D196bb361c30d0a506c39935c050131fd%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D13%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1603 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dab9bbf61c30d0a506c39935c0501311f%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D13%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1604 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=002035a4c3a1c6980317da3c050131a0

1605 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1891+-+Deployed+to+%5BINT%5D+on+10-04-24

1606 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=dae0f9e8c3a1c6980317da3c050131de

1607 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.202-beta+-+Deployed+to+%5BINT%5D+on+10-04-24

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
09 Apr 2024	1. CHG0059094[1608] APIM EPS FHIR Façade Updates v1.0.1868[1609] 2. CHG0059096[1610] s v1.0.201-APIM PfP FHIR API Apigee Update beta[1611]	1. ✅ 2. ✅	1. ✅ 2. ✅	
08 Apr 2024	1. CHG0059059 - APIM Infrastructure Updates (see page 1933)	1. ✅ Standard	1. ✅	
08 Apr 2024	1.CHG0059171 - APIM Infrastructure Updates (see page 1934)	1. ✅ Standard	1. ✅	
04 Apr 2024	1. CHG0058988 -[1612]Digital Onboarding Service (see page 1924)	1. ✅ Standard	1. ✅	
02 Apr 2024	1. CHG0058794 - APIM Infrastructure Updates[1613]	1. ✅	1. ✅	
27 Mar 2024	1. CHG0058838 - APIM Infrastructure Updates (see page 1931)	1. ✅ Standard	1. ✅	

1608 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=02c1d2aac35d4ed40317da3c050131f6
1609 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1869+-+Deployed+to+%5BINT%5D+on+04-04-24?src=contextnavpagetreemode
1610 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=52529226c39d4ed40317da3c050131ea
1611 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.201-beta+-+Deployed+to+%5BINT%5D+on+04-04-24?src=contextnavpagetreemode
1612 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D35591c02c3d10a946c39935c05013106%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull
1613 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0058794+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
26 Mar 2024	1. CHG0058566[1614] - APIM EPS FHIR Façade Updates v1.0.1842[1615] 2. CHG0058568[1616] - APIM PfP FHIR API Apigee Updates v1.0.198-beta[1617] 3. CHG0058779 - APIM Infrastructure Updates (see page 1929)	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ✅	
25 Mar 2024	1. CHG0058728 - APIM Infrastructure Updates[1618]	1. ✅ Standard	1. ✅	
25 Mar 2024	1. CHG0058712 - APIM Infrastructure Updates (see page 1926)	1. ✅ Standard	1. ✅	
22 Mar 2024	1. CHG0058624 - APIM Infrastructure Updates (see page 1925)	1. ✅ Standard	1. ✅	
21 Mar 2024	1. CHG0058536 - Digital Onboarding Service (see page 1924)	1. ✅ Standard	1. ✅	

1614 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=82a9a18ac30982906c39935c050131c3
1615 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1842+-+Deployed+to+%5BINT%5D+on+21-03-24
1616 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=88dae58ac34982906c39935c0501310f
1617 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.198-beta+-+Deployed+to+%5BINT%5D+on+21-03-24
1618 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0058728+-+APIM+Infrastructure+Updates?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
19 Mar 2024	1. CHG0057234[1619] NHS e-RS APIM Referral Release R1.30 Deployment (Prod) 2. CHG0057233[1620] NHS e-RS APIM Patient Care Release R1.22 Deployment (Prod) 3. CHG0058230[1621] APIM EPS FHIR Façade Updates v1.0.1813[1622] 4. CHG0058407 - APIM Infrastructure Updates (see page 1923)	1. ✅ 2. ✅ 3. ✅ 4. ✅ Standard	1. ✅ 2. ✅ 3. ✅ 4. ✅	
14 Mar 2024	1. CHG0058247 - APIM Infrastructure Updates - API Management - CDT Confluence (digital.nhs.uk) (see page 1922)	1. ✅ Standard		
13 Mar 2024	1. CHG0058130 - APIM Infrastructure Updates (see page 1920) 2. CHG0058183 - APIM Infrastructure Updates (see page 1921)	1. ✅ Standard 2. ✅ Standard	1. ✅ 2. ✅	
12 Mar 2024	1. CHG0057945[1623] APIM EPS FHIR Façade Updates v1.0.1794[1624]	1. ✅	1. ❌ (Release cancelled due to blocked pipeline)	

1619 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Ddd2f960183a806d000adf388beaad37a%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull
1620 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D70ee9a45836806d000adf388beaad3cd%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull
1621 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=527ca9bffb3c8210cfc6feceaeefdc20
1622 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1813+-+Deployed+to+%5BINT%5D+on+14-03-24
1623 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=4167ae2583f84e5800adf388beaad38d
1624 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1794+-+Deployed+to+%5BINT%5D+on+07-03-24

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
11 Mar 2024	1. CHG0057999 - APIM Infrastructure Updates (see page 1918) 2. CHG0058074 - APIM Infrastructure Updates (see page 1919)	1. ✅ Standard 2. ✅ Standard	1. 2. ✅	
07 Mar 2024	1. CHG0057788 (see page 1915) Digital Onboarding Service Code Release 2. CHG0057875 - APIM Infrastructure Updates (see page 1917)	1. Standard 2. ✅ Standard	1. ✅	
06 Mar 2024	1. CHG0057834 - APIM Infrastructure Updates (see page 1916)	1. ✅ Standard	1. ✅	
05 Mar 2024	1. CHG0057584[1625] APIM EPS FHIR Façade Updates v1.0.1782[1626] CHG0057585[1627] APIM PfP 2. s FHIR API Apigee Update v1.0.192-beta[1628] 3. CHG0057776 - APIM Infrastructure Updates (see page 1913)	1. ✅ 2. ✅	1. ✅ 2. ✅	
04 Mar 2024	1. CHG0057636 - APIM Infrastructure Updates (see page 1912)	1. Standard	1.	

1625 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ed15621783a8c69400adf388beaad362
1626 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1782+-+Deployed+to+%5BINT%5D+on+29-02-24
1627 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6685ae9b83a8c69400adf388beaad32d
1628 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.192-beta+-+Deployed+to+%5BINT%5D+on+29-02-24

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
29 Feb 2024	1. CHG0057559 - APIM Infrastructure Updates (see page 1911)	1. ✅ Standard	1.	
28 Feb 2024	1. CHG0057534 - APIM Infrastructure Updates (see page 1910)	1. ✅ Standard	1. ✅	
27 Feb 2024	1. CHG0057227[1629] - APIM EPS FHIR Façade Updates v1.0.1763[1630] CHG0057234 - APIM Infrastructure Updates (see page 1910) CHG0057228[1631] - APIM PfP FHIR API Apigee Updates v1.0.191-beta[1632] 3.	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ✅	
26 Feb 2024	1. CHG0057534 - APIM Infrastructure Updates (see page 1910)	1. ✅ Standard	1. ✅	
22 Feb 2024	1. CHG0057204[1633] - Digital Onboarding Content release for DAQ 14.2 Production	1. ✅ Standard	1. ✅	
21 Feb 2024	1. CHG0057120 (see page 1909) - Digital Onboarding Content release for DAQ 14.1 Production	1. ✅ Standard	1. ✅	

1629 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=db7856cd83a006d000adf388beaad326

1630 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1763+-+Deployed+to+%5BINT%5D+on+22-02-24

1631 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=22e85ac183e006d000adf388beaad325

1632 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.191-beta+-+Deployed+to+%5BINT%5D+on+22-02-24

1633 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D9e299589c3ec8e5442453fcb050131a6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
20 Feb 2024	1. CHG0055622[1634] NHS e-RS APIM Referral Release R1.29 Deployment (Prod) 2. CHG0055621[1635] NHS e-RS APIM Patient Care Release R1.21 Deployment (Prod) 3. CHG0056900[1636] s v1.0.187-APIM PfP FHIR API Apigee Update beta[1637]	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	
19 Feb 2024	1. CHG0056988- APIM Infrastructure Updates (see page 1908)	1. ✅ Standard	1. Pending	
15 Feb 2024	1. CHG0056746 - Digital Onboarding Service (see page 1907)	1. ✅	1. ✅	
13 Feb 2024	1. CHG0056553[1638] - APIM EPS FHIR Façade Updates v1.0.1724-beta[1639] 2. CHG0056554[1640] - APIM PfP FHIR API Apigee Updates v.1.186-beta[1641]	1. ✅ 2. ✅	1. ✅ 2. ✅	

1634 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dba70ff9fc3b3f51842453fcb05013183%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1635 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D75403797c3b3f51842453fcb050131da%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1636 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=31f32ef68310c61000adf388beaad34a

1637 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.187-beta+-+Deployed+to+%5BINT%5D+on+15-02-24

1638 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=0bea8e28c310421042453fcb0501310f

1639 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1724-beta+-+Deployed+to+%5BINT%5D+on+08-02-24

1640 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b89bc2e0c350421042453fcb0501316c

1641 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.186-beta+-+Deployed+to+%5BINT%5D+on+08-02-24

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
08 Feb 2024	1. CHG0056517[1642] Digital Onboarding / Developer Identity Keycloak configuration	1. ✅	1. ✅	
06 Feb 2024	1. CHG0056097[1643] - APIM EPS FHIR Façade Updates v1.0.1706-beta[1644] 2. CHG0056098[1645] - APIM PfP FHIR API Apigee Updates v1.0.185-beta[1646]	1. ✅ 2. ✅	1. ✅ 2. ✅	
01 Feb 2024	1. CHG0056020 - Digital Onboarding Service (see page 1906)	1. ✅ Standard	1. ✅	
31 Jan 2024	1. CHG0055691[1647] - APIM EPS FHIR Façade Updates v1.0.1672-beta[1648] 2. CHG0055699[1649] - APIM PfP FHIR API Apigee Updates v1.0.179-beta[1650]	1. ✅ 2. ✅	1. ✅ 2. ✅	
29 Jan 2024	1. CHG0055831- APIM Infrastructure Updates (see page 1905)	1. ✅ Standard	1. ✅	

1642 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db05e85a4839c865c00adf388beaad377%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull
1643 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=0569ce928300ca9800adf388beaad321
1644 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1706-beta+-+Deployed+to+%5BINT%5D+on+01-02-24
1645 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=abaa86d28340ca9800adf388beaad3bf
1646 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.185-beta+-+Deployed+to+%5BINT%5D+on+01-02-24
1647 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=5459b073c37b315842453fcb05013143
1648 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1672-beta+-+Deployed+to+%5BINT%5D+on+24-01-24
1649 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=986341bfc373715842453fcb050131d5
1650 https://nhsd-confluence.digital.nhs.uk/display/APIMC/PfP-APigee-v1.0.179-beta+-+Deployed+to+%5BINT%5D+on+24-01-24

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
24 Jan 2024	1. CHG0055559 - Digital Onboarding Service (see page 1904)	1. ✅ Standard	1. ✅	
23 Jan 2024	1. CHG0054270[1651] NHS e-RS APIM Referral Release R1.28 Deployment (Prod) 2. CHG0054268[1652] NHS e-RS APIM Patient Care Release R1.20 Deployment (Prod) 3. CHG0055379[1653] APIM EPS FHIR Façade Updates v1.0.1651-beta CHG0055382[1654] APIM PfP 4. FHIR API Apigee Updates v1.0.177-beta	1. ✅ 2. ✅ 3. ✅ 4. ✅	1. ✅ 2. ✅ 3. ✅ 4. ✅	
18 Jan 2024	1. CHG0055353 - Digital Onboarding Content release for DAQ 14.0 Production (see page 1903)	1. ✅ Standard	1. ✅	
17 Jan 2024	1. CHG0055343 - APIM Infrastructure Updates (see page 1901)	1. ✅ Standard	1. ✅	

1651 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8a79b1e2c383715042453fcb050131c3%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D17%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1652 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D83e8312ec343715042453fcb050131fe%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D17%26sysparm_record_list%3Dactive%253Dtrue%255Ebusiness_service%253Db3f5f2381bd6c590452bdce7b04bcb80%255EORDERBYDESCstart_date

1653 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=d2779d3dc333759442453fcb050131e5

1654 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b19851f9c373759442453fcb0501316e

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
16 Jan 2024	1. CHG0055051[1655] - APIM EPS FHIR Façade Updates v1.0.1622-beta[1656]	1. ✅	1. ✅	
11 Jan 2024	1. CHG0054748 APIM EPS FHIR Façade Updates v1.0.1603-beta[1657] 2. CHG0054750 APIM PfP FHIR API Apigee Updates v1.0.172-beta[1658] 3. CHG0055041 - Digital Onboarding Service (see page 1900)	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ✅	
09 Jan 2024	1. CHG0054951 - APIM Infrastructure Updates (see page 1899)	1. ✅ Standard	1. ✅	
08 Jan 2024	1. CHG0054686 - APIM Infrastructure Updates (see page 1897)	1. ✅	1. ✅	

1655 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ef36019fc32bbd9042453fcb05013102
1656 https://nhsd-confluence.digital.nhs.uk/display/APIMC/FHIR-v1.0.1622-beta+-+Deployed+to+%5BINT%5D+on+10-01-24
1657 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=5c59d625c3af791042453fcb050131a1
1658 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=5ffc5669c363b91042453fcb0501312f

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
04 Jan 2024	1. CHG0054263[1659] APIM EPS FHIR Façade Updates v1.0.1538-beta 2. CHG0054264[1660] APIM PfP FHIR API Apigee Updates v1.0.163-beta 3. CHG0054703 - APIM Infrastructure Updates (see page 1898)	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ✅	
22 Dec 2023 - 03 Jan 2023	CHANGE FREEZE - NO PRODUCTION RELEASES			
20 Dec 2023	1. CHG0054514 - APIM Infrastructure Updates (see page 1896)	1. ✅ Standard	1. ✅	
19 Dec 2023	1. CHG0054438 - APIM Infrastructure Updates (see page 1895)	1. ✅ Standard	1. ✅	

1659 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=172375aec30f315042453fcb05013159
1660 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6304356ec34f315042453fcb05013193

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Dec 2023	1. CHG0052637[1661] NHS e-RS APIM Referral Release R1.27 Deployment (Prod) 2. CHG0052632[1662] NHS e-RS APIM Patient Care Release R1.19 Deployment (Prod) 3. CHG0054160 - Digital Onboarding Service (see page 1893) 4. CHG0054271 - APIM Infrastructure Updates (see page 1894)	1. ✓ 2. ✓ 3. ✓ Standard 4. ✓ Standard	1. ✓ 2. ✓ 3. ✓ 4. ✓	
12 Dec 2023	1. CHG0054065 - APIM Infrastructure Updates (see page 1892)	1. ✓ Standard	1. ✓	
05 Dec 2023	1. CHG0053589[1663] APIM EPS FHIR Façade Updates v1.0.1480-beta 2. CHG0053590[1664] APIM PfP FHIR API Apigee Updates v1.0.158-beta 3. CHG0053705 - Identity Service Updates (see page 1890) 4. CHG0053719 - APIM Infrastructure Updates (see page 1891)	1. ✓ 2. ✓ 3. ✓ Standard 4. ✓ Standard	1. ✓ 2. ✓ 3. ✓ 4. ✓	
04 Dec 2023	1. CHG0053638 - Digital Onboarding Service (see page 1889)	1. ✓ Standard	1. ✓	

1661 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6d939ef8c3e2391042453fcb05013182
1662 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ee02d27cc362391042453fcb0501318b
1663 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=1000b5d6833635d400adf388beaad39d
1664 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=52a07d16837635d400adf388beaad3a9

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
29 Nov 2023	1. CHG0053407 - Digital Onboarding Service (see page 1888) (UAT only)	1. ✅ Standard	1. ✅	
28 Nov 2023	1. CHG0053201 - Digital Onboarding Service (see page 1881) 2. CHG0053287[1665] - APIM EPS FHIR Façade Updates v1.0.1475-beta 3. CHG0053288[1666]- APIM PfP FHIR API Apigee Updates v1.0.152-beta 4. CHG0053370 - APIM Infrastructure Updates (see page 1886) 5. CHG0053374 - APIM Infrastructure Updates (see page 1887)	1. ✅ Standard 2. ✅ 3. ✅ 4. ✅ Standard 5. ✅ Standard	1. ✅ 2. ✅ 3. ✅ 4. ✅ 5. ✅	
27 Nov 2023	1. CHG0053252 - APIM Infrastructure Updates (see page 1883)	1. ✅ Standard	1. ✅	Validate proxygen-cli secrets functionality works as expected in ptl before deploying to prod
23 Nov 2023	1. CHG0051804 - Booking and Referrals (see page 1845)	1. ✅	1. ✅	
22 Nov 2023	1. CHG0051994 - Digital Onboarding Service[1667]	1. ✅	1. ✅	

1665 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f71b01c1c3ba719442453fcb05013129
1666 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=99acc58dc3fa719442453fcb05013123
1667 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=10cd9b94c396bd1042453fcb050131d3

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
21 Nov 2023	1. CHG0052918 (see page 1879)APIM EPS FHIR Façade Updates v1.0.1467-beta 2. CHG0052605 (see page 1880) APIM PfP FHIR API Apigee Updates v1.0.150-beta	1. ✅ 2. ✅	1. ✅ 2. ✅	
20 Nov 2023	1. CHG0052735[1668] NHS e-RS APIM Referral Release R1.26 Deployment (Prod) 2. CHG0052734[1669] NHS e-RS APIM Patient Care Release R1.18 Deployment (Prod)	1. . 2. .	1. . 2. .	
17 Nov 2023	1. CHG0052866 - APIM Infrastructure Updates (see page 1878)	1. ✅ Standard	1. ✅	
16 Nov 2023	1. CHG0052687 - Digital Onboarding Content release (see page 1877)	1. ✅ Standard		
16 Nov 2023	1. CHG0052407 (see page 1873) APIM EPS FHIR Façade Updates v1.0.1454-beta 2. CHG0052605 (see page 1874) APIM PfP FHIR API Apigee Updates v1.0.148-beta	1. ✅ 2. ✅	1. ✅ 2. ✅	

1668 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=52ca3385c366f15042453fcb050131ca

1669 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=560a7b09c326f15042453fcb050131de

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Nov 2023	1. CHG0051389[1670] NHS e-RS APIM Referral Release R1.26 Deployment (Prod) 2. CHG0051390[1671] NHS e-RS APIM Patient Care Release R1.18 Deployment (Prod) 3. CHG0052689 - APIM Infrastructure Updates (see page 1875)	1. Cancelled due to An APIM/ Apigee (APM-4806 / Apigee case 1532790) issue 2. Cancelled due to An APIM/ Apigee (APM-4806 / Apigee case 1532790) issue 3. ✅ Standard	1. ❌ 2. ❌ 3. ✅	

1670 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3079b6484706bd14b93f4579516d43d6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

1671 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd98932884706bd14b93f4579516d43af

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
08 Nov 2023	1. CHG0052295 - Digital Onboarding Service (see page 1872) 2. CHG0052282 - Digital Onboarding Content release (see page 1871)	1. ✅ 2. ✅	1. ✅ 2.	
02 Nov 2023	1. APIM EPS FHIR Façade Updates v1.0.1443-beta (see page 1869)	1. ✅	1. ✅	
30 Oct 2023	1. CHG0051849 - APIM Infrastructure Updates (see page 1868)	1. ✅ standard	1. ✅	
26 Oct 2023	1. CHG0051704 - Digital Onboarding Service (see page 1867)	1. ✅ Standard	1. ✅	
26 Oct 2023	1. CHG0051205 - APIM EPS FHIR Façade Updates (see page 1864)	1. ✅	1. ✅	
18 Oct 2023	1. CHG0051432 - APIM Infrastructure Updates (see page 1865)	1. ✅ Standard	1. ✅	
17 Oct 2023	1. CHG0049934[1672] NHS e-RS APIM Referral Release R1.25 Deployment (Prod) 2. CHG0049935[1673] NHS e-RS APIM Patient Care Release R1.17 Deployment (Prod)	1. ✅ 2. ✅	1. ✅ 2. ✅	

1672 https://nhsdigitallive.service-now.com/change_request.do?sys_id=06a77d2a831d31d000adf388beaad341&sysparm_record_target=change_request&sysparm_record_row=4&sysparm_record_rows=8&sysparm_record_list=stateNOT+IN3%2C4%5Eassignment_groupDYNAMICd6435e965f510100a9ad2572f2b47744%5EORDERBYnumber

1673 https://nhsdigitallive.service-now.com/change_request.do?sys_id=c8f8712a835d31d000adf388beaad384&sysparm_record_target=change_request&sysparm_record_row=5&sysparm_record_rows=8&sysparm_record_list=stateNOT+IN3%2C4%5Eassignment_groupDYNAMICd6435e965f510100a9ad2572f2b47744%5EORDERBYnumber

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
13 Oct 2023	1. CHG0051201 - APIM Infrastructure Updates (see page 1863)	1. ✅ Standard	1. ✅	
12 Oct 2023	1. CHG0051070 - Digital Onboarding Service (see page 1862)	1. ✅ Standard	1. ✅	
10 Oct 2023	1. CHG0050994 - APIM Infrastructure Updates (see page 1861)	1. ✅ Standard	1. Failed - rolled back successfully	
03 Oct 2023	1. CHG0050564 - Digital Onboarding Content release (see page 1858) 2. CHG0050653 - APIM Infrastructure Updates (see page 1860)	1. ✅ Standard 2. ✅ Standard	1. ✅ 2. ✅	
02 Oct 2023	1. CHG0050342 - APIM Infrastructure Updates (see page 1856) 2. CHG0050574 - APIM Infrastructure Updates (see page 1859)	1. ✅ 2. ✅ Standard	1. ✅ 2. ✅	
28 Sep 2023	1. CHG0050379 - Digital Onboarding Service (see page 1857)	1. ✅ Standard	1. ✅	
26 Sep 2023	1. CHG0050127 - APIM - Add rate limiting custom attribute to app (see page 1854)	1. ✅	1. ✅	
21 Sep 2023	1. CHG0050060 - Digital Onboarding Service (see page 1853)	1. ✅ Standard	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
20 Sep 2023	1. CHG0050015 - Digital Onboarding Content release (see page 1852)	1. ✅ Standard	1. ✅	
18 Sep 2023	1. CHG0049916 - APIM Infrastructure Updates (see page 1851)	1. ✅ Standard	1. ✅	
15 Sep 2023	1. CHG0049829 - Modify rate limit for PDS FHIR proxy[1674]	1. ✅ Approved	1. ✅	
14 Sep 2023	1. CHG0049711 - Digital Onboarding Service (see page 1850)	1. ✅ Standard		
19 Sep 2023	1. CHG0048743 - NHS e-RS APIM Patient Care Release R1.16 Production Deployment (see page 1847) 2. CHG0048741 - NHS e-RS APIM Referral Release R1.24 Deployment (Prod) (see page 1848)	1. . 2. .	1. . 2. .	
14 Sep 2023	1. CHG0049159 - Booking and Referrals (see page 1844)	1. ✅ Standard	1. ✅	
13 Sep 2023	1. CHG0049615- Identity Service Changes (see page 1849)	1. ✅ Standard	1. .	
05 Sep 2023	1. CHG0049183 - APIM EPS Updates (see page 1842) 2. CHG0049315 - APIM Infrastructure Updates (see page 1846)	1. ✅ 2. ✅	1. ✅ 2. ✅	

1674 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=5cb0955dc315fd5042453fcb050131e9

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
04 Sep 2023	1. CHG0049251 - Digital Onboarding Content release (see page 1843)	1. ✅ Standard	1. ✅	
29 Aug 2023	1. CHG0049012 - APIM Infrastructure Updates (see page 1840)	1. ✅ Standard	1. ✅	
24 Aug 2023	1. CHG0048869 - Digital Onboarding Service (see page 1839)	1. ✅ Standard	1. ✅	
22 Aug 2023	1. CHG0047461 - NHS e-RS APIM Patient Care Release R1.15 Production Deployment (see page 1826) 2. CHG0047463 - NHS e-RS APIM Referral Release R1.23 Deployment (Prod) (see page 1827) 3. CHG0048714 - APIM Infrastructure Updates (see page 1838)	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ✅	
16th Aug 2023	1. CHG0048511 - Digital Onboarding Content release (see page 1837)	1. ✅ Standard	1. ✅	
15 Aug 2023	1. CHG0048455 - Digital Onboarding Content release (see page 1836)	1. ✅ Standard	1. ✅	
14 Aug 2023	1. CHG0048360- Digital Onboarding Content release (see page 1835)	1. ✅ Standard	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Aug 2023	1. CHG0048359 - APIM Infrastructure Updates (see page 1833) 2. CHG0048398 - APIM Infrastructure Updates (see page 1834)	1. Standard 2. Standard	1. ✓ 2. ✓	
10 Aug 2023	1. CHG0048266- Digital Onboarding Content release (see page 1832)	1. Standard	1. ✓	
09 Aug 2023	1. CHG0048062- Digital Onboarding Content release (see page 1830)	1. Standard	1. ✓	
08 Aug 2023	1. CHG0048068 - Digital Onboarding Service (see page 1829) 2. CHG0048116 - APIM Infrastructure Updates (see page 1831)	1. Standard 2. Standard	1. ✓ 2. ✓	
02 Aug 2023	1. CHG0047777 - Digital Onboarding Service (see page 1828) 2. CHG0047578 - Digital Onboarding Content release for DAQ 11.6 Production (see page 1824)	1. Standard 2. ✓	1. ✓ 2. ✗	
31 Jul 2023	1. CHG0047662 - Digital Onboarding Service (see page 1825)	1. Standard	1. ✓	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
25 Jul 2023	1. CHG0046042 - NHS e-RS APIM Patient Care Release R1.14 Production Deployment (see page 1817) 2. CHG0046041 NHS e-RS APIM Referral Release R1.22 Deployment (Prod) (see page 1819) 3. CHG0047419 - APIM Infrastructure Updates (see page 1823)	1. ✅ 2. ✅ 3. ✅ Standard	1. ✅ 2. ✅ 3. ❌	
24 Jul 2023	1. CHG0047223 - Digital Onboarding Service (see page 1822)	1. ✅ Standard	1. ✅	
13 Jul 2023	1. CHG0046863 - APIM EPS Updates (see page 1821)	1. ✅	1. ✅	
11 Jul 2023	1. CHG0046713 - Digital Onboarding Service (see page 1816)	1. ✅ Standard	1. ✅	
10 Jul 2023	1. CHG0046633 - APIM EPS Updates (see page 1820)	1. ✅	1. ✅	
03 Jul 2023	1. CHG0046325 - APIM Infrastructure Updates (see page 1814)	1. ✅ Standard	1. ✅	
30 Jun 2023	1. CHG0046184 - APIM Infrastructure Updates (see page 1813)	1. ✅ Standard	1. ✅	
28 Jun 2023	1. CHG0046077 - APIM EPS Updates (see page 1812)	1. ✅	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
27 Jun 2023	1. CHG0045027 - NHS e-RS APIM Patient Care Release R1.13 Production Deployment (see page 1799) 2. CHG0045025 - NHS e-RS APIM Referral Release R1.21 Production Deployment (see page 1800)	1. ✓ 2. ✓	1. ✓ 2. ✓	
23 Jun 2023	1. CHG0045793 - APIM EPS Updates (see page 1811)	1. ✓	1. ✓	
22 Jun 2023	1. CHG0045726 - Digital Onboarding Service (see page 1809)	1. ✓	1. ✓	
21 Jun 2023	1. CHG0045708 - APIM Infrastructure Updates (see page 1808)	1. ✓ Standard		1. Apigee only
20 Jun 2023	1. CHG0039918[1675] - APIM - NHS App Updates for Comms Hub: 2. CHG0045620- APIM EPS Update (see page 1807) 3. CHG0045708 - APIM Infrastructure Updates (see page 1808)	1. ✓ 2. Pending 3. ✓ Standard	1. ✓ 2. Pending 3. ✓	1. Self service release by NHS App N&M team
15 Jun 2023	1. CHG0045519 - APIM EPS Updates (see page 1806)	1. ✓	1. ✓	
15 Jun 2023	1. CHG0045108 - Digital Onboarding Service (see page 1803)	1. ✓	1. ✓	

1675 https://nhsdigitallive.service-now.com/change_request.do?sys_id=671528901b382d9093bfc88b274bcb94&sysparm_record_target=change_request&sysparm_record_row=52&sysparm_record_rows=365&sysparm_record_list=active%3Dtrue%5EORDERBYDESCstart_date

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
13 Jun 2023	1. CHG0045315 - APIM EPS Updates (see page 1802)	1. ✅	1. ✅	
13 Jun 2023	1. CHG0045293 - APIM Infrastructure Keycloak Upgrade Deployment	1. ✅	1. ✅	
07 Jun 2023 - 13 Jun 2023	NO PRODUCTION INFRASTRUCTURE RELEASES			
06 Jun 2023	1. CHG0044977 - APIM EPS Updates (see page 1798)	1. ✅	1. ✅	1. Self service release by EPS team
05 Jun 2023	1. CHG0045116 - APIM Infrastructure Updates (see page 1801)	1. ✅ Standard change	1. ✅	
01 Jun 2023	1. CHG0044062 - NHS e-RS APIM Patient Care Release R1.12 Production Deployment (see page 1790) 2. CHG0044059 - NHS e-RS APIM Referral Release R1.20 Production Deployment (see page 1791)	1. ✅ 2. ✅	1. ✅ 2. ✅	1. 2pm call for APIM to make the release, with e-RS people observing (Dial in details: Click here to join the meeting[1676]) 2. 2pm call for APIM to make the

1676 https://teams.microsoft.com/l/meetup-join/19%3ameeting_MjFkZGE5N2MtMTcxMS00YzY5LTgyNWMtY2VkZTc5NDI1Zjlk%40thread.v2/0?context=%7b%22Tid%22%3a%2250f6071f-bbfe-401a-8803-673748e629e2%22%2c%22Oid%22%3a%226e6f6805-998f-4327-b5d1-311f48eacf14%22%7d

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
				release, with e-RS people observing (Dial in details: Click here to join the meeting[1677])
25 May 2023	1. CHG0044790 - APIM Infrastructure Updates (see page 1797)	1. ✅ Standard change	1. ✅	
23 May 2023	1. CHG0044433 - Digital Onboarding Service (see page 1796)	1. ✅	1. ✅	1. Self service release by Onboarding team
17 May 2023	1. CHG0044202 - Digital Onboarding Content release for DAQ 11.5 Production (see page 1794)	1. ✅	1. ❌	Issues with release. Rolled back.

1677 https://teams.microsoft.com/l/meetup-join/19%3ameeting_MjFkZGE5N2MtMTcxMS00YzY5LTgyNWMtY2VkZTc5NDI1Zjlk%40thread.v2/0?context=%7b%22Tid%22%3a%2250f6071f-bbfe-401a-8803-673748e629e2%22%2c%22Oid%22%3a%226e6f6805-998f-4327-b5d1-311f48eacf14%22%7d

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
11 May 2023	1. CHG0043967 - Digital Onboarding Service (see page 1789) 2. CHG0044171 - APIM Infrastructure Updates (see page 1793)	1. ✅ 2. ✅ Standard change	1. ✅ 2. ✅	1. 2. Requires terraform state management as part of the process to avoid loss of data in S3; please read associated plan and perform state mgmt commands before applying terraform changes in pipeline
09 May 2023	1. CHG0042546 - NHS e-RS APIM Patient Care Release R1.11 Production Deployment (see page 1776) 2. CHG0042547 - NHS e-RS APIM Referral Release R1.19 Production Deployment (see page 1775)	1. ✅ 2. ✅	1. ✅ 2. ✅	1. 2.
04 May 2023	1. CHG0043908 - APIM Infrastructure Updates (see page 1787)	1. ✅ Standard change	1. ✅	
02 May 2023	1. CHG0043595 - Digital Onboarding Service (see page 1786)	1. ✅	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
27 Apr 2023	1. CHG0043151 - APIM EPS Updates (see page 1784) 2. CHG0043556 - APIM Infrastructure Updates (see page 1785)	1. ✅ 2. ✅ Standard change	1. ✅ 2. ✅	1. This will be a self service release by EPS team 2. No-change release to refresh/bring infra up to date
26 Apr 2023	1. CHG0043067 - Digital Onboarding Content release for DAQ 11.4 Production (see page 1783)	1. ✅	1. ✅	
20 Apr 2023	1. CHG0042923 - Digital Onboarding Service (see page 1782)	1. ✅	1. ✅	
17 Apr 2023	1. CHG0042721 - Digital Onboarding Content release for DAQ 11.3 Production[1678]	1. ✅	1. ✅	1. Self service release by the Digital Onboarding Service squads
13 Apr 2023	1. CHG0042712- Digital Onboarding Service (see page 1778)	1. ✅	1. ✅	

1678 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D9ec6bd831b4629d8d1817662164bcbfb%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
12 Apr 2023	1. CHG0042725 - APIM Infrastructure Updates (see page 1781)	1. ✅ Standard Change	1. ✅	1. To be deployed out of hours at 8pm 2. was deployed a day later than expected due to no approvals.
11 Apr 2023	1. CHG0042720 - APIM Infrastructure Updates (see page 1780)	1. ✅ Standard Change		Apigee only
06 Apr 2023	1. CHG0042663 - APIM Infrastructure Updates (see page 1777)	1. ✅ Standard Change	1. ❌	Apigee only Failed to a redundant resource that can't be cleanly deleted - platform not impacted
04 Apr 2023	1. CHG0042220 - Digital Onboarding Service (see page 1772)	1. ✅	1. ✅	Released by Alex H because Phil Mac on holiday
03 Apr 2023	1. CHG0042198 - Digital Onboarding Content release for DAQ 11.2 Production (see page 1770)	1. ✅	1. ✅	1. Self service release by the Digital Onboarding Service squads

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
30 Mar 2023	1. ~~CHG0042220 - Digital Onboarding Service (see page 1772)~~ 2. CHG0042299 - APIM Infrastructure updates (see page 1773)	1. ~~Pending~~ 2. ✅ Standard Change	1. ~~Pending~~ 2. Pending	1. Rescheduled as change not approved in time for release slot 2. 8pm release by ALex H
28 Mar 2023	1. CHG0041499 - NHS e-RS APIM Referral Release R1.18 Production Deployment (see page 1763) 2. CHG0041507 - NHS e-RS APIM Patient Care Release R1.10 Production Deployment (see page 1762) 3. CHG0042126 - APIM EPS Updates (see page 1769) 4. CHG0042217 - APIM Infrastructure updates (see page 1771)	1. ✅ 2. ✅ 3. Cancelled 4. ✅ Standard Change	1. ✅ 2. ✅ 3. Cancelled 4. Post-poned to thursday. Change cancelled so will raise new change	1. 2. 3. EPS Release has been cancelled due to issue found in testing 4. This will be done in the day by APIM
23 Mar 2023	1. CHG0042084 - APIM Infrastructure Updates (see page 1768)	1. ✅ Standard Change	1. ✅	
16 Mar 2023	1. Decommisioning of the Covid Medical Exemptions API	1. Pending	1. N/A	1. This is to remove the API proxy and decommision the API - placeholder in this schedule because it's not a release as such.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Mar 2023	1. CHG0041490 - APIM EPS Updates (see page 1760) 2. CHG0041524 - Digital Onboarding Service (see page 1764) 3. CHG0041622 - APIM Infrastructure Releases (see page 1767)	1. ✓ 2. ✓ 3. ✓ Standard Change	1. ✓ 2. ✓ 3. ✓	1. This will be a self service release by EPS team 2. This will be a self service release by the Digital Onboarding team 3. Releasing Apigee changes that could not be deployed y/day
13 Mar 2023	1. CHG0041597 - APIM Infrastructure Updates (see page 1766)	1. ✓ Standard Change	1. ✗	Apigee changes failed to deploy
09 Mar 2023	1. CHG0041498 - APIM Infrastructure Updates (see page 1761)	1. ✓ Standard Change	1. ✓	
07 Mar 2023	1. CHG0041319 - APIM Infrastructure updates (see page 1759)	1. ✓ Standard Change	1. ✓	
02 Mar 2023	1. CHG0041275 - APIM Infrastructure Updates (see page 1757)	1. ✓ Standard Change	1. ✓	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
01 Mar 2023	1. CHG0041198 - Identity Service Updates (see page 1756) 2. CHG0041210 - APIM Infrastructure Updates (see page 1757)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	
28 Feb 2023	1. CHG0040978 - APIM EPS Updates (see page 1754) 2. CHG0039377 - NHS e-RS APIM Referral Release R1.17 Production Deployment (see page 1740) 3. CHG0039378 - NHS e-RS APIM Patient Care Release R1.9 Production Deployment (see page 1741)	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	1. self service release by EPS team
23 Feb 2023	1. CHG0040876 - APIM Infrastructure Updates (see page 1753) 2. CHG0041013 - APIM Infrastructure Updates (see page 1755)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	
16 Feb 2023	1. CHG0040638 - Digital Onboarding Service (see page 1752)	1. ✅ Standard Change	1. ✅	
14 Feb 2023	1. CHG0040159 - APIM EPS Updates (see page 1748)	1. ✅	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
09 Feb 2023	1. CHG0040295 - Digital Onboarding Service (see page 1751)	1. Standard Change	1.	
08 Feb 2023	1. CHG0040263 - APIM Infrastructure Updates (see page 1750)	1. Standard Change	1.	
03 Feb 2023	1. CHG0040127 - APIM Infrastructure Updates (see page 1747)	1. Standard Change	1.	
02 Feb 2023	1. CHG0040020 - Digital Onboarding Service (see page 1745)	1. Standard Change	1.	
31 Jan 2023	1. CHG0039901 - APIM Infrastructure changes (see page 1744)	1. Standard Change	1.	
26 Jan 2023	CHG0039788 - APIM Infrastructure Changes (see page 1743)	1. Standard Change	1.	
19 Jan 2023	1. CHG0039474 - APIM Infrastructure Changes (see page 1738)	1. Standard Change	1.	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
17 Jan 2023	1. CHG0038100 - NHS e-RS APIM Referral Release R1.16 Production Deployment (see page 1731) 2. CHG0038099 - NHS e-RS APIM Patient Care Release R1.8 Production Deployment (see page 1729) 3. CHG0039266 - Digital Onboarding Service (see page 1739)	1. ✅ 2. ✅ 3. ✅ Standard Change	1. ✅ 2. ✅ 3. ✅	***Do not*** deploy to prod Apigee
12 Jan 2023				
10 Jan 2023	1. Infra release 2. CHG0038936 - Digital Onboarding Service (see page 1735) 3. CHG0038946 - Identity Service Updates (see page 1737)	1. N/A - covered by DOS release 2. Standard Change 3. Standard Change	1. N/A 2. ✅ 3. ✅	Placeholder for infra release
05 Jan 2023				
03 Jan 2023	Change Freeze			
29 Dec 2022	Change Freeze			
27 Dec 2022	Change Freeze			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
22 Dec 2022	Change Freeze			
20 Dec 2022	Change Freeze			
19 Dec 2022	1. CHG0038571: APIM Infrastructure Changes (see page 1735)	1. Standard change	1.	
15 Dec 2022	1. CHG0038358: Personal Demographics Service (see page 1733) 2. CHG0038470: APIM Infrastructure Changes (see page 1734)	1. Cancelled 2. Standard change	1.	Last change window before the change freeze
13 Dec 2022	1. CHG0038330: APIM Infrastructure Changes (see page 1732)	1. Standard change	1.	
08 Dec 2022				

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
06 Dec 2022	1. Infra release 2. CHG0036990 - NHS e-RS APIM Referral Release R1.15 Production Deployment (see page 1712) 3. CHG0036991 - NHS e-RS APIM Patient Care Release R1.7 Production Deployment (see page 1711) 4. CHG0037865 - APIM EPS Updates (see page 1727) 5. CHG0038027 - Digital Onboarding Service (see page 1728)	1. ⛔ 2. ✅ 3. ✅ 4. ✅ Normal change 5. ✅ Standard change	1. ⛔ (no release raised) 2. ✅ 3. ✅ 4. ✅ 5. ✅	1. Placeholder for infra release 2. 10-12 3. 10-12 4. In hours 5. 2-12
01 Dec 2022	1. CHG0037877 - APIM Identity Service Updates (see page 1725) 2. CHG0037892 - Digital Onboarding Service (see page 1726)	1. ✅ Standard Change 2. ✅ Standard Change	1. Pending 2. Pending	
29 Nov 2022	1. CHG0037410 - APIM EPS Updates (see page 1718) 2. CHG0037724 - Digital Onboarding Service (see page 1723) 3. CHG0037795: APIM Infrastructure Changes (see page 1724)	1. ✅ Normal change 2. ✅ Standard Change 3. ✅ Standard Change	1. ✅ 2. ✅ 3. ✅	
24 Nov 2022				

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
22 Nov 2022	1. CHG0037292 - Digital Onboarding Service (see page 1716) 2. CHG0037450: APIM Infrastructure Changes (see page 1721) 3. CHG0037404- APIM Identity Service Update (see page 1717)	1. ✅ Standard Change 2. ✅ Standard Change 3. ✅ Standard Change	1. ✅ 2. ✅ 3. ✅	
15 Nov 2022	1. CHG0037214: APIM Infrastructure Changes (see page 1715) - in hours 2. CHG0037210 - APIM Identity Service Updates (see page 1713)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	
10 Nov 2022	1. CHG0036796 - APIM Identity Service Updates (see page 1708)	1. ✅ Standard Change	1. Cancelled	Cancelled as needs a infra release. Re-rasied for 15 Nov 2022

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
08 Nov 2022	1. CHG0036171 - NHS e-RS APIM Referral Release R1.14 Production Deployment (see page 1705) 2. CHG0036172 - NHS e-RS APIM Patient Care Release R1.6 Production Deployment (see page 1706) 3. CHG0036917 - Digital Onboarding Service (see page 1710)	1. ✓ 2. ✓ 3. ✓ Standard Change	1. ✓ 2. ✓ 3. ✓	
27 Oct 2022	1. CHG0036522 - Digital Onboarding (see page 1707)	1. ✓ Standard Change	1. ✓	
20 Oct 2022	1. CHG0036192- Digital Onboarding Service (see page 1704)	1. ✓ Standard Change	1. ✓	
18 Oct 2022	1. CHG0036081 - Digital Onboarding Service (see page 1703)	1. ✓	1. ✓	
11 Oct 2022	1. CHG0035211 - NHS e-RS APIM Referral Release R1.13 Production Deployment (see page 1697) 2. CHG0035210 - NHS e-RS APIM Patient Care Release R1.5 Production Deployment (see page 1698) 3. CHG0035816 - Digital Onboarding Service (see page 1701)	1. ✓ 2. ✓ 3. ✓ Standard	1. ✓ 2. ✓ 3. ✓	Re: 1 and 2. Yes, we'd like your team to run these RFCs as we've still yet to agree with BJSS about them taking over these deployments.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
05 Oct 2022	1. CHG0035683: APIM Infrastructure Changes (see page 1700)	1. ✅ Standard Change	1. ✅	Hotfix for CSOC: **SIR0012362**
04 Oct 2022	1. CHG0035615 - Infrastructure Standard Change (see page 1699)	1. ✅ Standard Change	1. ✅	
30 Sep 2022	1. CHG0035191 - Electronic Signing Service - First of Type Readiness (see page 1689)	1. ✅	1. ✅	1. Self-release by Signing Service team. Deployment completed successfully.
27 Sep 2022	1. CHG0035334 - Digital Onboarding Service (see page 1695) 2. CHG0035412: APIM Infrastructure Changes (see page 1696)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	
22 Sep 2022	1. CHG0035187 - Digital Onboarding Service (see page 1694)	1. ✅ Standard Change	1. ✅	
20 Sep 2022	1. CHG0035002 - Digital Onboarding Service (see page 1688)	1. Pending		

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Sep 2022	1. CHG0033743 - GP Registration Management Information (see page 1687)	1. ✅		1. Technical go-live so done in business hours.
13 Sep 2022	1. CHG0034907: APIM Infrastructure Changes (see page 1686) 2. CHG0034039 - NHS e-RS APIM Referral Release R1.12.0 Production Deployment (see page 1683) 3. CHG0034041 - NHS e-RS APIM Patient Care Release R1.4.0 Production Deployment (see page 1685)	1. ✅ Standard Change 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	
08 Sep 2022	1. CHG0034608- Identity Service Changes - API Management - CDT Confluence (digital.nhs.uk) (see page 1682)	1. ✅		
06 Sep 2022	1. CHG0034532 - Initial deployment of Electronic Signing Service to production (see page 1677) 2. CHG0034551 - Digital Onboarding Service (see page 1671)	1. ✅ 2. ✅ Standard change		1. Completed successfully
01 Sep 2022	1. CHG0040981 - Initial deployment of Electronic Signing Service to production (see page 1673)	1. ✅ Normal change	1. ✅	Release process revealed some permissions errors so not run. Will re-schedule once resolved.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
30 Aug 2022	1. CHG 0034302 - Immunisation History API (see page 1670)	1. Normal change	1. ✓	
25 Aug 2022	1. CHG0033871 Sync-wrap API changes (see page 1663)	1. Normal change	1. ✓	
24 Aug 2022	1. CHG0034157 - Digital Onboarding Service (see page 1669)	1. Standard change	1. ✓	Ran a day late after missing the release window on Tuesday. (Had agreed to avoid overlapping an overnight process.)
23 Aug 2022	1. CHG0033762 - Apigee environment attributes[1679]	1. Normal change pending 2. Standard Change	1. ✓ 2. Pending	Agreed to run DOS release prior to 2300 so as not to clash with another overnight process. Could not meet this window due to Environment attributes change taking longer than expected.
18 Aug 2022	1. CHG0034015 - APIM Infrastructure Updates - Add patient care aggregator prod target server (see page 1668)	1. Standard Change	1. ✓	Agreed w/ Aubyn for in hours release

1679 https://nhsd-confluence.digital.nhs.uk/display/APM/CHG0033762+-+Apigee+data

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
16 Aug 2022	1. CHG0033220 - NHS e-RS APIM Referral Release R1.11.0 Production Deployment (see page 1664) 2. CHG0033321 - NHS e-RS APIM Patient Care Release R1.3.0 Production Deployment (see page 1666)	1. ✓ 2. ✓	1. ✓ 2. ✓	
09 Aug 2022	1. CHG0033763 - Digital Onboarding Service (see page 1662)	1. ✓ Standard Change	1. ✓	To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
02 Aug 2022	1. CHG0033524 - Digital Onboarding Service (see page 1661)	1. ✓ Standard Change	1. ✓	To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
28 Jul 2022				

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
26 Jul 2022	1. CHG0033306 - Digital Onboarding Service (see page 1660)	1. ✅ Standard Change	1. Pending	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
22 Jul 2022 Different from usual, APIM Referral and Patient Care between 14:00-17:00. There are no end users of the service yet and more support is available during work hours if something was to go wrong again.	1. CHG0033227 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment (see page 1656) 2. CHG0033228 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment (see page 1658)	1. ✅ 2. ✅	1. ✅ 2. ✅	NHSD e-RS Service Management will be in contact with Alex Carrie and Steve Walton about this given that there is no APIM release call on Friday 22 July. Any Warning and Error messages on the Azure DevOps pages can be safely ignored.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
19 Jul 2022 APIM Referral and Patient Care between 23:00-00:00, as usual.	1. RFC 0032744 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment (see page 1651) 2. RFC 0032733 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment (see page 1654) 3. CHG0033065 - APIM Infrastructure Updates (see page 1655)	1. ✅ 2. ✅ 3. ✅ Standard Change	1. ⛔ Failed, rescheduled for 22nd 2. ⛔ Not attempted, rescheduled for 22nd 3. ✅	(1) was attempted, failed and then the rollback failed. Because of this (2) was not attempted. The errors were caused by a thermal event at the Google London data centre which impacted the Apigee API host server. There were no faults with the code.
14 Jul 2022	1. CHG0032690 Immunisation History API (see page 1648)	1. ✅	1. ✅	
12 Jul 2022	1. CHG0032690 Immunisation History API (see page 1648) 2. CHG0032859 - APIM Infrastructure Updates (see page 1650)	1. Pending 2. ✅ Standard Change	1. ⛔ Cancelled as change is not approved 2. ✅	2. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
07 Jul 2022	1. CHG0032731: Digital Onboarding Service (see page 1649)	1. ✅ Standard Change	1. Pending	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
05 Jul 2022	1. CHG0032379: APIM Infrastructure Changes (see page 1647)	1. Standard Change	1. Pending	
30 Jun 2022	1. CHG0032484: Digital Onboarding Service (see page 1646)	1. Standard Change	1. Pending	
28 Jun 2022	1. CHG0032379: APIM Infrastructure Changes (see page 1645)	1. Standard Change	1. Pending	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
23 Jun 2022	1. RFC CHG0032105 - NHS App API change (see page 1642)	1. Standard Change	1.	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
21 Jun 2022	1. ~~CHG0032137 - Digital Onboarding Service (see page 1643)~~ **Cancelled see Release Status** 2. CHG0032180: APIM Infrastructure Changes (see page 1644)	1. ✅ Standard Change 2. ✅ Standard Change	1. **SUPERSEDED as release #2 contains the same change set plus additional changes requiring some manual updates to the terraform state file.** 2. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
16 Jun 2022	1. RFC 0031929 - NHS e-RS APIM Referral Release R1.9 Production Deployment (see page 1640) 2. RFC 0031930 - NHS e-RS APIM Patient Care Release R1.1.6 Production Deployment (see page 1639)	1. ✅ 2. ✅	1. Testing signed off. 2. Testing signed off.	Was originally for 14 June but BJSS advised to move this back due to some other work that came in late that has to be prioritised for 14 June, and which will involve an e-RS out of hours outage.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
14 Jun 2022	1. CHG0031941 - Digital Onboarding Service (see page 1641)	1. ✅ Standard Change	1. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
09 Jun 2022	1. RFC CHG0031779 - Digital Onboarding Service (see page 1637)	1. ✅ Standard Change	1. ✅	
07 Jun 2022	1. RFC CHG0031664 - Digital Onboarding Service (see page 1635) 2. RFC CHG0031705: APIM Infrastructure Changes (see page 1636)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	
26 May 2022	1. RFC CHG0031403: APIM Infrastructure Changes (see page 1634)	1. ✅ Standard Change	1. Pending	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
24 May 2022	1. RFC CHG0031039 - Digital Onboarding Service (see page 1632) 2. RFC CHG0031189: APIM Infrastructure Changes (see page 1633)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ⛔ Rolled back	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
17 May 2022	1. RFC 30032765 Covid Medical Exemptions (see page 1627) 2. RFC 30033031 - Digital Onboarding Service (see page 1631)	1. Pending 2. ✅ Standard Change	1. 2. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
12 May 2022	1. RFC 30032959 - Digital Onboarding Service (see page 1629)	1. ✅ Standard Change	1. Pending	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
10 May 2022	1. RFC 30032787 - NHS e-RS APIM Patient Care Release R1.0.5 Production Deployment (see page 1624)	1. ✅	1. Testing signed off, documentation ready. ✅	No outstanding issues. RFC can proceed.
05 May 2022	1. RFC 30032605 - NHS e-RS APIM Referral Release R1.8.1 Production Deployment (see page 1622) 2. RFC 30032835- Identity Service Changes - API Management - CDT Confluence (digital.nhs.uk) (see page 1628)	**NHS e-RS Programme**	**Areas Represented:**	
03 May 2022	1. RFC 30032821: APIM Infrastructure Changes (see page 1626) - RFC 30032815 - Digital Onboarding Service (see page 1625)	**NHSD EMT**	**NHS e-RS Programme**	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
28 Apr 2022	No releases	**NHSD Solutions Assurance (Functional)**	**NHSD EMT**	
26 Apr 2022	No releases	**NHSD Solutions Assurance (Non-Functional)**	**NHSD Solutions Assurance (Functional)**	Jon Williams
21 Apr 2022	No releases	**NHSD Service Management**	**NHSD Solutions Assurance (Non-Functional)**	Darren Wake
19 Apr 2022	1. RFC 30032569 - Digital Onboarding Service (see page 1621)	**NHSD Tech Ops**	**NHSD Service Management**	Mark Smith-Magee, Martin Ward
14 Apr 2022	No releases	**BJSS**	**NHSD Tech Ops**	Matt Holloway
12 Apr 2022	1. RFC 30032103 - NHS e-RS APIM Release R1.7 Production Deployment (see page 1616) 2. RFC 30032243: Digital Onboarding Service - My applications page (see page 1614) 3. RFC 30032474 - APIM Infrastructure Update (see page 1620)	1. Yes ✅ 2. Pending 3. ✅ Standard Change	**BJSS**	Kris Mouser-Brown, Dave Dawson, Beka Warriner

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
07 Apr 2022	1. RFC 30032406 - APIM Infrastructure Change (see page 1619)	1. ✅ Standard Change	1. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
05 Apr 2022	1. RFC 30032366 - APIM Infrastructure Change (see page 1618)	1. ✅ Standard Change	1. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.
31 Mar 2022	No releases			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
29 Mar 2022	1. RFC 30032221 APIM Infrastructure Change (DOS SSM Parameter) (see page 1617) 2. RFC 30032151 - NHS e-RS APIM Release R1.6.4 hotfix Production Deployment (see page 1612)	1. ✅ 2. ✅	1. ✅ 2. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this. 2. Testing signed off. RFC approved.
22 Mar 2022	1. RFC 30032113 - APIM infrastructure release (see page 1611)	1. ✅	1. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS) in the release plan, you will need permission to access the Firefly and Serenity board. Please contact Firefly and Serenity teams to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
17 Mar 2022	1. RFC 30032025: APIM Infrastructure Changes (see page 1608) 2. RFC 30031802 - NHS e-RS APIM Release R1.6 Production Deployment (see page 1610)	1. ✅ Standard Change 2. ✅	1. ✅ 2. ✅	1. To view Digital Onboarding Service JIRA tickets (DOS and ADZ) in the release plan, you will need permission to access the Turnstile and Tardis boards. Please contact Turnstile and Tardis teams to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
15 Mar 2022	1. RFC 30031983: APIM Infrastructure Changes (see page 1607)	1. ✅ Standard Change	1. ✅	1. Terraform will require manual imports of new s3 related imports as part of the uplift. This will be done in hours by Ben Strutt prior to release. To view Digital Onboarding Service JIRA tickets in the release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
10 Mar 2022	No releases			
08 Mar 2022	1. RFC 30031886: APIM Infrastructure Changes (see page 1606)	1. ✅ Standard Change	1. ✅	
01 Mar 2022	No releases			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
24 Feb 2022	1. RFC 30031685: APIM Infrastructure Changes (see page 1600)	1. ✅ Standard Change	1. ✅	1. Release to be done in hours as does not touch prod traffic
22 Feb 2022	1. RFC 30031092 - NHS e-RS APIM Release R1.5 Production Deployment (see page 1593) 2. RFC 30031501 - API Management Infrastructure: Switch to High Bandwith HSCN connection (see page 1598) 3. RFC 30031595 - APIM Infrastructure Changes (see page 1599)	1. ✅ 2. ✅ 3. ✅ Standard Change	1. ✅ 2. ✅ (in hours intervantion needed) 3. ✅ (successful after the above was resolved)	2. AWS + Apigee release for shared flow. 3. To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
17 Feb 2022	No releases			
15 Feb 2022	No releases			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
10 Feb 2022	1. RFC 30031345 - Add Developer Identity SSO to Production Apigee Portal (see page 1596) 2. RFC 30031392 - Digital Onboarding Service (see page 1597)	1. ✅ 2. ✅ Standard Change	1. ✅ (in hours configuration change) 2. ✅	1. This is an in-hours configuration change to Apigee Developer Portal 2. To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
08 Feb 2022	No releases			
03 Feb 2022	1. RFC 30031265 Digital Onboarding Service (see page 1595)	1. ✅ Standard Change		To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
01 Feb 2022	No releases			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
27 Jan 2022	1. RFC 30031095 - Digital Onboarding Service/Developer Identity SSO (see page 1594)	1. ✅	1. Pending	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
25 Jan 2022	No releases			
21 Jan 2022	1. RFC 30031068: APIM Infrastructure Changes (see page 1592)	1. ✅ Standard Change	1. ✅	Fix dashboard for TechOps: **done at 3pm**
20 Jan 2022	1. RFC 30031030: APIM Infrastructure Changes (see page 1591)	1. ✅ Standard Change	1. ✅	
18 Jan 2022	1. RFC 30030479 - NHS e-RS APIM Release R1.4 Production Deployment (see page 1583) 2. RFC 30030933 - PDS Add ASID and Reinstate ASID-Checking in Prod (see page 1589) 3. RFC 30030990: APIM Infrastructure Changes (see page 1590)	1. ✅ 2. Pending 3. ✅ Standard Change	1. ✅ 2. ✅ 3. ⛔ Rolled back	
Thursday 13th January 2022	1. RFC 30030422 - SDS API new query params + mapping change (see page 1579)	1. ✅ Standard Change		**Note - deploy at 2pm**

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
11 Jan 2022	1. RFC 30030831: APIM Infrastructure Changes (see page 1586) 2. RFC 30030852 - Identity Service (see page 1588)	1. ✅ Standard Change 2. ✅ Standard Change	1. ✅ 2. ✅	**MUST be done in order. Infrastructure first then Identity Service**
06 Jan 2022	1. RFC 30030787: APIM Infrastructure Changes (see page 1585)	1. ✅ Standard Change	1. Pending	I've marked myself Ben Strutt as the implementer in the RFC as I couldn't find Unknown User (jofi7) in the list.
04 Jan 2022	Change Freeze			
Thursday 30th December 2021	Change Freeze			
Tuesday 28th December 2021	Change Freeze			
Thursday 23rd December 2021	Change Freeze			
Tuesday 21st December 2021	Change Freeze			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday 16th December 2021	Urgent fixes only			
Wednesday 15th December 2021	RFC TBC - Alignment of infrastructure version in Production (see page 1584)	1. ✅ Standard Change	1. ✅	
Tuesday 14th December 2021	Urgent fixes only 1. RFC 2021-12-14 TBD: Upgrade netcompany app ratelimit in production (see page 1492) 2. Manual PDS API fix - update RateLimiting shared flow from rollingwindow to calendar	1. ✅ Raised by COVID team 2. ✅ 30030570	1. ✅ 2. ✅	
Thursday 9th December 2021	1. RFC 30030423 - NHS e-RS APIM Release R1.3 Production Deployment (see page 1580) 2. RFC 30030433 - Digital Onboarding Service (see page 1582)	1. ✅ 2. ✅ (Standard Change)	1. ✅ 2. ✅	1. This release has been agreed can be released in hours due to there being no production traffic. It will be released at 4pm. 2. To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Wednesday 9th December 2021	1. RFC 30030425: APIM Infrastructure - Add ERS Production Target Server (see page 1601)	1. ✅ (Standard Change)	1. ✅	1. Released in hours as purely additive infrastructure update.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday 7th December 2021	1. RFC 30030118 - NHS e-RS APIM Release R1.3 Production Deployment (see page 1573) 2. RFC 30030374 - Identity Service (see page 1577) 3. RFC 30030383 - Digital Onboarding Service (see page 1578) (Standard change)	1. ✅ 2. ✅ (Standard Change) 3. ✅ (Standard Change)	1. ⛔ Failed 2. ✅ 3. ✅	1. Note this is a Standard Change RFC 2. N/A 3. To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
				4. The failure of RFC 30030118 to deploy was due to a dependency on an Infrastructure service release that we missed. Whilst the deployment was unsuccessful it didn't cause any problems in production. It just meant we needed to schedule an Infra service release (to deploy a target server for eRS) and then re-run the eRS proxy deployment.
Thursday 2nd December 2021	1. RFC 30030299 - Digital Onboarding Service (see page 1576) (Standard change)	1. ✅	1. Pending	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this. Note this is a Standard Change RFC

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday 30th November 2021	1. RFC 30030217: Auth Services Changes - backchannel logout (see page 1575)	1. ✅	1. ✅	
Thursday 25th November 2021	No releases			
Tuesday 23rd November 2021	1. RFC 30030097 - Digital Onboarding Service (see page 1572)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Thursday 18th November 2021	No releases			
Tuesday 16th November 2021	1. RFC 30029982 - NHS App API (see page 1571)	1. ✅	1. ✅	In-hours release
Thursday November 11th	1. RFC 30029904 - Digital Onboarding Service (see page 1569) 2. RFC 30029940: APIM Infrastructure Release - Fix ApplyRateLimiting shared flow priority order of arguments [1680]	1. ✅ 2. ✅	1. ✅ 2. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.

1680 https://nhsd-confluence.digital.nhs.uk/display/APM/RFC+30029940%3A+APIM+Infrastructure+Release+-+Fix+ApplyRateLimiting+shared+flow+priority+order+of+arguments?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday November 9th	1. RFC 30029855: Auth Services Changes (see page 1566) 2. RFC 30029871: APIM Ambulance Data Submissions API Updates - v1.0.387-alpha (see page 1567)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Thursday November 4th	No releases			
Tuesday October 26th	1. RFC 30029702 - Digital Onboarding Service (see page 1565)	1. ✅	1. ✅	
Thursday October 21st	1. RFC 30029596: APIM Infrastructure Changes (see page 1564)	1. ✅	1. ✅	
Tuesday October 19th	1. RFC 30029506: APIM Ambulance Data Submissions API Updates - v1.0.376-alpha (see page 1557) a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50401&view=results 2. RFC 30029519 - EPS Primary care prescribing technical go live (see page 1558)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Thursday October 14th	1. RFC 30029451 - SDS API new query parameters (see page 1555) 2. RFC 30029482 - Set ApplyRateLimiting Quota policies to rollingwindow (see page 1556) 3. RFC 30029491 - APIM Infrastructure Changes (see page 1556)	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	2. Standard Infra Release 3. Emergency RFC

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday October 12th	1. RFC 30029377 - APIM Infrastructure Changes (see page 1553)	1. ✅	1. ✅	Standard Infra release
Thursday October 7th	1. RFC 30029301 - Convert Auth (Identity) Service to use ApplyRateLimiting shared flow	1. ✅	1. ✅	
Thursday September 23rd	1. RFC 30029047 - APIM NHS App API - Standard Release (first since 22 January 2021) (see page 1550)	1. ✅	1. Pending	First release in many months
Tuesday September 21st	1. RFC 30028975 - APIM Covid-19 Test Results API - Remove legacy product from Covid Pass APIGEE app configuration (see page 1549)	1. ✅	1. ✅	Unknown User (madu59) from Covid 19 test app is the primary contact for this release
Thursday September 16th	1. RFC 30028932 - Digital Onboarding Service fixes (see page 1548)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Tuesday September 14th	No releases			
10 Sep 2021	1. RFC 30028836 - Management UI Enable Azure AD SAML - In hours (see page 1546)	1. ✅	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
09 Sep 2021	1. RFC 30028795 - APIM Infrastructure Changes (see page 1544) 2. RFC 30028855 - APIM Covid-19 Test Results API - Enabling Token Exchange in the Covid Pass APIGEE app configuration (see page 1547)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Tuesday September 7th	1. RFC 30028763 - Digital Onboarding Service Fixes (see page 1543)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Thursday September 2nd	No releases			
Tuesday August 31st	1. RFC 30028609 - Digital Onboarding Service - disable user, resend invite, DAQ changelog, misc fixes (see page 1542)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Thursday August 26th	No releases			

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday August 24th	1. RFC 30028449: APIM PDS Patient (Citizen) Access Integration (see page 1540) 2. RFC 30028450 - APIM Infrastructure Changes [1681]	1. ✅ 2. ✅	1. ✅ 2. ✅	PDS Postponed from Thur 19th due to bug found post-RFC raising. RFC release version to be amended
Tues 24th - 26th	1. RFC 30028523 - Management UI SAML Testing - In hours (see page 1541)	1. Pending	1. Pending	This is management auth only with no run-time impact. Will be in-hours and there might be multiple short tests.
Thursday August 19th	No releases			
Tuesday August 17th	1. RFC 30028351 - APIM Covid-19 Test Results API - Token Exchange (see page 1535) 2. RFC 30028396 Apigee application restricted for immunisation history api (see page 1537) 3. RFC 30028431 - TPS raised to 270 - PDS API (see page 1538)	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	

1681 https://nhsd-confluence.digital.nhs.uk/display/APM/RFC+30028450+-++APIM+Infrastructure+Changes?src=contextnavpagetreemode

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday August 10th	1. RFC-30028297 Migrate Immunisation History Client Apps -> User Restricted (see page 1602) 2. RFC 30028313 - Digital Onboarding Service - GA/Hotjar fixes (see page 1533) 3. RFC 30028333 - API Management Infrastructure Updates (see page 1534)	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	1. Dependent on a successful release of 'Check My Coronavirus Record' (also planned for Tuesday 10th) 2. To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Thursday August 5th	1. RFC 30028231 - PDS FHIR API Conformance Resource URL Change (see page 1532)	1. ✅	1. ✅	1. This must happen on Thursday 5th August to keep the API Proxy in step with the Spine backend
Tuesday August 3rd	1. RFC 30028203 - Digital Onboarding Service - GA/Hotjar fixes (see page 1531)	1. ✅	1. ⛔ Failed	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday July 27th	1. RFC 30028033: APIM Identity Service Updates (see page 1531) a. IdentityService changes: v1.0.1156-alpha i. RELEASE: **v1.0.1156-alpha** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=40641&view=results ii. ROLLBACK: **v1.0.1144-alpha** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38550&view=results b. Infrastructure changes: v2021.07.20.09.49[1682] 2. RFC 30027868 - Verifying URID for user updates (see page 1527) a. PDS FHIR API: v1.2.1154-beta[1683] b. ROLLBACK: re-deploy **v1.2.1106-beta**[1684]	1. ✅ 2. ✅	1. ✅ 2. ✅	Requires an Infrastructiure release to be applied AFTER the Identity Service release, as well as a PDS release, which is to be applied AFTER the Infrastructure release Dependent on successful release and deployment of RFC 30028033 (see page 1531), prior to releasing this.

1682 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.07.20.09.49
1683 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=41508&view=results
1684 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday July 22nd	1. Immunisation History Updates - RFC 30027987 a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=40875&view=results and 'retry stage' b. ROLLBACK: re-deploy `v1.0.360-alpha`	1. ✅	1. ✅	
Tuesday July 20th	1. RFC 30027924 - Digital Onboarding Service - High Volume SMTP fix (see page 1530)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.
Tuesday July 13th	1. RFC 30027804: APIM Identity Service Updates a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38550&view=results b. ROLLBACK: Rebuild and deploy v1.0.1132-alpha 2. RFC 30027232 (see page 1520): APIM Run script in prod to update JWKS resource URLs for applications (see page 1520)	1. ✅ 2. ✅	1. ✅ 2. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday July 8th	1. RFC 30027655: APIM PDS Disable Async Update Mode (see page 1525) 2. RFC 30027682: APIM Infrastructure Updates to enable prod Spine connectivity (see page 1526)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Tuesday July 6th	1. RFC 30027574: APIM Identity Service Updates (v1.0.1132-alpha) a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38405&view=results b. ROLLBACK: Rebuild and redeploy v1.0.1096-alpha 2. RFC 30027613: PDS FHIR API Quota Update (see page 1525)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Thursday July 1st	1. RFC 30027510 - KVM Update to Identity Service Prod (see page 1524)	1. ✅	1. ✅	
Wednesday June 30th	1. RFC 30026932 - APIM Deprecation & Retirement of unused APIs	1. ✅	This is an RFC to deprecate a number of APIs that aren't in use - no out of hours deployment required.	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday June 24th	1. RFC 30027380 - APIM Infrastructure Updates to support future EPS changes (Tag v2021.06.23.15.17) a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38270&view=results b. ROLLBACK: Rebuild and redeploy v2021.06.22.10.58 TERRAFORM ONLY c. ROLLBACK: Rebuild and redeploy v2021.06.11.08.48 APIGEE ONLY	1. ✅	1. ✅	
Wednesday 23rd June	1. **EMERGENCY RELEASE** - RFC 30027382 - COVID-19 Test Results P5 Auth Changes (→ v1.0.153) a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38111&view=results b. ROLLBACK: Rebuild & Redeploy v1.0.134-alpha	1. ✅	1. ✅	
Tuesday 22nd June	1. RFC 30027332 - Digital Onboarding Service release (see page 1521)	1. ✅	1. ✅	To view JIRA tickets in release plan, you will need permission to access the Turnstile board. Please contact Turnstile to arrange this.

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Saturday 19th June	1. RFC 30027285 - Immunisation History API Update (Emergency release to support Covid related go-live on Monday)	1. ✅	1. ✅	Matt Mercer has agreed to do this deployment with support being available from the Immunisation History API squad if needed (Shaun).
Tuesday 15th June	1. RFC 30027067, 30027100 & 30027153: APIM - Infrastructure Release (see page 1515) (→ v2021.06.11.08.48) a. RFC 30027067 - Infrastructure service update (Tag v2021.06.08.14.33) b. RFC 30027100 - Infrastructure service update to support future EPS changes (Tag v2021.06.08.15.55) c. RFC 30027153 - Infrastructure service update for APM_1811 (Tag v2021.06.11.08.48) 2. RFC 30027083: APIM - PDS API - Sync in Production (see page 1518)	1. ✅ 2. ✅ 3. ✅ 4. ✅	1. ✅ 2. ✅ 3. ✅ 4. ✅	There are 3 separate Infrastructure service releases scheduled for the same night, to help us to break the releases down into smaller change sets (so rather than have 5 changes in one big bang release, it's one with 3 changes and then 2 single change releases). If one of the later releases fails that means we don't need to fully roll-back. **How to deploy infra releases from tags**: KOP-004 Release Process - Infrastructure[1685]

1685 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119755126/KOP-004+Release+Process+-+Infrastructure

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Monday 7th June	1. **EMERGENCY RELEASE** - RFC 30027001 - Add the Covid 19 Test Results product to the nc-vaccination-cache-prod application	1. ✅	1. ✅	
Thursday 3rd June	1. RFC 30026941 - Identity Service API Updates (v1.0.1096-alpha) a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=34846&view=results b. ROLLBACK: v1.0.1083-alpha - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=34240&view=results	1. ✅	1. ✅	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday 1st June	1. RFC 30026870 - Monitoring Discovery Service a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=34840&view=results b. ROLLBACK: v1.0.22-alpha - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25938&view=results 2. RFC 30026910 - API Management Infrastructure Updates to support SDS Go-Live a. Link https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=35618&view=results	1. ✅ 2. ✅	1. ✅ 2. ⚠️ Succeeded with caveats – cognito client secret issue still apparent, but not a goal of this deploy	
Thursday 27th May	1. RFC 30026784 - Authorisation Service Update a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=34240&view=results 2. RFC 30026822 (see page 1515) - Infrastructure Service Update a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=35132&view=results	1. ✅ 2. ✅	1. ✅ 2. ⛔ Failed (changes partially applied)	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday 20th May	1. RFC 30026690 - Immunisation History API Updates a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=34174&view=results 2. RFC30026707 - Identity Service API Updates a. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=33521&view=results	1. ✅ 2. ✅	1. ✅ 2. ✅	
Tuesday 18th May	1. RFC 30026623 - Infrastructure Service Updates	1. ✅	1. ✅	
Thursday 13th May	6553 must be done first. If it fails then 6495 cannot proceed. 1. RFC 30026553- API Management Infrastructure Updates (see page 1514) 2. RFC 30026495: APIM - Identity Service Updates (see page 1513)	1. ✅ 2. ✅	1. ✅ Apigee config succeeded (required for 26495). Terraform failed - but only new dap/waf objects not used in any live service. 2. ✅	
Tuesday 11th May	1. RFC 30025426 - Immunisation History API Updates (see page 1500)	1. ✅	Released early due to live traffic cutoff	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday 29th April	6140 must be done first, and is the priority for tonight. If 6140 fails, then 6104 cannot proceed: 1. RERELEASE RFC 30026140 - API Management Infrastructure Updates (see page 1512) 2. RFC 30026104: APIM Identity Service Updates (see page 1510)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Tuesday 27th April	6140 must be done first, and is the priority for tonight. If 6140 fails, then 6104 cannot proceed: 1. RERELEASE RFC 30026140 - API Management Infrastructure Updates (see page 1512) 2. RFC 30026104: APIM Identity Service Updates (see page 1510)	1. ✅ 2. ✅	1. ⛔ Failed 2. ⛔ Not attempted	
Thursday 22nd April	6140 must be done first, and is the priority for tonight. If 6140 fails, then 6104 cannot proceed: 1. RFC 30026140 - API Management Infrastructure Updates (see page 1512) 2. RFC 30026104: APIM Identity Service Updates (see page 1510)	1. ✅ 2. ✅	1. ⛔ Failed 2. ⛔ Not attempted	
Tuesday 20th April	1. RFC 30026054: Fix Spike Arrest Error Message (see page 1510)	1. ✅	1. ✅	
Tuesday 13th April	1. RFC 30025905: Immunisation History API - Move into private beta - No Deployment Change	1. tbc	1. No release required	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday 8th April	1. RFC 30025798: Splunk logging improvements and Changes to support Identity Service (see page 1507) 2. RFC 30025874: APIM Identity Service Updates (see page 1508) 3. RFC 30025842: No deployment change (see page 1390) 4. RFC 30025803: APIM PDS Base URL Deprecation Updates (see page 1508)	1. ✅ 2. ✅ 3. ✅ 4. ✅	1. ✅ 2. ✅ 3. No release required. 4. ✅	
Wednesday 7th April	1. RFC 30025529: APIM EPS Updates (see page 1503)	1. tbc	1. tbc	
Thursday 1st April	1. RFC 30025726: AWS into production (no deployment required)	1. tbc	1. No release required - RFC for approval purposes only	
Tuesday 30th March	1. RFC 30025695: APIM Identity Service Update - v1.0.946-alpha[1686] 2. RFC 30025723: APIM Identity Service Updates (see page 1506)	1. ✅ 2. ✅	1. ✅ 2. ✅	
Thursday 25th March				
Tuesday 23rd March	1. RFC 30025588: APIM Apigee Config - Update Production Certificate (see page 1503)	1. tbc	1. tbc	

1686 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.946-alpha

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Thursday 18 March	1. RFC 30025432: APIM Ambulance Data Submissions API Updates - v1.0.330-alpha (see page 1501) 2. RFC 30025379: APIM Identity Service Updates - v1.0.859-alpha (see page 1499)	1. ✅ 2. ✅	1. ✅ (mistakenly deployed a 2nd time; failed smoke tests) 2. ⛔ Failed	
Tuesday 16th March	1. RFC 30025432: APIM Ambulance Data Submissions API Updates - v1.0.330-alpha (see page 1501) 2. RFC 30025379: APIM Identity Service Updates - v1.0.859-alpha (see page 1499)	1. ✅ 2. ✅	1. ✅ (failed smoke tests) 2. ⛔ Not attempted	
Thursday 11th March	1. RFC 30025357: APIM Infrastructure: Add scope variable to Splunk output	1. ✅	1. ✅	
Tuesday 9th March	1. RFC 30025111 - Monitoring Service Discovery in Prod (see page 1495) (RESUME)	1. ✅	1. ✅	
Thursday 4th March	1. RFC 30025111 - Monitoring Service Discovery in Prod (see page 1495) (defered)	1. ✅	1. ⚠️	
Tuesday 2nd March	1. RFC 30025111 - Monitoring Service Discovery in Prod (see page 1495) (defered)	1. ✅	1. ⚠️	
Thursday 25th Feb	1. RFC 30025111 - Monitoring Service Discovery in Prod (see page 1495) (defered)	1. ✅	1. ⚠️	

Release Window	Release Plans (in order)	RFCs Approved?	Release Status	Notes
Tuesday 23rd Fed	No releases scheduled	-	-	
Thursday 18th Feb	No releases scheduled	-	-	
Tuesday 16th Feb	1. 2021-02-16 identity-service-api deployment plan (see page 1489)	1. ✅	1. ✅	
21st January	30024381 - Apigee Config - Update APIgee Key Value Maps 30024419 - APIM Config Change - Adding Splunk Logging to Ambulance Analytics API 30024430 - Apigee Config - Update API Product descriptions and requirements	1. ✅ 2. ✅ 3. ✅	1. ✅ 2. ✅ 3. ✅	
Monday 13th Jan	CHG0070517	1. ✅	1. ✅	
27 Mar 2025	CHG0073626	1. ✅	1. Scheduled	

9.2 CHG NumberHere - Template

<Link to ServiceNow Change request>

9.2.1 Tickets

CHG created by <**NAME of change RAISER**>. The Release implements the changes in the following tickets:

- **<List of tickets ideally as links to JIRA>**

9.2.2 Prepared Release

Release Tag: <**eg. v2021.07.06.11.20**>

<**Link to Azure** devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results>

9.2.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.2.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1687]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.2.4 Roll-Back

- Redeploy:
  - Release Tag: <**eg. v2021.06.23.15.17**>
  - <**Link** to Azure devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results
- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.3 Archive

This is for old releases. Move releases into this folder once they have been completed.

- 2021-02-16 identity-service-api deployment plan (see page 1489)
- Release Plan APIM-2021-03-02 (see page 1492)
- RFC 2021-12-14 TBD: Upgrade netcompany app ratelimit in production (see page 1492)
- RFC 30025111 - Monitoring Service Discovery in Prod (see page 1495)
- RFC 30025379: APIM Identity Service Updates - v1.0.859-alpha (see page 1499)
- RFC 30025426 - Immunisation History API Updates (see page 1500)
- RFC 30025432: APIM Ambulance Data Submissions API Updates - v1.0.330-alpha (see page 1501)
- RFC 30025529: APIM EPS Updates (see page 1503)
- RFC 30025588: APIM Apigee Config - Update Production Certificates (see page 1503)
- RFC 30025611: APIM - Remove API Platform caveat for limiting the number of connecting parties (see page 1504)
- RFC 30025620: APIM - Remove API Platform caveat for TPS rate limit of 5 TPS (see page 1505)
- RFC 30025723: APIM Identity Service Updates (see page 1506)

1687 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- RFC 30025798: Splunk logging improvements and Changes to support Identity Service (see page 1507)
- RFC 30025803: APIM PDS Base URL Deprecation Updates (see page 1508)
- RFC 30025874: APIM Identity Service Updates (see page 1508)
- RFC 30026054: Fix Spike Arrest Error Message (see page 1510)
- RFC 30026104: APIM Identity Service Updates (see page 1510)
- RFC 30026140 - API Management Infrastructure Updates (see page 1512)
- RFC 30026495: APIM - Identity Service Updates (see page 1513)
- RFC 30026553- API Management Infrastructure Updates (see page 1514)
- RFC 30026822 - Infrastructure Service Update (see page 1515)
- RFC 30027067, 30027100 & 30027153: APIM - Infrastructure Release (see page 1515)
- RFC 30027083: APIM - PDS API - Sync in Production (see page 1518)
- RFC 30027232: APIM Run script in prod to update JWKS resource URLs for applications (see page 1520)
- RFC 30027332 - Digital Onboarding Service Release (see page 1521)
- RFC 30027481 PDS API - Sync Wrap PROD (see page 1522)
- RFC 30027510 - API Management Infrastructure - Identity Service KVM (see page 1524)
- RFC 30027613: APIM PDS Production Quota Update (see page 1525)
- RFC 30027655: APIM PDS Disable Async Update Mode (see page 1525)
- RFC 30027682: APIM Infrastructure Updates to enable prod Spine connectivity (see page 1526)
- RFC 30027868 - Verifying URID for user updates (see page 1527)
- RFC 30027924 - Digital Onboarding Service - High Volume SMTP fix (see page 1530)
- RFC 30028033: APIM Identity Service Updates (see page 1531)
- RFC 30028203 - Digital Onboarding Service - GA/Hotjar fixes (see page 1531)
- RFC 30028231 - PDS FHIR API Conformance Resource URL Change (see page 1532)
- RFC 30028313 - Digital Onboarding Service - GA/Hotjar fixes (see page 1533)
- RFC 30028333 - API Management Infrastructure Updates (see page 1534)
- RFC 30028351 - APIM Covid-19 Test Results API - Token Exchange (see page 1535)
- RFC 30028396 Apigee application restricted for immunisation history api (see page 1537)
- RFC 30028431 - TPS raised to 270 - PDS API (see page 1538)
- RFC 30028449: APIM PDS Patient (Citizen) Access Integration (see page 1540)
- RFC 30028450 - APIM Infrastructure Changes (see page 1540)
- RFC 30028523 - Management UI SAML Testing - In hours (see page 1541)
- RFC 30028609 - Digital Onboarding Service - disable user, resend invite, DAQ changelog, misc fixes (see page 1542)
- RFC 30028763 - Digital Onboarding Service Fixes (see page 1543)
- RFC 30028795 - APIM Infrastructure Changes (see page 1544)
- RFC 30028836 - Management UI Enable Azure AD SAML - In hours (see page 1546)
- RFC 30028855 - APIM Covid-19 Test Results API - Enabling Token Exchange in the Covid Pass APIGEE app configuration (see page 1547)
- RFC 30028932 - Digital Onboarding Service fixes (see page 1548)
- RFC 30028975 - APIM Covid-19 Test Results API - Remove legacy product from Covid Pass APIGEE app configuration (see page 1549)
- RFC 30029047 - APIM NHS App API - Standard Release (first since 22 January 2021) (see page 1550)
- RFC 30029377 - APIM Infrastructure Changes (see page 1553)
- RFC 30029381 - SDS API forward reliable and forward express address fix and new actor mappings (see page 1554)

- RFC 30029451 - SDS API new query parameters (see page 1555)
- RFC 30029482 - Set ApplyRateLimiting Quota policies to rollingwindow (see page 1556)
- RFC 30029491 - APIM Infrastructure Changes (see page 1556)
- RFC 30029506: APIM Ambulance Data Submissions API Updates - v1.0.376-alpha (see page 1557)
- RFC 30029519 - EPS Primary care prescribing technical go live (see page 1558)
- RFC 30029596: APIM Infrastructure Changes (see page 1564)
- RFC 30029702 - Digital Onboarding Service (see page 1565)
- RFC 30029855: Auth Services Changes (see page 1566)
- RFC 30029871: APIM Ambulance Data Submissions API Updates - v1.0.387-alpha (see page 1567)
- RFC 30029904 - Digital Onboarding Service (see page 1569)
- RFC 30029940: APIM Infrastructure Release - Fix ApplyRateLimiting shared flow priority order of arguments (see page 1570)
- RFC 30029982- APIM NHS App API - Apply per-application rate limits (see page 1571)
- RFC 30030097 - Digital Onboarding Service (see page 1572)
- RFC 30030118 - NHS e-RS APIM Release R1.3 Production Deployment (see page 1573)
- RFC 30030152 - EPS bug fix (see page 1574)
- RFC 30030217: Auth Services Changes - backchannel logout (see page 1575)
- RFC 30030299 - Digital Onboarding Service (see page 1576)
- RFC 30030374 - Identity Service (see page 1577)
- RFC 30030383 - Digital Onboarding Service (see page 1578)
- RFC 30030422 - SDS API new query params + mapping change (see page 1579)
- RFC 30030423 - NHS e-RS APIM Release R1.3 Production Deployment (see page 1580)
- RFC 30030433 - Digital Onboarding Service (see page 1582)
- RFC 30030479 - NHS e-RS APIM Release R1.4 Production Deployment (see page 1583)
- RFC 30030574 - Alignment of infrastructure version in Production (see page 1584)
- RFC 30030787: APIM Infrastructure Changes (see page 1585)
- RFC 30030831: APIM Infrastructure Changes (see page 1586)
- RFC 30030852 - Identity Service (see page 1588)
- RFC 30030933 - PDS Add ASID and Reinstate ASID-Checking in Prod (see page 1589)
- RFC 30030990: APIM Infrastructure Changes (see page 1590)
- RFC 30031030: APIM Infrastructure Changes (see page 1591)
- RFC 30031068: APIM Infrastructure Updates (see page 1592)
- RFC 30031092 - NHS e-RS APIM Release R1.5 Production Deployment (see page 1593)
- RFC 30031095 - Digital Onboarding Service/Developer Identity SSO (see page 1594)
- RFC 30031265 Digital Onboarding Service (see page 1595)
- RFC 30031345 - Add Developer Identity SSO to Production Apigee Portal (see page 1596)
- RFC 30031392 - Digital Onboarding Service (see page 1597)
- RFC 30031501 - API Management Infrastructure: Switch to High Bandwith HSCN connection (see page 1598)
- RFC 30031595 - APIM Infrastructure Changes (see page 1599)
- RFC 30031685: APIM Infrastructure Changes (see page 1600)
- RFC 30030425: APIM Infrastructure - Add ERS Production Target Server (see page 1601)
- RFC-30028297 Migrate Immunisation History Client Apps -> User Restricted (see page 1602)
- RFC - APIM Infrastructure Updates - Feature Flag (see page 1603)
- RFC TBD - Digital Onboarding Service (see page 1605)

9.3.1 2021-02-16 identity-service-api deployment plan

As discussed in today's meeting here are the steps to follow in order to implement the APM-1701 changes in the identity service API.

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1688] when deploying these changes!

9.3.1.1 Plan Summary

- ☑ 30024937 - Utils changes
  - ☑ Then have to run each of the following deploys in prod (create a new BUILD on the build pipeline most recent tag, and then babysit through to prod release)
    - ☑ Monitoring-Service-Discovery v1.0.21-alpha[1689] **(CORRECTION: v1.0.19-alpha)**
      - ☑ Run a new build of latest tag https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=23220
      - ☑ Watch thru to prod - https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
    - ☑ NHS-App v1.0.70-alpha[1690] **(CORRECTION: v1.0.73-alpha)**
      - ☑ Run a new build of latest tag https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=23179
      - ☑ Watch thru to prod - https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
- ☑ 30024951 - Ambulance _status v1.0.290-alpha[1691]
  - ☑ Run a new build of latest tag https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=23265
  - ☑ Watch thru to prod - https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
- ☑ 30024940 - PDS Updates v1.2.612-beta[1692]
  - ☑ Run a new build of latest tag https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=140

1688 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1689 https://github.com/NHSDigital/monitoring-service-discovery/tree/v1.0.21-alpha
1690 https://github.com/NHSDigital/nhs-app-api/tree/v1.0.70-alpha
1691 https://github.com/NHSDigital/ambulance-analytics-api/tree/v1.0.290-alpha
1692 https://github.com/NHSDigital/personal-demographics-service-api/releases/tag/v1.2.612-beta

- [x] Watch thru to prod - https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
- [x] 30024941 - Identity service v1.0.797-alpha[1693]
  - [x] This is the riskiest change.
  - [x] As before, run a new build of latest tag https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=23260
  - [x] Watch thru to prod - https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
    - [x] Watch for at least 20 mins
    - [ ] If significantly higher 401s/403s/5XX, then roll back
      - [ ] Call Aubyn
      - [ ] Mitigate IMMEDIATELY
      - [ ] Report incident:
        - [ ] Contact service bridge to inform them OR raise an incident on Cherwell (DO THIS AFTER STARTING THE FAILURE PLAN)
  - [ ] FAILURE PLAN:
    - [ ] Run build of tag: v1.0.718-alpha[1694]

9.3.1.1.1 KOP - Build from tag

KOP-019 Release process - Individual API[1695]

9.3.1.1.2 KOP - Rollback

KOP-017 Rolling back to a stable release[1696]

9.3.1.2 Detail

9.3.1.2.1 APM-1701

1. Add default scopes to API's (all but PDS). Initial testing has shown it works with the currently deployed version of identity service, but need to apply it more widely as part of integration testing. In case the manifest file is introduced to this API's while we are driving these changes, the manifest file

1693 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.797-alpha
1694 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.718-alpha
1695 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169715007/KOP-019+Release+process+-+Individual+API
1696 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708113/KOP-017+Rolling+back+to+a+stable+release

will add the scopes by default for every API (as discussed with Ben Strutt).

a. step 1:
    i. Merge utils changes into edge branch. (DONE)
b. step 2:
    i. Run latest releases on ptl environments of the following API's
        1. Ambulance Analytics (DONE)
        2. Electronic-Prescription-Service (DONE)
        3. Hello-World (DONE)
        4. Key-Locator (need assistance)
        5. Mock-NHSDID (not sure)
        6. Monitoring-Service-Discovery (DONE)
        7. NHS-App (DONE)
        8. Reasonable-Adjustments (release failing in internal-dev since jan 21)
        9. Risk-Stratification (DONE)
        10. Spine-Directory-Service (DONE)
        11. Summary-Care-Record (DONE)
        12. Sync-Wrap
        13. E-Referrals-Service-API

2. Add default Scopes to PDS (new ones) using the current manifest file.
    a. Step 1:
        i. Add new scopes and to the manifest file. (DONE)
        ii. Implement the changes to PDS to also use the old scopes and new scopes inside the proxy. (DONE)
        iii. Review changes. (DONE)
    b. Step 2:
        i. Merge changes and release (DONE)
        ii. Perform automated tests, for an app-restricted app, to show that PDS continues to work. (DONE)

3. Identity Service deployment
    a. Step 1:
        i. Deploy Identity-Service in ptl environments and test features can coexist with recent changes already merged into master.(DONE)
        ii. Perform any other non-pipeline automated tests on PDS for both user and app restricted on INT
        iii. Perform manual tests on PDS with Smartcards and Yubikeys on INT (Tested by Gurdeep with no issues found)
        iv. Tests AMB-495 and APM-1814
        v. If all done - then can ask API producers to check their APIs on INT (which we might have to wait a few days for)

Lucas *- Crossed out the last two, since the release to Production has to repeat the above sequence... is that correct? (Although 1.b. has a much smaller lists of proxies)*

9.3.1.2.2 AMB-495

These changes were merged into master and they are deployed in ptl. (Pipelines - Run v1.0.708-alpha+22554 (azure.com)[1697]) they should be okay to be released

9.3.1.2.3 APM-1814

These changes were merged into master and they are deployed in ptl. (Pipelines - Run v1.0.685-alpha+22146 (azure.com)[1698]) they should be okay to be released

9.3.2 Release Plan APIM-2021-03-02

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution

Jira project doesn't exist or you don't have permission to view it.
View these issues in Jira[1699]

9.3.3 RFC 2021-12-14 TBD: Upgrade netcompany app ratelimit in production

9.3.3.1 Release plan

App name: nc-vaccination-cache-prod

App ID: 2940830f-243c-4bd9-bb7c-306130cce541

Rate limit (s): 3,500

Rate limit (m): **210000**

9.3.3.1.1 Authenticate

Grab a temporary passcode from https://nhs-digital-prod.login.apigee.com/passcode

Then:

```
export PASSCODE="<PASSCODE_FROM_APIGEE>"
```

1697 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=22554&view=results
1698 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=22146&view=results
1699 https://nhsd-jira.digital.nhs.uk/secure/IssueNavigator.jspa?reset=true&jqlQuery=fixVersion+%3D+APIM-2021-03-02+&src=confmacro

```
curl "https://nhs-digital-prod.login.apigee.com/oauth/token" -H "Accept: application/
json" -d "grant_type=password&response_type=token&passcode=$PASSCODE" -H
"Authorization: Basic ZWRnZWNsaTplZGdlY2xpc2VjcmV0"
```

Take the access token from the resulting JSON string and copy it.

9.3.3.1.2 Log on to a build agent

~~See KOP-021 Logging in to EC2 instances~~[1700]. **(NOTE: this is not working for me right now, but using AWS SSM is a workaround)**

9.3.3.1.3 Follow instructions to set app rate limit

Rate limiting shared flow (see page 1039) - from 'Adding custom rate limiting for an individual application'

```
curl https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/nhsd-
prod/apps/2940830f-243c-4bd9-bb7c-306130cce541/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' | jq
```

View the existing rate limiting information, which should look like this:

```
{
  "identity-service-prod": {
    "quota": {
      "interval": 1,
      "limit": 120000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "ratelimit": "120000pm"
    }
  }
}
```

The new limits should look like this:

```
{
  "identity-service-prod": {
    "quota": {
      "interval": 1,
      "limit": 210000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "ratelimit": "210000pm"
```

1700 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/193011226/KOP-021+Logging+in+to+EC2+instances

```
    }
  }
}
```

Then send the new limits in a command as follows:

```
curl -X PUT https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/
nhsd-prod/apps/2940830f-243c-4bd9-bb7c-306130cce541/attributes/ratelimiting/identity-
service-prod -H 'Authorization: bearer <your_apigee_access_token>' -H 'content-type:
application/json' --data '
{
  "identity-service-prod": {
    "quota": {
      "interval": 1,
      "limit": 210000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "ratelimit": "210000pm"
    }
  }
}'
```

9.3.3.1.4 Monitor and verify

Confirm the changes in the apigee admin panel.

Stay on past the 10PM announcement and watch for 429s.

If traffic is reaching the new limits or triggering 429s then a conference call (e.g. via HSSI) with the relevant stakeholders will be called to manually increase the limits (if agreed).

9.3.3.1.5 Keep it secret, keep it safe

On the build agent:

```
cat /dev/null > ~/.bash_history
history -c
```

This will delete any secrets posted in the bash history.

9.3.3.2 Rollback plan

Run the above plan, with the ratelimits dialed down.

9.3.4 RFC 30025111 - Monitoring Service Discovery in Prod

As discussed in today's meeting here are the steps to follow in order to implement the AMB-527 changes in the identity service API.

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1701] when deploying these changes!

9.3.4.1 Revised Plan - Tuesday 9th March

Deploy to Prod the following services. Verify that each service has registered and appeared in the Prod Grafana.

NOTE: Services MUST be rebuilt during out-of-office hours, as rebuilding a specific Release will cause it to be deployed to each environment in the PTL as part of the release pipeline.

This will impact anyone currently testing or deploying work to the PTL environments during working hours.

As a courtesy, please rebuild and deploy services out side of office hours, and THEN REBUILD AND DEPLOY TO INT ONLY the most recent build, so that the PTL environments are back to the state they were in ready for tomorrows work day.

API	Version	Build
Monitoring Service Discovery Service	v1.0.22-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25554&view=results
Identity Service	v1.0.797-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24869&view=results
NHS App	v1.0.84-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24871&view=results

1701 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

API	Version	Build
PDS	v1.2.699-beta	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24719&view=results
Syncwrap	v1.0.100-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24874&view=results

Verify that each service in the above table is present inthe Prod Grafanan API Moniotring dashboard: https://grafana.prod.api.platform.nhs.uk/d/ZKp1gjsGk/api-status?orgId=1&refresh=5m

---

9.3.4.2 Revised Plan - Thursday 4th March

Deploy to Prod the following services. Verify that each service has registered and appeared in the Prod Grafana.

NOTE: Services MUST be rebuilt during out-of-office hours, as rebuilding a specific Release will cause it to be deployed to each environment in the PTL as part of the release pipeline.

This will impact anyone currently testing or deploying work to the PTL environments during working hours.

As a courtesy, please rebuild and deploy services out side of office hours, and THEN REBUILD AND DEPLOY TO INT ONLY the most recent build, so that the PTL environments are back to the state they were in ready for tomorrows work day.

API	Version
Monitoring Service Discovery Service	v1.0.22-alpha
Identity Service	v1.0.797-alpha
Ambulance Analytics API	v1.0.290-alpha
NHS App	v1.0.84-alpha
PDS	v1.2.699-beta

API	Version
Syncwrap	v1.0.100-alpha

Verify that each service in the above table is present inthe Prod Grafanan API Moniotring dashboard: https://grafana.prod.api.platform.nhs.uk/d/ZKp1gjsGk/api-status?orgId=1&refresh=5m

---

9.3.4.3 Revised Plan - Tuesday 2nd March

Deploy to Prod the following services. Verify that each service has registered and appeared in the Prod Grafana.

API	Version	Release Link
Monitoring Service Discovery Service	v1.0.22-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25225&view=results
Identity Service	v1.0.797-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25251&view=results
Ambulance Analytics API	v1.0.290-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24875&view=results
NHS App	v1.0.84-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25257&view=results
PDS	v1.2.699-beta	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25260&view=results
Syncwrap	v1.0.100-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25261&view=results

Verify that each service in the above table is present inthe PRod Grafanan API Moniotring dashboard.

9.3.4.4 Original Plan Summary

Merge APIM Utils changes into edge

- waiting for confirmation on api key is present for prod env for status endpoints ✅ Confirmed by Matt Mercer
- waiting for Matt Mercer to confirm all the required pieces are present ✅
- grafana dashboard is pointing to prod prometheus ✅

9.3.4.4.1 RFC: 30025111 - Release monitoring service discovery service into prod

- KVM in Prod for monitoring-service-discovery
    - Add to KVM "monitoring-sd-service" key "entries" = value is "{}" (empty JSON struct)
- Approve Release Monitoring-Service-Discovery into Prod
    - Release: v1.0.22-alpha: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=24725&view=results
    - Previous Release incase of rollback: v1.0.19-alpha

9.3.4.4.1.1 Rebuild and Release existing Prod services

Rebuilding to pick up changes to APIM Utils.

- Rebuild **current** live release **Identity Service v1.0.797-alpha**
    - Release to prod
    - verify in grafana dashboard for prod env that AMB has been registered and is being monitored
- Rebuild **current** live release **APIM Ambulance v1.0.290-alpha**
    - Release to prod
    - verify in grafana dashboard for prod env that AMB has been registered and is being monitored
- Rebuild **current** live release **APIM NHS-App v1.0.84-alpha**
    - Release to prod
    - verify in grafana dashboard for prod env that AMB has been registered and is being monitored
- Rebuild **current** live release **APIM PDS**
    - Release to prod
    - verify in grafana dashboard for prod env that AMB has been registered and is being monitored
- Rebuild **current** live release **APIM Syncwrap**
    - Release to prod
    - verify in grafana dashboard for prod env that AMB has been registered and is being monitored

9.3.4.4.1.2 Post-Release (during office hours - not as part of the evening release)

For the following services in PTL Environment:

- Identity Service
- APIM Ambulance
- APIM NHS-App
- APIM PDS
- APIM Syncwrap

Do the following:

- verify in grafana dashboard for prod env that AMB has been registered and is being monitored
- remove _ping and _status from prod from PTL env
    - see Adding status monitoring to a service (see page 1065)

9.3.4.4.2 RFC: 30025064

Deploy v1.2.699-beta PDS FHIR API into production

9.3.5 RFC 30025379: APIM Identity Service Updates - v1.0.859-alpha

As discussed in today's meeting here are the steps to follow in order to implement the following tickets in the identity service API:

- AMB-514[1702]
- AMB-548[1703]
- APM-1750[1704]
- AMB-549[1705]

Remember to follow KOP-018 Releasing changes to Production[1706] when deploying these changes!

1702 https://nhsd-jira.digital.nhs.uk/browse/AMB-514
1703 https://nhsd-jira.digital.nhs.uk/browse/AMB-548
1704 https://nhsd-jira.digital.nhs.uk/browse/APM-1750
1705 https://nhsd-jira.digital.nhs.uk/browse/AMB-549
1706 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.5.1 Release Plan - Tuesday 16th March

API	Version	Build Link
Identity Service	v1.0.859-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25566&view=results

9.3.5.2 Release Process

- Click the following link for the specific build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=25566&view=results
- Click "Run New"
- Ensure "Branch/Tag" is set to "v1.0.859-alpha"
- Click "Run"
- Once the build complete, the Identity-Service-Release pipeline will begin
- Once the release pipeline completed the PTL and INT environment deploys, the Release pipeline will wait for manual approval before deploying to Prod
- Manually approve the deployment to Prod
- Verify the Identity Service _ping endpoint in the Prod Grafana dashboard: https://grafana.prod.api.platform.nhs.uk/d/-yYVfaUMk/api-status?viewPanel=6&orgId=1&refresh=5m

9.3.6 RFC 30025426 - Immunisation History API Updates

9.3.6.1 Tickets in this release

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
APM-2068[1707] - Token exchange / integration DONE	Token Exchange	Low	Low	Low

1707 https://nhsd-jira.digital.nhs.uk/browse/APM-2068?src=confmacro

9.3.6.2 Release Procedure

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=32719&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.6.3 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Build and deploy immunisation-history as at v1.0.322-alpha
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.3.7 RFC 30025432: APIM Ambulance Data Submissions API Updates - v1.0.330-alpha

This Release Plan is to cover the deployment of the three Ambulance tickets that introduce breaking changes to the YAS and NWAS clients. The goal is to combine the deployment of these three ticketas into a single release, co-ordinated with YAS and NWAs in order to minimise the impact of the breaking changes.

Remember to follow KOP-018 Releasing changes to Production[1708] when deploying these changes!

1708 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.7.1 Status of tickets to be deployed

Ticket	Name	Code Reviewed	SA Reviewed	Merged to Master	Release Version
AMB-494[1709]	Rename the / submission endpoin	✅	✅	✅	
AMB-507[1710]	Migrate to Identity Services Auth	✅	✅	✅	
AMB-552[1711]	Rename Base URL	✅	✅	✅	

9.3.7.2 RFC Change Request Details

Release Version to be deployed: v1.0.330-alpha

9.3.7.3 Manual Changes to be applied

AMB-494: None

AMB-507: None

AMB-552: None

9.3.7.4 Dependencies

Any other RFC Change Requests this release plan is dependent on? none

9.3.7.5 Release Plan Summary

How to deploy the RFC, in clear steps:

The v1.0.330-alpha release has been rebuilt to pick up the latest changes to APIM Utils, and successfully promoted through the PTL environments. The Release is now awaiting deployment to Production.

1709 https://nhsd-jira.digital.nhs.uk/browse/AMB-494
1710 https://nhsd-jira.digital.nhs.uk/browse/AMB-507
1711 https://nhsd-jira.digital.nhs.uk/browse/AMB-552

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=26897&view=results
- Click "Review" to approve the deployment of v1.0.330-alpha into the Prod environment
- The deployment should be successful (smoke tests are executed as part of the deployment process to prod)
- Ensure that Ambulance API is green for _ping and _status in the Prod Grafana monitoring: https://grafana.prod.api.platform.nhs.uk/d/-yYVfaUMk/api-status?orgId=1&refresh=5m

9.3.8 RFC 30025529: APIM EPS Updates

9.3.8.1 Tickets

The Release implements the changes in the following tickets:

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1712] when deploying these changes!

9.3.8.2 Release Plan - Wednesday 7th April

API	Version	Build Link

9.3.8.3 Release Process

9.3.9 RFC 30025588: APIM Apigee Config - Update Production Certificates

9.3.9.1 Tickets

The Release implements the changes in the following tickets:

1712 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

⚠ Remember to follow KOP-018 Releasing changes to Production[1713] when deploying these changes!

9.3.9.2 Release Plan - Tuesday 23rd March

API	Version	Build Link

9.3.9.3 Release Process

9.3.10 RFC 30025611: APIM - Remove API Platform caveat for limiting the number of connecting parties

9.3.10.1 Tickets

The Release implements the changes in the following tickets:

⚠ Remember to follow KOP-018 Releasing changes to Production[1714] when deploying these changes!

1713 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

1714 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.10.2 Release Plan - Tuesday 23rd March

API	Version	Build Link

9.3.10.3 Release Process

9.3.11 RFC 30025620: APIM - Remove API Platform caveat for TPS rate limit of 5 TPS

9.3.11.1 Tickets

The Release implements the changes in the following tickets:

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1715] when deploying these changes!

9.3.11.2 Release Plan - Tuesday 23rd March

API	Version	Build Link

1715 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.11.3 Release Process

9.3.12 RFC 30025723: APIM Identity Service Updates

9.3.12.1 Tickets

The Release implements the changes in the following tickets:

- APM-1572[1716]
- APM-1750[1717]
- APM-1876[1718]
- APMSPII-620[1719]
- AMB-514[1720]
- AMB-517[1721]
- AMB-519[1722]
- AMB-520[1723]
- AMB-548[1724]
- AMB-549[1725]
- AMB-558[1726]
- AMB-563[1727]
- AMB-574[1728]
- AMB-582[1729]

1716 https://nhsd-jira.digital.nhs.uk/browse/APM-1572
1717 https://nhsd-jira.digital.nhs.uk/browse/APM-1750
1718 https://nhsd-jira.digital.nhs.uk/browse/APM-1876
1719 https://nhsd-jira.digital.nhs.uk/browse/APMSPII-620
1720 https://nhsd-jira.digital.nhs.uk/browse/AMB-514
1721 https://nhsd-jira.digital.nhs.uk/browse/AMB-517
1722 https://nhsd-jira.digital.nhs.uk/browse/AMB-519
1723 https://nhsd-jira.digital.nhs.uk/browse/AMB-520
1724 https://nhsd-jira.digital.nhs.uk/browse/AMB-548
1725 https://nhsd-jira.digital.nhs.uk/browse/AMB-549
1726 https://nhsd-jira.digital.nhs.uk/browse/AMB-558
1727 https://nhsd-jira.digital.nhs.uk/browse/AMB-563
1728 https://nhsd-jira.digital.nhs.uk/browse/AMB-574
1729 https://nhsd-jira.digital.nhs.uk/browse/AMB-582

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1730] when deploying these changes!

9.3.12.2 Release Plan - Tuesday 30th March

API	Version	Release Link
Identity Service API	v1.0.946-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=28479&view=results

9.3.13 RFC 30025798: Splunk logging improvements and Changes to support Identity Service

9.3.13.1 Tickets

APM-1943[1731] - Imm Hist API - User identity token check DONE

AMB-605[1732] - Modify Token Exchange flow to incorporate NHS login DONE

9.3.13.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=29469&view=results

9.3.13.3 Instructions

1. Go to prepared release
2. Give prod approval
3. Check apigee diffs - abort if changes are not as expected (shared flows, kvms, target servers)
4. Give apigee approval

1730 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1731 https://nhsd-jira.digital.nhs.uk/browse/APM-1943?src=confmacro
1732 https://nhsd-jira.digital.nhs.uk/browse/AMB-605?src=confmacro

5. Allow deploy to run through
6. If deploy succeeds:
    a. Monitor health dashboard for 15 mins and observe error rates
    b. Check splunk logs for presence of correlation ID
    c. Check splunk logs to make sure logs are still coming through
7. If deploy fails:
    a. Rollback

9.3.14 RFC 30025803: APIM PDS Base URL Deprecation Updates

9.3.14.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-741[1733]

9.3.14.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=29149&view=results

9.3.14.3 Release Process

1. Go to prepared release
2. Give prod approval
3. Allow deploy to run through
4. If deploy succeeds:
    a. Check splunk logs to make sure logs are still coming through using new URL
    b. **MANUALLY** undeploy the old URL from envs in apigee (both can be deployed simultaneously, need to deprecate old base URL).
    c. Check splunk logs to make sure requests using old base URL are no longer successful.
5. If deploy fails:
    a. Rollback

9.3.15 RFC 30025874: APIM Identity Service Updates

9.3.15.1 Tickets

The Release implements the changes in the following tickets:

1733 https://nhsd-jira.digital.nhs.uk/browse/APMSPII-741

- APM-2008[1734]
- AMB-608[1735]

Uplift: https://github.com/NHSDigital/identity-service-api/compare/v1.0.946-alpha...v1.0.950-alpha

Remember to follow KOP-018 Releasing changes to Production[1736] when deploying these changes!

9.3.15.2 Release Plan - Thursday 8th April

API	Version	Release Link
Identity Service	v1.0.950-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=29259&view=results

9.3.15.3 Release Process

1. Ensure the APIM Infrastructure Release preceeding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
3. Verify via Grafana and Splunk dashboards that Identity Service is still healthy and service requests without a noticable uptick in error responses

1734 https://nhsd-jira.digital.nhs.uk/browse/APM-2008
1735 https://nhsd-jira.digital.nhs.uk/browse/AMB-608
1736 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.16 RFC 30026054: Fix Spike Arrest Error Message

9.3.16.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-627[1737]

9.3.16.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=30955&view=results

9.3.16.3 Release Process

1. Go to prepared release
2. Give prod approval
3. Allow deploy to run through
4. If deploy succeeds:
    a. Check splunk logs and wait ~15 mins to check service health
5. If deploy fails:
    a. Rollback

9.3.17 RFC 30026104: APIM Identity Service Updates

9.3.17.1 Tickets

The Release implements the changes in the following tickets:

- AMB-687[1738]
- AMB-605[1739]
- AMB-522[1740]
- AMB-664[1741]
- AMB-665[1742]

1737 https://nhsd-jira.digital.nhs.uk/secure/RapidBoard.jspa?rapidView=2266&view=detail&selectedIssue=APMSPII-627
1738 https://nhsd-jira.digital.nhs.uk/browse/AMB-687
1739 https://nhsd-jira.digital.nhs.uk/browse/AMB-605
1740 https://nhsd-jira.digital.nhs.uk/browse/AMB-522
1741 https://nhsd-jira.digital.nhs.uk/browse/AMB-664
1742 https://nhsd-jira.digital.nhs.uk/browse/AMB-665

- AMB-691[1743]
- APM-2036[1744]
- AEA-1189[1745]

Uplift: https://github.com/NHSDigital/identity-service-api/compare/v1.0.950-alpha...v1.0.1033-alpha

Remember to follow KOP-018 Releasing changes to Production[1746] when deploying these changes!

9.3.17.2 Release Plan - Tuesday 19th April

API	Version	Release Link
Identity Service	v1.0.1033-alpha	https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=30870&view=results

9.3.17.3 Release Process

1. Ensure the APIM Infrastructure Release preceeding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
3. Verify via Grafana and Splunk dashboards that Identity Service is still healthy and service requests without a noticable uptick in error responses

1743 https://nhsd-jira.digital.nhs.uk/browse/AMB-691
1744 https://nhsd-jira.digital.nhs.uk/browse/APM-2036
1745 https://nhsd-jira.digital.nhs.uk/browse/AEA-1189
1746 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.18 RFC 30026140 - API Management Infrastructure Updates

9.3.18.1 Tickets in this RFC

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
APM-1162	Various small fixes	Low	Low	Low
APM-2002	Add jwks-helper service	Low	Low	Low
AMB-673	Point dev target server for nhs-login to sandpit	Low	Low	Low
AMB-688	CIS2 Domain name updates - PTL envs, KVM variables to support	Low	Low	Low
APM-2025	APIM Slackbot improvements & fixes	Low	Low	Low
AMB-665	Add kvm for internal dev int copy	Low	Low	Low
APM-1486	Deprecate old egress pattern	Low	Low	Low
AMB-605	Add NHS Login and CIS2 target servers	Low	Low	Low
APM-1943	Add ID Token Issuer to KVM, use NHS login JWKS path, add id token issuer to kvm	Low	Low	Low

9.3.18.2 Release Procedure

1. Release messaging in line with KOP-018

2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=32347&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.18.3 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Build and deploy api-management-infrastructure as at 2021-04-08
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.3.19 RFC 30026495: APIM - Identity Service Updates

9.3.19.1 Tickets

The Release implements the changes in the following tickets:

- AMB-573[1747]
- AMB-673[1748]

Uplift: https://github.com/NHSDigital/identity-service-api/compare/v1.0.1033-alpha...v1.0.1068-alpha

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1749] when deploying these changes!

1747 https://nhsd-jira.digital.nhs.uk/browse/AMB-573
1748 https://nhsd-jira.digital.nhs.uk/browse/AMB-673
1749 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.20 RFC 30026553- API Management Infrastructure Updates

9.3.20.1 Tickets in this RFC

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
APM-2139		Low	Low	Low
APM-2110		Low	Low	Low
APM-2137		Low	Low	Low
AMB-643		Low	Low	Low
AMB-577		Low	Low	Low
AEA-1287		Low	Low	Low
DRM-441		Low	Low	Low
APM-2097		Low	Low	Low
APM-2040		Low	Low	Low
APM-2047		Low	Low	Low

9.3.20.2 Release Procedure

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=33507&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.20.3 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Build and deploy api-management-infrastructure as at 2021-04-08
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.3.21 RFC 30026822 - Infrastructure Service Update

9.3.21.1 Tickets in this RFC

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
APM-2214[1750] - covid test results - provision live tls keystore DONE	This is a high impact and high urgency RFC because it needed to get the Covid Test Results API into production.	Low	Low	Low

Notes about emergency hot fix:

- Some 504's and 502's were seen on PDS at around the time of the target server, but there were on no other proxies impacted. The 5xx's were within tolerance.

9.3.22 RFC 30027067, 30027100 & 30027153: APIM - Infrastructure Release

9.3.22.1 Tickets in this RFC

APM-1811[1751] - Apigee data backups - automate DONE

APM-2156[1752] - Swap base URL of NHSD jwks store to github pages DONE

1750 https://nhsd-jira.digital.nhs.uk/browse/APM-2214?src=confmacro
1751 https://nhsd-jira.digital.nhs.uk/browse/APM-1811?src=confmacro
1752 https://nhsd-jira.digital.nhs.uk/browse/APM-2156?src=confmacro

APM-2216[1753] - Move DAP Secrets out of Terraform control DONE

APM-2141[1754] - NHS login token exchange - feature toggle on in prod DONE

APM-1871[1755] - Re-jig the frequency of polling for prod and PTL environments DONE

9.3.22.2 Purpose of release

- Toggle on Token Exchange
- Change polling frequency for monitoring
- Create encrypted DynamoDB table for backups
- Backup Apigee apps to DynamoDB table
- Decouple secret/secret-value relationship in Terraform for the DAP Cognito client

APM-2156 is included in this release, but will have no effect on Production at this time.

9.3.22.3 Release Number:

v2021.06.11.08.48[1756]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=37271&view=results

9.3.22.4 Release Procedure

1. Release messaging in line with KOP-018 Releasing changes to Production[1757]
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36904&view=results
    a. Both items can be run in parallel
3. Observe logs for 15 mins to catch any potential issues
    a. PDS FHIR API
    b. Identity Service
    c. API Management Dashboard for all prod environment
4. Success/Failure messaging in line with KOP-018

1753 https://nhsd-jira.digital.nhs.uk/browse/APM-2216?src=confmacro
1754 https://nhsd-jira.digital.nhs.uk/browse/APM-2141?src=confmacro
1755 https://nhsd-jira.digital.nhs.uk/browse/APM-1871?src=confmacro
1756 https://github.com/NHSDigital/api-management-infrastructure/tree/v2021.06.11.08.48
1757 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.22.4.1 Notes for APM-1811 Apigee app data back ups

Providing the terraform for creating the table in DynamoDB does not fail the Apply this should have no destructive effect on production as it will only 'GET' the app infomation from apigee and write the data in the DynamoDB table.

1. Check DynamoDB in the AWS prod account to see if 'apigee_apps' table has been created (If not checked in APM-2216 terraform changes).
    a. Check in Terraform Apply stage: module.dynamodb_tables.aws_dynamodb_table.apigee_apps has been created
2. ~~Run back-up pipeline with 'Backup prod Apigee Apps' ticked and verify success.~~
3. ~~Check up prod Apigee apps have been backed up by looking at the 'apigee_apps' table again in AWS and see if the data has been added to the table.~~

9.3.22.4.2 Notes for APM-2216 Terraform Changes

There is a risk that this change will cause the Terraform Apply stage to fail on Developer Authentication (*.auth or *.dap resource), in the event that this occurs follow this procedure to check if the environment is left in a good state:

1. Ensure that the only error in the apply output is related to the module `module.developer-authentication.*`.
2. Check DynamoDB in the AWS prod account to see if 'apigee_apps' table has been created.
3. Send the error output to Unknown User (jofi7) to make note of the issue on the ticket.
4. If this is the only error and the `apigee_apps` table has been created, the deploy has otherwise succeeded.

~~In the event that the Terraform runs successfully, ensure that the secret has a value by running:~~

```
aws secretsmanager get-secret-value --secret-id prod/cognito-pools/developer-
authentication/dap/credentials
```

9.3.22.5 Rollback Procedure

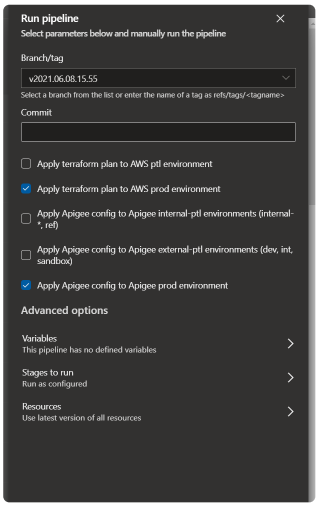
KOP-017 Rolling back to a stable release[1758]

1. Follow KOP-004 Release Process - Infrastructure[1759] for the following tags until a stable (non-fail) state is reached. You may need to rebuild the tags if the builds for them have not been retained (however there should be a fresh build of each...).

1758 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708113/KOP-017+Rolling+back+to+a+stable+release

1759 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119755126/KOP-004+Release+Process+-+Infrastructure

a. v2021.06.08.15.55
b. If (a) fails: v2021.06.08.14.33
c. If (b) fails: Call Laurence on 07944748276



9.3.23 RFC 30027083: APIM - PDS API - Sync in Production

DONE

9.3.23.1 Tickets in this RFC

APMSPII-921

APMSPII-924

APMSPII-832

APMSPII-797

APMSPII-676

APMSPII-529

9.3.23.2 Purpose of release

All changes that release sync wrap behaviour to the PDS API.

Note - we do not presently have a production partner, therefore this is **not going to break anyone** in the event it is unsuccessful.

9.3.23.3 Release Number:

`#v1.2.1043-beta+36888`

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36888&view=results

9.3.23.4 Release Procedure

1. Release messaging in line with KOP-018 Releasing changes to Production[1760]
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36888&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.23.5 Rollback Procedure

KOP-017 Rolling back to a stable release[1761]

1. Build and release API at v1.2.812-beta through to production
2. Re-run release https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36888&view=results to everything *except* production

9.3.23.6 Release prep checklist

(From Raising RFCs - Cheatsheet (see page 2091))

- [x] Determine whether you need to run an APIM or a Spine release plan - it affects all of the steps/links below, the rest of this process will refer to APIM releases
- [x] Write up a change summary and checklist in confluence in new, nested page in Release Plans[1762]
- [x] Utilise our KOPs (Key Operating Procedures) wherever possible - Key Operating Procedures[1763] - the standard implementation plans are KOP-018[1764] and KOP-019[1765]; standard rollback plans are KOP-016[1766] and KOP-017[1767]
- [x] Raise RFC in Cherwell - see How to raise a Release RFC Change Request in Cherwell[1768]
- [x] The dev/tech lead should attend the "war room" session (organised by Steven Walton , usually these are Tues/Thurs 12pm) to explain what needs to be done
- [x] Ensure RFC has been fully signed off - and marked off as signed off on Release Schedule (see page 1390)

1760 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1761 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708113/KOP-017+Rolling+back+to+a+stable+release
1762 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1763 https://nhsd-confluence.digital.nhs.uk/display/APM/Key+Operating+Procedures
1764 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1765 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1766 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1767 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1768 https://nhsd-confluence.digital.nhs.uk/display/APM/How+to+raise+a+Release+RFC+Change+Request+in+Cherwell

9.3.24 RFC 30027232: APIM Run script in prod to update JWKS resource URLs for applications

9.3.24.1 Tickets in this RFC

APM-2156[1769] - Swap base URL of NHSD jwks store to github pages DONE

9.3.24.2 Purpose of RFC

This is an RFC to run a script in the *api-management-infrastructure* repo to swap the *jwks-resource-url* custom attributes on apps that have them from the *https://raw.githubusercontent.com/NHSDigital/identity-service-jwks/** url to the *https://nhsdigital.github.io/identity-service-jwks/** url format. This is because the new URL uses GitHub pages which has much higher availability and will improve resilience for production applications.

9.3.24.3 Release Number:

No release needed. This is a script to be run against production applications and requires an RFC for this reason.

9.3.24.4 Release Procedure

1. Release messaging in line with KOP-018 Releasing changes to Production[1770]
2. Back up Apigee apps for the 'prod' organisation by running the *APIM-Apigee-App-Backup* pipeline. See the 'Backing up Apigee Apps' section of KOP-015 Restore Apigee App from Back-up[1771] for how to run.
3. In the *api-management-infrastructure* repo cd into *scripts/apigee-state-scripts*
4. Run following command with your apigee access token: *poetry run python update-jwks-attributes.py "nhsd-prod" <APIGEE_ACCESS_TOKEN>*
5. In the *prod* Apigee organisation check for some example cases of apps with a *jwks-resource-url* custom attribute to see if it has the format *https://nhsdigital.github.io/identity-service-jwks/**
6. If the the attribute has the old format of *https://raw.githubusercontent.com/NHSDigital/identity-service-jwks/** the script has failed. This doesn't mean there will be negative effect on production apps but should be reported as a failure.

1769 https://nhsd-jira.digital.nhs.uk/browse/APM-2156?src=confmacro
1770 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1771 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708706/KOP-015+Restore+Apigee+App+from+Back-up

7. Observe logs for 15 mins to catch any potential issues
8. If any potential issues with an app follow the rollback procedure
9. Success/Failure messaging in line with KOP-018

9.3.24.5 Rollback Procedure

- If anything goes wrong with an app you can restore an App by using the *APIM-Apigee-App-Restore* pipeline using the App Id. See the 'Restoring an Apigee app or all apps from a Back-up' section of KOP-015 Restore Apigee App from Back-up[1772] for how to run.
- If there is a worst case scenario where all apps have been effected the pipeline can also be run for all apps in the prod organisation. This will have some risk outcomes see 'Considerations' section of KOP-015 Restore Apigee App from Back-up[1773].

9.3.25 RFC 30027332 - Digital Onboarding Service Release

9.3.25.1 Tickets in this RFC

key	summary	type	status

Jira project doesn't exist or you don't have permission to view it.
View these issues in Jira[1774]

Note - tickets above are in the Turnstile project and backlog. Please ask for permission from Turnstile to access them if you are unable to.

9.3.25.2 Purpose of release

- Push latest application code for Digital Onboarding Service to production to enable release of application code

1772 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708706/KOP-015+Restore+Apigee+App+from+Back-up
1773 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708706/KOP-015+Restore+Apigee+App+from+Back-up
1774 https://nhsd-jira.digital.nhs.uk/secure/IssueNavigator.jspa?reset=true&jqlQuery=project+%3D+DRM+AND+issuetype+%3D+Epic+AND+fixVersion+%3D+%22DOS+%7C+MVP%22+&src=confmacro

9.3.25.3 Release Number:

v2021.06.22.10.58[1775]

Prod release pipeline: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=37953&view=results

9.3.25.4 Release Procedure

1. Release messaging in line with KOP-018 Releasing changes to Production[1776]
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=37692&view=results
    a. Only a terraform apply to prod is required, no Apigee changes
    b. Show plan should create a new task definition for drm-dap and drm-clamav with image name 37689-sha86b8df2[1777].
3. Observe logs for 15 mins to catch any potential issues
    a. API Management Dashboard for all prod environment
    b. Digital Onboarding Service containers (dap-staging, dap-daq, dap-prod)
4. Success/Failure messaging in line with KOP-018

9.3.25.5 Rollback Procedure

KOP-017 Rolling back to a stable release[1778]

1. Follow KOP-004 Release Process - Infrastructure[1779] for the following tags until a stable (non-fail) state is reached. You may need to rebuild the tags if the builds for them have not been retained (however there should be a fresh build of each...).
    a. v2021.06.11.08.48

9.3.26 RFC 30027481 PDS API - Sync Wrap PROD

Status

CANCELLED

1775 https://github.com/NHSDigital/api-management-infrastructure/tree/v2021.06.22.10.58
1776 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1777 https://eu-west-2.console.aws.amazon.com/ecr/repositories/private/958002497996/drm-dap/image/sha256:e2dd10bafb635ddac9a201ab49a3e6192d195eeb5db96d6f33f494c8ce86ee4b/details/?region=eu-west-2
1778 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1779 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-004+Release+Process+-+Infrastructure

9.3.26.1 Cell G Checklist

- [x] Pipelines checked to ensure build is successful, including all environments
- [x] Are you running an APIM, or a Spine release plan - it affects all of the steps/links below, the rest of this process will refer to APIM releases
- [x] Change summary written up, and this checklist has been copied in to a new, nested page in Release Plans[1780]
- [x] KOPs (Key Operating Procedures[1781]) used wherever possible - the standard implementation plans are KOP-018[1782] and KOP-019[1783]; standard rollback plans are KOP-016[1784] and KOP-017[1785]
- [x] RFC raised in Cherwell - see How to raise a Release RFC Change Request in Cherwell[1786]
- [x] Release has been scheduled within the Release Plans[1787] page
- [ ] The dev/tech lead has attended the "war room" session (organised by Steven Walton , usually these are Tues/Thurs 12pm) to explain what needs to be done
- [ ] Ensure RFC has been fully signed off - and marked off as signed off on Release Plans[1788]

9.3.26.2 Release Number

v1.2.1043-beta+36888

9.3.26.3 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36888&view=results

9.3.26.3.1 Release Process

1. Go to prepared release
2. Give prod approval
3. Allow deploy to run through
4. If deploy succeeds:
    a. Check splunk logs and wait ~15 mins to check service health
5. If deploy fails:

1780 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1781 https://nhsd-confluence.digital.nhs.uk/display/APM/Key+Operating+Procedures
1782 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1783 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1784 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1785 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1786 https://nhsd-confluence.digital.nhs.uk/display/APM/How+to+raise+a+Release+RFC+Change+Request+in+Cherwell
1787 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1788 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans

a. Rollback

9.3.27 RFC 30027510 - API Management Infrastructure - Identity Service KVM

9.3.27.1 Tickets in this RFC

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
APMSPII-527	Identity Service Prod KVM Update Application	Low	Low	Low

This is a re-build and re-release of the last Infrastructure release to go into productions (v2021.06.23.15.17) to apply an updated value in AWS Secrets manager.

- Build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38799&view=results
- Release: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38802&view=results

9.3.27.2 Release Procedure

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38802&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.27.3 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Redeploy api-management-infrastructure Release https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38270&view=results
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.3.28 RFC 30027613: APIM PDS Production Quota Update

9.3.28.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-798[1789]

9.3.28.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results

9.3.28.3 Release Process

1. Go to prepared release
2. Give prod approval

9.3.28.3.1 Post Release Checks

1. If deploy succeeds:
    a. Check splunk logs to make sure logs are still coming through
    b. Log into Apigee Edge UI and verify that both API Products for PDS FHIR API have the quota set to 14400

9.3.28.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=36888&view=results
- Redploy *last successful non-production release* (to all envs except production)

9.3.29 RFC 30027655: APIM PDS Disable Async Update Mode

9.3.29.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-1043

1789 https://nhsd-jira.digital.nhs.uk/browse/APMSPII-798

9.3.29.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results

9.3.29.3 Release Process

1. Go to prepared release
2. Give prod approval

9.3.29.3.1 Post Release Checks

1. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.29.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results
- Redploy *last successful non-production release* (to all envs except production)

9.3.30 RFC 30027682: APIM Infrastructure Updates to enable prod Spine connectivity

9.3.30.1 Tickets

The Release implements the changes in the following tickets:

- AEA-1517
- APM-2351
- AMB-751
- APM-2016
- ERSSUP-55868
- APM-2337
- APM-2112
- AMB-722

9.3.30.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39726&view=results
(v2021.07.06.11.20)

9.3.30.3 Release Process

1. Go to prepared release
2. Give prod approval

9.3.30.3.1 Post Release Checks

1. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.30.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38270&view=results (v2021.06.23.15.17)
- Redeploy *last successful non-production release* (to all envs except production)

9.3.31 RFC 30027868 - Verifying URID for user updates

9.3.31.1 High level checklist

- [x] Pipelines checked to ensure build is successful, including all environments
- [x] Are you running an APIM, or a Spine release plan - it affects all of the steps/links below, the rest of this process will refer to APIM releases
- [x] Change summary written up,
- [x] and this checklist has been copied in to a new, nested page in Release Plans[1790]
- [x] KOPs (Key Operating Procedures[1791]) used wherever possible - the standard implementation plans are KOP-018[1792] and KOP-019[1793]; standard rollback plans are KOP-016[1794] and KOP-017[1795]
- [x] RFC raised in Cherwell - see How to raise a Release RFC Change Request in Cherwell[1796]
- [x] Release has been scheduled within the Release Plans[1797] page
- [x] The dev/tech lead has attended the "war room" session (organised by Steven Walton , usually these are Tues/Thurs 12pm) to explain what needs to be done
- [ ] Ensure RFC has been fully signed off - and marked off as signed off on Release Plans[1798]

1790 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1791 https://nhsd-confluence.digital.nhs.uk/display/APM/Key+Operating+Procedures
1792 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1793 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1794 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1795 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1796 https://nhsd-confluence.digital.nhs.uk/display/APM/How+to+raise+a+Release+RFC+Change+Request+in+Cherwell
1797 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1798 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans

9.3.31.1.1 Tickets

RFC created by Unknown User (dohu3)

The Release implements the changes in the following tickets for PDS FHIR API:

- the change: AMB-612[1799] - Default user role - add to PDS API DONE

Prepared Release

Release Tag: **v1.2.1154-beta**

Link to Azure devops: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=41508&view=results

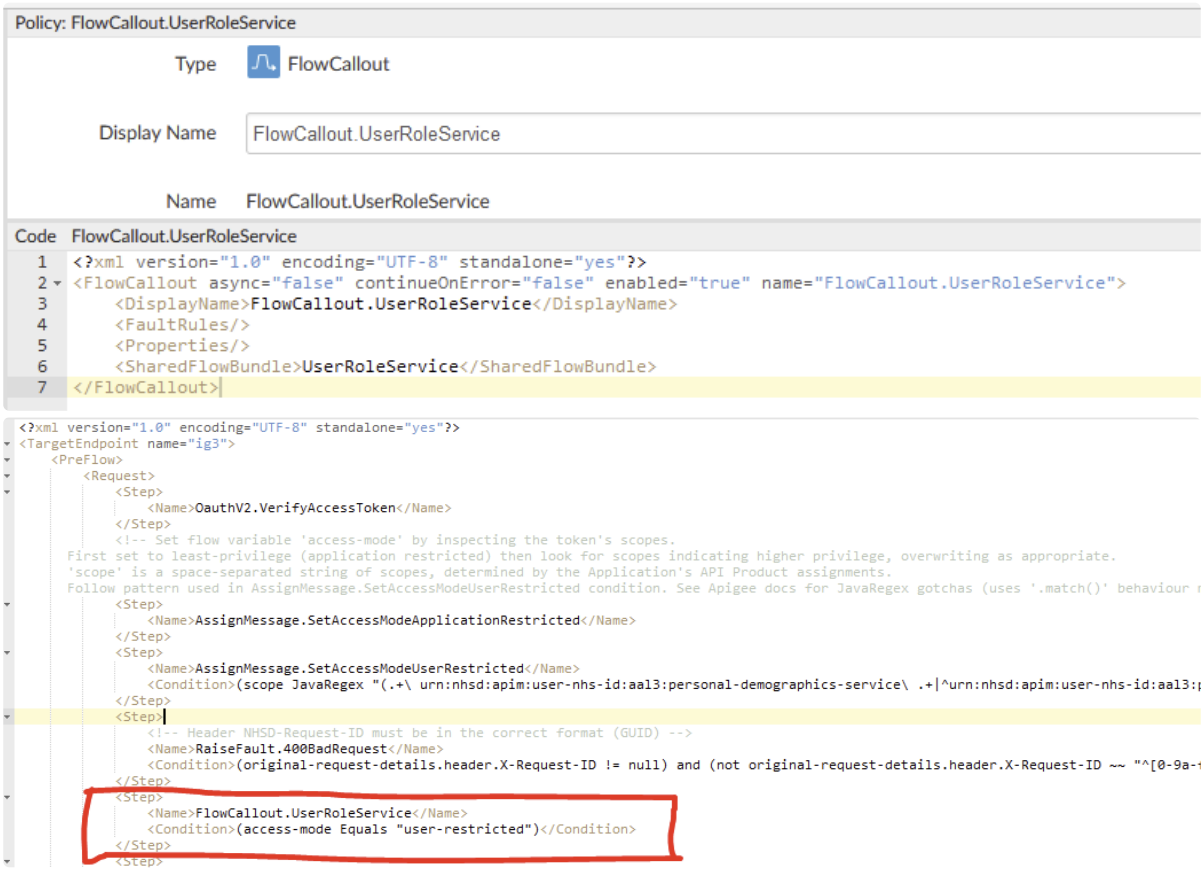
9.3.31.2 Release Process

1. Ensure that RFC 30028033: APIM Identity Service Updates (see page 1531) (a. IdentityService changes: v1.0.1156-alpha b.Infrastructure changes: v2021.07.20.09.49[1800]) has been succesfully released and deployed, prior to releasing this
2. Click the Release Link[1801] approve the manual step to deploy the release to prod (see KOP-019 Release process - Individual API[1802])

9.3.31.2.1 Post Release Checks

1. Log in to Apigee portal and check that the `UserRoleService` Shared Flow exists,
2. still on Apigee, check Personal Demographics proxy production environment - target endpoints - ig3 contains the preflow step "FlowCallout.UserRoleService" and Policy: "FlowCallout.UserRoleService" exists

1799 https://nhsd-jira.digital.nhs.uk/browse/AMB-612?src=confmacro
1800 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.07.20.09.49
1801 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=41508&view=results
1802 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169715007/KOP-019+Release+process+-+Individual+API



3. Monitor Grafana Prod dashboard[1803] to make sure all services are still running as expected.
4. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
5. Check production traffic for elevated levels HTTP Status code 400 BAD REQUEST - indicator that this is being caused by the new URID validation shared flow

9.3.31.3 Roll-Back

- Redeploy:
  - Release Tag: **v1.2.1106-beta**
  - link to Azure devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results
- Redeploy *last successful non-production releases* (to all envs except production)
- We will follow our KOP for rolling back: https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release

1803 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.32 RFC 30027924 - Digital Onboarding Service - High Volume SMTP fix

9.3.32.1 Tickets

RFC created by **Steven Walton.** The Release implements the changes in the following tickets:

- DRM-1167
- DRM-1019

Note - to access the above you need access to the Digital Onboarding Service JIRA project. Please ask Turnstile for access if required.

9.3.32.2 Prepared Release

Release Tag: **v2021.07.14.14.32**

https://dev.azure.com/NHSD-APIM/API Platform/_build/results?buildId=40419[1804]

Note only required to deploy to Prod - no Apigee changes required

9.3.32.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

   Check the following:

   New ECS Security Group added to DAP containers that allows access to port 80 on CIDR blocks 104.18.21.226/32 104.18.20.226/32
   Task definitions for dap-daq, dap-staging, and dap-prod are of drm-dap:40259 and drm-clamav:40259 respectively.

9.3.32.3.1 Post Release Checks

1. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.32.4 Roll-Back

- Redeploy:
  - Release Tag: **v2021.07.06.11.20**

1804 https://dev.azure.com/NHSD-APIM/API Platform/_build/results?buildId=40419

    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=38270&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.33 RFC 30028033: APIM Identity Service Updates

9.3.33.1 Tickets

The Release implements the changes in the following tickets:

- AMB-612[1805]

- APMSPII-1084 - Identity Service: Add issuer attribute to access tokens

> ⚠ Remember to follow KOP-018 Releasing changes to Production[1806] when deploying these changes!

9.3.33.2 Release Plan - Tuesday 30th March

API	Version	Release Link
Identity Service API	v1.0.1156-alpha	
Infrastructure	v2021.07.20.09.49[1807]	

9.3.34 RFC 30028203 - Digital Onboarding Service - GA/Hotjar fixes

9.3.34.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

1805 https://nhsd-jira.digital.nhs.uk/browse/AMB-612
1806 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1807 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.07.20.09.49

- DRM-1175
- DRM-1176

Please note to access the above you will need access to the Digital Onboarding Service project. Please contact Turnstile for access.

9.3.34.2 Prepared Release

Release Tag: **v2021.07.30.13.48**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=43096&view=results

No Apigee changes required

9.3.34.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

9.3.34.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1808]
2. Monitor APIM Dashboard generally ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.34.4 Roll-Back

- Redeploy:
    - Release Tag: v2021.07.20.09.49
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=42612&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.35 RFC 30028231 - PDS FHIR API Conformance Resource URL Change

9.3.35.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-575

1808 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.35.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=43312&view=results

9.3.35.3 Release Process

1. Go to prepared release
2. Give prod approval

9.3.35.3.1 Post Release Checks

1. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.35.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=41508&view=results
- Redploy *last successful non-production release* (to all envs except production)

9.3.36 RFC 30028313 - Digital Onboarding Service - GA/Hotjar fixes

9.3.36.1 Tickets

RFC created by **Ben Fozard.** The Release implements the changes in the following tickets:

- DRM-1175
- DRM-1176

Please note to access the above you will need access to the Digital Onboarding Service project. Please contact Turnstile for access.

9.3.36.2 Prepared Release

Release Tag: **v2021.08.05.12.20**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44021&view=results

No Apigee changes required

Prepared Build for tonights release: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44376&view=results

9.3.36.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

9.3.36.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1809]
2. Monitor APIM Dashboard generally ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

> RELEASE NOTE: During release, the Prod Grafana dashboard was out of action with 503 errors across the board for a good 10mins - panic inducing!

9.3.36.4 Roll-Back

- Redeploy:
  - Release Tag: v2021.07.21.15.27
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=43714&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.37 RFC 30028333 - API Management Infrastructure Updates

9.3.37.1 Release Version: v2021.07.29.10.39

9.3.37.2 Tickets in this RFC

Ticket Name	Description	Outage Impact	Outage Likelihood	Risk
AMB-776[1810]	Identity data in Authorisation Service logging to Splunk	Low	Low	Low

1809 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1810 https://nhsd-jira.digital.nhs.uk/browse/AMB-776

Apigee changes only.

9.3.37.3 Release Procedure

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=42814&view=results
    a. Click "Run New"
    b. Untick checkbox.
    c. Tick "Apply Apigee config to Apigee prod environment"
    d. Click "Run"
    e. Review and approve release to Prod
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.3.37.4 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Build and deploy api-management-infrastructure:
    a. AWS: v2021.07.21.15.27
    b. Apigee: v2021.07.20.09.49
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.3.38 RFC 30028351 - APIM Covid-19 Test Results API - Token Exchange

9.3.38.1 Tickets

RFC created by Unknown User (joba20) (on behalf of Unknown User (madu59)) The Release implements the changes in the following tickets:

- https://bluesquirrel.atlassian.net/browse/HTC-3804
- https://bluesquirrel.atlassian.net/browse/HTC-3803[1811]

Lower risk change as this is adding configuration to enable future use of Token Exchange by switching over API Consumer App Products.

1811 https://bluesquirrel.atlassian.net/browse/HTC-3803?src=confmacro

9.3.38.2 Prepared Release

Release Tag: v1.0.169-alpha+44589[1812]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=266

9.3.38.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.38.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1813]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.38.4 Roll-Back

- Redeploy:
    - Release Tag: <**eg. v2021.06.23.15.17** >
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=266
- Redeploy *last successful non-production release* (to all envs except production)

9.3.38.5 Results Service 8.9.3 - Release Notes

Release Description	**Natural Immunity Letter support (Part 1 of 2). This release will introduce support for token exchange authentication flow. Current COVID-19 Test Results API exposed over API Management platform is using tactical authentication flow not allowing to implement "Unattended access" that's why migration to strategical solution using Token Exchange flow is required.**
Release Version	8.9.3
Release Date	12.08.2021 TBA

1812 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=44589
1813 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

Release Description	**Natural Immunity Letter support (Part 1 of 2). This release will introduce support for token exchange authentication flow. Current COVID-19 Test Results API exposed over API Management platform is using tactical authentication flow not allowing to implement "Unattended access" that's why migration to strategical solution using Token Exchange flow is required.**
Status	in progress
Reason for change	The COVID Pass/Certificate application will need to switch to stategical authentication flow securing API exposing COVID 19 test results. This requires a change to the Test Results API to support both token exchange and allow to start development on "Unattended access".
Non-functional impact	None
Testing Summary	Testing will be splitted into two parts: • E2E testing with support from NetCompany (owners of COVID pass app) • Internal testing performed by development team

Tickets:

HTC-3803 R1 - Add Token Exchange OAuth flow to Apigee Proxy

HTC-3804 R1 - FHIR API - Change of Access Headers

9.3.39 RFC 30028396 Apigee application restricted for immunisation history api

9.3.39.1 Tickets

RFC created by Tim Wang The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/DALA-380

9.3.39.2 Prepared Release

Release Tag: v1.0.400-alpha+44010

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44011&view=results

9.3.39.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.39.3.1 Post Release Checks

1. Monitor Splunk Dashboard[1814]
2. Monitor APIM Dashboard for Immunisation History API ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.39.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.362-alpha+40875
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=40875&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.40 RFC 30028431 - TPS raised to 270 - PDS API

9.3.40.1 Tickets

RFC created by Alex Lord

The Release implements the changes in the following tickets:

- APMSPII-1127

9.3.40.2 Prepared Release

Release Tag: **v1.2.1309-beta**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44686&view=results

9.3.40.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link[1815] approve the manual step to deploy the release to prod (see KOP-019 Release process - Individual API[1816])

1814 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/health?form.index=summary_apm_prod&form.proxy=immunisation-history-prod&form.app=*&form.env=prod

1815 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=41508&view=results

1816 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API

9.3.40.3.1 Post Release Checks

1. Check PDS TPS limit is set at 270
2. Monitor Grafana Prod dashboard[1817] to make sure all services are still running as expected.
3. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.40.4 Roll-Back

- Redeploy:
    - link to Azure devops results: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=43312&view=results
    - Release Tag: **v1.2.1207-beta**
- Redeploy *last successful non-production release* (to all envs except production)
- We will follow our KOP for rolling back: https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release

9.3.40.5 High level checklist

- [ ] Pipelines checked to ensure build is successful, including all environments
- [x] Are you running an APIM, or a Spine release plan - it affects all of the steps/links below, the rest of this process will refer to APIM releases
- [x] Change summary written up, and this checklist has been copied in to a new, nested page in Release Plans[1818]
- [x] KOPs (Key Operating Procedures[1819]) used wherever possible - the standard implementation plans are KOP-018[1820] and KOP-019[1821]; standard rollback plans are KOP-016[1822] and KOP-017[1823]
- [x] RFC raised in Cherwell - see How to raise a Release RFC Change Request in Cherwell[1824]
- [x] Release has been scheduled within the Release Schedule (see page 1390) page
- [ ] The dev/tech lead has attended the "war room" session (organised by Steven Walton , usually these are Tues/Thurs 12pm) to explain what needs to be done
- [ ] Ensure RFC has been fully signed off - and marked off as signed off on Release Plans[1825]

1817 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1818 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
1819 https://nhsd-confluence.digital.nhs.uk/display/APM/Key+Operating+Procedures
1820 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
1821 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
1822 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
1823 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
1824 https://nhsd-confluence.digital.nhs.uk/display/APM/How+to+raise+a+Release+RFC+Change+Request+in+Cherwell
1825 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans

9.3.41 RFC 30028449: APIM PDS Patient (Citizen) Access Integration

9.3.41.1 Tickets

The Release implements the changes in the following tickets:

- AMB-756[1826]

9.3.41.2 Prepared Release

Release version: v1.2.1315-beta[1827]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45153&view=results

9.3.41.3 Release Process

1. Go to prepared release
2. Give prod approval

9.3.41.3.1 Post Release Checks

1. If deploy succeeds:
    a. Check splunk logs to make sure logs are still coming through

9.3.41.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44686&view=results
- Redploy *last successful non-production release* (to all envs except production)

9.3.42 RFC 30028450 - APIM Infrastructure Changes

9.3.42.1 Tickets

RFC created by Ben Strutt. The Release implements the changes in the following tickets:

- APM-2398[1828] - Rate limiting - easily re-usable across APIs (shared flow) DONE

1826 https://nhsd-jira.digital.nhs.uk/browse/AMB-756
1827 https://github.com/NHSDigital/personal-demographics-service-api/releases/tag/v1.2.1315-beta
1828 https://nhsd-jira.digital.nhs.uk/browse/APM-2398?src=confmacro

- Small infrastructure tagging changes for AWS.

9.3.42.2 Prepared Release

Release Tag: v2021.08.13.14.07[1829]

<**Link to Azure** devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results>

9.3.42.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.42.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1830]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.42.4 Roll-Back

- Redeploy:
  - Release Tag: v2021.08.05.12.20
  - <**Link** to Azure devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.43 RFC 30028523 - Management UI SAML Testing - In hours

9.3.43.1 Tickets

RFC created by Aubyn Crawford The Release implements the changes in the following tickets:

- APM-1250[1831] - Apigee Admin SAML - Apigee, Azure configuration DONE

1829 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.08.13.14.07
1830 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1831 https://nhsd-jira.digital.nhs.uk/browse/APM-1250?src=confmacro

9.3.43.2 Prepared Release

N/A - manual Apigee config changes

9.3.43.3 Release Process

1. Login with a Zone Admin account in Apigee
2. Navigate to https://apigee.com/sso
3. In the dropdown for the Prod org select the "prod" Zone
4. Save - this turns on SAML
5. Test and diagnose SAML login
6. Follow roll-back to disable

9.3.43.3.1 Post Release Checks

N/A - just carry out testing

9.3.43.4 Roll-Back

- Login with a Zone Admin account in Apigee
- Navigate to https://apigee.com/sso and connect the "prod" Zone to the Prod Org
- In the dropdown for the Prod org select the no zone option
- Save - this disables SAML

9.3.44 RFC 30028609 - Digital Onboarding Service - disable user, resend invite, DAQ changelog, misc fixes

9.3.44.1 Tickets

RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- DOS-475
- DOS-480
- DOS-481
- DOS-482
- DOS-473

Please note to access the above you will need access to the Digital Onboarding Service project. Please contact Turnstile for access.

9.3.44.2 Prepared Release

Release Tag: **v2021.08.24.13.17**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45546&view=results

9.3.44.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.44.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1832]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.44.4 Roll-Back

- TODO - Update with rollback version
- Redeploy:
    - Release Tag: v2021.08.13.14.07[1833]
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45569&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.45 RFC 30028763 - Digital Onboarding Service Fixes

9.3.45.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/DOS-489
- https://nhsd-jira.digital.nhs.uk/browse/DOS-513

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

1832 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1833 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.08.13.14.07

9.3.45.2 Prepared Release

Release Tag: **v2021.09.02.14.15**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=46435&view=results

Note - only AWS Prod required - no Apigee

9.3.45.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.45.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1834]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.45.4 Roll-Back

- Redeploy:
    - Release Tag: **v2021.08.24.13.17**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=46129&view=results
    - Note - only AWS Prod required - no Apigee
- Redeploy *last successful non-production release* (to all envs except production)

9.3.46 RFC 30028795 - APIM Infrastructure Changes

9.3.46.1 Tickets

RFC created by MattM. The Release implements the changes in the following tickets:

- AMB-804[1835] - Identity data in Authorisation Service logging to Splunk DONE
- APM-2191[1836] - Rate limiting - convert auth service DONE

1834 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1835 https://nhsd-jira.digital.nhs.uk/browse/AMB-804?src=confmacro
1836 https://nhsd-jira.digital.nhs.uk/browse/APM-2191?src=confmacro

- APM-2450[1837] - Ship Cloudwatch Metrics / Logs to splunk DONE
- APM-2467[1838] - Investigate ECS Under Provisioned warning in Grafana after terraform deploy DONE

9.3.46.2 Prepared Release

Release Tag: v2021.09.03.14.39[1839]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=47298&view=results

9.3.46.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.46.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1840]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.46.4 Roll-Back

- Redeploy:
  - Rollback AWS:
    - to: v2021.09.02.14.15 → https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=46435&view=results
  - Rollback APIgee
    - to: v2021.08.13.14.07 → https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=46717&view=results

1837 https://nhsd-jira.digital.nhs.uk/browse/APM-2450?src=confmacro
1838 https://nhsd-jira.digital.nhs.uk/browse/APM-2467?src=confmacro
1839 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2021.09.03.14.39
1840 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.47 RFC 30028836 - Management UI Enable Azure AD SAML - In hours

9.3.47.1 Tickets

RFC created by Aubyn Crawford The Release implements the changes in the following ticket:

- APM-2348[1841] - Apigee Admin SAML - Switch over to SAML DONE

9.3.47.2 Prepared Release

N/A - manual Apigee config changes

9.3.47.3 Release Process

1. Login with a Zone Admin account in Apigee
2. Navigate to https://apigee.com/sso
3. In the dropdown for the Prod org select the "prod" Zone
4. Save - this turns on SAML
5. Test and diagnose SAML login
6. Follow roll-back to disable

9.3.47.3.1 Post Release Checks

N/A - just carry out testing

9.3.47.4 Roll-Back

- Login with a Zone Admin account in Apigee
- Navigate to https://apigee.com/sso and connect the "prod" Zone to the Prod Org
- In the dropdown for the Prod org select the no zone option
- Save - this disables SAML

1841 https://nhsd-jira.digital.nhs.uk/browse/APM-2348?src=confmacro

9.3.48 RFC 30028855 - APIM Covid-19 Test Results API - Enabling Token Exchange in the Covid Pass APIGEE app configuration

9.3.48.1 Tickets

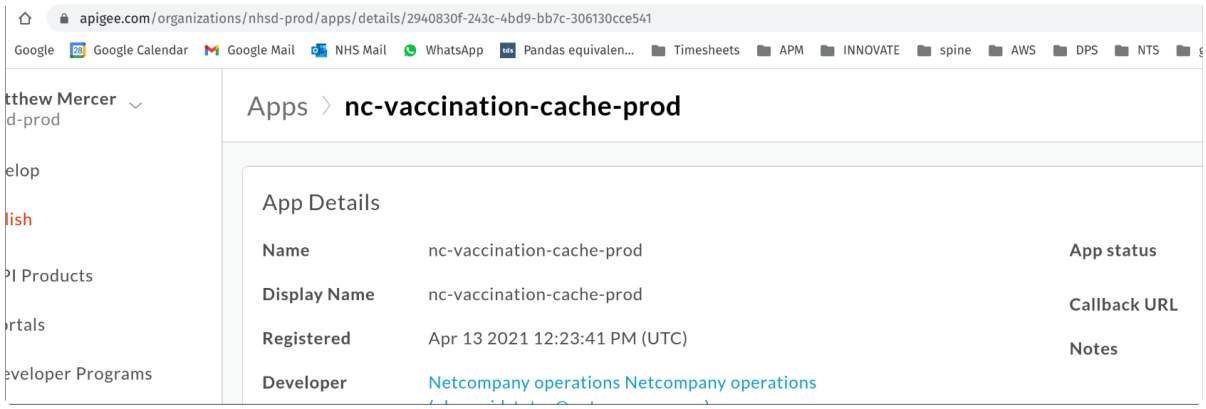
RFC created by Alex Carrie on behalf of Marcin Duszynski The Release implements the changes in the following tickets:

9.3.48.2 Prepared Release

Release Tag: v1.0.170-alpha

(orginal release) - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45615&view=results

9.3.48.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
2. Add COVID-19 Test Results API user-restricted APIGEE product to the NetCo Covid Pass application configuration (APIGEE Client configuration)
3. App ID:
4. 

9.3.48.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1842]
2. Monitor APIM Dashboard for Covid-19 Test Results API ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. nc-vaccination-cache-prod traffic received and processed successfully

1842 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.48.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.169-alpa
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=44590&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.49 RFC 30028932 - Digital Onboarding Service fixes

9.3.49.1 Tickets

RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- **DOS-533**
- **DOS-439**
- **DOS-542**
- **DOS-515**
- **DOS-514**
- **DOS-512**
- **DOS-511**
- **DOS-510**
- **DOS-449**
- **DOS-469**
- **DOS-438**

Note: to access the above tickets you will need to be a member of the Digital Onboarding Service Project. Please contact Turnstile for access.

9.3.49.2 Prepared Release

Release Tag: **v2021.09.13.16.17**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=47784&view=results

Note - only AWS Prod apply required - no Apigee.

9.3.49.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.49.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1843]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.49.4 Roll-Back

- Redeploy:
  - Release Tag: **v2021.09.02.14.15**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=46435&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.50 RFC 30028975 - APIM Covid-19 Test Results API - Remove legacy product from Covid Pass APIGEE app configuration

9.3.50.1 Description

RFC created by Alex Carrie on behalf of Marcin Duszynski

Final release for Natural Immunity Letters team containing application restricted mode and removing legacy access mode.
Legacy APIGEE product will be removed from the API Management platform and from the API Catalogue in the release.

9.3.50.2 Prepared Release

Release Tag: v1.0.185-alpha+46653

(orginal release) - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=47992&view=results

9.3.50.3 Release Process

- Approve access to the COVID-19 Test Result API (Production Environment) User-Restricted APIGEE product on the client configuration with App ID 2940830f-243c-4bd9-bb7c-306130cce541 and App Name nc-vaccination-cache-prod
- Approve PROD deployment of selected version in the DevOps (https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=47992&view=results)

1843 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.50.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1844]
2. Monitor APIM Dashboard for Covid-19 Test Results API ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. nc-vaccination-cache-prod traffic received and processed successfully

9.3.50.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.170-alpa
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45615&view=results

9.3.51 RFC 30029047 - APIM NHS App API - Standard Release (first since 22 January 2021)

9.3.51.1 Description

RFC created by Gareth Sprenger on behalf of Ian Nelson

Second production release of NHS App API proxy (previous release was on 22 January 2021).

No third parties are yet using the API in production, but several are onboarding at present and expected to be using the API from October 2021.

This release therefore bundles all changes made since the original release:

- NHSO-16754: Return 404 from internal/runtime endpoints
- NHSO-0000: Clarify NHS Login integration environment name
- NHSO-16578: Specification changes for custom 429 error
- NHSO-16054: Custom error
- NHSO-16213: Remove integration tests for obsolete non-FHIR endpoint
- NHSO-16314: Update specification examples for 403 forbidden errors
- NHSO-16197: Adding integration tests for Event and Patient reports.
- NHSO-15830: Add expired code to operation outcome schema
- NHSO-15845: Report specification tweaks
- NHSO-16281: change report paths
- NHSO-15845: Patient Report and Event Report specification
- NHSO-15835: update documentation for FHIR communication endpoints
- NHSO-16198: Postman collection endpoint for patient report
- NHSO-15781: Add integration tests

1844 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- NHSO-15833: Mention postman collection and c# integration tests in docs
- NHSO-15782: rewrite link header
- NHSO-15838: Configurable push notification content documentation
- NHSO-15384: Quiet time documentation
- Bump y18n from 3.2.1 to 3.2.2
- NHSO-15784: Postman collection update
- Bump prismjs from 1.21.0 to 1.24.1
- Bump json-pointer from 0.6.0 to 0.6.1
- Bump postcss from 7.0.26 to 7.0.36
- Bump hosted-git-info from 2.8.5 to 2.8.9
- Bump striptags from 3.1.1 to 3.2.0
- Bump set-getter from 0.1.0 to 0.1.1
- NHSO-15856: Increase rate limit and quota to 2400pm
- NHSO-16028: remove non-fhir spec from api documentation
- NHSO-15830: Differentiate error responses
- Bump path-parse from 1.0.6 to 1.0.7
- NHSO-15972: Mapped to ADZ-894 - Remove readOnly label from response schema
- NHSO-13473: Return 404 to apigee from test endpoints
- NHSO-15624: Postman collection update
- NHSO-15526: Add dev environment to manifest template
- NHSO-15006: Update documentation to explain that comms will only be relayed to P9 users
- NHSO-0000: Documentation tweak
- NHSO-14892: Update documentation to show line breaks in markdown messages
- NHSO-14952: Fix digital developer hub url
- Bump lodash from 4.17.20 to 4.17.21 in /sandbox
- NHSO-13807: Added diagnostics to example response
- NHSO-13807: Simplify error response formats
- NHSO-13816: Documentation update
- NHSO-0000: Allow variant casing of legacy communication endpoint 2eed34
- NHSO-10867: Event Report Retrieval Function
- NHSO-10867: Add event report endpoint to postman collection
- ADZ-960: NHS Number capitalisation in documentation
- NHSO-0000: Test assertion improvement
- NHSO-13255: Catch-all error integration test
- NHSO-13250: Test correlation ID mirrored back to consumer
- NHSO-13109: Documentation update
- NHSO-13294: postman collection changes
- NHSO-13248: Return FHIR resource in success response body
- NHSO-13083: Mirror X-Correlation-Id on fault responses
- NHSO-13110 : added new policy AssignMessage.CatchallErrorMessage, modified proxies, target file to call new ess code 400 401 403
- APM-1572 Add manifest_template.yml
- NHSO-11327: Setup postman collections to support jwt authentication
- NHSO-13078: Increase max in-app message body length to 5000 characters
- NHSO-11327: Signed JWT Authentication

- NHSO-13097: Docmentation changes for jwt auth
- NHSO-0000: Documentation update - we support push notifications
- NHSO-12652: API specification changes for HL7 FHIR R4 endpoints
- NHSO-12920: postman fix
- NHSO-0000: Remove deployment to sandbox
- NHSO-0000: Documentation change - API status
- NHSO-12920: Add postman collection and environments
- NHSO-13027: Derive sender name from FHIR sender property, not requester
- NHSO-12651: FHIR validation integration tests
- NHSO-0000: Correct the spelling of 'patients'
- NHSO-12650: Test refactoring

9.3.51.2 Prepared Release

Release Tag: v1.0.133-alpha+48331

(orginal release) - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=48331&view=results

9.3.51.3 Release Process

- Approve PROD deployment of selected version in DevOps (https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=48331&view=results)

9.3.51.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1845]
2. Unknown User (iane1) to try sending a communication to himself as a smoke test?

9.3.51.4 Roll-Back

- Redeploy:
  - Release Tag: ?? Don't know which built - _status endpoint shows build number 39673 but this has long since been cleared down from DevOps.
  - Commit hash of last: 908088a

1845 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.52 RFC 30029377 - APIM Infrastructure Changes

9.3.52.1 Tickets

RFC created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-1953[1846] - Add all unique internal names to API registry + build time checks DONE
- APM-2486
- DOS-464
- APM-2466[1847] - Convert Azure DevOps Services to OAuth2 DONE
- APM-2519[1848] - Back channel logout - Create caches for session-id and revoked session-id DONE
- APM-2535[1849] - Broken cert check pipeline DONE

9.3.52.2 Prepared Release

Release Tag: v2021.10.04.10.22[1850]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50408&view=results

9.3.52.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.52.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1851]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

1846 https://nhsd-jira.digital.nhs.uk/browse/APM-1953?src=confmacro
1847 https://nhsd-jira.digital.nhs.uk/browse/APM-2466?src=confmacro
1848 https://nhsd-jira.digital.nhs.uk/browse/APM-2519?src=confmacro
1849 https://nhsd-jira.digital.nhs.uk/browse/APM-2535?src=confmacro
1850 https://github.com/NHSDigital/api-management-infrastructure/tree/v2021.10.04.10.22
1851 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.52.4 Roll-Back

- Redeploy:
  - Rollback AWS:
    - to: v2021.09.13.16.17 → https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=48208&view=results
  - Rollback APIgee
    - to: v2021.08.13.14.07 → https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=45569&view=results

9.3.53 RFC 30029381 - SDS API forward reliable and forward express address fix and new actor mappings

9.3.53.1 Tickets

RFC created by **Chitrali Chaturvedi** The Release implements the changes in the following tickets:

- https://gpitbjss.atlassian.net/browse/NIAD-1753
  If LDAP returns a forward reliable or forward express interaction, an additional LDAP query has to be made to retrieve proper "address"
- https://gpitbjss.atlassian.net/browse/NIAD-1752
  Add nhsMhsActor mappings. Field was missing in the /Endpoint response
- https://gpitbjss.atlassian.net/browse/NIAD-1857
  Bug fix to NIAD-1752 where /Endpoint returned an error if nhsMhsActor was an empty array

9.3.53.2 Prepared Release

Release Tag: **v1.0.441-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=53326&view=results

9.3.53.3 Release Process

1. Approve PROD deployment of selected version in DevOps https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=53326&view=results

9.3.53.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1852]

[1852] https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for Spine Directory Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.53.4 Roll-Back

- Redeploy:
    - Release Tag: **v1.0.421-alpha**

9.3.54 RFC 30029451 - SDS API new query parameters

9.3.54.1 Tickets

RFC created by **Chitrali Chaturvedi** The Release implements the changes in the following tickets:

- https://gpitbjss.atlassian.net/browse/NIAD-1634 and https://gpitbjss.atlassian.net/browse/NIAD-1636
  Allowed to query for /Endpoint using only interaction_id + party_key parameters
- https://gpitbjss.atlassian.net/browse/NIAD-1611
  API documentation updated on how to use api-key
- https://gpitbjss.atlassian.net/browse/NIAD-1620
  Updated SDS documentation and made spec available on prod
- https://gpitbjss.atlassian.net/browse/NIAD-1477
  Updated SDS documentation on the onboarding process

9.3.54.2 Prepared Release

Release Tag: **v1.0.421-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50990&view=results

9.3.54.3 Release Process

1. Approve PROD deployment of selected version in DevOps https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50990&view=results

9.3.54.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1853]
2. Monitor APIM Dashboard for Spine Directory Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

1853 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.54.4 Roll-Back

- Redeploy:
    - Release Tag: **v1.0.397-alpha**

9.3.55 RFC 30029482 - Set ApplyRateLimiting Quota policies to rollingwindow

> ⚠ This release plan has been superseded by the emergency RFC which includes all these changes and the emergency changes!
> Please follow the emergency infra RFC plan (see page 1556) instead.
> I am leaving the list of tickets here for our auditing trail.

9.3.55.1 Tickets

RFC created by Ben Strutt. The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/APM-1896: AWS Pentest Finding - HSTS / Strict Transport Security headers not implemented
- https://nhsd-jira.digital.nhs.uk/browse/APM-2558: Set ApplyRateLimiting Quota policies to rollingwindow
- https://nhsd-jira.digital.nhs.uk/browse/APM-2489: Schedule job for refreshing Azure Oauth Token
- https://nhsd-jira.digital.nhs.uk/browse/APM-2548: AWS egress-haproxy - apply OS updatesn

9.3.56 RFC 30029491 - APIM Infrastructure Changes

9.3.56.1 Tickets

Emergency RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/DOS-593

9.3.56.2 Prepared Release

Release Tag: **v2021.10.14.15.31**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=51304&view=results

9.3.56.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.56.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1854]
2. Monitor APIM Dashboard for 15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.56.4 Roll-Back

- Redeploy:
  - Release Tag: ***v2021.10.04.10.22***
- Redeploy *last successful non-production release* (to all envs except production)

9.3.57 RFC 30029506: APIM Ambulance Data Submissions API Updates - v1.0.376-alpha

Remember to follow KOP-018 Releasing changes to Production[1855] when deploying these changes!

9.3.57.1 Status of tickets to be deployed

Ticket	Name	Code Reviewed	SA Reviewed	Merged to Master	Release Version
AMB-866[1856]	Rate limiting	✅	✅	✅	v1.0.376-alpha

1854 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1855 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1856 https://nhsd-jira.digital.nhs.uk/browse/AMB-866

9.3.57.2 RFC Change Request Details

Release Version to be deployed: v1.0.376-alpha

9.3.57.3 Manual Changes to be applied

None

9.3.57.4 Dependencies

Any other RFC Change Requests this release plan is dependent on? none

9.3.57.5 Release Plan Summary

How to deploy the RFC, in clear steps:

The v1.0.376-alpha release has been rebuilt to pick up the latest changes to APIM Utils, and successfully promoted through the PTL environments. The Release is now awaiting deployment to Production.

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50401&view=results
- Click "Review" to approve the deployment of v1.0.376-alpha into the Prod environment
- The deployment should be successful (smoke tests are executed as part of the deployment process to prod)
- Ensure that Ambulance API is green for _ping and _status in the Prod Grafana monitoring: https://grafana.prod.api.platform.nhs.uk/d/-yYVfaUMk/api-status?orgId=1&refresh=5m

9.3.58 RFC 30029519 - EPS Primary care prescribing technical go live

9.3.58.1 Tickets

RFC created by Toby Moore.

Second production release of the EPS API (previous release approved for technical go live in November 2020).

No third parties are yet using the API in production, but several are onboarding at present and expected to be using the API from October 2021.

This release therefore bundles all changes made since the original release, referenced by the following tickets:

- Address audit warnings (#576)
- Aea 1713 change manual approval (#579)
- AEA-1715: Handle some missing fields in the release response. (#578)

- added an additional manual approval step for releasing to int (#577)

Everything below this point has already been released to prod, but the RFC is seeking approval for it to be used:

- Address dependabot alert (#575)
- Aea 000 comment audit behaviour (#574)
- Address NPM audit warnings (#573)
- remove auth from metadata (#572)
- update intent and basedOn fields, add tests (#570)
- Enable status monitoring. (#571)
- add more info to verify (#569)
- Support diffing release notes against an arbitrary tag (#568)
- AEA-1644: Extend GET /Task sandbox interaction. (#567)
- AEA-1644: Add response fields to GET /Task interaction in sandbox. (#565)
- AEA-1486 Add repeat-dispensing dispense smoke-test (#562)
- update dispense notification translations to map prescription and line item repeat info (#563)
- Merge pull request #561 from NHSDigital/aea-update-oas
- Ensure linting is consistent across eps modules (#557)
- update medicationRequest in response translation to include previous prescription info and update intent field (#560)
- AEA-1486: Add previous issue date to release response translations. (#558)
- AEA-1228: Enable withdraw pact tests. (#559)
- Add tracker sandbox (#556)
- AEA-1630: Fix the order of pertinent information within a line item. (#554)
- Update spine url for internal-qa (#555)
- AEA-1542: Prescribing Refactor Branch (#533)
- AEA-1626 Add internal-qa environment to release pipeline (#553)
- Aea 1603 metadata capability statement endpoint (#550)
- AEA-1555: Fixes to translations for professional & prescribing codes. (#549)
- AEA-1527: Add separate products for prescribing, dispensing and tracker. (#525)
- AEA-1387: Support dispense notification amendments. (#544)ADZ-894: Remove readOnly label from response schema (#548)
- Add documentation for submodules (#546)AEA-1396: 'No more prescriptions available' returns 200-informational… (#545)
- AEA-1533: adding a verify-signature endpoint to the API (#529)
- AEA-1552: Added authoredOn and providedBy fields to the release response, tests… (#542)
- Update release changes script. (#543)
- Merge pull request #535 from NHSDigital/oas-changes
- AEA-1558: Use correct SDS IDs in ControlActEvent author. (#537)
- AEA-1561: Remove int-preview deployment as this breaks int. (#541)
- rename pact files for int preview (#540)
- AEA-1561: Update service name to pass check in manifest validation. (#539)
- AEA-1561: Update pipeline vars for int preview to pass validation. (#538)
- implement int preview (#536)
- Add patient release request sandbox response to match nominated pharmacy release (#534)

- Merge pull request #530 from NHSDigital/AEA-1496_make_practitioner_role_telecom_mandatory
- implement provenance translations (#531)
- Merge pull request #528 from NHSDigital/ms-AEA-999-hotlink-smartcards
- update healthcareService response translations to allow missing telecom fields, reflect in tests. also update org tests to reflect (#527)
- AEA-1471: Change the translation to use a more appropriate source for dispenser notes. (#526)
- AEA-1520: Improve OpenAPI specification. (#519)
- status endpoint always returns a 200 so that the response isn't overwritten by apigee re-formatting 500 responses (#523)
- AEA-1229: added return task operation to pact test list (#518)
- Merge pull request #521 from NHSDigital/AEA-1516-smoke-test-timeouts
- Merge pull request #522 from NHSDigital/AEA-1517-unblock-release
- Merge pull request #520 from NHSDigital/AEA-1517-improve-status-error-response
- AMB-468: Disable status monitoring (#516)
- AEA-1517: Add lightweight healthcheck for ECS deploy (#517)
- AEA-1219: Enable dispense smoke tests. (#515)
- removing the placholder values for the to-asid and party-key now in s… (#514)
- AEA-1499: Make demographics conversion more resilient. (#512)
- AEA-1505: Add dose to text conversion (part 7). (#510)
- AEA-1460: Add dose to text conversion (part 6). (#509)
- Add secondary-care no-nominated-pharmacy example (#511)
- Merge pull request #507 from NHSDigital/aea-1416-spine-healthcheck
- AEA-1464: Add dose to text conversion (part 5). (#508)
- AEA-1444: Update to-asid and to-party-key in prod. (#499)update response format to platform standard (#506)
- Merge pull request #501 from NHSDigital/AEA-1339-add-default-party-key
- AEA-1459: Add dose to text conversion (part 4). (#505)
- AEA-1458: Add dose to text conversion (part 3). (#504)
- AEA-1457: Add dose to text conversion (part 2). (#503)
- AEA-1456: Add dose to text conversion (part 1). (#502)Merge pull request
- #496 from NHSDigital/dependabot/npm_and_yarn/tests/e2e/pact/ws-5.2.3
- Bump postcss from 7.0.35 to 7.0.36 in /tests/e2e/pact (#498)
- AEA-1339: Populate ASID from Apigee app custom attribute. (#473)
- Update OAS for sandbox claims (#497)
- Add stubbed dispense-claim-information interaction to sandbox (#495)
- AEA-1381: Add MedicationRequest.dispenseRequest.numberOfRepeatsAllowed to the consistency validation. (#493)
- AEA-1376: Add translations for more Spine errors. (#489)
- added a handler for the cancellation response to return operationOutcome rather than a Bundle in some circumstances (#494)
- Update first cancellation smoke-test to be cancelled by an admin (#492)
- AEA-1381: Map high repeat number from two possible locations. (#485)
- Aea update authorisation in oas (#488)
- Merge pull request #487 from NHSDigital/ms-AEA-999-tidy-spec-heading-levels
- Update electronic-prescription-service-api.yaml (#484)

- Revert "AMB-714 add UserRoleService SharedFlow" (#480)
- Merge pull request #483 from NHSDigital/AEA-1423-translate-patient-release-request
- Fix secondary care clinical practitioner examples (#481)Aea 1301 feature toggle on dispense prescribe (#479)
- Move conversion tests from smoke-tests to unit-tests (478)
- Merge pull request #469 from NHSDigital/AMB-714-add-default-user-role
- Bump browserslist from 4.16.3 to 4.16.6 in /tests/e2e/pact (#477)
- update prepare endpoint schemas, fix makefile and package json (#476)
- Use request.logger.error instead of console.error (#474)
- Fix obtaining access tokens for testing in the ref environment. (#475)
- Remove requirement for user auth. (#472)
- Further linting changes (#470)
- Update dispensing link in spec (#471)
- Aea linting (#468)
- Update linter to include dot-notation, eqeqeq and no-multi-spaces (#467)
- Added tests for switchContentTypeForSmokeTest server extension (#464)
- Revert package/package lock changes from when instability was introduced (#465)
- AEA-1327: Use the correct API URL when checking deployment to prod. (#463)
- add debug endpoints to README (#462)
- AEA-1383: Only publish smoke test output if smoke tests have been run. (#461)
- Merge pull request #460 from NHSDigital/aea-update-app-tests
- Add smoke tests for mixed course of therapy types (#454)
- Replace prod ASID and party key values with placeholders (#457)
- AEA-1378: Fix translation of multiple prescriber endorsements. (#458)
- add x-smoke-test to invalid prod header list and update test descriptions (#455)
- Rebrand NHS Identity to NHS CIS2 (#456)
- added smoke test (#446)
- stop certain headers from being available in production (#452)
- AEA-1218: Fixes for release interaction. (#445)
- allow xml responses from spine, add smoke test on process-message (#425)
- Use more appropriate install command for node (#450)
- AEA-1356 / AEA-1361: Handle release rejections + translate errors to FHIR. (#449)
- Disable debug routes for prod environment (#443)
- update validation on medicationRequest and medicationDispense to reje… (#448)
- notes on how to get started on repo, update to lockfile version 2 (#447)
- Translate Error Codes (#428)
- create Medication resource translations & breaking change to CommunicationRequest validation (#436)
- Create output file for test-pack and publish as build artifact (#441)
- Deprecate postman collections (#442)
- Add prescribing cancellation example smoke-tests (#437)
- AEA-1218: Fix interaction and CPA IDs. (#439)
- Update electronic-prescription-service-api.yaml (#438)
- ADO Signing Fixes (#435)
- Test production releases (#434)

- Test production releases (#433)
- Add preview version to test releaseRelease from feature branch to test new ADO release pipeline
- Handle missing/optional extensions in bundle validation (#432)
- validate endpoint (#430)
- Aea 1296 map unknown professional code (#429)
- Update validityPeriod in smoke-tests (#431)
- Cleanup smoke tests (#427)
- Fix sandbox cancel response. (#422)
- Bump y18n from 3.2.1 to 3.2.2 in /specification (#421)
- Add util to calculate release notes (#420)
- aea-964 (#419)
- Move models from coordinator to models module. Add dispense notification to update_prescriptions.py (#417)
- AEA-1081 Add primary-care repeat-prescribing smoke-test (#415)
- AEA-1160 Fix update_prescriptions + allow skipping broker for smoke-tests (#411)
- AEA-1181: Remove organization details from withdraw message. (#416)
- Remove coordinator-models symlinks (#412)
- Fix error in bundle validator dispense-notification unique performer by type test (#414)
- AEA-1162 Sign prescriptions in ADO / AEA-1160 Share code between pact and coordinator to test all examples (#408)
- add Task path and info about Auth to spec (#407)
- AEA-1239 / APM-1486: Switch to using egress proxy for Spine and ODS communication. (#403)
- make use of wrapper functions to handle user auth and external FHIR validator response (#402)
- make use of auth level header in prescription journey (#395)
- AEA-1230: Fix manifest template to allow overriding rate limit. (#400)
- AEA-1125: Tidy up shared translation code. (#398)
- AEA-1131 Enrich dispense notification from ods (#394)
- Aea 1071 send access level to backend (#392)
- Unit tests for Task validator (#396)[WIP]
- AEA-1128 add sandbox responses for task withdraw and return (#391)
- AEA-1131: Prescription Dispense Translation (#381)
- Aea 1201 Handle Unknown CPA IDs (#389)
- AEA-1165: Look up organization details in ODS. (#387)
- Aea 1182 primary care repeat dispensing (#386)
- make use of translations in sandbox release response (#385)
- move messageId inside toSpineRequest (#384)
- APM-1572: Add manifest_template.yml (#360)
- AEA-1073: Stubbed Release Request (#358)
- AEA-1178: Update validator. (#382)
- AEA-1037: Fix translation errors. (#380)
- Bump prismjs from 1.21.0 to 1.23.0 in /specification (#383)
- AEA-1037: Refactor Spine response translation. (#377)
- Aea 1163 primary care dispense fixes (#375)
- AEA-1176: Spec fixes. (#378)
- update dependencies (#376)

- AEA-1037: Fix incorrect translation of performer site type. (#374)
- Testing tool updates (#369)
- AEA-1037: Review changes. Tidying up. (#372)
- Aea 990 unique medication identifiers (#373)
- AEA-1037: Restructure project. (#368)
- AEA-1037: Translate $release responses. (#356)
- re-add components to spec file (#367)
- AEA-1143 - Add dispensing TBC and update spec (#365)
- Aea 1142 primary care dispense check (#364)
- Eps spec changes rob (#357)
- Primary-care: update-prescriptions and sign-prescriptions (#362)
- split datetime functions into separate file (#363)
- AEA-1123 Updates to manual testing tools and docs (#361)
- Add Nizoral cream example (#300)
- AEA-1121 Add tests to update prescriptions and Add cancelation smoke-tests back (#355)
- +setstatus beta - Set API status to beta. Update readme. Rename files/folders. (#359)
- AEA-816: Add translations for medication list. (#353)
- Aea release validation update (#354)
- AEA-1059: Add GP acute and repeat dispensing examples. (#344)
- Aea update spec from rob review (#347)
- AEA-1120: Fix proxy route rule. Change smoke tests to look for success. (#350)
- Additional postman scripts (#324)
- AEA-949: New UK Core version. Removing schema from OpenAPI spec. (#343)
- Improve testing tooling (#333)
- AEA-1046 update base url (#348)
- Handle dispense messages in validator. (#346)
- uppercase UUID to be accepted by spine (#345)
- Aea 550 add X-Request-ID header (#341)
- AEA-1052 dispense sandbox (#338)
- edit spec to include dispense info and edit pipeline to make pr spec … (#339)
- AEA-1045: Return 500 from _status endpoint if service is not healthy. (#340)
- AEA-1025: Add autoscaling policy in ref and prod. (#337)
- AEA-1000: Require API key for subpaths of validator metrics endpoint. (#336)
- Expose validator metrics. Update validator. (#335)
- AEA-1024: Increase task memory to 3GB. Switch to latest validator version. (#332)
- AEA-1005 add log lines for hashes (#330)
- Merge pull request #331 from NHSDigital/AEA-737-and-824
- AEA-831: Update validator. (#327)
- Added logging (#322)
- AEA-995: Increase the CPU of our ECS task. (#321)
- remove the spine message from a process-message response (#318)
- removed message from logger (#320)
- AEA-995: Allow FHIR validator callout to be skipped. (#319)
- AEA-909 (#284)
- AEA-965: Add tests to verify prescription signatures (#317)

- Aea idk push multiple prescriptions to int (#315)
- Bump axios from 0.19.2 to 0.21.1 in /coordinator (#316)
- aea - remove bundle entry duplication (#314)

9.3.58.2 Prepared Release

Release Tag: v1.0.677-beta

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=51688&view=results

9.3.58.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual_approval_prod step to deploy the release to prod

9.3.58.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1857]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.58.4 Roll-Back

- Redeploy:
  - Release Tag: **v1.0.671-beta**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=50385&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.59 RFC 30029596: APIM Infrastructure Changes

9.3.59.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:

1857 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

    - DOS-536[1858] - Integrate the Digital Onboarding Service application with KeyCloak in PTL
    - DOS-596[1859] - HotJar WAF rules are in the wrong order
    - APM-2520[1860] - Back channel logout - mock oidc/cis2 and id/logout tokens for testing
- Minor config changes to support the future release of the following in-flight tickets:
    - AMB-879[1861] - Register and add ClientCert to truststore
    - APM-2521[1862] - Back channel logout - Add endpoint to Identity Service

9.3.59.2 Release Process

Start a release pipeline with the **refs/tags/v2021.10.20.11.50**, click **Review** and **Approve** the prod release.

- Release Tag: **v2021.10.20.11.50**
- Release Pipeline[1863]

9.3.59.3 Post Release Checks

- Monitor Grafana Prod dashboard[1864] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1865] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.59.4 Roll-Back

Redeploy:

- Release Tag: **v2021.10.14.15.31**
- Rollback Pipeline[1866]

9.3.60 RFC 30029702 - Digital Onboarding Service

9.3.60.1 Tickets

Emergency RFC created by **Hannah Royal**. The release implements the changes in the following tickets:

1858 https://nhsd-jira.digital.nhs.uk/browse/DOS-536
1859 https://nhsd-jira.digital.nhs.uk/browse/DOS-596
1860 https://nhsd-jira.digital.nhs.uk/browse/APM-2520
1861 https://nhsd-jira.digital.nhs.uk/browse/AMB-879
1862 https://nhsd-jira.digital.nhs.uk/browse/APM-2521
1863 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=52470
1864 https://grafana.prod.api.platform.nhs.uk
1865 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1866 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=52469

- https://nhsd-jira.digital.nhs.uk/browse/DOS-606
- https://nhsd-jira.digital.nhs.uk/browse/DOS-608
- https://nhsd-jira.digital.nhs.uk/browse/DOS-640

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

9.3.60.2 Prepared Release

Release Tag: **v2021.10.25.11.27**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=52886&view=results

9.3.60.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this has been successfully deployed.
2. Click the Release Link above and approve the manual step to deploy the release to prod.

9.3.60.4 Post Release Checks

1. Monitor Grafana Prod Dashboard[1867]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.60.5 Roll-Back

- Redeploy:
    - Release tag: **v2021.10.20.11.50**
- Redeploy *last successful non-production release* (to all envs except production)

9.3.61 RFC 30029855: Auth Services Changes

9.3.61.1 Tickets

RFC created by **Jodie Barnsley**

The release contains:

- Code changes for the following completed tickets:

1867 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- APM-2541[1868] - Bolierplate SECURITY.md for all repos
- APM-2522[1869] - Back channel logout - implement sid/token caching on authorization code flow

9.3.61.2 Release Process

Start a release pipeline with the **refs/tags/v1.0.1211-alpha**, click **Review** and **Approve** the prod release.

- Release Tag: **v1.0.1211-alpha**
- Release Pipeline (see page 1566)

9.3.61.3 Post Release Checks

- Monitor Grafana Prod dashboard[1870] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1871] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.61.4 Roll-Back

Redeploy:

- Release Tag: **v1.0.1206-alpha**
- Rollback Pipeline (see page 1566)

9.3.62 RFC 30029871: APIM Ambulance Data Submissions API Updates - v1.0.387-alpha

Remember to follow KOP-018 Releasing changes to Production[1872] when deploying these changes!

1868 https://nhsd-jira.digital.nhs.uk/browse/APM-2541
1869 https://nhsd-jira.digital.nhs.uk/browse/APM-2522
1870 https://grafana.prod.api.platform.nhs.uk
1871 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1872 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production

9.3.62.1 Status of tickets to be deployed

Ticket	Name	Code Reviewed	Merged to Master	Release Version
AMB-887[1873] - Merge and test dependency pull requests DONE	Merge and test dependency pull requests	✓	✓	v1.0.387-alpha

9.3.62.2 RFC Change Request Details

Release Version to be deployed: v1.0.387-alpha

9.3.62.3 Manual Changes to be applied

None

9.3.62.4 Dependencies

Any other RFC Change Requests this release plan is dependent on? none

9.3.62.5 Release Plan Summary

How to deploy the RFC, in clear steps:

The v1.0.387-alpha release has been rebuilt to pick up the latest changes to APIM Utils, and successfully promoted through the PTL environments. The Release is now awaiting deployment to Production.

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=52835&view=results
- Click "Review" to approve the deployment of v1.0.376-alpha into the Prod environment
- The deployment should be successful (smoke tests are executed as part of the deployment process to prod)
- Ensure that Ambulance API is green for _ping and _status in the Prod Grafana monitoring: https://grafana.prod.api.platform.nhs.uk/d/-yYVfaUMk/api-status?orgId=1&refresh=5m=

1873 https://nhsd-jira.digital.nhs.uk/browse/AMB-887?src=confmacro

9.3.62.6 Rollback

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=52020&view=results

9.3.63 RFC 30029904 - Digital Onboarding Service

9.3.63.1 Tickets

RFC created by **Hannah Royal**. The release implements the changes in the following tickets:

- DOS-589
- DOS-530
- DOS-590
- DOS-609
- DOS-421
- DOS-552
- DOS-634
- DOS-635
- DOS-633
- DOS-632
- DOS-630

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

9.3.63.2 Prepared Release

Release Tag: **v2021.10.26.10.29**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=54847&view=results

9.3.63.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this has been successfully deployed.
2. Click the Release Link above and approve the manual step to deploy the release to prod.

9.3.63.4 Post Release Checks

1. Monitor Grafana Prod Dashboard[1874]

1874 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.63.5 Roll-Back

- Redeploy:
    - Release tag:
    - **v2021.10.25.11.27**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=52886&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.64 RFC 30029940: APIM Infrastructure Release - Fix ApplyRateLimiting shared flow priority order of arguments

9.3.64.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-2642[1875] - Debug ApplyRateLimiting app-override not applied
    - AMB-879[1876] - Register and add ClientCert to truststore
    - APM-2520[1877] - Back channel logout - mock oidc/cis2 and id/logout tokens for testing
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-2521[1878] - Back channel logout - Add endpoint to Identity Service
    - APMSPII-1206[1879] - Rate Limiting objectives and implementation

9.3.64.2 Release Process

Start a release pipeline with the **refs/tags/v2021.11.09.10.15**, click **Review** and **Approve** the prod release.

- Release Tag: **v2021.11.09.10.15**
- Release Pipeline[1880]

1875 https://nhsd-jira.digital.nhs.uk/browse/APM-2642
1876 https://nhsd-jira.digital.nhs.uk/browse/AMB-879
1877 https://nhsd-jira.digital.nhs.uk/browse/APM-2520
1878 https://nhsd-jira.digital.nhs.uk/browse/APM-2521
1879 https://nhsd-jira.digital.nhs.uk/browse/APMSPII-1206
1880 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=54998

9.3.64.3 Post Release Checks

- Monitor Grafana Prod dashboard[1881] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1882] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.64.4 Roll-Back

Redeploy:

- Release Tag: **v2021.10.20.11.50**
- Rollback Pipeline[1883]

9.3.65 RFC 30029982- APIM NHS App API - Apply per-application rate limits

9.3.65.1 Description

No third parties are yet using the API in production, but several are onboarding at present and expected to be using the API from December 2021.

This release includes only change NHSO-17240 to apply per-application rate limits to the proxy.

NHSO-17240[1884] - NHS App API: Apply per-application rate limits DONE

9.3.65.2 Prepared Release

Release Tag: v1.0.143-alpha+55040

(orginal release) - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=55040&view=results

9.3.65.3 Release Process

- Approve PROD deployment of selected version in DevOps (https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=55040&view=results)

1881 https://grafana.prod.api.platform.nhs.uk
1882 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1883 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=54997
1884 https://nhsd-jira.digital.nhs.uk/browse/NHSO-17240?src=confmacro

9.3.65.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1885]
2. Unknown User (iane1) to try sending a communication to himself as a smoke test

9.3.65.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.133-alpha

9.3.66 RFC 30030097 - Digital Onboarding Service

9.3.66.1 Tickets

RFC created by Phillip Macdonald. The release implements the changes in the following tickets:

- DOS-673
- DOS-665
- DOS-671

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

New UAT environment URL: https://dos-uat.prod.api.platform.nhs.uk/

9.3.66.2 Prepared Release

Release Tag: **v2021.11.18.16.50**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=55916&view=results

9.3.66.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this has been successfully deployed.
2. Click the Release Link above and approve the manual step to deploy the release to prod.

1885 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.66.4 Post Release Checks

1. Monitor Grafana Prod Dashboard[1886]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.66.5 Roll-Back

- Redeploy:
  - Release tag: **v2021.11.09.10.15**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=54998&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.67 RFC 30030118 - NHS e-RS APIM Release R1.3 Production Deployment

9.3.67.1 Tickets

- ERSSUP-58531[1887]: API-M add production product to manitest-template.yml
- ERSSUP-58314[1888]: Update XAPI spec w.r.t. ERSSUP-56694[1889] (DICOM Content-Type)

Further details on APIM 1.3 Release Page[1890]

9.3.67.2 Prepared Release

Release Tag: 1.3.2-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=56503&view=results

9.3.67.3 Release Process

1. **Please ensure all comms for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the Release preceding this, has been successfully deployed

---

1886 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1887 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58531
1888 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58314
1889 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-56694
1890 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/328963333/APIM+1.3+Release+Page

3. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.67.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[1891]

9.3.67.4 Roll-Back

- As it wasn't previously deployed to production there isn't a "Last successful release". The rollback required is to un-deploy 1.3.2-beta to restore the environment to the pre-deployment state - KOP-007 Emergency Single-Service Shutdown[1892]

9.3.68 RFC 30030152 - EPS bug fix

9.3.68.1 Tickets

RFC created by Toby Moore the release implements the changes in the following tickets:

- AEA-1578

9.3.68.2 Prepared Release

Release Tag: v1.0.679-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=51922&view=results

9.3.68.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.68.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1893]

1891 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1892 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119764388/KOP-007+Emergency+Single-Service+Shutdown
1893 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.68.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.677-beta
  - **Link to Azure DevOps** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=51688&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.69 RFC 30030217: Auth Services Changes - backchannel logout

9.3.69.1 Tickets

RFC created by Alex Carrie

9.3.69.1.1 Overview

This release will add the backchannel_logout endpoint to the auth service. This endpoint is solely for receiving notifications from cis2 (restricted by cis2 allowed ips). When the endpoint receives a logout notification from cis2 with a logout token it with invalidate the matching id token. This is done by caching the matching sid with the access token.

This work brings our auth service in line with cis2 standards and is implementing expected behaviour. See cis2 documentation on backchannel logout here - https://digital.nhs.uk/services/identity-and-access-management/nhs-care-identity-service-2/care-identity-authentication/guidance-for-developers/detailed-guidance/session-management#back-channel-logout

The release contains:

- Code changes for the following completed tickets:
  - APM-2524[1894] - Back channel logout - register backchannel logout endpoint with cis2 DONE
    - Backchannel logout - register backchannel logout endpoint with cis2.
  - APM-2521[1895] - Back channel logout - Add endpoint to Identity Service DONE - Back channel logout - Add endpoint to Identity Service

9.3.69.2 Release Process

Click the link to the release pipeline below, retry the manual-approval stage, click **Review** and **Approve** the prod release.

- Release Tag: **v1.0.1338-alpha**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=57415&view=results

---

1894 https://nhsd-jira.digital.nhs.uk/browse/APM-2524?src=confmacro

1895 https://nhsd-jira.digital.nhs.uk/browse/APM-2521?src=confmacro

9.3.69.3 Post Release Checks

- Monitor Grafana Prod dashboard[1896] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1897] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.
- NOTE: from testing in INT, it appears that the Auth Service could start to return the occasional 501s -this is **not** an issue and can be ignored

9.3.69.4 Roll-Back

Redeploy:

- Release Tag: **v1.0.1325-alpha**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=56349&view=results

9.3.70 RFC 30030299 - Digital Onboarding Service

9.3.70.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- DOS-647
- DOS-627

9.3.70.2 Prepared Release

Release Tag: **v2021.11.25.16.02**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58322&view=results

9.3.70.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

1896 https://grafana.prod.api.platform.nhs.uk
1897 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.3.70.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1898]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.70.4 Roll-Back

- Redeploy:
  - Release Tag: **v2021.11.18.16.50**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=55916&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.71 RFC 30030374 - Identity Service

9.3.71.1 Tickets

RFC created by Alex Carrie. The Release implements the changes in the following tickets:

APM-2722[1899] - Add User-Agent header to identity-service JWKS Service Callout DONE

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

9.3.71.2 Prepared Release

Release Tag: v1.0.1344-alpha[1900]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58714&view=results

9.3.71.3 Release Process

1. Click the link to the release pipeline above, retry the manual-approval stage, click **Review** and **Approve** the prod release.

1898 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1899 https://nhsd-jira.digital.nhs.uk/browse/APM-2722?src=confmacro
1900 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1344-alpha

9.3.71.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1901] (note that some of the monitoring stack may be down during deployment).
2. Monitor Splunk APIM Dashboard[1902] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.71.4 Roll-Back

- Redeploy:
  - Release Tag: **v1.0.1338-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=57415&view=results

9.3.72 RFC 30030383 - Digital Onboarding Service

9.3.72.1 Tickets

RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- DOS-532
- DOS-701

9.3.72.2 Prepared Release

Release Tag: **v2021.11.25.16.02+58739 (redeployment of same version to update DOS application)**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58739&view=results

9.3.72.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.72.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1903]

1901 https://grafana.prod.api.platform.nhs.uk
1902 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1903 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.72.4 Roll-Back

- Redeploy:
  - Release Tag: **v2021.11.25.16.02+58322**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58322&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.73 RFC 30030422 - SDS API new query params + mapping change

9.3.73.1 Tickets

RFC created by Jason Edwards. The Release implements the changes in the following tickets:

- ~~https://gpitbjss.atlassian.net/browse/NIAD-1634~~ and ~~https://gpitbjss.atlassian.net/browse/NIAD-1636~~
  ~~Allowed to query for /Endpoint using only interaction_id + party_key parameters~~
- ~~https://gpitbjss.atlassian.net/browse/NIAD-1611~~
  ~~API documentation updated on how to use api-key~~
- ~~https://gpitbjss.atlassian.net/browse/NIAD-1620~~
  ~~Updated SDS documentation and made spec available on prod~~
- ~~https://gpitbjss.atlassian.net/browse/NIAD-1477~~
  ~~Updated SDS documentation on the onboarding process~~
- https://nhsd-jira.digital.nhs.uk/browse/APM-2016
  Invoke pytest from python rather than shell script
- https://gpitbjss.atlassian.net/browse/NIAD-1753
  Forward reliable address handler
- https://gpitbjss.atlassian.net/browse/NIAD-1752
  Adding nhsMhsActor mappings
- https://gpitbjss.atlassian.net/browse/NIAD-1857
  Fix for all cases if LDAP returns empty arrays
- https://gpitbjss.atlassian.net/browse/NIAD-1891
  Production URL added to doc
- https://gpitbjss.atlassian.net/browse/NIAD-1939
  LDAP → FHIR Manufacturer mapping . Note test results showing against NIAD-1939, but commits performed against NIAD-1936[1904]
- https://gpitbjss.atlassian.net/browse/NIAD-1955
  Change Apigee cache TTL from 15mins to 30mins

1904 https://gpitbjss.atlassian.net/browse/NIAD-1936

9.3.73.2 Prepared Release

Release Tag: **v1.0.446-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58895&view=results

9.3.73.3 Release Process

1. Approve PROD deployment of selected version in DevOps https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58895&view=results
2. Please note there are 4 warnings listed by the deploy job relating to deprecated parameters. These will be resolved in the next release;
    - Warn product_display_name is a deprecated pipeline parameter
    - Warn product_description is a deprecated pipeline parameter
    - Warn portal_api_requires_callback_url is a deprecated pipeline parameter
    - Warn spec_file is a deprecated pipeline parameter

9.3.73.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1905] .
2. Monitor APIM Dashboard for Spine Directory Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.73.4 Roll-Back

- Redeploy:
    - Release Tag: **v1.0.421-alpha**

9.3.74 RFC 30030423 - NHS e-RS APIM Release R1.3 Production Deployment

9.3.74.1 Tickets

- ERSSUP-58531[1906]: API-M add production product to manitest-template.yml
- ERSSUP-58314[1907]: Update XAPI spec w.r.t. ERSSUP-56694[1908] (DICOM Content-Type)

1905 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1906 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58531
1907 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58314
1908 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-56694

Further details on APIM 1.3 Release Page[1909]

Note: This is a clone of a previous release. The release failed due to a missing target server in Apigee not being present. A release of the Infrastruture service has now added the target server so the this RFC is to attempt the previous RFC release again.

9.3.74.2 Prepared Release

Release Tag: 1.3.2-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=56503&view=results

9.3.74.3 Release Process

1. **Please ensure all comms for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.74.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[1910]

9.3.74.4 Roll-Back

- As it wasn't previously deployed to production there isn't a "Last successful release". The rollback required is to un-deploy 1.3.2-beta to restore the environment to the pre-deployment state - KOP-007 Emergency Single-Service Shutdown[1911]

1909 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/328963333/APIM+1.3+Release+Page
1910 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1911 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119764388/KOP-007+Emergency+Single-Service+Shutdown

9.3.75 RFC 30030433 - Digital Onboarding Service

9.3.75.1 Tickets

Standard Change RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-678[1912]
- DOS-692[1913]

9.3.75.2 Prepared Release

Release Tag: **v2021.12.06.14.57**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58915&view=results

9.3.75.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.75.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1914]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.75.4 Roll-Back

- Redeploy:
    - Release Tag: **v2021.11.09.16.42**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58888&view=logs&j=b59d050b-0bf9-5803-ec04-7f124888d112&t=44151db8-30b5-585e-a489-dae1d6050f3d/results?buildId=39373&view=results[1915]
- Redeploy *last successful non-production release* (to all envs except production)

1912 https://nhsd-jira.digital.nhs.uk/browse/DOS-678
1913 https://nhsd-jira.digital.nhs.uk/browse/DOS-692
1914 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1915 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results

9.3.76 RFC 30030479 - NHS e-RS APIM Release R1.4 Production Deployment

9.3.76.1 Tickets

- Defects observed during APIM 1.3 release and splitting products for App-Restricted & User-Restricted

Further details on APIM 1.4 Release Page[1916]

9.3.76.2 Prepared Release

Release Tag: 1.4.2-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60138&view=results

9.3.76.3 Release Process

1. **Please ensure all comms for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.76.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[1917]

9.3.76.3.2 Post Release Actions

As part of ERSSUP-57341 the e-RS product was split so that user and application restricted access modes could be permissioned separately.

The existing product "e-Referrals-Service (Production)" has been replaced by "e-Referral Service - Healthcare Worker (Production)". The Application Restricted access mode is not yet supported so no product is generated yet.

1. Check that the "e-Referral Service - Healthcare Worker (Production)" product has been created
2. Check that no Applications are associated with the "e-Referrals-Service (Production)" Product

---

1916 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/354255052/APIM+1.4+Release+Page
1917 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

    - There are currently no live applications so there should be no associations
3. If no Applications are associated with the "e-Referrals-Service (Production)" Product then delete it

A separate request will be raised in the platforms-apim-producer-support Slack channel to remove the product in the Integration Testing (INT) environment.

9.3.76.4 Roll-Back

- Redeploy:
    - Release Tag: v1.3.2-beta
    - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=56503&view=results

9.3.77 RFC 30030574 - Alignment of infrastructure version in Production

9.3.77.1 Tickets

RFC created by Unknown User (lasm2) .

The purpose of this release is to ensure that the manual emergency change from rollingWindow to calendar for the "ApplyRateLimiting" Shared Flow.

- Calendar has been running successfully in INT for a number of weeks
- In Prod, the Shared Flow, it was on Revision 5, and manually updated to Revision 9 (v2021.12.09.18.07)
- Latest on PTL is at v2021.12.09.18.07
    - The cert change will be drawn into the release, with a check on secrets

The Release implements the changes in the following tickets:

- APM-2742: NHS App Prod Cert update
- No ticket for the emergency release yesterday
- APMSPII-1261: Rate limit change on PDS

The release will involve:

- ~~Terraform release to update Grafana dashboard~~
- Infra deploy

9.3.77.2 Prepared Release

Release Tag: v2021.12.09.18.07

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=59350&view=results

9.3.77.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.77.3.1 Post Release Checks

1. ~~Monitor Grafana Prod dashboard~~[1918] **~~Not done~~ – more changes required before this change is useful and don't want to bring down monitoring during a critical period.**
2. Monitor APIM Dashboard for all proxies ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.77.4 Roll-Back

- WARNING - in this instance rollback will result in the "rollingWindow" bug being reintroduced
    - Create a hotfix tag with just the "calendar" change in the shared flow
- Redeploy:
    - Release Tag: v2021.11.09.16.42
    - <**Link** to Azure devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results

9.3.78 RFC 30030787: APIM Infrastructure Changes

9.3.78.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-2796[1919] - Grafana infrastructure version dashboard displays no information
    - APM-2732[1920] - Enable OAuth2-mock to int environment

9.3.78.2 Release Process

Start a release pipeline with the **refs/tags/v2021.12.16.15.30**, click **Review** and **Approve** the prod release.

1918 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1919 https://nhsd-jira.digital.nhs.uk/browse/APM-2796
1920 https://nhsd-jira.digital.nhs.uk/browse/APM-2732

- Release Tag: **v2021.12.16.15.30**
- Release Pipeline[1921]

9.3.78.3 Post Release Checks

- Monitor Grafana Prod dashboard[1922] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1923] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.78.4 Roll-Back

Redeploy:

- Release Tag: **v2021.12.06.14.57**
- Rollback Pipeline[1924]

9.3.79 RFC 30030831: APIM Infrastructure Changes

9.3.79.1 Tickets

RFC created by **Alex Carrie**

The release contains:

- Code changes for the following completed tickets:
  - APM-657[1925] - Auth - extend combined auth refresh token from 1 to 12 hours DONE - Updates and adds identity service KVMs.
  - APM-363[1926] - Add a security policy and guidelines to repository DONE - Adding mark down file, no risk.
  - **APM-TRIAGE Remove IP restriction on keycloak** - keycloak is PTL so should be no risk.
  - DOS-702 - Disabling the old authentication in DOS for the new Dev Identity login, and content/ keycloak changes for the final release.
  - APM-2774[1927] - CIS2 mock - sign in with user ID, not email address DONE - Keycloak front end change PTL only.

1921 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60480&view=results
1922 https://grafana.prod.api.platform.nhs.uk
1923 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1924 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=60571
1925 https://nhsd-jira.digital.nhs.uk/browse/APM-657?src=confmacro
1926 https://nhsd-jira.digital.nhs.uk/browse/APM-363?src=confmacro
1927 https://nhsd-jira.digital.nhs.uk/browse/APM-2774?src=confmacro

- APM-2819[1928] - Improved testability - e-RS 500 error DONE - Keycloak users change - PTL only.
- APM-2782[1929] - Improvements to API setup self-serve DONE - Improvements to the self serve api set up - PTL only.
- DOS-646 - Adds cross section conditional logic to DOS question sections.
- DOS-785 - Fixes ability to download attachments from historical answers in DOS.
- DOS-685 - Removes use of RAWSQL statements in DOS.
- DOS-694 - Enables the use of Multi-API DAQ in DOS.
- DOS-727
- DOS-650
- DOS-676

9.3.79.2 Release Process

Start a release pipeline with the **refs/tags/v2022.01.07.16.52**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.01.07.16.52**
- Release Pipeline[1930] -
  - **NOTE: The `terraform-prod` stage of the release pipeline may fail on the first apply due to moving an SSM parameter between modules. If this happens, restart a new copy of the pipeline. The second "fresh" pipeline should succeed.**
  - If the second application fails, delete the SSM parameter **/prod/monitoring/platform_version** before running the pipeline a third time, this will certainly succeed. This will require AWS prod admin permissions.

9.3.79.3 Post Release Checks

- Monitor Grafana Prod dashboard[1931] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1932] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.79.4 Roll-Back

Redeploy:

- Release Tag: **v2021.12.16.15.30**

1928 https://nhsd-jira.digital.nhs.uk/browse/APM-2819?src=confmacro
1929 https://nhsd-jira.digital.nhs.uk/browse/APM-2782?src=confmacro
1930 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60722&view=results
1931 https://grafana.prod.api.platform.nhs.uk
1932 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

- Rollback Pipeline[1933]

9.3.80 RFC 30030852 - Identity Service

9.3.80.1 NOTE - Must be done after successful release of Infrastructure (planned for 11th Jan) - RFC 30030831: APIM Infrastructure Changes

9.3.80.2 Tickets

RFC created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-657[1934] - Auth - extend combined auth refresh token from 1 to 12 hours DONE - Adds policy to check auth provider on refresh token then selects kvm accordingly. Currently kvms have the same value but adds logic to refresh token flow. Tested manually. Some risk to refreshing tokens but fairly low.
- APM-2751[1935] - Identity-Service - External Access Token is null on a valid request in int DONE - Update userinfo error message on unsupported auth provider - Not risky
- APM-2732[1936] - Enable OAuth2-mock to int environment DONE - Removes unused config from identity service. Not risky

9.3.80.3 Prepared Release

Release Tag: v1.0.1369-alpha[1937]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60363&view=results

9.3.80.4 Release Process

1. Click the link to the release pipeline above, retry the manual-approval stage, click **Review** and **Approve** the prod release.

9.3.80.4.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1938] (note that some of the monitoring stack may be down during deployment).

1933 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60480&view=results
1934 https://nhsd-jira.digital.nhs.uk/browse/APM-657?src=confmacro
1935 https://nhsd-jira.digital.nhs.uk/browse/APM-2751?src=confmacro
1936 https://nhsd-jira.digital.nhs.uk/browse/APM-2732?src=confmacro
1937 https://github.com/NHSDigital/identity-service-api/tree/v1.0.1369-alpha
1938 https://grafana.prod.api.platform.nhs.uk

2. Monitor Splunk APIM Dashboard[1939] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.80.5 Roll-Back

- Redeploy:
  - Release Tag: **v1.0.1344-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58714&view=results

9.3.81 RFC 30030933 - PDS Add ASID and Reinstate ASID-Checking in Prod

9.3.81.1 Tickets

The Release implements the changes in the following tickets:

- APMSPII-1268

9.3.81.2 Prepared Release

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60976&view=results

9.3.81.3 Release Process

9.3.81.3.1 Add ASIDs to Apps

1. Add `asid` attribute to Evacc - PROD - NEW[1940] - 200000028306
2. Check Splunk to ensure traffic is flowing for these apps - misconfigured ASID may cause problems
   a. **Check app still working:** index="requests_apm_prod" source="apigee:nhsd-prod:prod:personal-demographics-prod:*" meta.application_id="5608c504-f3f7-436a-a226-c3abdbcdae67" | `apim-macro-set-status-code` | timechart span=10s count by status_code_returned
   b. **Check ASID is now being sent:** index="requests_apm_prod" source="apigee:nhsd-prod:prod:personal-demographics-prod:*" meta.application_id="5608c504-f3f7-436a-a226-c3abdbcdae67" | table meta.ASID

⛔**If the app isn't working - remove the ASID and don't proceed with the rest of this release!!**

1939 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

1940 https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/5608c504-f3f7-436a-a226-c3abdbcdae67

9.3.81.3.2 Release ASID-Checking to Production

1. Go to prepared release
2. Give prod approval

9.3.81.3.3 Post Release Checks

1. Monitor APIM Dashboard for PDS Production for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.81.4 Roll-Back

- Redeploy https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=59232&view=results
- Redpoy *last successful non-production release* (to all envs except production)

9.3.82 RFC 30030990: APIM Infrastructure Changes

9.3.82.1 Tickets

RFC created by **Hannah Ibberson**

The release contains:

- Code changes for the following completed tickets:
  - DOS-682[1941] - Combining several pages for easier navigation in DOS
  - DOS-732[1942] - Coloured status tags for internal assessing roles in DOS
  - DOS-836[1943] - Adding Serbia and Germany to the DOS WAF
  - DOS-841[1944] - Adding Spain to the DOS WAF
  - DOS-704[1945] - Deploying blank keycloak infrastructure to production

9.3.82.2 Release Process

Start a release pipeline with the **refs/tags/v2022.01.18.09.59**, click **Review** and **Approve** the prod release.

1941 https://nhsd-jira.digital.nhs.uk/browse/DOS-682
1942 https://nhsd-jira.digital.nhs.uk/browse/DOS-732
1943 https://nhsd-jira.digital.nhs.uk/browse/DOS-836
1944 https://nhsd-jira.digital.nhs.uk/browse/DOS-841
1945 https://nhsd-jira.digital.nhs.uk/browse/DOS-704

- Release Tag: v2022.01.18.09.59
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=61386&view=results

9.3.82.3 Post Release Checks

- Monitor Grafana Prod dashboard[1946] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1947] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.82.4 Roll-Back

Redeploy:

- Release Tag: v2022.01.07.16.52
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60842&view=results

9.3.83 RFC 30031030: APIM Infrastructure Changes

9.3.83.1 Tickets

RFC created by **Phillip Macdonald**

The release contains:

- Code changes for the following completed tickets:
  - DOS-682[1948] - Combining several pages for easier navigation in DOS
  - DOS-732[1949] - Coloured status tags for internal assessing roles in DOS
  - DOS-836[1950] - Adding Serbia and Germany to the DOS WAF
  - DOS-841[1951] - Adding Spain to the DOS WAF
  - DOS-704[1952] - Deploying blank keycloak infrastructure to production

9.3.83.2 Release Process

Start a release pipeline with the **refs/tags/v2022.01.18.09.59**, click **Review** and **Approve** the prod release.

1946 https://grafana.prod.api.platform.nhs.uk/
1947 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
1948 https://nhsd-jira.digital.nhs.uk/browse/DOS-682
1949 https://nhsd-jira.digital.nhs.uk/browse/DOS-732
1950 https://nhsd-jira.digital.nhs.uk/browse/DOS-836
1951 https://nhsd-jira.digital.nhs.uk/browse/DOS-841
1952 https://nhsd-jira.digital.nhs.uk/browse/DOS-704

- Release Tag: v2022.01.18.09.59
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=61614&view=results
- **This release is expected to fail on the first terraform apply, due to ACM/ELB creation order. A re-run of the full pipeline, i.e. re-plan and re-apply, should allow a succesful release. 2 failures should result in a rollback.**

9.3.83.3 Post Release Checks

- Monitor Grafana Prod dashboard[1953] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1954] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.83.4 Roll-Back

Redeploy:

- Release Tag: v2022.01.07.16.52
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60842&view=results

9.3.84 RFC 30031068: APIM Infrastructure Updates

9.3.84.1 Tickets

RFC created by **Ben Strutt** The Release implements the changes in the following tickets:

- APM-2016 Fix ECS component of grafana dashboard for techops.

9.3.84.2 Prepared Release

Release Tag: **v2022.01.21.14.13**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62012&view=results

9.3.84.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

1953 https://grafana.prod.api.platform.nhs.uk/
1954 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.3.84.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1955]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.85 RFC 30031092 - NHS e-RS APIM Release R1.5 Production Deployment

9.3.85.1 Tickets

- An APIM team change to add a security reference document
- An APIM team change to update the e-referrals-service-api.yaml file
- An APIM team change to allow products access to mock oauth (test harness replacement)
- OAS file example validation using the openapi-core Python library

Further details on APIM 1.5 Release Page[1956]

9.3.85.2 Prepared Release

Release Tag: 1.5.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62222&view=results

9.3.85.3 Release Process

1. **Please ensure all comms for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.85.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[1957]

1955 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
1956 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/373755032/APIM+1.5+Release+Page
1957 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.85.4 Roll-Back

- Redeploy:
  - Release Tag: v1.4.2-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60138&view=results

9.3.86 RFC 30031095 - Digital Onboarding Service/Developer Identity SSO

9.3.86.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- DOS-790
- DOS-842
- DOS-749
- DOS-748
- DOS-739
- DOS-843
- DOS-855
- DOS-830
- DOS-740
- DOS-742
- DOS-705
- DOS-817

9.3.86.2 Prepared Release

Release Tag: **v2022.01.24.14.57**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62289&view=results

9.3.86.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.86.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1958]

[1958] https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.86.4 Roll-Back

- Redeploy:
  - Release Tag: v2022.01.21.14.13
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62012&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.87 RFC 30031265 Digital Onboarding Service

9.3.87.1 Tickets

RFC created by Unknown User (gipa6). The Release implements the changes in the following tickets:

- DOS-741
- DOS-743
- DOS-933
- DOS-744[1959]
- DOS-932[1960]

9.3.87.2 Prepared Release

Release Tag: **v2022.02.14.43**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=64031&view=results

9.3.87.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.87.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1961]

1959 https://nhsd-jira.digital.nhs.uk/browse/DOS-744
1960 https://nhsd-jira.digital.nhs.uk/browse/DOS-932
1961 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.87.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.01.24.14.57**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62289&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.3.88 RFC 30031345 - Add Developer Identity SSO to Production Apigee Portal

9.3.88.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- DOS-891

9.3.88.2 Prepared Release

N/A

Note this is an Apigee Configuration change - as such, it doesn't have an explicit release.

Release Process

Will be implemented by Phillip Macdonald in hours

1. Follow the steps in the JIRA ticket documented

9.3.88.2.1 Post Release Checks

1. Check a user can log into Apigee with SAML:
   a. And can create:
      i. An app
      ii. A team

9.3.88.3 Roll-Back

- Switch off SAML integration for Apigee portal
- Re-deploy custom domain integration

9.3.89 RFC 30031392 - Digital Onboarding Service

9.3.89.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-903[1962]
- DOS-908[1963]
- DOS-904[1964]
- DOS-760[1965]

9.3.89.2 Prepared Release

Release Tag: **v2022.02.08.16.01**

// TODO: add production plan

9.3.89.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.89.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1966]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.89.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.02.14.43**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=64031&view=results
- Redeploy *last successful non-production release* (to all envs except production)

1962 https://nhsd-jira.digital.nhs.uk/browse/DOS-903
1963 https://nhsd-jira.digital.nhs.uk/browse/DOS-908
1964 https://nhsd-jira.digital.nhs.uk/browse/DOS-904
1965 https://nhsd-jira.digital.nhs.uk/browse/DOS-760
1966 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.90 RFC 30031501 - API Management Infrastructure: Switch to High Bandwith HSCN connection

9.3.90.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
  - APM-2906[1967] - AWS to HSCN routing change
  - APM-2897[1968] - Hello World - XML code gen proxy - MVP
  - DOS-891[1969] - Keycloak: Apigee Prod and Non-Prod Client set-up and configuration
- Minor config changes to support the future release of the following in-flight tickets:
  - APM-2861[1970] - Fix logic in shared flow for user_role

9.3.90.2 Release Process

Start a release pipeline with the **refs/tags/v2022.02.14.15.16**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.02.14.15.16**
- Release Pipeline[1971]

9.3.90.3 Post Release Checks

- Monitor Grafana Prod dashboard[1972] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1973] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.3.90.4 Roll-Back

Redeploy:

- Release Tag: **v2022.02.08.16.01**
- Rollback Pipeline (see page 1598)

1967 https://nhsd-jira.digital.nhs.uk/browse/APM-2906
1968 https://nhsd-jira.digital.nhs.uk/browse/APM-2897
1969 https://nhsd-jira.digital.nhs.uk/browse/DOS-891
1970 https://nhsd-jira.digital.nhs.uk/browse/APM-2861
1971 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=67277&view=results
1972 https://grafana.prod.api.platform.nhs.uk
1973 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.3.91 RFC 30031595 - APIM Infrastructure Changes

9.3.91.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-924[1974]
- DOS-905[1975]
- DOS-1006[1976]
- DOS-1007[1977]

9.3.91.2 Prepared Release

Release Tag: **v2022.02.18.11.59**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=66945&view=results

9.3.91.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.91.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1978]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.91.4 Roll-Back

If RFC 30031501 - API Management Infrastructure: Switch to High Bandwith HSCN connection (see page 1598) is successful:

- Redeploy:
    - Release Tag: **v2022.02.14.15.16**

1974 https://nhsd-jira.digital.nhs.uk/browse/DOS-924
1975 https://nhsd-jira.digital.nhs.uk/browse/DOS-905
1976 https://nhsd-jira.digital.nhs.uk/browse/DOS-1006
1977 https://nhsd-jira.digital.nhs.uk/browse/DOS-1007
1978 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

    - Release Pipeline[1979]

Otherwise:

- Redeploy:
    - Release Tag: **v2022.02.08.16.01**
    - Rollback Pipeline[1980]
- Redeploy *last successful non-production release* (to all envs except production)

9.3.92 RFC 30031685: APIM Infrastructure Changes

9.3.92.1 Tickets

RFC created by **Ben Strutt**. This release falls under our standard change process (see page 2093).

The release contains:

- Code changes for the following completed tickets:
    - APM-2884[1981] - Monitor Splunk ingestion health from prometheus/grafana

9.3.92.2 Release Process

Start a release pipeline with the **refs/tags/v2022.02.21.15.34**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.02.21.15.34**
- Release Pipeline[1982]

9.3.92.3 Post Release Checks

- Monitor Grafana Prod dashboard[1983] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[1984] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

1979 https://nhsd-confluence.digital.nhs.uk/pages/createpage.action?fromPageId=387843497&linkCreation=true&spaceKey=APM&title=Not+yet+started
1980 https://nhsd-confluence.digital.nhs.uk/pages/createpage.action?fromPageId=387843497&linkCreation=true&spaceKey=APM&title=Not+yet+started
1981 https://nhsd-jira.digital.nhs.uk/browse/APM-2884
1982 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=67629&view=results
1983 https://grafana.prod.api.platform.nhs.uk
1984 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.3.92.4 Roll-Back

Redeploy:

- Release Tag: **v2022.02.18.11.59**
- Rollback Pipeline (see page 1600)

9.3.93 RFC 30030425: APIM Infrastructure - Add ERS Production Target Server

9.3.93.1 Tickets

RFC created by Ben Strutt. The Release implements the changes in the following tickets:

- ERSSUP-57222 - Adding e-RS FT servers as targets, including e-RS CA

9.3.93.2 Prepared Release

Release Tag: v2021.11.09.16.42

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=58888&view=logs&j=b59d050b-0bf9-5803-ec04-7f124888d112&t=44151db8-30b5-585e-a489-dae1d6050f3d/results?buildId=39373&view=results[1985]

9.3.93.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.93.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1986]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.93.4 Roll-Back

- Redeploy:
    - Release Tag: v2021.11.09.10.15

1985 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results
1986 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.3.94 RFC-30028297 Migrate Immunisation History Client Apps -> User Restricted

9.3.94.1 Advise RFC starting in slack

9.3.94.2 Check traffic on Immunisation History, make everything looks ok.

```
index="requests_apm_prod" source="apigee:nhsd-prod:prod:immunisation-history-prod:*"
| rename target.status_code as target_status
| rename error.status_code as error_status
| rename response.status_code as response_status
| rename proxy.suffix as suffix
| rename meta.application as app
| eval status=if(response_status!="",response_status, if(error_status!="",error_statu
s, target_status))
| search suffix!="/_status"
| search suffix!="/_ping"
| eval app_status=app + "_" + status
| timechart count by app_status
```

https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/search?earliest=-15m&latest=now&q=search%20index%3D%22requests_apm_prod%22%20source%3D%22apigee%3Anhsd-prod%3Aprod%3Aimmunisation-history-prod%3A*%22%0A%7C%20rename%20target.status_code%20as%20target_status%0A%7C%20rename%20error.status_code%20as%20error_status%0A%7C%20rename%20response.status_code%20as%20response_status%20%0A%7C%20rename%20proxy.suffix%20as%20suffix%20%0A%7C%20rename%20meta.application%20as%20app%20%0A%7C%20eval%20status%3Dif(response_status!%3D%22%22%2Cresponse_status%2C%20if(error_status!%3D%22%22%2Cerror_status%2C%20target_status))%0A%7C%20search%20suffix!%3D%22%2F_status%22%0A%7C%20search%20suffix!%3D%22%2F_ping%22%0A%7C%20eval%20app_status%3Dapp%20%2B%20%22_%22%20%2B%20status%20%0A%7C%20timechart%20count%20by%20app_status&display.page.search.mode=smart&dispatch.sample_ratio=1&display.page.search.tab=visualizations&display.general.type=visualizations&sid=1628176132.1540992_7A3A6B4B-80C6-47BF-8A50-EDE9841C029D

9.3.94.3 Make changes in APIgee

1. Login to APIgee UI
2. locate the affected production client applications:
    - https://apigee.com/organizations/nhsd-prod/apps/details/2940830f-243c-4bd9-bb7c-306130cce541
    - https://apigee.com/organizations/nhsd-prod/apps/details/defa5e86-73c0-4054-8891-4bc4db95f1a8

3. for each affected application:
    a. click Edit (top right)
    b. Click Add Product
    c. Select **Immunisation History - User Restricted (Production)**
    d. Click Add
    e. Remove **Immunisation History - Legacy (Production)** (click the X)
    f. click Save

> RELEASE NOTE: Simply adding the new products was not enough - newly added Products are set to "Pending" even if set to "Approved" when adding! Had to pop back into each Application and manually "Approve" the newly added products.

9.3.94.4 Observe API traffic

using Splunk, observe API traffic, 'for a few minutes'? Ensure applications are still able to access the API as before.

9.3.94.4.1 Rollback (ONLY IF REQUIRED)

If there is a problem observed in #4 Revert the steps in #3

9.3.94.5 Update Live Access Logbook

https://nhsd-confluence.digital.nhs.uk/display/APM/Live+access+logbook

9.3.94.6 Notify RFC Complete In Slack

9.3.95 RFC - APIM Infrastructure Updates - Feature Flag

This Release Plan is to cover the deployment of the Feature Flag feature in APIM Infrastructure. This change consists of the addition of a new KVM variable and deployment of the changes to the APIGee Config.

⚠ Remember to follow KOP-018 Releasing changes to Production[1987] when deploying these changes!

9.3.95.1 Status of tickets to be deployed

Ticket	Name	Merged to Master	Release Version
APM-1866[1988]	APIGee feature toggles support	✅	

9.3.95.2 RFC Change Request Details

Release Version to be deployed: n/a

9.3.95.3 Manual Changes to be applied (Pre-Deploy)

KVM Key to be added to Prod:

- Add KVM "clear-caribou"
  - Add key "feature_toggles"
  - Value = "{"enable-token-exchange": false}"

9.3.95.4 Deployment Process

Deploy APIGee Config Changes to Prod environment

9.3.95.5 Dependencies

Any other RFC Change Requests this release plan is dependent on?

1987 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169714940/KOP-018+Releasing+changes+to+Production
1988 https://nhsd-jira.digital.nhs.uk/browse/APM-1866

9.3.95.6 Manual Changes to be applied (Post-Deploy)

N/A

9.3.95.7 Post-Deployment Checks

How to verify that the deployment was successful

9.3.96 RFC TBD - Digital Onboarding Service

9.3.96.1 Tickets

RFC created by Giles Papworth. The Release implements the changes in the following tickets:

- DOS-701

Note to access the above please request access to the Digital Onboarding Service (Turnstile Project)

9.3.96.2 Prepared Release

Release Tag: <**eg. v2021.07.06.11.20** >

<**Link to Azure** devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39373&view=results>

9.3.96.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.3.96.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[1989]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.3.96.4 Roll-Back

- Redeploy:
    - Release Tag: <**eg. v2021.06.23.15.17** >

1989 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

    - <**Link** to Azure devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.4 RFC 30031886: APIM Infrastructure Changes

9.4.1 Tickets

RFC created by ***Ben Strutt***

The release contains:

- Code changes for the following completed tickets:
    - APM-2899[1990] - Hello World - Add spec micro-service to paas
    - APM-2927[1991] - S3 artefact repository for proxy bundle
    - ADZ-1520[1992] - DOS: App list: DOS to use its "machine user" account to talk to Apigee
    - DOS-660[1993] - DOS .NET 6 runtime upgrade
    - DOS-666[1994] - sort out unused .NET using statements in DOS
    - DOS-1036[1995] - Add Netherlands to DOS/Dev Identity WAF
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-2016[1996] - API producer support 2021/2022

9.4.2 Release Process

Start a release pipeline with the **refs/tags/v2022.03.08.13.16**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.03.08.13.16**
- Release Pipeline[1997]

9.4.3 Post Release Checks

- Monitor Grafana Prod dashboard[1998] (note that some of the monitoring stack may be down during deployment).

1990 https://nhsd-jira.digital.nhs.uk/browse/APM-2899
1991 https://nhsd-jira.digital.nhs.uk/browse/APM-2927
1992 https://nhsd-jira.digital.nhs.uk/browse/ADZ-1520
1993 https://nhsd-jira.digital.nhs.uk/browse/DOS-660
1994 https://nhsd-jira.digital.nhs.uk/browse/DOS-666
1995 https://nhsd-jira.digital.nhs.uk/browse/DOS-1036
1996 https://nhsd-jira.digital.nhs.uk/browse/APM-2016
1997 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69121&view=results
1998 https://grafana.prod.api.platform.nhs.uk

- Monitor Splunk APIM Dashboard[1999] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.4.4 Roll-Back

Redeploy:

- Release Tag: **v2022.02.21.15.34**
- Rollback Pipeline[2000]

9.5 RFC 30031983: APIM Infrastructure Changes

9.5.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
  - APM-2915[2001] - Upgrade terraform & terraform providers, uplift s3
  - APM-2943[2002] - Test text xml generated auth
  - APM-2980[2003] - Keycloak - create API Producer realm
- Minor config changes to support the future release of the following in-flight tickets:
  - APM-2016[2004] - API producer support 2021/2022
- Changes to the Digital Onboarding Service:
  - DOS-938[2005] - Support section restructuring
  - DOS-1017[2006] - Remove "reject section" action
  - DOS-1049[2007] - Bug fix: developer could not download a file in the answer history
  - DOS-863[2008] - Change the styling on the section status component
  - DOS-994[2009] - Adding a validation rule for an invalid "starting worksheet"
  - DOS-995[2010] - Adding a validation rule for an invalid "api worksheet"

1999 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2000 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69129&view=results
2001 https://nhsd-jira.digital.nhs.uk/browse/APM-2915
2002 https://nhsd-jira.digital.nhs.uk/browse/APM-2943
2003 https://nhsd-jira.digital.nhs.uk/browse/APM-2980
2004 https://nhsd-jira.digital.nhs.uk/browse/APM-2016
2005 https://nhsd-jira.digital.nhs.uk/browse/DOS-938
2006 https://nhsd-jira.digital.nhs.uk/browse/DOS-1017
2007 https://nhsd-jira.digital.nhs.uk/browse/DOS-1049
2008 https://nhsd-jira.digital.nhs.uk/browse/DOS-863
2009 https://nhsd-jira.digital.nhs.uk/browse/DOS-994
2010 https://nhsd-jira.digital.nhs.uk/browse/DOS-995

- DOS-906[2011] - Test mode: reset (roll forward) products when a DAQ is imported
- DOS-1081[2012] - Bug fix: user could not resubmit a section in certain scenarios

9.5.2 Release Process

Start a release pipeline with the **refs/tags/v2022.03.14.11.27**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.03.14.11.27**
- Release Pipeline[2013]

9.5.3 Post Release Checks

- Monitor Grafana Prod dashboard[2014] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2015] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.5.4 Roll-Back

Redeploy:

- Release Tag: **v2022.03.08.21.08**
- Rollback Pipeline (see page 1607)

9.6 RFC 30032025: APIM Infrastructure Changes

9.6.1 Tickets

RFC created by Unknown User (phma5) The Release implements the changes in the following tickets:

- Configuration update in DOS:
  - ADZ-1816[2016] - settings to be used by Dev Portal re-implementation; not yet used by any deployed code
  - DOS-1054[2017] - Add Finland to the Dev Identity/DOS WAF
  - DOS-1079[2018] - Add Poland to the Dev Identity/DOS WAF

2011 https://nhsd-jira.digital.nhs.uk/browse/DOS-906
2012 https://nhsd-jira.digital.nhs.uk/browse/DOS-1081
2013 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69941&view=results
2014 https://grafana.prod.api.platform.nhs.uk
2015 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2016 https://nhsd-jira.digital.nhs.uk/browse/ADZ-1816
2017 https://nhsd-jira.digital.nhs.uk/browse/DOS-1054
2018 https://nhsd-jira.digital.nhs.uk/browse/DOS-1079

- APM-2980[2019] - Keycloak - create API Producer realm DONE Keycloak - create API Producer realm
- APM-2927[2020] - S3 artefact repository for proxy bundle DONE S3 artefact repository for proxy bundle
- APM-2899[2021] - Hello World - Add spec micro-service to paas DONE Hello World - Add spec micro-service to paas
- APM-2935[2022] - Use Ratelimiting Shared Flow in template repo. DONE Use Ratelimiting Shared Flow in template repo.
- APM-2915[2023] - Upgrade terraform & terraform providers, uplift s3 DONE Upgrade terraform & terraform providers, uplift s3
- APM-2943[2024] - Test text xml generated auth DONE Test text xml generated auth
- APM-2930[2025] - Swap over remaining expiring certs DONE Swap over remaining expiring certs

- Some small config changes to support some producer support triage work under the ticket:
  - APM-2016

9.6.2 Prepared Release

Release Tag: v2022.03.16.12.22

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70040&view=results

9.6.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.6.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2026]

2019 https://nhsd-jira.digital.nhs.uk/browse/APM-2980?src=confmacro
2020 https://nhsd-jira.digital.nhs.uk/browse/APM-2927?src=confmacro
2021 https://nhsd-jira.digital.nhs.uk/browse/APM-2899?src=confmacro
2022 https://nhsd-jira.digital.nhs.uk/browse/APM-2935?src=confmacro
2023 https://nhsd-jira.digital.nhs.uk/browse/APM-2915?src=confmacro
2024 https://nhsd-jira.digital.nhs.uk/browse/APM-2943?src=confmacro
2025 https://nhsd-jira.digital.nhs.uk/browse/APM-2930?src=confmacro
2026 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.6.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.03.14.11.27**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69941&view=results
- Redeploy *last successful non-production release* (to all envs except production)
- In case of rollback the following prod server-certs for api.service.nhs.net will need to be rolled back too:
  - prod/server-certs/api.service.nhs.uk/key - https://eu-west-2.console.aws.amazon.com/secretsmanager/home?region=eu-west-2#!/secret?name=prod%2Fserver-certs%2Fapi.service.nhs.uk%2Fkey
  - prod/server-certs/api.service.nhs.uk/crt - https://eu-west-2.console.aws.amazon.com/secretsmanager/home?region=eu-west-2#!/secret?name=prod%2Fserver-certs%2Fapi.service.nhs.uk%2Fcrt
  - This is done using the AWS CLI:
    - list-secrets[2027] can be used to find the secret entried current value and AWSPREVIOUS version_id
    - get-secret-value[2028] can then be used with the AWSPREVIOUS version_id to retrieve the previous version
    - the previous version can then be restored using put-secret-value[2029] or the AWS Secrets Manager GUI

9.7 RFC 30031802 - NHS e-RS APIM Release R1.6 Production Deployment

9.7.1 Tickets

RFC created by Mark Smith-Magee, the Release implements the changes in the following tickets:

- API modelling and OAS documentation for our 11.5 GPIT Futures endpoints.
- Fixes for some example scenarios in the OAS sandbox application.
- Wording updates for existing OAS documentation.

2027 https://docs.aws.amazon.com/cli/latest/reference/secretsmanager/list-secrets.html
2028 https://docs.aws.amazon.com/cli/latest/reference/secretsmanager/get-secret-value.html
2029 https://docs.aws.amazon.com/cli/latest/reference/secretsmanager/put-secret-value.html

Further details on APIM 1.6 Release Page[2030].

9.7.2 Prepared Release

Release Tag: 1.6.2-beta

Link to Azure devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=68364&view=results

9.7.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.7.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2031]

9.7.4 Roll-Back

- Redeploy:
  - Release Tag: 1.5.0-beta
  - **Link to Azure** devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=62222&view=results

9.8 RFC 30032113 - APIM Infrastructure Changes

9.8.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-980[2032] - add authorisation check in front of all pages and redirect to keycloak if not logged in

2030 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/386401400/APIM+1.6+Release+Page
2031 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2032 https://nhsd-jira.digital.nhs.uk/browse/DOS-980

- DOS-990[2033] - change the default sort to last updated descending on the product tables
- DOS-909[2034] - test mode: add the ability to delete test organisations and products

Prepared Release

Release Tag: **v2022.03.16.12.22+70387**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70392&view=results

9.8.2 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.8.2.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2035]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.8.3 Roll-Back

- Redeploy:
    - Release Tag: **v2022.03.16.12.22+70040**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70040&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.9 RFC 30032151 - NHS e-RS APIM Release R1.6.4 hotfix Production Deployment

9.9.1 Tickets

RFC created by Mark Smith-Magee. The Release implements the changes in the following tickets (from the APIM v1.6.4 hotfix Release build page, add to the bullet list below details from the High level summary of deliverables):

2033 https://nhsd-jira.digital.nhs.uk/browse/DOS-990
2034 https://nhsd-jira.digital.nhs.uk/browse/DOS-909
2035 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- ERSSUP-61171 - API-M Sandbox application's _status endpoint does not return a 200 HTTP status code

Further details on APIM 1.6.4[2036] hotfix Release build page.

9.9.2 Prepared Release

Release Tag: 1.6.4-beta (confirm this version with BJSS - BJSS to insert new link below)

Link to Azure devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70900&view=results

9.9.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.9.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2037]

9.9.4 Roll-Back

- Redeploy:
  - Release Tag: 1.6.2-beta
  - **Link to Azure** devops results https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=68364&view=results

9.10 RFC 30032161: APIM Infrastructure Changes

9.10.1 Tickets

RFC created by **Ben Strutt**

2036 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/406882631/APIM+1.6.4+Hotfix+Release+Page
2037 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

The release contains:

- Code changes for the following completed tickets:
    - APM-2959[2038] - Hello world products to S3
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-2016[2039] - API producer support 2021/2022
    - APM-2982[2040] - Micro service inspects and verifies Access Token

9.10.2 Release Process

Start a release pipeline with the **refs/tags/v2022.03.23.11.51**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.03.23.11.51**
- Release Pipeline[2041]

9.10.3 Post Release Checks

- Monitor Grafana Prod dashboard[2042] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2043] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.10.4 Roll-Back

Redeploy:

- Release Tag: **v2022.03.16.12.22**
- Rollback Pipeline[2044]

9.11 RFC 30032243: Digital Onboarding Service - My applications page

9.11.1 Tickets

RFC created by **Rafal Kocinski (TBC)**. The Release implements the changes in the following tickets:

2038 https://nhsd-jira.digital.nhs.uk/browse/APM-2959
2039 https://nhsd-jira.digital.nhs.uk/browse/APM-2016
2040 https://nhsd-jira.digital.nhs.uk/browse/APM-2982
2041 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70703&view=results
2042 https://grafana.prod.api.platform.nhs.uk
2043 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2044 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70703&view=results

Note: to access the tickets ask for view access to JIRA boards of Tardis (ADZ-* tickets) and Onboarding (DOS-* tickets) teams.

- Stories from epic ADZ-1535 - the actual feature being released into use in prod: re-implementation of 'My applications' page from Developer Portal[2045] on DOS platform:

Key	Summary	Status

No issues found[2046]

- DOS-965

- Stories from epic ADZ-1536 - a feature to be released later, whose partial code is included in the current deployment, but not enabled/not made available for actual use: support for self-serve uploading of the URL pointing to public JWKS keys self-hosted by external developers:

Key	Summary	Status

No issues found[2047]

9.11.2 Prepared Release

Release Tag: **v2022.04.05.07.59+72756**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72756&view=results

9.11.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2045 https://portal.developer.nhs.uk/

2046 https://nhsd-jira.digital.nhs.uk/secure/IssueNavigator.jspa?reset=true&jqlQuery=key+in+%28ADZ-1503%2C+ADZ-1504%2C+ADZ-1518%2C+ADZ-1519%2C+ADZ-1583%2C+ADZ-1688%2C+ADZ-1601%2C+ADZ-1520%2C+ADZ-1717%2C+ADZ-1515%2C+ADZ-1770%2C+ADZ-1771%2C+ADZ-1816%2C+ADZ-1808%2C+ADZ-1813%2C+ADZ-1807%2C+ADZ-1960%29++&src=confmacro

2047 https://nhsd-jira.digital.nhs.uk/secure/IssueNavigator.jspa?reset=true&jqlQuery=key+in+%28ADZ-1512%2CADZ-1510%2CADZ-1508%2CADZ-1507%29&src=confmacro

9.11.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2048]
2. Monitor Splunk APIM Dashboard[2049] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.
3. Verify that DOS is up and running and that it can be signed in to:
    a. Navigate to production DOS: https://onboarding.prod.api.platform.nhs.uk/
    b. Sign in with your production DOS/KeyCloak credentials.
    c. Verify that the sign-in was successful and that you are redirected to the 'Products' page in DOS.

9.11.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.04.05.07.59+72318**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72318&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.12 RFC 30032103 - NHS e-RS APIM Release R1.7 Production Deployment

9.12.1 Tickets

- OAS elements to support GPIT Futures updates with new endpoints as well as amendments to existing ones

Further details on APIM 1.7[2050] Release build page.

9.12.2 Prepared Release

Release Tag: 1.7.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=71283&view=results

2048 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2049 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2050 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/402853377/APIM+1.7+Release+Page

9.12.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.12.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2051]

9.12.4 Roll-Back

- Redeploy:
  - Release Tag: 1.6.4-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70900&view=results

9.13 RFC 30032221 - APIM Infrastructure Changes (DOS SSM Parameter)

9.13.1 Tickets

RFC created by Unknown User (gipa6). The Release implements the changes in the following tickets:

- DOS-1053
- **DOS-1052**
- **DOS-1090**
- **DOS-1051**

9.13.2 Prepared Release

Release Tag: **v2022.03.28.10.46**

2051 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

PTL Plan: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=71299&view=logs&j=9fe29e38-4226-5214-14c5-2bef50b7391f&t=9fe29e38-4226-5214-14c5-2bef50b7391f

PROD Plan: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=71368&view=results

9.13.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.13.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2052]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.13.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.03.23.11.51**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=70703&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.14 RFC 30032366 - APIM Infrastructure Change

9.14.1 Tickets

RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- DOS-1046
- DOS-1048
- DOS-847
- DOS-1207
- DOS-663
- DOS-1212
- DOS-1212
- DOS-1206
- DOS-1211

2052 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- DOS-1213
- DOS-1224

9.14.2 Prepared Release

Release Tag: **v2022.04.05.07.59+72020**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72020&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d

9.14.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.14.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2053]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.14.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.03.28.10.46**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=71368&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.15 RFC 30032406 - APIM Infrastructure Change

9.15.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- DOS-1246
- DOS-862

2053 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.15.2 Prepared Release

Release Tag: **v2022.04.05.07.59+72318**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72318&view=results

9.15.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.15.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2054]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.15.4 Roll-Back

- Redeploy:
  Release Tag: **v2022.04.05.07.59+72020**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72020&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d
- Redeploy *last successful non-production release* (to all envs except production)

9.16 RFC 30032474 - APIM Infrastructure Update

9.16.1 Tickets

RFC created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- DOS-1254
- DOS-1111

9.16.2 Prepared Release

v2022.04.05.07.59+72766

2054 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72766&view=results

9.16.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.16.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2055]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.16.4 Roll-Back

- Redeploy:
  - **v2022.04.05.07.59+72318**
    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72318&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.17 RFC 30032569 - Digital Onboarding Service

9.17.1 Tickets

RFC created by **Hannah Ibberson.**The Release implements the changes in the following tickets:

- DOS-976[2056] - last updated column gets overall last updated time for the product + assessment
- DOS-1249[2057] - mailto link on "no account" page updated
- DOS-1271[2058] - fix for cross-conditional questions causing saving issues
- DOS-1019[2059] - including time in last updated column
- DOS-1260[2060] - questionnaire repo calls made async

2055 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2056 https://nhsd-jira.digital.nhs.uk/browse/DOS-976
2057 https://nhsd-jira.digital.nhs.uk/browse/DOS-1249
2058 https://nhsd-jira.digital.nhs.uk/browse/DOS-1271
2059 https://nhsd-jira.digital.nhs.uk/browse/DOS-1019
2060 https://nhsd-jira.digital.nhs.uk/browse/DOS-1260

9.17.2 Prepared Release

Release Tag: **v2022.04.05.07.59+73298**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=73298&view=results

9.17.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.17.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2061]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.17.4 Roll-Back

- Redeploy:
  - **v2022.04.05.07.59+72766**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=72766&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.18 RFC 30032605 - NHS e-RS APIM Referral Release R1.8 Production Deployment

9.18.1 Tickets

- ERSSUP-61490[2062]: APIM - 55 second timeout on backend responses. Bug fix.
- ERSSUP-61143[2063]: API-M: Update inert library version utilised by the sandbox application. Security vulnerability patch.
- ERSSUP-61789[2064]: Ability to use Business Function 'AUTHORISED_APPLICATION' via User Restricted/APIM route to make successful endpoint calls

2061 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2062 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61490
2063 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61143
2064 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61789

Further details on APIM 1.8[2065] Release build page.

9.18.2 Prepared Release

Release Tag: 1.8.1

Link to Azure DevOps: Pipelines - E-Referrals-Service-API[2066]

Build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74665&view=results

9.18.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (RFC 30032821: APIM Infrastructure Changes (see page 1626))
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.18.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2067]

9.18.4 Roll-Back

- Redeploy:
  - Release Tag: 1.7.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=71283&view=results

2065 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/410565278/APIM+1.8+Release+Page
2066 https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionScope=%5CE-Referrals-Service-API
2067 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.19 RFC 30032787 - NHS e-RS APIM Patient Care Release R1.0.5 Production Deployment

9.19.1 Tickets

- New patient Care APIM Proxy including ping and status end points. Deploying this as an enabler to the 11.8 release which will launch the Wayfinder project API's.

Further details on APIM-PC 1.0[2068] Release build page.

9.19.2 Prepared Release

Release Tag: 1.0.5-alpha

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=73690&view=results

9.19.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.19.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2069]

9.19.4 Roll-Back

- Redeploy:
  - Release Tag: N/A
  - **Link to Azure DevOps:**

2068 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/410552231/APIM-PC-1.0+Release+Page
2069 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

As the Patient Care version of APIM has not been previously deployed to production before, there isn't a "Last successful release".

The rollback required is to un-deploy 1.0.5-alpha to restore the environment to the pre-deployment state - KOP-007 Emergency Single-Service Shutdown[2070].

9.20 RFC 30032815 - Digital Onboarding Service

9.20.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-979[2071] - radio button with comment question type
- DOS-1112[2072] - recourse journey: add status tags to questions
- DOS-1287[2073] - make ApplicationRepository async
- ADZ-1788 - style ApplicationDetails page
- ADZ-1933 - code changes to support ApplicationDetails go-live
- ADZ-1602 - enforce Apigee authorization for app reads / updates
- ADZ-1607 - validate JWKS url for applications

9.20.2 Prepared Release

~~Release Tag: **v2022.04.05.07.59+74609**~~

~~https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74609&view=results~~

Same release pipeline as RFC 30032821: APIM Infrastructure Changes (see page 1626)

9.20.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.20.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2074]

2070 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/119764388/KOP-007+Emergency+Single-Service+Shutdown
2071 https://nhsd-jira.digital.nhs.uk/browse/DOS-979
2072 https://nhsd-jira.digital.nhs.uk/browse/DOS-1112
2073 https://nhsd-jira.digital.nhs.uk/browse/DOS-1287
2074 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.20.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.04.05.07.59+73298**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=73298&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.21 RFC 30032821: APIM Infrastructure Changes

9.21.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - AEA-2209[2075] - Deploy EPSAT to sandbox environment with a placeholder dose to text page
    - APM-3148[2076] - Update apim-audit-bot deprecated integration by 20-09-2022
    - APM-1676[2077] - identity service - _status
    - AMB-1053[2078] - Create a Certificate Sign Request
    - APM-3061[2079] - Create ECS resources in proxygen microservice
    - APM-3149[2080] - add kvm for hrdc-api
    - APM-3109[2081] - Wayfinder create deploy target servers
    - APM-2996[2082] - Create fixture to automate proxy generation before running tests on proxygen
    - APM-2973[2083] - Tidy up github actions
    - APM-3049[2084] - Create /specs endpoint

2075 https://nhsd-jira.digital.nhs.uk/browse/AEA-2209
2076 https://nhsd-jira.digital.nhs.uk/browse/APM-3148
2077 https://nhsd-jira.digital.nhs.uk/browse/APM-1676
2078 https://nhsd-jira.digital.nhs.uk/browse/AMB-1053
2079 https://nhsd-jira.digital.nhs.uk/browse/APM-3061
2080 https://nhsd-jira.digital.nhs.uk/browse/APM-3149
2081 https://nhsd-jira.digital.nhs.uk/browse/APM-3109
2082 https://nhsd-jira.digital.nhs.uk/browse/APM-2996
2083 https://nhsd-jira.digital.nhs.uk/browse/APM-2973
2084 https://nhsd-jira.digital.nhs.uk/browse/APM-3049

    - APM-3167[2085] - Add ECS resources to S3 object
- Minor config changes to support the future release of the following in-flight tickets:
    - AEA-1571[2086] - Allow suppliers to exchange a dose syntax for a text string
    - AEA-2118[2087] - Stop getting user details from cache, and replace with a lookup
    - APM-2291[2088] - Template repo tidy up FY22-23
    - APM-3095[2089] - API producer support FY22-23

9.21.2 Release Process

Start a release pipeline with the **refs/tags/v2022.04.29.10.22**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.04.29.10.22**
- Release Pipeline[2090]

9.21.3 Post Release Checks

- Monitor Grafana Prod dashboard[2091] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2092] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.21.4 Roll-Back

Redeploy:

- Release Tag: **v2022.04.05.07.59**
- Rollback Pipeline[2093]

9.22 RFC 30032765 Covid Medical Exemptions

9.22.1 Tickets

RFC created by Joe Walker. The Release implements the changes in the following tickets:

2085 https://nhsd-jira.digital.nhs.uk/browse/APM-3167
2086 https://nhsd-jira.digital.nhs.uk/browse/AEA-1571
2087 https://nhsd-jira.digital.nhs.uk/browse/AEA-2118
2088 https://nhsd-jira.digital.nhs.uk/browse/APM-2291
2089 https://nhsd-jira.digital.nhs.uk/browse/APM-3095
2090 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=74662
2091 https://grafana.prod.api.platform.nhs.uk
2092 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2093 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=74661

- ☑ DALA-822[2094] - Rate Limiting Deprecated - COVID Medical Exemptions DONE
- ☑ DALA-858[2095] - Exemptions API - Documentation In Beta DONE

9.22.2 Prepared Release

Release Tag: **v1.0.58-alpha+73282**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=73282&view=results

9.22.3 Release Procedure

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=32719&view=results
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018

9.22.4 Rollback Procedure

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Redeploy last successful production release:
    a. Release Tag: **v1.0.52-alpha+69019**
    b. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69019&view=result[2096]
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.23 RFC 30032835- Identity Service Changes

9.23.1 Tickets

RFC was created by Unknown User (lufa1) The Release implements the changes in the following tickets:

- Code changes for the following tickets

2094 https://nhsd-jira.digital.nhs.uk/browse/DALA-822?src=confmacro
2095 https://nhsd-jira.digital.nhs.uk/browse/DALA-858?src=confmacro
2096 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=69019&view=results

- [APM-1676] identity service - _status - CDT JIRA (digital.nhs.uk)[2097] Add _status endpoint and enable status monitoring

9.23.2 Prepared Release

Release Tag: v1.0.1388-alpha[2098]v1.0.1388-alpha

Pipelines - Run v1.0.1388-alpha+74426 (azure.com)[2099]

9.23.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.23.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2100]
2. Verify the creation of a new entry on the dashboard for Identity Service _status
3. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.23.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1369-alpha[2101]v1.0.1369-alphav1.0.1369-alphav1.0.1369-alphav1.0.1369-alpha
  - Pipelines - Run v1.0.1369-alpha+60363 (azure.com)[2102]
- Redeploy *last successful non-production release* (to all envs except production)

9.24 RFC 30032959 - Digital Onboarding Service

9.24.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

2097 https://nhsd-jira.digital.nhs.uk/browse/APM-1676
2098 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74426&view=results
2099 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74426&view=results
2100 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2101 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=60362
2102 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=60363&view=results

- DOS-1113[2103] - keep the answer history thread expanded until the question is approved
- DOS-1114[2104] - hide the optional comment box for approved questions + provide an "add comment" button
- DOS-1115[2105] - hide the optional file upload + provide an "attach file" button
- DOS-977[2106] - only allow section approval if all the questions are approved
- DOS-1290[2107] - remove CaseNotesRepository
- DOS-1291[2108] - remove ContentLinksRepository

9.24.2 Prepared Release

Release Tag: **v2022.04.29.10.22+75472**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=75472&view=results

9.24.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.24.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2109]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.24.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.04.29.10.22+74662**
    - Release Pipeline[2110]
- Redeploy *last successful non-production release* (to all envs except production)

2103 https://nhsd-jira.digital.nhs.uk/browse/DOS-1113
2104 https://nhsd-jira.digital.nhs.uk/browse/DOS-1114
2105 https://nhsd-jira.digital.nhs.uk/browse/DOS-1115
2106 https://nhsd-jira.digital.nhs.uk/browse/DOS-977
2107 https://nhsd-jira.digital.nhs.uk/browse/DOS-1290
2108 https://nhsd-jira.digital.nhs.uk/browse/DOS-1291
2109 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2110 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=74662

9.25 RFC 30033031 - Digital Onboarding Service

9.25.1 Tickets

RFC created by Lee Richardson . The release implements the changes in the following tickets:

ADZ-2068

ADZ-2085

ADZ-1936

9.25.2 Prepared Release

Release Tag: v2022.05.13.09.54+75702

Pipelines - Run v2022.04.29.10.22+75980 (azure.com)[2111]

9.25.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.25.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2112]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.25.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.04.29.10.22+75472**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=75472&view=results
- Redeploy *last successful non-production release* (to all envs except production)

2111 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=75980&view=results
2112 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.26 RFC CHG0031039 - Digital Onboarding Service

9.26.1 Tickets

RFC created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-1392[2113] - fixing bug where pre-multi-api products have all the sections available
- DOS-1243[2114] - only allowing request more info for a section if it contains questions requiring more info
- DOS-1389[2115] - fixing visual bug with sort icons in table headers
- DOS-1116[2116] - adjusting the style of the answer history items

9.26.2 Prepared Release

Release Tag: **v2022.04.29.10.22+76810**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=76810&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d

9.26.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.26.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2117]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.26.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.04.29.10.22+75980**

2113 https://nhsd-jira.digital.nhs.uk/browse/DOS-1392
2114 https://nhsd-jira.digital.nhs.uk/browse/DOS-1243
2115 https://nhsd-jira.digital.nhs.uk/browse/DOS-1389
2116 https://nhsd-jira.digital.nhs.uk/browse/DOS-1116
2117 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

Pipelines - Run v2022.04.29.10.22+75980 (azure.com)[2118]

- Redeploy *last successful non-production release* (to all envs except production)

9.27 RFC CHG0031189: APIM Infrastructure Changes

9.27.1 Tickets

RFC created by *Ben Strutt*

The release contains:

- Code changes for the following completed tickets:
  - APM-2998[2119] - Deploy proxygen on AWS
  - APM-3065[2120] - Structure proxygen requests using 3-way diff request/s3/apigee
  - APM-3225[2121] - Use app display name not app name in splunk dashboards
  - APM-3199[2122] - Fix back-up apps pipeline

9.27.2 Release Process

Start a release pipeline with the **refs/tags/v2022.05.23.08.42**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.05.23.08.42**
- Release Pipeline[2123]

9.27.3 Post Release Checks

- Monitor Grafana Prod dashboard[2124] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2125] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.27.4 Roll-Back

Redeploy:

2118 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=75980&view=results
2119 https://nhsd-jira.digital.nhs.uk/browse/APM-2998
2120 https://nhsd-jira.digital.nhs.uk/browse/APM-3065
2121 https://nhsd-jira.digital.nhs.uk/browse/APM-3225
2122 https://nhsd-jira.digital.nhs.uk/browse/APM-3199
2123 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=76760
2124 https://grafana.prod.api.platform.nhs.uk
2125 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

- Release Tag: **v2022.04.29.10.22**
- Rollback Pipeline[2126]

9.28 RFC CHG0031403: APIM Infrastructure Changes

9.28.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
  - APM-3225[2127] - Use app display name not app name in splunk dashboards
  - APM-2998[2128] - Deploy proxygen on AWS
  - APM-3258[2129] - Fix cross-account AMI sharing (infra-release post-mortem)
  - APM-3199[2130] - Fix back-up apps pipeline
  - APM-3065[2131] - Structure proxygen requests using 3-way diff request/s3/apigee
  - APM-2588[2132] - 2021 Platform Pen Test - EPB-04 - Potentially unnecessary users permitted to login
- Minor config changes to support the future release of the following in-flight tickets:
  - APM-3082[2133] - Technical debt - Phase 2b FY22-23

9.28.2 Release Process

Start a release pipeline with the **refs/tags/v2022.05.26.15.33**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.05.26.15.33**
- Release Pipeline[2134]

9.28.3 Post Release Checks

- Monitor Grafana Prod dashboard[2135] (note that some of the monitoring stack may be down during deployment).

2126 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=76759
2127 https://nhsd-jira.digital.nhs.uk/browse/APM-3225
2128 https://nhsd-jira.digital.nhs.uk/browse/APM-2998
2129 https://nhsd-jira.digital.nhs.uk/browse/APM-3258
2130 https://nhsd-jira.digital.nhs.uk/browse/APM-3199
2131 https://nhsd-jira.digital.nhs.uk/browse/APM-3065
2132 https://nhsd-jira.digital.nhs.uk/browse/APM-2588
2133 https://nhsd-jira.digital.nhs.uk/browse/APM-3082
2134 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=77143
2135 https://grafana.prod.api.platform.nhs.uk

- Monitor Splunk APIM Dashboard[2136] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.28.4 Roll-Back

Redeploy:

- Release Tag: **v2022.04.29.10.22**
- Rollback Pipeline[2137]

9.29 RFC CHG0031664 - Digital Onboarding Service

9.29.1 Tickets

RFC created by **Lee Richardson**. The Release implements the changes in the following tickets:

- DOS-1342[2138] - Parentheses are not supported properly in the Radio Button > Question Settings
- ADZ-2137[2139] - Create app owner guidance
- update-to-chrome-102-e2e
- ADZ-2094[2140] - Create app - specify owner
- DOS-1341[2141] - Presentation of radio field question when indented
- DOS-1292[2142] - Correct use of async in DAQRepository
- ADZ-2124[2143] - JWKS URL validation as a warning
- DOS-1288[2144] - Correct use of async in AnswerRepository

9.29.2 Prepared Release

Release Tag: v2022.05.26.15.33[2145]

2136 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2137 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=77142
2138 https://nhsd-jira.digital.nhs.uk/browse/DOS-1342
2139 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2137
2140 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2094
2141 https://nhsd-jira.digital.nhs.uk/browse/DOS-1341
2142 https://nhsd-jira.digital.nhs.uk/browse/DOS-1292
2143 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2124
2144 https://nhsd-jira.digital.nhs.uk/browse/DOS-1288
2145 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77924&view=logs&s=02512434-2a68-53f3-8a0e-f48d4d50f045

9.29.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.29.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2146]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.29.4 Roll-Back

- Redeploy:
    - Release tag: v2022.05.26.15.33
    - Pipeline run: v2022.05.26.15.33+77159[2147]
- Redeploy *last successful non-production release* (to all envs except production)

9.30 RFC CHG0031705: APIM Infrastructure Changes

9.30.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-3268[2148] - Proxygen get all specs endpoint not giving the correct output
    - APM-3276[2149] - Update DevOps permissions
    - APM-3257[2150] - Support EPS to connect their new SDS container to LDAP
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-2586[2151] - 2021 Platform Pen Test - EPB-01 - Weak SSH configuration

2146 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2147 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77159&view=results
2148 https://nhsd-jira.digital.nhs.uk/browse/APM-3268
2149 https://nhsd-jira.digital.nhs.uk/browse/APM-3276
2150 https://nhsd-jira.digital.nhs.uk/browse/APM-3257
2151 https://nhsd-jira.digital.nhs.uk/browse/APM-2586

- APM-2587[2152] - 2021 Platform Pen Test - EPB-03 - Excessive permissions configured for shadow file
- APM-3095[2153] - API producer support FY22-23

9.30.2 Release Process

Start a release pipeline with the **refs/tags/v2022.06.07.14.56**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.06.07.14.56**
- Release Pipeline[2154]

9.30.3 Post Release Checks

- Monitor Grafana Prod dashboard[2155] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2156] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.30.4 Roll-Back

Redeploy:

- Release Tag: **v2022.05.26.15.33**
- Rollback Pipeline[2157]

9.31 RFC CHG0031779 - Digital Onboarding Service

9.31.1 Tickets

RFC created by **Lee Richardson**. The Release implements the changes in the following tickets:

- ADZ-2182[2158] - Link create app page from app list page
- DOS-1248[2159] - section submission
- DOS-978[2160] - extend "edit product name" permissions

2152 https://nhsd-jira.digital.nhs.uk/browse/APM-2587
2153 https://nhsd-jira.digital.nhs.uk/browse/APM-3095
2154 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77963&view=results
2155 https://grafana.prod.api.platform.nhs.uk
2156 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2157 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=77960
2158 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2182
2159 https://nhsd-jira.digital.nhs.uk/browse/DOS-1248
2160 https://nhsd-jira.digital.nhs.uk/browse/DOS-978

- DOS-1281[2161] - simplify the question partials
- DOS-1306[2162] - spelling mistake fix
- DOS-1333[2163] - fix rollforward section status bug

9.31.2 Prepared Release

Release Tag: v2022.06.07.14.56+78143

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78143&view=results

9.31.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.31.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2164]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.31.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.06.07.14.56+77963**
    - Release Pipeline[2165]
- Redeploy *last successful non-production release* (to all envs except production)

2161 https://nhsd-jira.digital.nhs.uk/browse/DOS-1281
2162 https://nhsd-jira.digital.nhs.uk/browse/DOS-1306
2163 https://nhsd-jira.digital.nhs.uk/browse/DOS-1333
2164 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2165 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77963&view=results

9.32 RFC 0031930 - NHS e-RS APIM Patient Care Release R1.1.6 Production Deployment

9.32.1 Tickets

- Building on top of the APIM enablers released in APIM-PC 1.0 Release Page - e-RS Live - CDT Confluence (digital.nhs.uk)[2166], this release (APIM-PC 1.1) will make the Patient Care proxy production ready.
- This release will satisfy the Business Requirement [ERSSUP-61037] APIM Client makes an API call to an e-RS Patient Care Endpoint AND e-RS Receives and Routes an API Call - CDT JIRA (digital.nhs.uk)[2167]

Further details on APIM-PC 1.1.6[2168] Release build page.

9.32.2 Prepared Release

Release Tag: 1.1.6-alpha

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77283&view=results

9.32.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.32.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2169]

2166 https://nhsd-confluence.digital.nhs.uk/display/ERS/APIM-PC+1.0+Release+Page
2167 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61037
2168 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/423305425/APIM-PC-1.1+Release+Page
2169 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.32.4 Roll-Back

- Redeploy:
  - Release Tag: 1.0.5-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=73690&view=results

9.33 RFC 0031929 - NHS e-RS APIM Referral Release R1.9 Production Deployment

9.33.1 Tickets

- ERSSUP-56595[2170] - Discovery: APIGEE configuration for ERSSUP-56026: APIM Client makes an Application-restricted Open API call to a read-only e-RS Endpoint[2171]
- ERSSUP-61927[2172] - REVERT disallowing HTML file changes[2173]
- ERSSUP-58836[2174] - Add Application-restricted Apigee product[2175]
- ERSSUP-60546[2176] - Productionisation - e-RS Alpha Checklist[2177]

Further details on APIM 1.9 Release[2178] build page.

9.33.2 Prepared Release

Release Tag: 1.9.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=76521&view=results

9.33.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.

2170 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-56595?src=confmacro
2171 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-56595?src=confmacro
2172 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61927?src=confmacro
2173 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61927?src=confmacro
2174 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58836?src=confmacro
2175 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58836?src=confmacro
2176 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60546?src=confmacro
2177 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60546?src=confmacro
2178 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/427890844/APIM+1.9+Release+Page

2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.33.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2179]

9.33.4 Roll-Back

- Redeploy:
  - Release Tag: 1.8.1-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74665&view=results

9.34 CHG0031941 - Digital Onboarding Service

9.34.1 Tickets

RFC created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-1463[2180] - developers answers hidden on radio buttons in review view

9.34.2 Prepared Release

Release Tag: **v2022.06.07.14.56+78425**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78425&view=results

9.34.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2179 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2180 https://nhsd-jira.digital.nhs.uk/browse/DOS-1463

9.34.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2181]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.34.4 Roll-Back

- Redeploy:
    - Release Tag: v2022.06.07.14.56+78143

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78143&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.35 RFC CHG0032105 - NHS App API change

9.35.1 Tickets

RFC created by Christopher Speers The Release implements the changes in the following tickets:

- NHSO-20789: Increase rate limits
- NHSO-18276: Removes 404 internal endpoints filters

9.35.2 Prepared Release

Release Tag: v1.0.182-alpha+78021

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78021&view=results

9.35.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2181 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.35.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2182]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.35.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.143-alpha

9.36 CHG0032137 - Digital Onboarding Service

9.36.1 Tickets

RFC created by **Phillip Macdonald.** The Release implements the changes in the following tickets:

- DOS-1368 - Developer can view independent API approvals
- DOS-1350 - Onboarding approval can be given to APIs seperately
- DOS-1348 - Fixes issue with UAT product re-configuration causing accessible products
- DOS-1306 - spelling mistake in Keycloak account creation screen

9.36.2 Prepared Release

Release Tag: **v2022.06.09.09.20**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78984&view=results

9.36.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.36.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2183]

2182 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2183 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.36.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.06.07.14.56+78425**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=78425&view=results
- Redeploy *last successful non-production release* (to all envs except production)

9.37 CHG0032180: APIM Infrastructure Changes

9.37.1 Tickets

RFC CHG0032180[2184] created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-3252[2185] - Wire up SDS DEP ldap environment to SDS DEV apigee environment
    - APM-3227[2186] - Help SCR FHIR API team fix separate auth pattern
    - DALA-979[2187] - Update the service search key value map paths
    - AMB-1132[2188] - Find NHS Root CA certificate
    - APM-2586[2189] - 2021 Platform Pen Test - EPB-01 - Weak SSH configuration
    - DOS-1306[2190] - Spelling mistake in an error on the developer account "create account" page
- Minor config changes to support the future release of the following in-flight tickets:
    - AMB-1137[2191] - Extend separate auth refresh token maximum time to 12 hours
    - APM-3095[2192] - API producer support FY22-23
    - ADZ-2069[2193] - Create app - specify environment

2184 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do%3Fsys_id=3c5f7e581bdcd9105189855bd34bcb9f%26sysparm_stack=change_request_list.do%3Fsysparm_query=active=true
2185 https://nhsd-jira.digital.nhs.uk/browse/APM-3252
2186 https://nhsd-jira.digital.nhs.uk/browse/APM-3227
2187 https://nhsd-jira.digital.nhs.uk/browse/DALA-979
2188 https://nhsd-jira.digital.nhs.uk/browse/AMB-1132
2189 https://nhsd-jira.digital.nhs.uk/browse/APM-2586
2190 https://nhsd-jira.digital.nhs.uk/browse/DOS-1306
2191 https://nhsd-jira.digital.nhs.uk/browse/AMB-1137
2192 https://nhsd-jira.digital.nhs.uk/browse/APM-3095
2193 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2069

9.37.2 Release Process

Start a release pipeline with the **refs/tags/v2022.06.17.10.11**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.06.17.10.11**
- Release Pipeline[2194]

9.37.3 Post Release Checks

- Monitor Grafana Prod dashboard[2195] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2196] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.37.4 Roll-Back

Redeploy:

- Release Tag: **v2022.06.07.14.56**
- Rollback Pipeline[2197]

9.38 CHG0032379: APIM Infrastructure Changes

9.38.1 Tickets

CHG0032379[2198] created by **Phillip Macdonald**

The release contains:

- DOS-1102 - self serve signup to Digital Onboarding
- DOS-1312 - APIs can have statuses that affect the journey - limited/discontinued/available/retired
- DOS-1496 - An API using a section that is referenced by other section questions that are not included in the API question set should still be able to be answered without needing to update questions it cannot access.
- DOS-1526 - Fix that prevents DAQ publishing in production
- DOS-1530 - Add Sweden to the DOS/Apigee WAF

2194 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=79068&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d

2195 https://grafana.prod.api.platform.nhs.uk

2196 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

2197 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=79064

2198 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do%3Fsys_id=ac6c96961bd4111093bfc88b274bcbc9%26sysparm_stack=change_request_list.do%3Fsysparm_query=active=true

- ADZ-2176 - Rewords JWKS errors and warnings
- ADZ-2175 - Changes when JWKS warnings show
- ADZ-2069 - Adds environment field to create app page

9.38.2 Release Process

Start a release pipeline with the **refs/tags/v2022.06.27.09.49**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.06.27.09.49**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=79962&view=results

9.38.3 Post Release Checks

- Monitor Grafana Prod dashboard[2199] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2200] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.38.4 Roll-Back

Redeploy:

- Release Tag: **v2022.06.17.10.11**
- Release Pipeline[2201]

9.39 CHG0032484: Digital Onboarding Service

9.39.1 Tickets

CHG0032484[2202] created by **Lee Richardson**

The release contains:

- ADZ-2087 - Public key is read only for prod apps
- DOS-1461- Changing sort of Assessment Tracker to use 24hr clock
- DOS-1339- Uploaded file is not removed when review is cleared or changed
- DOS-1237- Validation message for Select question type is inaccurate

2199 https://grafana.prod.api.platform.nhs.uk

2200 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

2201 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=79068&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d

2202 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b104f4cf1bd855105189855bd34bcb2f

9.39.2 Release Process

Start a release pipeline with the **refs/tags/v2022.06.27.09.49**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.06.27.09.49**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80319&view=results

9.39.3 Post Release Checks

- Monitor Grafana Prod dashboard[2203] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2204] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.39.4 Roll-Back

Redeploy:

- Release Tag: **v2022.06.27.09.49**
- Release Pipeline[2205]

9.40 CHG0032533: APIM Infrastructure Changes

9.40.1 Tickets

CHG created by Dominic Platt The Release implements the changes in the following tickets:

- APM-3119[2206] - Onboard AWS WAF logs DONE

9.40.2 Release Process

1. Start a release pipeline with the **refs/tags/v2022.06.30.14.12**, click **Review** and **Approve** the prod release.
    - Release Tag: **v2022.06.30.14.12**

2203 https://grafana.prod.api.platform.nhs.uk
2204 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2205 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80216&view=results
2206 https://nhsd-jira.digital.nhs.uk/browse/APM-3119?src=confmacro

9.40.2.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2207] (note that some of the monitoring stack may be down during deployment).
2. Monitor Splunk APIM Dashboard[2208] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.40.3 Roll-Back

- Redeploy:
  - Release Tag: **v2022.06.27.09.49**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=79962&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.41 CHG0032690 Immunisation History API

9.41.1 Tickets

RFC created by Joe Walker. The Release implements the changes in the following tickets

- DALA-967[2209] - Request / Target Claim Authorisation Model In DDA DONE
- DALA-972[2210] - Request / Target Claim Authorisation Model In APIGee DONE
- DALA-1038[2211] - Align custom attribute implementation to APIM's spec DONE
- DALA-1042[2212] - Proxy: Consume authorised_targets from array and coerce into header DONE

9.41.2 Prepared Release

Release Tag: v1.0.443-alpha[2213]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81552&view=results

2207 https://grafana.prod.api.platform.nhs.uk/
2208 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2209 https://nhsd-jira.digital.nhs.uk/browse/DALA-967?src=confmacro
2210 https://nhsd-jira.digital.nhs.uk/browse/DALA-972?src=confmacro
2211 https://nhsd-jira.digital.nhs.uk/browse/DALA-1038?src=confmacro
2212 https://nhsd-jira.digital.nhs.uk/browse/DALA-1042?src=confmacro
2213 https://github.com/NHSDigital/immunisation-history-api/tree/v1.0.443-alpha

9.41.3 Release Process

1. Release messaging in line with KOP-018
2. Approve and run https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81552&view=logs&s=c2f23cc9-b49b-5971-bdfa-5e6430153f6c
3. Observe logs for 15 mins to catch any potential issues
4. Success/Failure messaging in line with KOP-018
5. Decorate all of our consumer apps with the following Custom Attribute named "apim-app-flow-vars" and value

```
{"immunisation-history": {"authorised_targets": ["COVID19"]}}
```

9.41.4 Roll-Back

1. Failure messaging in line with KOP-018
2. Escalate to 2nd line
3. Redeploy last successful production release:
    a. Release Tag: **v1.0.400-alpha** (Aug 2021)
4. Observe logs for 15 mins to catch any potential issues
5. Service restoration messaging as in KOP-018

9.42 CHG0032731: Digital Onboarding Service

9.42.1 Tickets

CHG0032731[2214] created by **Lee Richardson**

The release contains:

- ADZ-2087 - Public key is read only for prod apps
- ADZ-2274- Public key is read only if hosted by us
- DOS-1461- Changing sort of Assessment Tracker to use 24hr clock
- DOS-1339- Uploaded file is not removed when review is cleared or changed
- DOS-1237- Validation message for Select question type is inaccurate
- DOS-1295- Correct use of async in ProductAssessmentRepository
- DOS-1289-Correct use of async in APIDefinitionRepository
- DOS-1519-Tool available for DAQ testing
- approve-api-updates-last-updated-time

2214 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6df0e05d1be4d11093bfc88b274bcb2c

- DOS-1352Surface approved APIs on assessment tracker
- DOS-1293Correct use of async in EventLogRepository
- ADZ-2274Public key is read only if hosted by us

9.42.2 Release Process

Start a release pipeline with the **refs/tags/v2022.06.30.14.12**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.06.30.14.12**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81521&view=results

9.42.3 Post Release Checks

- Monitor Grafana Prod dashboard[2215] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2216] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.42.4 Roll-Back

Redeploy:

- Release Tag: v2022.06.30.14.12
- Release Pipeline[2217]

9.43 CHG0032859 - APIM Infrastructure Updates

9.43.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-1238[2218] - request more info for a section without full review
- DOS-1371[2219] - promote pipeline can no longer be run without manual approval
- DOS-1294[2220] - correct use of async in Organisation repository
- DOS-1296[2221] - correct use of async in Static repository
- DOS-1576[2222] - add Kenya to the WAF

2215 https://grafana.prod.api.platform.nhs.uk
2216 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2217 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81224&view=results
2218 https://nhsd-jira.digital.nhs.uk/browse/DOS-1238
2219 https://nhsd-jira.digital.nhs.uk/browse/DOS-1371
2220 https://nhsd-jira.digital.nhs.uk/browse/DOS-1294
2221 https://nhsd-jira.digital.nhs.uk/browse/DOS-1296
2222 https://nhsd-jira.digital.nhs.uk/browse/DOS-1576

9.43.2 Prepared Release

Release Tag: **v2022.07.06.16.06**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82037&view=results

9.43.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.43.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2223]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.43.4 Roll-Back

- Start a release pipeline with the **refs/tags/v2022.06.30.14.12**, click **Review** and **Approve** the prod release.
    - Release Tag: **v2022.06.30.14.12**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81521&view=results

9.44 RFC 0032744 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment

9.44.1 Tickets

SPCA work relating to APIM Client Retrieves Service Provider Clinicians Available to a Service Provider Clinician Admin. Specifically:

- ERSSUP-62066[2224] - Change default branch to develop on e-referrals-api github[2225]
- ERSSUP-61146[2226] - API-M: Update eslint version depended on by the sandbox application[2227]

2223 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2224 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62066?src=confmacro
2225 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62066?src=confmacro
2226 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61146?src=confmacro
2227 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61146?src=confmacro

- ERSSUP-61831[2228] - APIM Stack - Investigate and fix Dependabot Critical Alerts[2229]
- ERSSUP-61223[2230] - API-M: Add dependabot configuration to the referrals APIM repository[2231]
- ERSSUP-61500[2232] - API-M: add Sandbox tests for the Appointment endpoints[2233]
- ERSSUP-61498[2234] - API-M: add Sandbox tests for the Advice and Guidance endpoints[2235]
- ERSSUP-61502[2236] - API-M: add Sandbox tests for the Attachment endpoints[2237]
- ERSSUP-61503[2238] - API-M: add Sandbox tests for the Service endpoints[2239]
- ERSSUP-61504[2240] - API-M: add Sandbox tests for the Reference Data endpoint[2241]
- ERSSUP-58534[2242] - API-M Add _status implementation to the sandbox application[2243]
- ERSSUP-62632[2244] - Update apigee tool to fix json schema vulnerability[2245]
- ERSSUP-62312[2246] - APIM Stack - Fix Base npm High Alerts[2247]
- ERSSUP-61495[2248] - API-M: Add Sandbox tests for the worklist endpoints[2249]
- ERSSUP-61496[2250] - API-M: add Sandbox tests for the referral endpoints[2251]
- ERSSUP-62313[2252] - APIM Stack - Fix Base poetry High Alerts[2253]
- ERSSUP-61345[2254] - Development of ERSSUP-60244: OAS File Overview and Examples for 'Retrieve Service Provider Clinicians Available to a Service Provider Clinician Admin' Endpoint[2255]
- ERSSUP-61006[2256] - API-M: Add internal-dev-sandbox deployment target to the release process[2257]

2228 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61831?src=confmacro
2229 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61831?src=confmacro
2230 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61223?src=confmacro
2231 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61223?src=confmacro
2232 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61500?src=confmacro
2233 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61500?src=confmacro
2234 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61498?src=confmacro
2235 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61498?src=confmacro
2236 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61502?src=confmacro
2237 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61502?src=confmacro
2238 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61503?src=confmacro
2239 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61503?src=confmacro
2240 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61504?src=confmacro
2241 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61504?src=confmacro
2242 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58534?src=confmacro
2243 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58534?src=confmacro
2244 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62632?src=confmacro
2245 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62632?src=confmacro
2246 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62312?src=confmacro
2247 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62312?src=confmacro
2248 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61495?src=confmacro
2249 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61495?src=confmacro
2250 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61496?src=confmacro
2251 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61496?src=confmacro
2252 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62313?src=confmacro
2253 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62313?src=confmacro
2254 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61345?src=confmacro
2255 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61345?src=confmacro
2256 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61006?src=confmacro
2257 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61006?src=confmacro

- ERSSUP-60076[2258] - API-M Test Harness : Mock FT user ID setup[2259]

Further details on APIM 1.10.0[2260] Release build page.

9.44.2 Prepared Release

Release Tag: 1.10.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80140&view=results

9.44.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.44.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2261]

9.44.4 Roll-Back

- Redeploy:
  - Release Tag: 1.9.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=76521&view=results

2258 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60076?src=confmacro
2259 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60076?src=confmacro
2260 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/427908185/APIM-R-1.10+Release+Page
2261 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.45 RFC 0032733 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment

9.45.1 Tickets

APIM-PC release containing a new Service API Cache in APIM and several items that were part of the Wayfinder MVP, but were not critical for APIM-PC 1.1. Specifically:

- ERSSUP-61771[2262] - > APIM - Update README file[2263]
- ERSSUP-62165[2264] - Patient Care Sending a request with an invalid NHSD-Target-Identifier leads to a 500 error response.[2265]
- ERSSUP-62570[2266] - APIM - Patient Care API Rate limit incorrect for Prod[2267]
- ERSSUP-62493[2268] - APIM - Increase Rate Limiting (Patient API)[2269]

Further details on APIM 1.2.0[2270] Release build page.

9.45.2 Prepared Release

Release Tag: 1.2.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80008&view=results

9.45.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

2262 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61771?src=confmacro
2263 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61771?src=confmacro
2264 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62165?src=confmacro
2265 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62165?src=confmacro
2266 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62570?src=confmacro
2267 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62570?src=confmacro
2268 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62493?src=confmacro
2269 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62493?src=confmacro
2270 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/444894765/APIM-PC-1.2+Release+Page

9.45.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2271]

9.45.4 Roll-Back

- Redeploy:
  - Release Tag: 1.1.6-alpha
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77283&view=results

9.46 CHG0033065 - APIM Infrastructure Updates

9.46.1 Tickets

CHG created by Dominic Platt**.** The Release implements the changes in the following tickets:

- APM-2590[2272] - 2021 Platform Pen Test - EPB-02 - IP Forwarding enabled DONE - Disable IP Forwarding

9.46.2 Prepared Release

Release Tag: **v2022.07.13.15.56**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82656&view=lo[…]-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d[2273]

Terraform only release.

9.46.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2271 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2272 https://nhsd-jira.digital.nhs.uk/browse/APM-2590?src=confmacro

2273 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82656&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d

9.46.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2274]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.46.4 Roll-Back

- Start a release pipeline with the **refs/tags/v2022.07.06.16.06**, click **Review** and **Approve** the prod release.
    - Release Tag: **v2022.07.06.16.06**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82072&view=results

9.47 CHG0033227 - NHS e-RS APIM Referral Release R1.10.0 Production Deployment

9.47.1 Tickets

SPCA work relating to APIM Client Retrieves Service Provider Clinicians Available to a Service Provider Clinician Admin. Specifically:

- ERSSUP-62066[2275] - Change default branch to develop on e-referrals-api github[2276]
- ERSSUP-61146[2277] - API-M: Update eslint version depended on by the sandbox application[2278]
- ERSSUP-61831[2279] - APIM Stack - Investigate and fix Dependabot Critical Alerts[2280]
- ERSSUP-61223[2281] - API-M: Add dependabot configuration to the referrals APIM repository[2282]
- ERSSUP-61500[2283] - API-M: add Sandbox tests for the Appointment endpoints[2284]
- ERSSUP-61498[2285] - API-M: add Sandbox tests for the Advice and Guidance endpoints[2286]

2274 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2275 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62066?src=confmacro
2276 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62066?src=confmacro
2277 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61146?src=confmacro
2278 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61146?src=confmacro
2279 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61831?src=confmacro
2280 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61831?src=confmacro
2281 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61223?src=confmacro
2282 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61223?src=confmacro
2283 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61500?src=confmacro
2284 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61500?src=confmacro
2285 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61498?src=confmacro
2286 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61498?src=confmacro

- ERSSUP-61502[2287] - API-M: add Sandbox tests for the Attachment endpoints[2288]
- ERSSUP-61503[2289] - API-M: add Sandbox tests for the Service endpoints[2290]
- ERSSUP-61504[2291] - API-M: add Sandbox tests for the Reference Data endpoint[2292]
- ERSSUP-58534[2293] - API-M Add _status implementation to the sandbox application[2294]
- ERSSUP-62632[2295] - Update apigee tool to fix json schema vulnerability[2296]
- ERSSUP-62312[2297] - APIM Stack - Fix Base npm High Alerts[2298]
- ERSSUP-61495[2299] - API-M: Add Sandbox tests for the worklist endpoints[2300]
- ERSSUP-61496[2301] - API-M: add Sandbox tests for the referral endpoints[2302]
- ERSSUP-62313[2303] - APIM Stack - Fix Base poetry High Alerts[2304]
- ERSSUP-61345[2305] - Development of ERSSUP-60244: OAS File Overview and Examples for 'Retrieve Service Provider Clinicians Available to a Service Provider Clinician Admin' Endpoint[2306]
- ERSSUP-61006[2307] - API-M: Add internal-dev-sandbox deployment target to the release process[2308]
- ERSSUP-60076[2309] - API-M Test Harness : Mock FT user ID setup[2310]

Further details on APIM 1.10.0[2311] Release build page.

9.47.2 Prepared Release

Release Tag: 1.10.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80140&view=results

---

2287 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61502?src=confmacro
2288 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61502?src=confmacro
2289 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61503?src=confmacro
2290 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61503?src=confmacro
2291 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61504?src=confmacro
2292 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61504?src=confmacro
2293 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58534?src=confmacro
2294 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-58534?src=confmacro
2295 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62632?src=confmacro
2296 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62632?src=confmacro
2297 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62312?src=confmacro
2298 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62312?src=confmacro
2299 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61495?src=confmacro
2300 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61495?src=confmacro
2301 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61496?src=confmacro
2302 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61496?src=confmacro
2303 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62313?src=confmacro
2304 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62313?src=confmacro
2305 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61345?src=confmacro
2306 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61345?src=confmacro
2307 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61006?src=confmacro
2308 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61006?src=confmacro
2309 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60076?src=confmacro
2310 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60076?src=confmacro
2311 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=427908185

9.47.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.47.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2312]

9.47.4 Roll-Back

- Redeploy:
  - Release Tag: 1.9.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=76521&view=results

9.48 CHG0033228 - NHS e-RS APIM Patient Care Release R1.2.0 Production Deployment

9.48.1 Tickets

APIM-PC release containing a new Service API Cache in APIM and several items that were part of the Wayfinder MVP, but were not critical for APIM-PC 1.1. Specifically:

- ERSSUP-61771[2313] - > APIM - Update README file[2314]
- ERSSUP-62165[2315] - Patient Care Sending a request with an invalid NHSD-Target-Identifier leads to a 500 error response.[2316]

2312 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2313 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61771?src=confmacro
2314 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61771?src=confmacro
2315 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62165?src=confmacro
2316 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62165?src=confmacro

- ERSSUP-62570[2317] - APIM - Patient Care API Rate limit incorrect for Prod[2318]
- ERSSUP-62493[2319] - APIM - Increase Rate Limiting (Patient API)[2320]

Further details on APIM 1.2.0[2321] Release build page.

9.48.2 Prepared Release

Release Tag: 1.2.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80008&view=results

9.48.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.48.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2322]

9.48.4 Roll-Back

- Redeploy:
  - Release Tag: 1.1.6-alpha
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=77283&view=results

2317 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62570?src=confmacro
2318 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62570?src=confmacro
2319 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62493?src=confmacro
2320 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62493?src=confmacro
2321 https://nhsd-confluence.digital.nhs.uk/display/ERS/APIM-PC+1.2+Release+Page
2322 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.49 CHG0033306 - Digital Onboarding Service

9.49.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following ti

- DOS-1651-Don't (sic) use the negative contraction on the sign-in page
- ADZ-2276-Error when editing a newly created App
- DOS-1377-Unable to promote an approver/assessor to onboarding lead
- ADZ-2099-DOS: Key mgmt: generate keys in the browser
- DOS-1253-Move GA properties into a page filter
- ADZ-2186-Edit my app - issue with Apigee edit page
- ADZ-2283-Maintenance mode exclusion
- DOS-1357-An invite can be created for someone with a pending invite in the same organisation
- ADZ-2145-API tests for unsupported API call
- DOS-1646-Add Singapore to the WAF
- DOS-1705-Add Denmark to the WAF
- DOS-1620-Add has to the self-serve journey content
- DOS-1456-How long does a developer take to complete a section metrics trigger
- DOS-1605-How long does it take for a product to be fully approved metrics trigger

9.49.2 Prepared Release

Release Tag: **v2022.07.25.14.26**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83199&view=results

9.49.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.49.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2323]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2323 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.49.4 Roll-Back

- Release Tag: **v2022.07.13.15.56**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82656&view=lo[…]-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d[2324]

9.50 CHG0033524 - Digital Onboarding Service

9.50.1 Tickets

CHG created by **Gabriel Schimanski** The Release implements the changes in the following tickets:

- DOS-1611[2325] - Answer submission metric
- DOS-1655[2326] - Remove allow email access screen
- ADZ-2282[2327] - Jwks url wording
- ADZ-2271[2328] - Disable legacy create app flow
- DOS-1335[2329] - Enable developers to add products
- DOS-1603[2330] - Increase Fargate CPU size to prevent 100% behaviour during a roll forward

9.50.2 Prepared Release

Release Tag: **v2022.08.01.11.28**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=84271&view=results

9.50.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2324 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=82656&view=logs&j=e0d7ee14-ed89-56f4-440c-1a63d87c013a&t=4fc9994a-d8ae-5b3f-26ee-03d9d9e95c5d
2325 https://nhsd-jira.digital.nhs.uk/browse/DOS-1611
2326 https://nhsd-jira.digital.nhs.uk/browse/DOS-1655
2327 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2282
2328 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2271
2329 https://nhsd-jira.digital.nhs.uk/browse/DOS-1335
2330 https://nhsd-jira.digital.nhs.uk/browse/DOS-1603

9.50.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2331]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.50.4 Roll-Back

- Redeploy:
  Release Tag: **v2022.07.25.14.26**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83199&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.51 CHG0033763 - Digital Onboarding Service

9.51.1 Tickets

CHG created by **Giles Papworth** The Release implements the changes in the following tickets:

- DOS-1638[2332] - Turn off maintenance mode automatically if roll forward successful.
- DOS-1602[2333] - Improve roll forwards speed
- DOS-1614[2334] - How many times is a section approved/not approved
- ADZ-2146[2335] - JWKS URL - Positive confirmation of validation
- ADZ-2273[2336] - Public key read only for prod apps

9.51.2 Prepared Release

Release Tag: **v2022.08.01.11.28**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=85209&view=results

9.51.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

2331 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2332 https://nhsd-jira.digital.nhs.uk/browse/DOS-1638
2333 https://nhsd-jira.digital.nhs.uk/browse/DOS-1602
2334 https://nhsd-jira.digital.nhs.uk/browse/DOS-1614
2335 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2146
2336 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2273

2. Click the Release Link above and approve the manual step to deploy the release to prod

9.51.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2337]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.51.4 Roll-Back

- DOS-Rollback Procedure: Release Rollback Procedure[2338]
    - Roll back production environment using instructions
- Redeploy:
  Release Tag: **v2022.08.01.11.28**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=84271&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.52 CHG0033871 Sync-wrap API changes

9.52.1 Tickets

- APM-3416[2339]

9.52.2 Prepared Release

Release Tag: v1.0.126-alpha

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=85523&view=results

9.52.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.**
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2337 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2338 https://nhsd-confluence.digital.nhs.uk/spaces/DRM/pages/402858080/Release+Rollback+Procedure
2339 https://nhsd-jira.digital.nhs.uk/browse/APM-3416

9.52.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2340]

9.52.4 Roll-Back

Redeploy:
Release Tag: v1.0.117-alpha
Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=32340&view=results

9.53 CHG0033320 - NHS e-RS APIM Referral Release R1.11.0 Production Deployment

9.53.1 Content

High-level summary:

- OAS Documentation change to accommodate the addition of service details to the A008 endpoint response, along with API design.
- A&G Clinical Information endpoint now documented in OAS.
- HTML and HTM files can no longer be uploaded, and this is now reflected in the OAS docs.
- Practitioner profile URL corrected in A040 response in OAS.
- Sandbox automation tests for A043 A&G Clinical Information endpoint.

Tickets:

- ERSSUP-61006[2341] - API-M: Add internal-dev-sandbox deployment target to the release process
- ERSSUP-62971[2342] - APIM-R Update dependabot config to not rebase all branches
- ERSSUP-62716[2343] - OAS modelling for ERSSUP-62621: e-RS provides service information in response for A008 (Referrals for review) endpoint
- ERSSUP-62804[2344] - API-M Add sandbox test for the get Practitioner endpoint
- ERSSUP-63226[2345] - xAPI APIM: Poetry unable to find installation candidates for regex package

2340 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2341 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61006?src=confmacro
2342 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62971?src=confmacro
2343 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62716?src=confmacro
2344 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62804?src=confmacro
2345 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-63226?src=confmacro

- ERSSUP-62678[2346] - Update XAPI spec w.r.t. ERSSUP-60430 (Remove HTML & HTMContent-Type) - Merge into Release 12.0
- ERSSUP-62742[2347] - API-M: update Sandbox tests for A043 Retrieve A&G Clinical Information (/STU3/CommunicationRequest/{ubrn}/$ers.generateCRI)
- ERSSUP-60513[2348] - API-M: Move integration testing from eRS Alpha environment
- ERSSUP-61551[2349] - [OAS Documentation] - Retrieve A&G Clinical Information endpoint not documented in OAS (/STU3/CommunicationRequest/{ubrn}/$ers.generateCRI)
- ERSSUP-62787[2350] - A040 OAS documentation - practitioner profile URL incorrect on example response

Further details on the APIM 1.11.0[2351] Release build page.

9.53.2 Prepared Release

Release Tag: 1.11.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83320&view=results

9.53.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.53.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2352]

2346 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62678?src=confmacro
2347 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62742?src=confmacro
2348 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-60513?src=confmacro
2349 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-61551?src=confmacro
2350 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62787?src=confmacro
2351 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/457616266/APIM-R-1.11+Release+Page
2352 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.53.4 Roll-Back

- Redeploy:
  - Release Tag: 1.10.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80140&view=results

9.54 CHG0033321 - NHS e-RS APIM Patient Care Release R1.3.0 Production Deployment

9.54.1 Content

High-level summary:

- A rate limit change to improve the accuracy of rate limiting as advised by APIM
- TRM updates

Tickets:

- ERSSUP-63261[2353] - APIM-PC Builds Failing Installing python
- ERSSUP-62801[2354] - Investigate and Fix Critical Alerts in Patient Care Repo
- ERSSUP-62802[2355] - Investigate and Fix High Severity Alerts in Patient Care Repo
- ERSSUP-63248[2356] - Wayfinder - "fix" APIM rate limiting configuration inline with APIM team advice
- ERSSUP-62936[2357] - Removed unused APIM-PC policies and fix JSON error response when Auth failure
- ERSSUP-62237[2358] - APIM Patient Care - Automated tests to cover APIM policies/logic
- ERSSUP-63281[2359] - Administer APIM-PC 1.3 cut

Further details on APIM 1.3.0[2360] Release build page.

9.54.2 Prepared Release

Release Tag: 1.3.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83630&view=results

2353 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-63261?src=confmacro
2354 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62801?src=confmacro
2355 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62802?src=confmacro
2356 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-63248?src=confmacro
2357 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62936?src=confmacro
2358 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-62237?src=confmacro
2359 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-63281?src=confmacro
2360 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/457616341/APIM-PC-1.3+Release+Page

9.54.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.54.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2361]

9.54.4 Roll-Back

- Redeploy:
  - Release Tag: 1.2.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=80008&view=results

9.55 CHG0033762 - Apigee environment attributes

9.55.1 Tickets

CHG created by **Lee Richardson** The Release implements the changes in the following tickets:

- ADZ-2070

9.55.2 Prepared Release

https://github.com/NHSDigital/drm-dap/tree/ADZ-2070-populate-apigee-envs/scripts/populateApigeeEnvs

2361 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.55.3 Release Process

1. Ensure your Apigee user has the permissions required to read all apps, read all products, read all developers, and write app attributes.
2. Take note of some existing attributes. See Post Release Checks.
3. Take backup. https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-015+Restore+Apigee+App+from+Back-up
4. Run `populateApigeeEnvs.py`. See readme. https://github.com/NHSDigital/drm-dap/tree/ADZ-2070-populate-apigee-envs/scripts/populateApigeeEnvs
5. Attach contents of generated "log" directory to this RFC.

9.55.3.1 Post Release Checks

Check that the following apps have their original attributes as well the new one.

https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/a7e03ceb-2570-42e6-a7c9-624ea15269c8
environment - int

https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/defa5e86-73c0-4054-8891-4bc4db95f1a8
environment - prod

9.55.4 Roll-Back

If apps have lost their attributes, restore backup. KOP-015 Restore Apigee App from Back-up[2362]
A roll back script is provided but should not be relied on as it uses the same functionality to replace the lost attributes and there's no reason to believe that it wouldn't encounter the same bug which caused the loss of attributes.

9.56 CHG0034015 - APIM Infrastructure Updates - Add patient care aggregator prod target server

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D73f928721bd11518c994206fe54bcba7%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.56.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

2362 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/169708706/KOP-015+Restore+Apigee+App+from+Back-up

- APM-3436[2363] - Assist patient-care-aggregator with prod cert/key DONE

9.56.2 Prepared Release

Release Tag: v2022.08.17.07.57

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86131&view=results

9.56.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.56.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2364]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.56.4 Roll-Back

- Redeploy:
    - Release Tag: v2022.06.27.09.49 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=79962&view=results

9.57 CHG0034157 - Digital Onboarding Service

ServiceNow CHG request[2365]

9.57.1 Tickets

CHG created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-1709[2366] DAQ validator rule for no answerable questions
- DOS-1609[2367] Added user email row to user details screen

2363 https://nhsd-jira.digital.nhs.uk/browse/APM-3436?src=confmacro

2364 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2365 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do%3Fsys_id=5d67b5981b2d1510d1817662164bcb22%26sysparm_stack=change_request_list.do%3Fsysparm_query=active=true

2366 https://nhsd-jira.digital.nhs.uk/browse/DOS-1709

2367 https://nhsd-jira.digital.nhs.uk/browse/DOS-1609

- DOS-1775[2368] Replace WAF allow list with deny list
- DOS-1506[2369] Turn off emails in test mode
- ADZ-2294[2370]Alteration of My Applications page

9.57.2 Prepared Release

Release Tag: **v2022.08.08.14.59**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86575&view=results

9.57.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.57.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2371]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.57.4 Roll-Back

- Release Tag: **v2022.08.01.11.28**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=85209&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.58 CHG 0034302 - Immunisation History API

<Link to ServiceNow Change request>

9.58.1 Tickets

CHG created by Joe Walker The Release implements the changes in the following tickets:

- ☑ DALA-1110[2372] - Pass full request URL from proxy to backend DONE

2368 https://nhsd-jira.digital.nhs.uk/browse/DOS-1775
2369 https://nhsd-jira.digital.nhs.uk/browse/DOS-1506
2370 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2294
2371 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2372 https://nhsd-jira.digital.nhs.uk/browse/DALA-1110?src=confmacro

9.58.2 Prepared Release

Release Tag: **v1.0.458-alpha+86878**

Link to Azure devops results, https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86878&view=results

9.58.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.58.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2373]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.58.4 Roll-Back

- Redeploy:
    - Release Tag: **v1.0.443-alpha+81552**
    - <**Link** to Azure devops https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=81552&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.59 CHG0034551 - Digital Onboarding Service

Servicenow CHG Request[2374]

9.59.1 Tickets

CHG created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-1801[2375] Release Quicksight security group and users to production

---

2373 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2374 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Da168f7371b6991d0d1817662164bcba6%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D398%26sysparm_record_list%3Dactive%253Dtrue%255EORDERBYDESCnumber

2375 https://nhsd-jira.digital.nhs.uk/browse/DOS-1801

- DOS-1767[2376] Add a parser for excel styled questions
- DOS-1828[2377] Allow previously invited user to be re-invited after their invitation has expired
- DOS-1663[2378] Reordered the columns in the DAQ
- DOS-1797[2379] Make assessment tracker show last developer update
- DOS-1802[2380] Enable Quicksight in production

9.59.2 Prepared Release

Release 1 Tag: v2022.09.02.10.13

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=87772&view=results

Release 2 Tag: v2022.09.02.10.22

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=87787&view=results

9.59.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release 1 Link above and approve the manual step to deploy the release to prod
3. Click the Release 2 link above and approve the manual step to deploy the release to prod
    a. It is likely that Release 1 will cause the plan to be invalid and the whole plan will need to be run again. In that event, run a new pipeline with the tag **v2022.09.02.10.22** to update the plan and approve the manual deploy step.

9.59.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2381]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.59.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.08.08.14.59**

2376 https://nhsd-jira.digital.nhs.uk/browse/DOS-1767
2377 https://nhsd-jira.digital.nhs.uk/browse/DOS-1828
2378 https://nhsd-jira.digital.nhs.uk/browse/DOS-1663
2379 https://nhsd-jira.digital.nhs.uk/browse/DOS-1797
2380 https://nhsd-jira.digital.nhs.uk/browse/DOS-1802
2381 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86575&view=results

- Redeploy *last successful non-production release* (to all envs except Production)

9.60 CHG0040981 - Initial deployment of Electronic Signing Service to production

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D5be41fad1be15950c994206fe54bcb3c

9.60.1 Tickets

CHG created by Neil Sproston. The Release implements the changes in the following tickets:

- BAU-11215

9.60.2 Prepared Release

Release Tag: **v1.0.480-beta**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86081&view=results

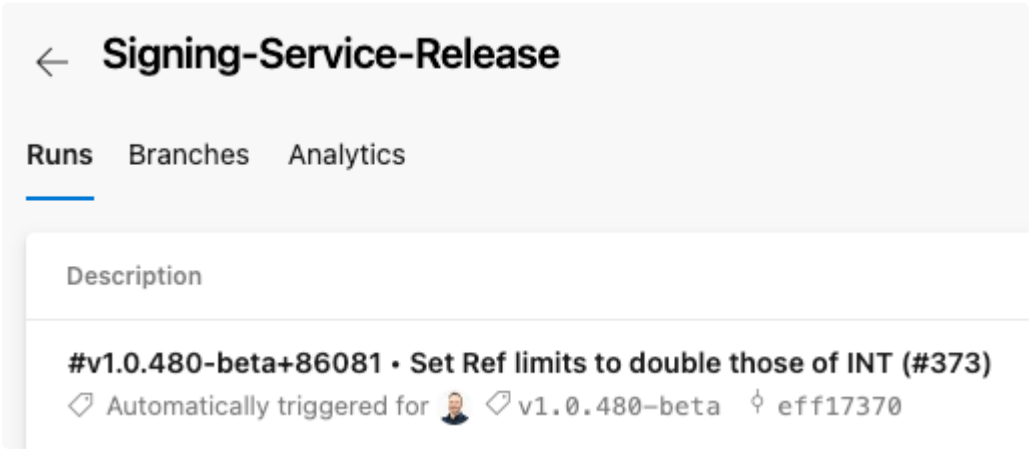
9.60.3 Release Process

1. See also - ESS Deployment[2382]

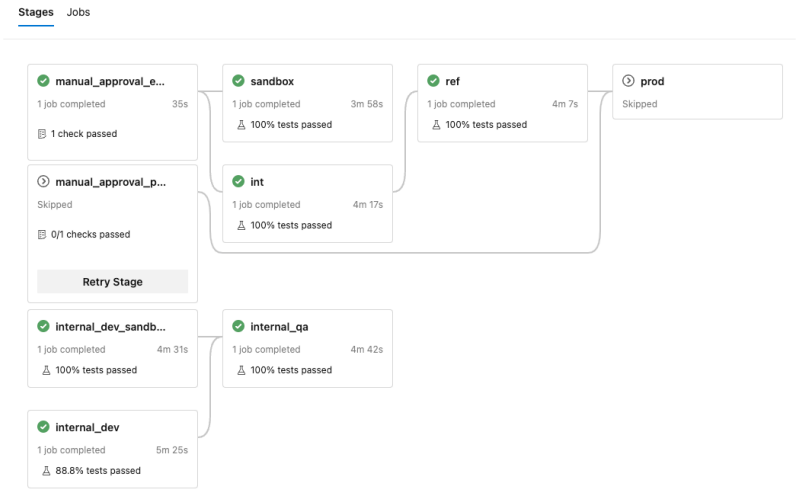
2. ### 9.60.3.1 Technical Steps to Deploy to Live

   - Prior to Deployment of Change to Live
     i. Merging a Pull Request will automatically deploy to the internal-dev, internal-qa and internal-dev-sandbox environments
     ii. If no issues noted, there is a manual approval gate (manual_approval_external_ptl) in the pipeline which will allow deployment to proceed to the Sandbox, INT and REF environments - liaise with EPS before doing this, they have suppliers on INT
     iii. If all well, an RFC should now raised for the final deployment to Live
   - Azure DevOps Pipeline - Signing Service Release[2383]
   - There will be a pipeline run with a name based on the Git tag which was created when the Pull Request was merged - e.g. v1.0.480-beta in the following screengrab:

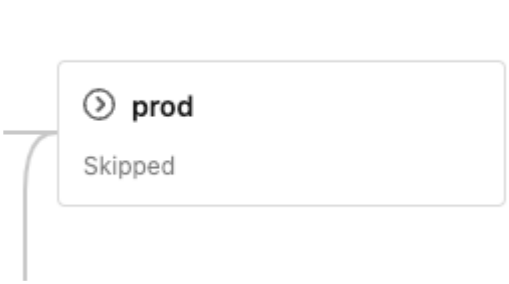
2382 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/483235694/ESS+Deployment
2383 https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=343

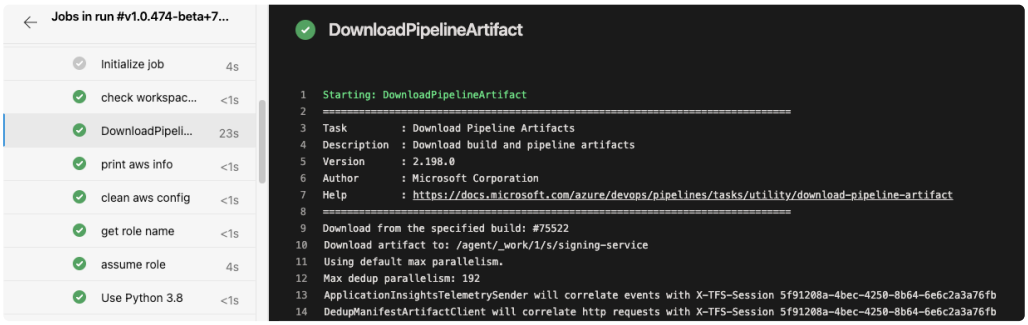
- 

- If deployment to all Dev and PTL environments was successful, the pipeline diagram should look similar to the the following:

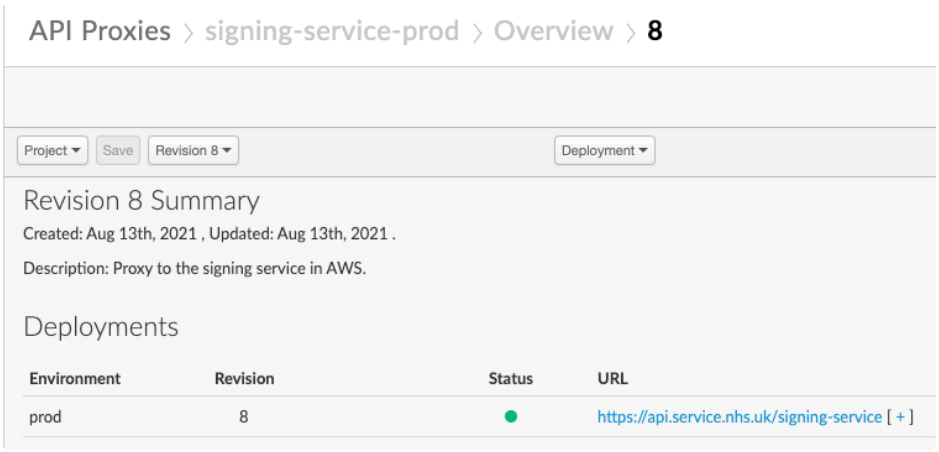
- 

- Clicking on the 'Retry Stage' (manual_approval_prod) will prompt to confirm approval
- The deployment should now proceed, logs can be followed by clicking on the 'prod' icon:

- 

- 

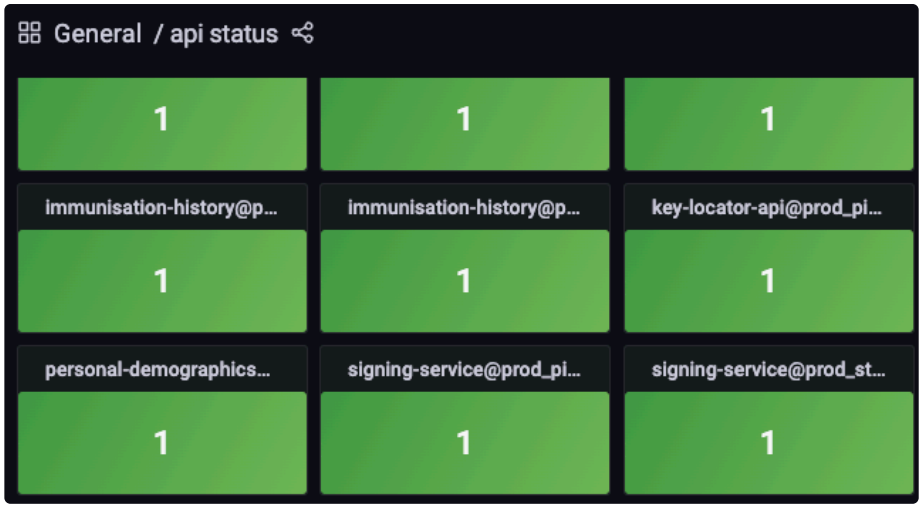
9.60.3.2 Post Release Checks

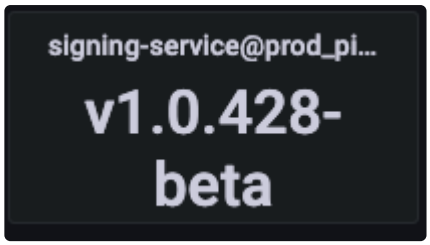
1. Monitor APIM Dashboard for Signing Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions - there is currently no traffic hitting this as it's not in Production
    - Once the pipeline completes successfully, check status of the Apigee Proxy and AWS service components
        - Check the Apigee proxy status in the Apigee console or via curl
            - Apigee console
            - 

2.
            - Curl should show the new version tag

Check Apigee Proxy Status

```
curl https://api.service.nhs.uk/signing-service/_ping
{"version":"v1.0.428-
beta","revision":"8","releaseId":"44641","commitId":"3efd73196a50835f0b4dfae8ee9
d8a15b2da7b6a"}%
```

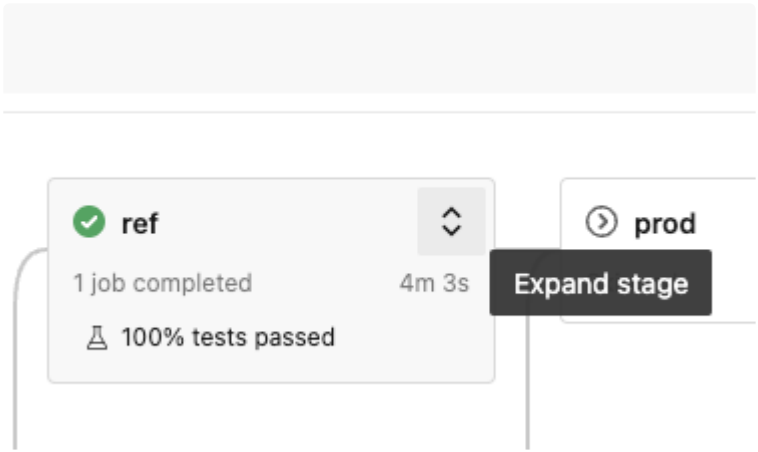
- Check the APIM Grafana Dashboard[2384]
- 

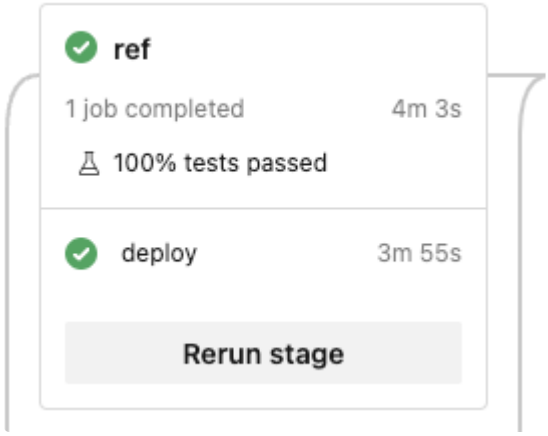
- The deployed version tag should also be present on this dashboard
- 

9.60.4 Roll-Back

- Re-deploy the last successful run
    - Go to the last successful pipeline run which successfully deployed to Prod

2384 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Click on the icon in the top right of the Prod bubble to expand it (Ref shown)
- 

- Click on 'Rerun stage' (Ref shown)
- 

9.61 CHG0034532 - Initial deployment of Electronic Signing Service to production

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Dc5db8b371ba951d0d1817662164bcb42

9.61.1 Tickets

Note: This service is not currently in use and has no end users.
The release is to validate the deployment process.

CHG created by Ed Breslin The Release implements the changes in the following tickets:

- BAU-10584

9.61.2 Prepared Release

Release Tag: **v1.0.480-beta**

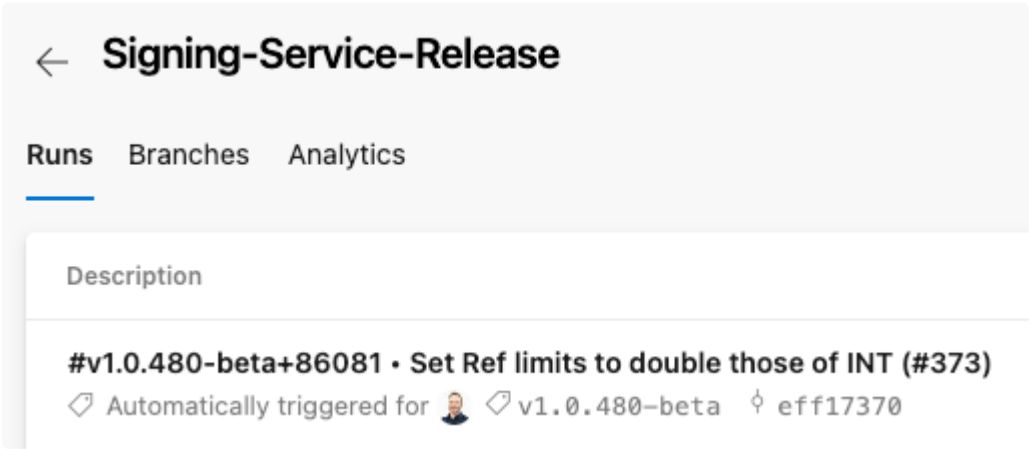
https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86081&view=results

Release Process

1. See also - ESS Deployment[2385]

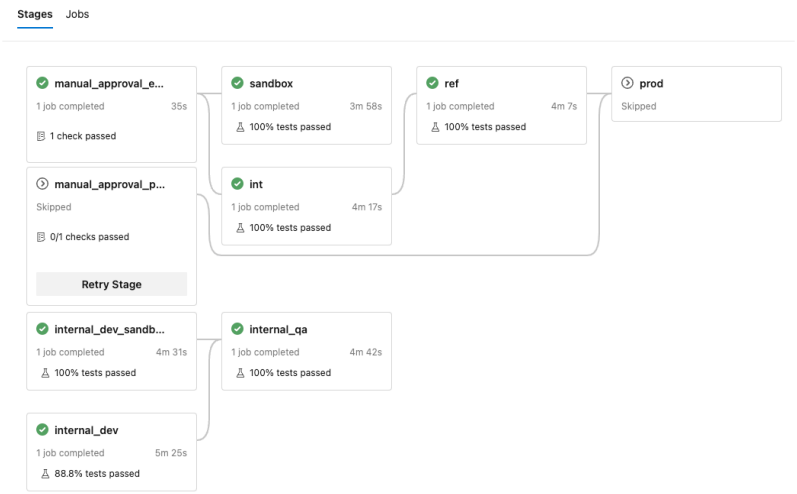
2. #### 9.61.2.1 Technical Steps to Deploy to Live

   - Prior to Deployment of Change to Live
     i. Merging a Pull Request will automatically deploy to the internal-dev, internal-qa and internal-dev-sandbox environments
     ii. If no issues noted, there is a manual approval gate (manual_approval_external_ptl) in the pipeline which will allow deployment to proceed to the Sandbox, INT and REF environments - liaise with EPS before doing this, they have suppliers on INT
     iii. If all well, an RFC should now raised for the final deployment to Live
   - Azure DevOps Pipeline - Signing Service Release[2386]
   - There will be a pipeline run with a name based on the Git tag which was created when the Pull Request was merged - e.g. v1.0.480-beta in the following screengrab:

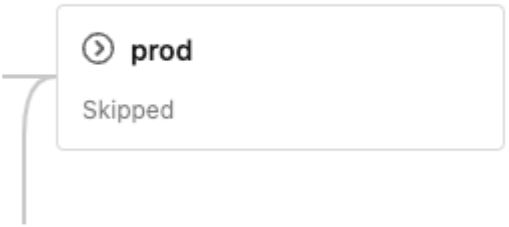


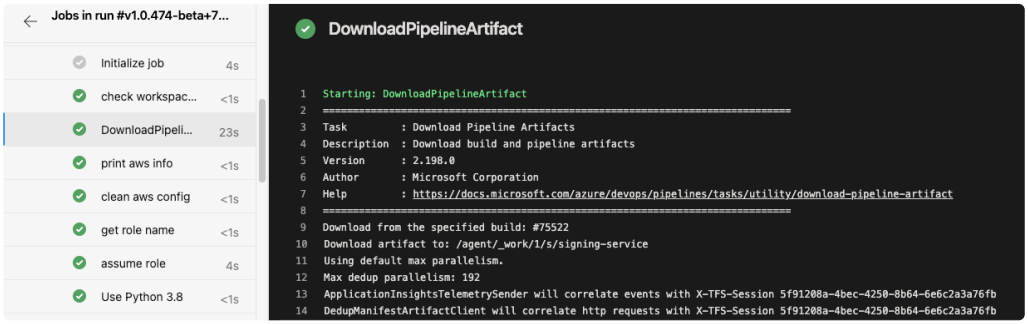
   - If deployment to all Dev and PTL environments was successful, the pipeline diagram should look similar to the the following:

2385 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/483235694/ESS+Deployment
2386 https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=343

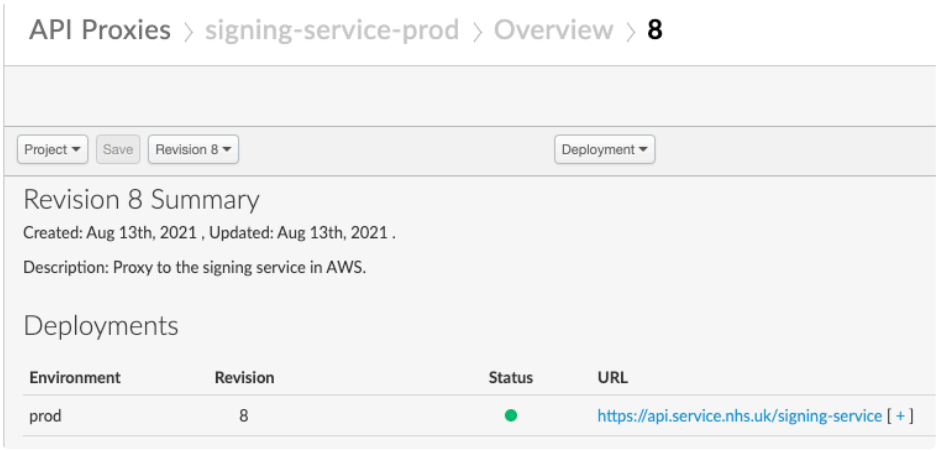
- 

- Clicking on the 'Retry Stage' (manual_approval_prod) will prompt to confirm approval
- The deployment should now proceed, logs can be followed by clicking on the 'prod' icon:

- 

- 

9.61.2.2 Post Release Checks

1. Monitor APIM Dashboard for Signing Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions - there is currently no traffic hitting this as it's not in Production
    - Once the pipeline completes successfully, check status of the Apigee Proxy and AWS service components
        - Check the Apigee proxy status in the Apigee console or via curl
            - Apigee console
            - 

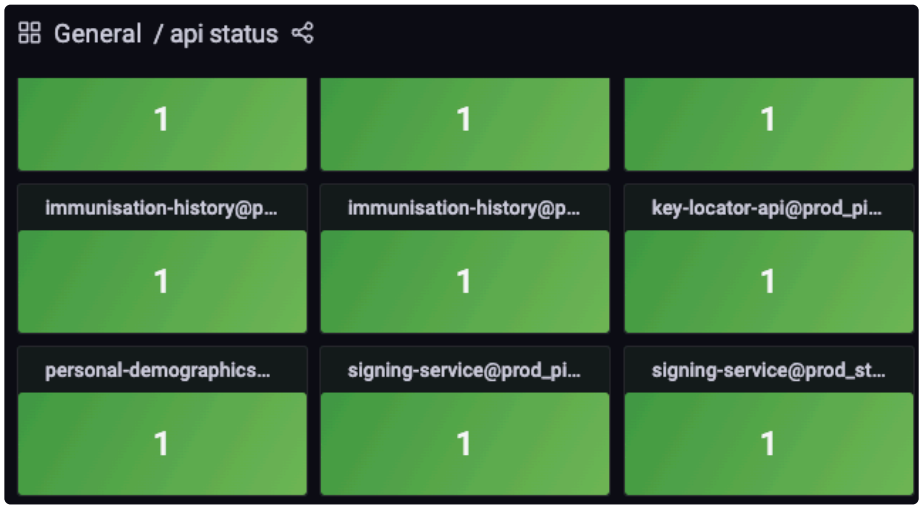
2.
        - Curl should show the new version tag

Check Apigee Proxy Status

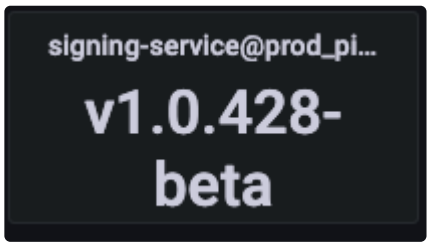
```
curl https://api.service.nhs.uk/signing-service/_ping
{"version":"v1.0.428-
beta","revision":"8","releaseId":"44641","commitId":"3efd73196a50835f0b4dfae8ee9
d8a15b2da7b6a"}%
```

        - Check the APIM Grafana Dashboard[2387]

2387 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

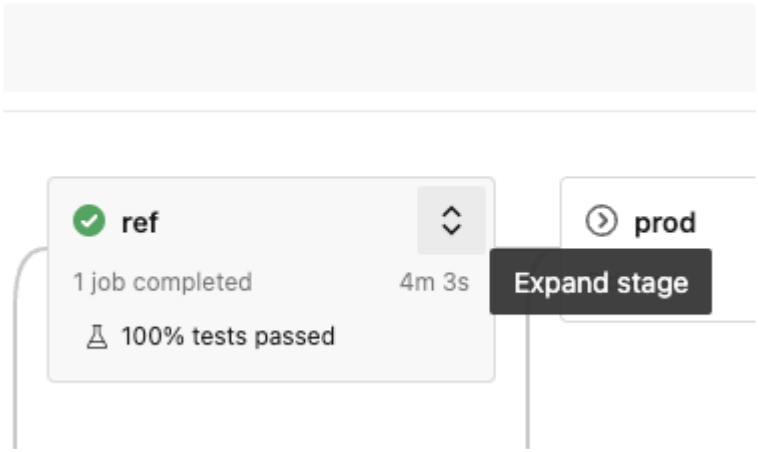
- 

- The deployed version tag should also be present on this dashboard

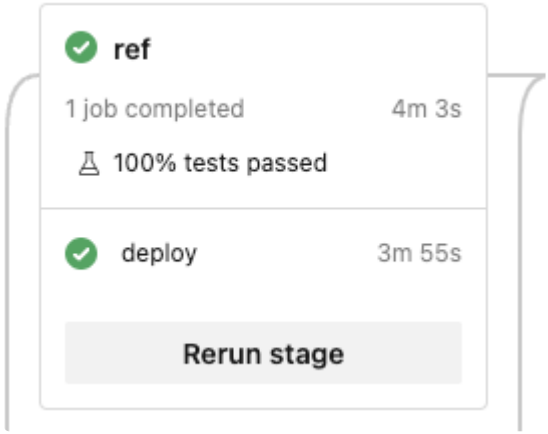
- 

9.61.3 Roll-Back

- Re-deploy the last successful run
    - Go to the last successful pipeline run which successfully deployed to Prod
    - Click on the icon in the top right of the Prod bubble to expand it (Ref shown)
-



- Click on 'Rerun stage' (Ref shown)



9.62 CHG0034608- Identity Service Changes

9.62.1 Tickets

RFC was created by Unknown User (lufa1) The Release implements the changes in the following tickets:

- Code changes for the following tickets
  - [APM-2975] IdentityService: migrate client_credentials tests to use TestUtils 2.0 - CDT JIRA (digital.nhs.uk)[2388] Refactor tests and improve caching for client credentials

2388 https://nhsd-jira.digital.nhs.uk/browse/APM-2975

9.62.2 Prepared Release

Release Tag: v1.0.1496-alpha[2389]

Pipelines - Run v1.0.1496-alpha+87824 (azure.com)[2390]

9.62.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.62.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2391]
2. Verify the creation of a new entry on the dashboard for Identity Service _status
3. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.62.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1388-alpha
    - Pipelines - Run v1.0.1388-alpha+74426 (azure.com)[2392]
- Redeploy *last successful non-production release* (to all envs except production)

9.63 CHG0034039 - NHS e-RS APIM Referral Release R1.12.0 Production Deployment

9.63.1 Content

High-level summary:

- Contains updates to Docker dependencies.
- Split APIM repo structures according to FHIR version.
- Rate limiting changes.

Tickets:

2389 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74426&view=results
2390 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=87824&view=results
2391 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2392 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=74426&view=results

- ERSSUP-62899 - Upgrade docker dependencies in APIM-R repo
- ERSSUP-63493 - APIM poetry fails to install due to https://github.com/python-poetry/poetry/issues/6133
- ERSSUP-63215 - xAPI APIM: Split APIM repo structure according to FHIR version
- ERSSUP-62204 - xAPI APIM: Implement new rate limit model
- ERSSUP-63625 - APIM-R 1.12 Version Upgrades
- ERSSUP-61832 - APIM-R 1.12 - Vulnerability Updates
- ERSSUP-59607 - APIM - Follow APIM design to use Title Case for HTTP headers

Further details on the APIM 1.12.0[2393] Release build page.

9.63.2 Prepared Release

Release Tag: 1.12.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86605&view=results

9.63.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.63.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2394]

9.63.4 Roll-Back

- Redeploy:
  - Release Tag: 1.11.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83320&view=results

2393 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/461268784/APIM-R-1.12+Release+Page
2394 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.64 CHG0034041 - NHS e-RS APIM Patient Care Release R1.4.0 Production Deployment

9.64.1 Content

High-level summary:

- TRM updates

Tickets:

- ERSSUP-63293 - APIM-PC Implement dependabot config
- ERSSUP-63296 - APIM-PC 1.4 Version Upgrades
- ERSSUP-63295 - APIM-PC 1.4 Vulnerability Fixes
- ERSSUP-63702 - Administer APIM-R-1.12 and APIM-PC-1.4 cuts
- ERSSUP-63610 - APIM-PC Integration Tests failing to get token

Further details on APIM 1.4.0[2395] Release build page.

9.64.2 Prepared Release

Release Tag: 1.4.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86604&view=results

9.64.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.64.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2396]

2395 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/461268932/APIM-PC-1.4+Release+Page
2396 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.64.4 Roll-Back

- Redeploy:
    - Release Tag: 1.3.0-beta
    - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=83630&view=results

9.65 CHG0034907: APIM Infrastructure Changes

9.65.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-3476[2397] - Support GP Registration Management Information go-live
    - AMB-1274[2398] - keycloak-sandbox for NHS Login Separated Auth
    - AMB-1266[2399] - Create a truststore by running ansible playbook

9.65.2 Release Process

Start a release pipeline with the **refs/tags/v2022.09.09.10.48**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.09.09.10.48**
- Release Pipeline[2400]

9.65.3 Post Release Checks

- Monitor Grafana Prod dashboard[2401] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2402] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

2397 https://nhsd-jira.digital.nhs.uk/browse/APM-3476
2398 https://nhsd-jira.digital.nhs.uk/browse/AMB-1274
2399 https://nhsd-jira.digital.nhs.uk/browse/AMB-1266
2400 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88356&view=results
2401 https://grafana.prod.api.platform.nhs.uk
2402 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.65.4 Roll-Back

Redeploy:

- Release Tag: **v2022.09.02.10.22**
- Rollback Pipeline (see page 1686)

9.66 CHG0033743 - GP Registration Management Information

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D9fda723b1b451194c994206fe54bcb91%26sysparm_stack%3D%26sysparm_view%3D

9.66.1 Tickets

CHG created by **Jen Hughes** The Release implements the technical go-live for GP Registration Management Information.

9.66.2 Prepared Release

Release Tag: **v1.0.120-alpha**

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88503&view=results

9.66.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.66.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2403]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.66.4 Roll-Back

- Manually disable prod-proxy (since no previous good state for first release).
- Redeploy *last successful non-production release* (to all envs except Production)

2403 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.67 CHG0035002 - Digital Onboarding Service

Servicenow CHG Request[2404]

9.67.1 Tickets

CHG created by **Harry Coulter**. The Release implements the changes in the following tickets:

- ADZ-2232
- ADZ-2322
- ADZ-2293
- DOS-1829
- DOS-1667
- ADZ-2500
- ADZ-2070

9.67.2 Prepared Release

Release Tag: v2022.09.13.15.16

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88564&view=results

9.67.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.67.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2405]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Verify new flow

- [ ]

2404 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D2bb6b9881bce5110c994206fe54bcbbb%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

2405 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

"My applications and teams" should lead to the legacy developer portal.

"Get your software onboarded to go live" should lead to "ExternalLoginDisabled", assuming the user does not have any onboarding products associated with their account. If they do, it will instead lead to the /Products page.

"Go back to the developer hub" will link to the NHSD website-hosted developer hub.

9.67.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.09.09.10.48**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88356&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.68 CHG0035191 - Electronic Signing Service - First of Type Readiness

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=bb57a1c61b06d990d1817662164bcbf8

9.68.1 This will be self-released by the Signing Service team

9.68.2 Tickets

Note: This service is not currently in use and has no end users.
JIRA for Deployment - BAU-10740

The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/BAU-10668 - ESS - Send Cloudwatch Alerts to Slack
- https://nhsd-jira.digital.nhs.uk/browse/BAU-10717 - ESS - Add approver group for prod deployments
- https://nhsd-jira.digital.nhs.uk/browse/BAU-10714 - ESS - Add manual approval for consumers of the prod API
- https://nhsd-jira.digital.nhs.uk/browse/BAU-10773 - ESS - Set Prod Apigee Rate Limits to Match REF

9.68.3 Prepared Release

Release Tag: **v1.0.484-beta**

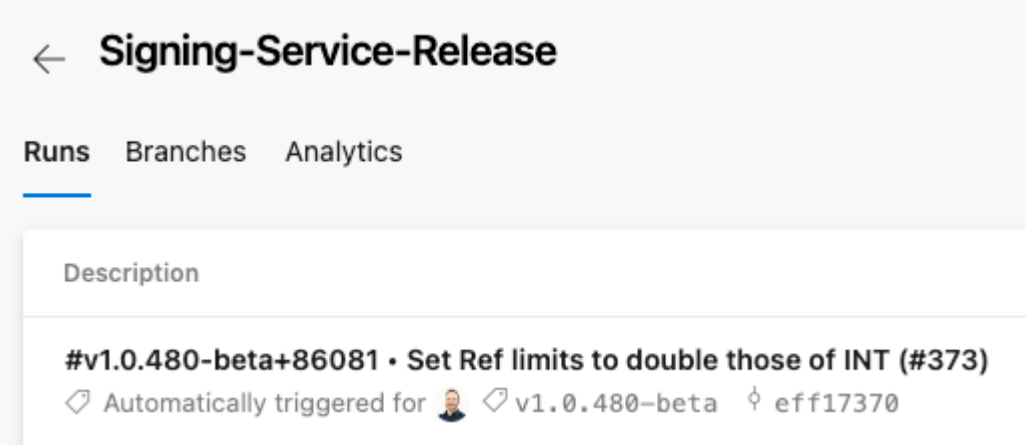
https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89400&view=results

Release Process

1. See also - ESS Deployment[2406]

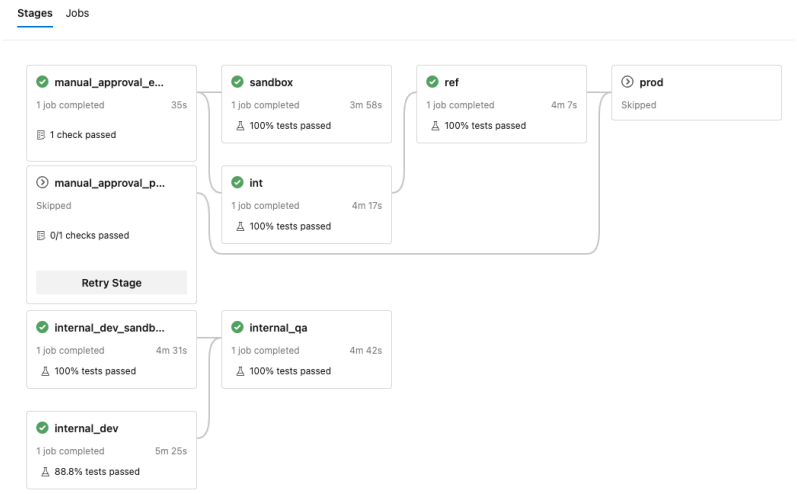
2. 9.68.3.1 Technical Steps to Deploy to Live

- Prior to Deployment of Change to Live
    i. Merging a Pull Request will automatically deploy to the internal-dev, internal-qa and internal-dev-sandbox environments
    ii. If no issues noted, there is a manual approval gate (manual_approval_external_ptl) in the pipeline which will allow deployment to proceed to the Sandbox, INT and REF environments - liaise with EPS before doing this, they have suppliers on INT
    iii. If all well, an RFC should now raised for the final deployment to Live
- Azure DevOps Pipeline - Signing Service Release[2407]
- There will be a pipeline run with a name based on the Git tag which was created when the Pull Request was merged - e.g. v1.0.480-beta in the following screengrab:

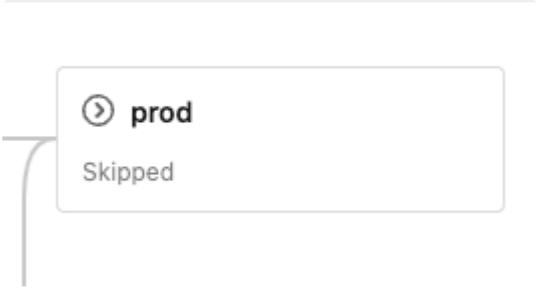


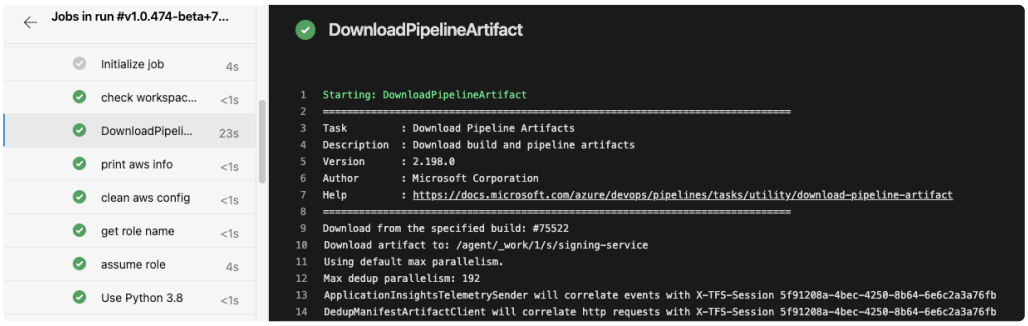
- If deployment to all Dev and PTL environments was successful, the pipeline diagram should look similar to the the following:

2406 https://nhsd-confluence.digital.nhs.uk/spaces/CIS/pages/483235694/ESS+Deployment
2407 https://dev.azure.com/NHSD-APIM/API%20Platform/_build?definitionId=343



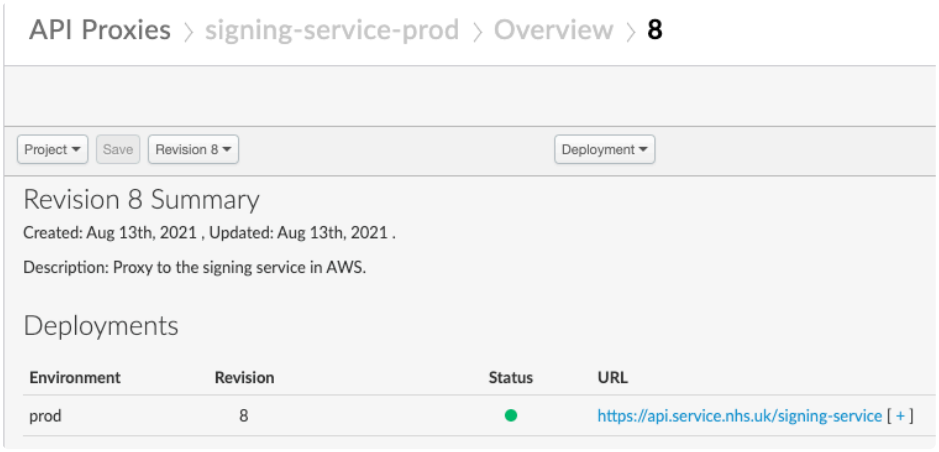
- Clicking on the 'Retry Stage' (manual_approval_prod) will prompt to confirm approval
- The deployment should now proceed, logs can be followed by clicking on the 'prod' icon:





9.68.3.2 Post Release Checks

1. Monitor APIM Dashboard for Signing Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions - there is currently no traffic hitting this as it's not in Production

- Once the pipeline completes successfully, check status of the Apigee Proxy and AWS service components
    - Check the Apigee proxy status in the Apigee console or via curl
        - Apigee console
        - 

2.

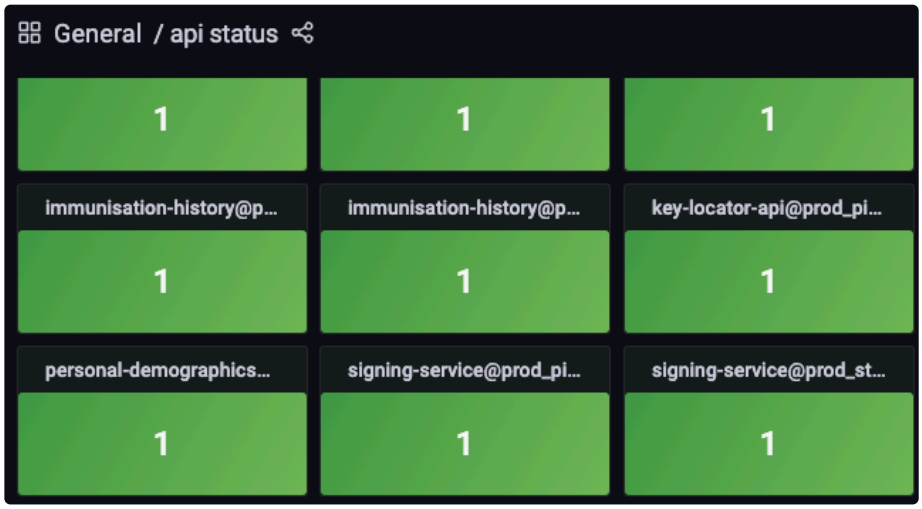
- Curl should show the new version tag

Check Apigee Proxy Status

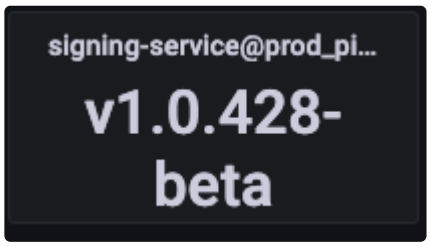
```
curl https://api.service.nhs.uk/signing-service/_ping
{"version":"v1.0.428-
beta","revision":"8","releaseId":"44641","commitId":"3efd73196a50835f0b4dfae8ee9
d8a15b2da7b6a"}%
```

- Check the APIM Grafana Dashboard[2408]

2408 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

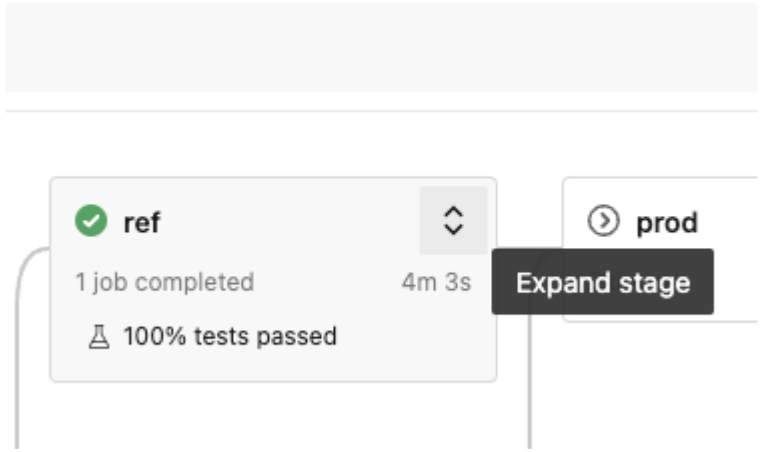
- 

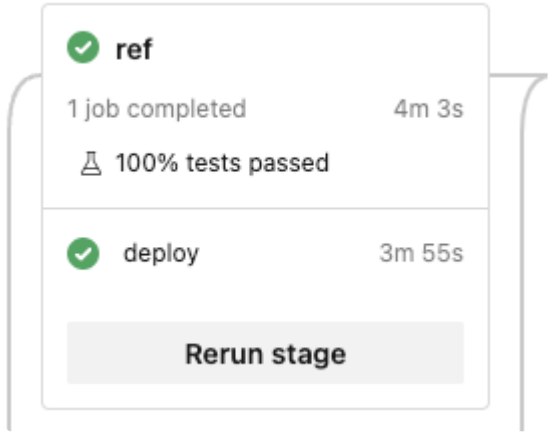
- The deployed version tag should also be present on this dashboard

- 

9.68.4 Roll-Back

- Re-deploy the last successful run
    - Go to the last successful pipeline run which successfully deployed to Prod
    - Click on the icon in the top right of the Prod bubble to expand it (Ref shown)
-

- 

- Click on 'Rerun stage' (Ref shown)
- 

9.69 CHG0035187 - Digital Onboarding Service

Servicenow CHG Request[2409]

9.69.1 Tickets

CHG created by **Lee Richardson**. The Release implements the changes in the following tickets:

- DOS-1668
- ADZ-2501
- ADZ-2449
- ADZ-2540

2409 https://nhsdigitallive.service-now.com/change_request.do?sys_id=1f2a91021b825550c994206fe54bcba8&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.69.2 Prepared Release

Release Tag: v2022.09.21.15.53

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88854&view=results

9.69.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.69.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2410]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Verify new flow

9.69.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.09.09.10.48**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88564&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.70 CHG0035334 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D525bea6f1bced9d0d1817662164bcbb0

9.70.1 Tickets

CHG created by **Hannah Ibberson**.The Release implements the changes in the following tickets:

- DOS-915[2411] - parse links in the DAQ

2410 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2411 https://nhsd-jira.digital.nhs.uk/browse/DOS-915

- DOS-1938[2412] - remove unneeded link validation

9.70.2 Prepared Release

Release Tag: **v2022.09.21.15.53+89170**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89170&view=results

9.70.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.70.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2413]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.70.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.09.21.15.53+88854**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88854&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.71 CHG0035412: APIM Infrastructure Changes

9.71.1 Tickets

RFC created by **Ben Strutt**

NOTE: JIRA was down while pulling this release plan together so was unable to get a summary of the tickets. I will update it in the morning once JIRA is fixed.

The release contains:

- Code changes for the following completed tickets:
- Minor config changes to support the future release of the following in-flight tickets:

2412 https://nhsd-jira.digital.nhs.uk/browse/DOS-1938

2413 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.71.2 Release Process

Start a release pipeline with the **refs/tags/v2022.09.21.15.53**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.09.21.15.53**
- Release Pipeline[2414]

9.71.3 Post Release Checks

- Monitor Grafana Prod dashboard[2415] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2416] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.71.4 Roll-Back

Redeploy:

- Release Tag: **v2022.09.21.15.53**
- Rollback Pipeline[2417]

9.72 CHG0035211 - NHS e-RS APIM Referral Release R1.13 Production Deployment

9.72.1 Content

High-level summary:

- ERSSUP-61283 - [xAPI APIM] System URLs in sandbox examples
- ERSSUP-63797 - APIM-R 1.13 Version Upgrades

Tickets:

- TBC

Further details on the APIM 1.13.0[2418] Release build page.

2414 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89222&view=results
2415 https://grafana.prod.api.platform.nhs.uk
2416 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2417 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89224&view=results
2418 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/493096347/APIM-R-1.13+Release+Page

9.72.2 Prepared Release

Release Tag: 1.13.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88956&view=results

9.72.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.72.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2419]

9.72.4 Roll-Back

- Redeploy:
  - Release Tag: 1.12.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86605&view=results

9.73 CHG0035210 - NHS e-RS APIM Patient Care Release R1.5 Production Deployment

9.73.1 Content

High-level summary:

- ERSSUP-63798 - APIM-PC 1.5 Version Upgrades

Tickets:

2419 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- ERSSUP-63798 - APIM-PC 1.5 Version Upgrades

Further details on APIM 1.5.0[2420] Release build page.

9.73.2 Prepared Release

Release Tag: 1.5.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88969&view=results

9.73.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.73.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2421]

9.73.4 Roll-Back

- Redeploy:
  - Release Tag: 1.4.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=86604&view=results

9.74 CHG0035615 - Infrastructure Standard Change

https://nhsdigitallive.service-now.com/task_ci_list.do?sysparm_query=task%3Df753a78a1b5e9d54d1817662164bcbd5%5E&sysparm_view=

2420 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/493096357/APIM-PC-1.5+Release+Page
2421 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.74.1 Tickets

CHG created by Unknown User (dabr15) The Release implements the changes in the following tickets:

- APM-3122[2422] - Onboard Apigee Audit Logs DONE

9.74.2 Prepared Release

Release Tag: v2022.10.04.07.16[2423]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89724&view=results

9.74.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.74.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2424]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.74.4 Roll-Back

- Redeploy:
  - Release Tag: v2022.09.27.09.09
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89282&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.75 CHG0035683: APIM Infrastructure Changes

https://nhsdigitallive.service-now.com/change_request.do?sys_id=ee5c05aa1bde1d14c994206fe54bcb32&sysparm_view=&sysparm_domain=null&sysparm_domain_scope=null

2422 https://nhsd-jira.digital.nhs.uk/browse/APM-3122?src=confmacro
2423 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2022.10.04.07.16
2424 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.75.1 Tickets

CHG created by **Ben Strutt**. The Release implements the changes in the following tickets:

- APM-3525[2425] - Turn off YACE logging DONE

9.75.2 Prepared Release

Release Tag: v2022.10.05.08.41

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89847&view=results

9.75.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.75.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2426]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.75.4 Roll-Back

- Redeploy:
    - Release Tag: v2022.10.04.07.16
- Redeploy *last successful non-production release* (to all envs except Production)

9.76 CHG0035816 - Digital Onboarding Service

Servicenow CHG Request[2427]

9.76.1 Tickets

CHG created by **Lee Richardson**. The Release implements the changes in the following tickets:

2425 https://nhsd-jira.digital.nhs.uk/browse/APM-3525?src=confmacro

2426 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2427 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=194ceec81b2add54c994206fe54bcb0e

- ADZ-2349
- ADZ-2541
- DOS-1836 - Add approval date to ApplicationApiAssignment
- DOS-1880 - Add ability to assign onboarding leads to APIs
- DOS-1777 - Improve loading speed on products page
- ADZ-2545
- DOS-1637[2428] - Add how many users have answered a question (OBL/developer counts)

9.76.2 Prepared Release

Release Tag: v2022.10.05.08.41

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90171

9.76.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.76.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2429]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Verify new flow

9.76.4 Roll-Back

- Redeploy:
    - Release Tag: v2022.10.05.08.41
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=89847
- Redeploy *last successful non-production release* (to all envs except Production)

2428 https://nhsd-jira.digital.nhs.uk/browse/DOS-1637
2429 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.77 CHG0036081 - Digital Onboarding Service

Link to ServiceNow Change request[2430]

9.77.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-2054[2431] - remove ability to add api to non-onboarding lead and pending onboarding leads
- DOS-2049[2432] - improve db query for products landing page
- ADZ-2505[2433] - prevent email casing bug

9.77.2 Prepared Release

Release Tag**: v2022.10.05.08.41**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90713&view=results

9.77.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.77.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2434]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

With regards to ADZ-2505[2435] - this release will break the developer portal for the user who initially raised the bug in ADZ-2465[2436] because we changed the casing of their email in Apigee to match the email in DOS. As the release fixes the email casing mismatch issue, we need to revert the initial email change in Apigee. Specifically, for this user we need to delete the account with the lowercase email (newest one with no apps), then edit the account with the uppercase email and apps attached to it to be lowercase. This can be done in

2430 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Dce46e3961b269d58d1817662164bcbb4
2431 https://nhsd-jira.digital.nhs.uk/browse/DOS-2054
2432 https://nhsd-jira.digital.nhs.uk/browse/DOS-2049
2433 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2505
2434 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2435 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2505
2436 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2465

hours before the release - Sophie Clayton will walk through the steps with someone who has prod Apigee access. **Update** - this has been done.

9.77.4 Roll-Back

- Redeploy:
    - Release Tag: v2022.10.05.08.41
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90171
- Redeploy *last successful non-production release* (to all envs except Production)

9.78 CHG0036192- Digital Onboarding Service

ServiceNow Change request[2437]

9.78.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-1838[2438] - section status history

9.78.2 Prepared Release

Release Tag: **v2022.10.05.08.41**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90842&view=results

9.78.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.78.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2439]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2437 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D1740814f1b225998d1817662164bcb82
2438 https://nhsd-jira.digital.nhs.uk/browse/DOS-1838
2439 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.78.4 Roll-Back

- Redeploy:
    - Release Tag**: v2022.10.05.08.41**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90713&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.79 CHG0036171 - NHS e-RS APIM Referral Release R1.14 Production Deployment

9.79.1 Content

High-level summary:

- Refactoring of e-referrals-service-api.yml file (the file that contains the OAS doc content)
- Minor vulnerability patches in the APIM stack.

Tickets:

- NA

Further details on the APIM-R-1.14 Release Page[2440] Release build page.

9.79.2 Prepared Release

Release Tag: 1.14.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90753&view=results

9.79.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

2440 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/512760236/APIM-R-1.14+Release+Page

9.79.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2441]

9.79.4 Roll-Back

- Redeploy:
    - Release Tag: 1.13.0-beta
    - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88956&view=results

9.80 CHG0036172 - NHS e-RS APIM Patient Care Release R1.6 Production Deployment

9.80.1 Content

High-level summary:

- Creation and publication of the Wayfinder OAS documentation
- Minor vulnerability patches in the APIM stack.

Tickets:

- NA

Further details on APIM-PC-1.6 Release Page[2442] Release build page.

9.80.2 Prepared Release

Release Tag: 1.6.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90752&view=results

2441 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2442 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/512760251/APIM-PC-1.6+Release+Page

9.80.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.80.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2443]

9.80.4 Roll-Back

- Redeploy:
    - Release Tag: 1.5.0-beta
    - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=88969&view=results

9.81 CHG0036522 - Digital Onboarding

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D37eefd191bbad550d1817662164bcb13

9.81.1 Tickets

CHG created by **Hannah Ibberson**.The Release implements the changes in the following tickets:

- DOS-1975[2444] - assessment tracker "latest awaiting review" column
- ADZ-2473[2445] - apps without an env treated as prod

2443 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2444 https://nhsd-jira.digital.nhs.uk/browse/DOS-1975
2445 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2473

9.81.2 Prepared Release

Release Tag: **v2022.10.05.08.41**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=91615&view=results

9.81.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.81.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2446]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.81.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.10.05.08.41**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90842&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.82 CHG0036796 - APIM Identity Service Updates

<Link to ServiceNow Change request>[2447]

9.82.1 Tickets

CHG created by Unknown User (joba20) on behald of Ed Feldman-bright

The Release implements the changes in the following tickets:

2446 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2447 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Dd8f906331bba195093bfc88b274bcb59%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull%26sysparm_record_row%3D5%26sysparm_record_rows%3D5%26sysparm_record_list%3Dassignment_group%253djavascript%253agetMyGroups()%255eactive%253dtrue%255esys_class_name%253dchange_request%255eORsys_class_name%253dchange_task%255eORDERBYnumber

- APM-3489[2448] - Identity service errors - invalid exp claim in jwt DONE
- AMB-1276[2449] - Java User Rst Separate Auth NHS Login example project DONE
- AMB-1254[2450] - Response returns error message about client_assertion when the issue lies with the subject_token DONE
- AMB-1281[2451] - Refactor KeyCloak CIS2 Realms DONE
- APM-3430[2452] - Identity Service: refactor token_exchange tests DONE

9.82.2 Prepared Release

Release Tag: v1.0.1641-alpha

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=92077&view=results

9.82.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.82.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2453]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.82.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1496-alpha
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=87824&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2448 https://nhsd-jira.digital.nhs.uk/browse/APM-3489?src=confmacro
2449 https://nhsd-jira.digital.nhs.uk/browse/AMB-1276?src=confmacro
2450 https://nhsd-jira.digital.nhs.uk/browse/AMB-1254?src=confmacro
2451 https://nhsd-jira.digital.nhs.uk/browse/AMB-1281?src=confmacro
2452 https://nhsd-jira.digital.nhs.uk/browse/APM-3430?src=confmacro
2453 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.83 CHG0036917 - Digital Onboarding Service

Link to service now CHG request[2454]

9.83.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-1939[2455] - add initial answer to answer history
- DOS-1412[2456] - make emails API specific
- DOS-1960[2457] - final approval email changed to API specific approval email
- DOS-1860[2458] - new message question type
- DOS-2198[2459] - increase fargate memory size
- DOS-667[2460] - deploy S3 bucket for lambda code
- ADZ-2536[2461] - edit API keys page
- ADZ-2326[2462] - application details view API keys

9.83.2 Prepared Release

Release Tag: **v2022.11.08.10.49**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=92579&view=results

9.83.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.83.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2463]

2454 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D00c66bc11bc7d99093bfc88b274bcbad
2455 https://nhsd-jira.digital.nhs.uk/browse/DOS-1939
2456 https://nhsd-jira.digital.nhs.uk/browse/DOS-1412
2457 https://nhsd-jira.digital.nhs.uk/browse/DOS-1960
2458 https://nhsd-jira.digital.nhs.uk/browse/DOS-1860
2459 https://nhsd-jira.digital.nhs.uk/browse/DOS-2198
2460 https://nhsd-jira.digital.nhs.uk/browse/DOS-667
2461 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2536
2462 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2326
2463 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.83.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.10.05.08.41**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=91615&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.84 CHG0036991 - NHS e-RS APIM Patient Care Release R1.7 Production Deployment

9.84.1 Content

High-level summary:

- Version upgrades in the tech stack (as is custom for most of our APIM releases. Security patches etc.)
- A few very minor changes to the status terminology in the patient-care OAS document.

  Update to the status URL that appears in the OAS documentation (reaction to URL change made by APIM team)

Tickets:

- NA

Further details on APIM-PC-1.7 Release Page[2464] Release build page.

9.84.2 Prepared Release

Release Tag: 1.7.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93444&view=results

9.84.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.

2464 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/518653674/APIM-PC-1.7+Release+Page

2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.84.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2465]

9.84.4 Roll-Back

- Redeploy:
  - Release Tag: 1.6.0-beta
  - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90752&view=results

9.85 CHG0036990 - NHS e-RS APIM Referral Release R1.15 Production Deployment

9.85.1 Content

High-level summary:

- Introduction of new App Restricted end points (A024, A025 & A043)
- OAS documentation to support new end points

Tickets:

- NA

Further details on the APIM-R-1.15 Release Page[2466] Release build page.

9.85.2 Prepared Release

Release Tag: 1.15.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93442&view=results

2465 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2466 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/518653663/APIM-R-1.15+Release+Page

9.85.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.85.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2467]

9.85.4 Roll-Back

- Redeploy:
  - Release Tag: 1.14.0-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=90753&view=results

9.86 CHG0037210 - APIM Identity Service Updates

<**Link to ServiceNow Change request**[2468]>

9.86.1 Tickets

CHG created by Unknown User (joba20) on behald of Ed Feldman-bright

The Release implements the changes in the following tickets:

- APM-3489[2469] - Identity service errors - invalid exp claim in jwt DONE

2467 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2468 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D79167f231b4fdd1493bfc88b274bcb73%26sysparm_record_target%3Dtask%26sysparm_record_row%3D2%26sysparm_record_rows%3D2%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

2469 https://nhsd-jira.digital.nhs.uk/browse/APM-3489?src=confmacro

- AMB-1276[2470] - Java User Rst Separate Auth NHS Login example project DONE
- AMB-1254[2471] - Response returns error message about client_assertion when the issue lies with the subject_token DONE
- AMB-1281[2472] - Refactor KeyCloak CIS2 Realms DONE
- APM-3430[2473] - Identity Service: refactor token_exchange tests DONE

9.86.2 Prepared Release

Release Tag: v1.0.1641-alpha

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=92077&view=results

9.86.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.86.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2474]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.86.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1496-alpha
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=87824&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2470 https://nhsd-jira.digital.nhs.uk/browse/AMB-1276?src=confmacro
2471 https://nhsd-jira.digital.nhs.uk/browse/AMB-1254?src=confmacro
2472 https://nhsd-jira.digital.nhs.uk/browse/AMB-1281?src=confmacro
2473 https://nhsd-jira.digital.nhs.uk/browse/APM-3430?src=confmacro
2474 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.87 CHG0037214: APIM Infrastructure Changes

9.87.1 Tickets

RFC created by *Ben Strutt*

The release contains:

- Code changes for the following completed tickets:
  - APM-3502[2475] - Integration tests for new proxygen endpoints and APIKey security
  - APM-3501[2476] - Exempt "paas--" prefixed entries in secret-squirrel from being managed by infra ansible role
  - APM-3489[2477] - Identity service errors - invalid exp claim in jwt
  - APM-3508[2478] - Proxygen deployment adds proxy to monitoring
  - APM-3565[2479] - Clean up JWKs repo / fix pipeline
  - AMB-1276[2480] - Java User Rst Separate Auth NHS Login example project
  - APM-3400[2481] - Implement OpenAPI as payload for Proxygen
  - APM-3510[2482] - API version in spec and plan module
  - APM-2947[2483] - SPIKE - Design rollback
  - APM-3470[2484] - Fix PaaS product deploys
  - APM-3406[2485] - Generate ping and status endpoint
  - AMB-1311[2486] - Pytest NHSD APIM fixture returns a 400 error when attempting to authorize with combined auth
- Minor config changes to support the future release of the following in-flight tickets:
  - DOS-667[2487] - Move background email work to a lambda processing layer
  - APM-3095[2488] - API producer support FY22-23

2475 https://nhsd-jira.digital.nhs.uk/browse/APM-3502
2476 https://nhsd-jira.digital.nhs.uk/browse/APM-3501
2477 https://nhsd-jira.digital.nhs.uk/browse/APM-3489
2478 https://nhsd-jira.digital.nhs.uk/browse/APM-3508
2479 https://nhsd-jira.digital.nhs.uk/browse/APM-3565
2480 https://nhsd-jira.digital.nhs.uk/browse/AMB-1276
2481 https://nhsd-jira.digital.nhs.uk/browse/APM-3400
2482 https://nhsd-jira.digital.nhs.uk/browse/APM-3510
2483 https://nhsd-jira.digital.nhs.uk/browse/APM-2947
2484 https://nhsd-jira.digital.nhs.uk/browse/APM-3470
2485 https://nhsd-jira.digital.nhs.uk/browse/APM-3406
2486 https://nhsd-jira.digital.nhs.uk/browse/AMB-1311
2487 https://nhsd-jira.digital.nhs.uk/browse/DOS-667
2488 https://nhsd-jira.digital.nhs.uk/browse/APM-3095

9.87.2 Release Process

Start a release pipeline with the **refs/tags/v2022.11.08.10.49**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.11.08.10.49**
- Release Pipeline[2489]

9.87.3 Post Release Checks

- Monitor Grafana Prod dashboard[2490] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2491] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.87.4 Roll-Back

Redeploy:

- Release Tag: **v2022.10.04.07.16**
- Rollback Pipeline (see page 1715)

9.88 CHG0037292 - Digital Onboarding Service

link to service now request[2492]

9.88.1 Tickets

CHG created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-1842[2493] - do not send section back to developer if their conditional path has not changed
- DOS-2222[2494] - disable ability to add questions with cross-sectional conditionality
- DOS-2245[2495] - add Linux AWS managed rule to WAFs
- ADZ-2548[2496] -[2497] update interim homepage copy

2489 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93029&view=results
2490 https://grafana.prod.api.platform.nhs.uk
2491 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2492 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Dd4d5210c1b97d55493bfc88b274bcb1b
2493 https://nhsd-jira.digital.nhs.uk/browse/DOS-1842
2494 https://nhsd-jira.digital.nhs.uk/browse/DOS-2222
2495 https://nhsd-jira.digital.nhs.uk/browse/DOS-2245
2496 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2548
2497 https://nhsdigital-platforms.slack.com/team/U0397EPG6UC

- ADZ-2570[2498] - invalid json warning for public key file upload
- ADZ-2327[2499] - edit basic app details page

9.88.2 Prepared Release

Release Tag: **v2022.11.18.16.43**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94272&view=results

9.88.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.88.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2500]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.88.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.11.08.10.49**
  - Release Pipeline [2501]
- Redeploy *last successful non-production release* (to all envs except Production)

9.89 CHG0037404- APIM Identity Service Update

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D4f090fe11b5719105189855bd34bcb8b%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.89.1 Tickets

CHG created by Fayez Ahmed The Release implements the changes in the following tickets:

2498 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2570
2499 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2327
2500 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2501 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93029&view=results

- APM-1832[2502] - Identity service errors - jwt signature verification fails DONE

9.89.2 Prepared Release

Release Tag: v1.0.1646-alpha

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94195&view=results

9.89.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.89.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2503]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.89.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1641-alpha[2504]
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=92077&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.90 CHG0037410 - APIM EPS Updates

Link to ServiceNow Change request[2505]

2502 https://nhsd-jira.digital.nhs.uk/browse/APM-1832?src=confmacro

2503 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2504 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=92076

2505 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D07ad8ba51b1b951093bfc88b274bcb68%26sysparm_record_target%3Dtask%26sysparm_record_row%3D1%26sysparm_record_rows%3D1%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

9.90.1 Tickets

CHG0037410 created by Michael Walters.

The release contains the following completed tickets:

- To meet the EPS FHIR API Platinum Service Level.
  - AEA-2228[2506] - Update Node & NPM to a later version
- To complete 'Sign a Prescription'
  - AEA-2180[2507] - Signing-service update max batch size from 200 to 250
  - AEA-2704[2508] - API should not expect mandatory SDS user ID for ResponsiblePractitioner
- To fix a bug in 'Send/Amend a Dispense Notification'
  - AEA-2636[2509] - Supplied Material determiner code is incorrect in HL7v3 dispense notification
- To improve validation and translations in 'Cancel a Prescription'
  - AEA-2262[2510] - Update and check translations of cancel display
  - AEA-2210[2511] - Tighten Validation around Telecom
- Removing ODS, LDAP, SDS lookups
  - AEA-2117[2512] - Enable suppliers to use separate auth in integration
- Improved validation around professional codes supplied when prescriptions are created
  - AEA-2714[2513] - Remove leading C from GMC consultant codes
  - AEA-2527[2514] - Population of Prescriber Codes
- Added validation to clarify the different resources we would expect based on the prescription type
  - AEA-2493[2515] - Differentiating between Prescribing Settings by Prescription Type (Organisation Type)
- Added validation to expect SDS user identifiers within cancel requests and responses
  - AEA-2559[2516] - Population of the SDS User Identifier within Cancellation Requests and Responses
- Removed the Organisation.type field within FHIR Cancel Responses
  - AEA-2560[2517] - Representation of Organisation Type in Cancel Requests and Responses
- Amended invalid FHIR URLs within cancel responses

2506 https://nhsd-jira.digital.nhs.uk/browse/AEA-2228
2507 https://nhsd-jira.digital.nhs.uk/browse/AEA-2180
2508 https://nhsd-jira.digital.nhs.uk/browse/AEA-2704
2509 https://nhsd-jira.digital.nhs.uk/browse/AEA-2636
2510 https://nhsd-jira.digital.nhs.uk/browse/AEA-2262
2511 https://nhsd-jira.digital.nhs.uk/browse/AEA-2210
2512 https://nhsd-jira.digital.nhs.uk/browse/AEA-2117
2513 https://nhsd-jira.digital.nhs.uk/browse/AEA-2714
2514 https://nhsd-jira.digital.nhs.uk/browse/AEA-2527
2515 https://nhsd-jira.digital.nhs.uk/browse/AEA-2493
2516 https://nhsd-jira.digital.nhs.uk/browse/AEA-2559
2517 https://nhsd-jira.digital.nhs.uk/browse/AEA-2560

    - AEA-2694[2518] - URLs in FHIR extensions are incorrect
- Updated display values within cancel responses
    - AEA-2262[2519] - Update and check translations of cancel display
- To release functionality of the FHIR Façade Dispensing API
    - AEA-2550[2520] - Identify the time that a cancellation request is created
    - AEA-2092[2521] - Amend Dispense Notification
    - AEA-1938[2522] - Full and Partial Dispense notification
    - AEA-2121[2523] - Verify Signature when Releasing a prescription
    - AEA-2113[2524] - To prove the Data Integrity Risk Items
    - AEA-1946[2525] - Claim and Claim Amend for a prescription
    - AEA-1942[2526] - Withdraw a dispense notification
    - AEA-2178[2527] - Add current dispenser details to the negative release response
    - AEA-2114[2528] - The API does meet the original requirement for Review Date in the Compliance spec
    - AEA-1978[2529] - Produce correct Release response messages
    - AEA-2115[2530] - Signature Verification fails for an Endorsed Prescription
    - AEA-2122[2531] - To prove the Solution Redundancy and IG conformance Risk Items
    - AEA-2130[2532] - Ensure we can mitigate risks for Delayed Asynchronous Communication
    - AEA-2153[2533] - Continuous Repeat Dispensing Prescription Fails to Dispense in EPSAT
    - AEA-2198[2534] - Unable to Release Controlled Drug examples
    - AEA-2090[2535] - Dispense Notification message fails TKW Validation
    - AEA-2208[2536] - Change Task so that it takes a user provided telecom

---

2518 https://nhsd-jira.digital.nhs.uk/browse/AEA-2694
2519 https://nhsd-jira.digital.nhs.uk/browse/AEA-2262
2520 https://nhsd-jira.digital.nhs.uk/browse/AEA-2550
2521 https://nhsd-jira.digital.nhs.uk/browse/AEA-2092
2522 https://nhsd-jira.digital.nhs.uk/browse/AEA-1938
2523 https://nhsd-jira.digital.nhs.uk/browse/AEA-2121
2524 https://nhsd-jira.digital.nhs.uk/browse/AEA-2113
2525 https://nhsd-jira.digital.nhs.uk/browse/AEA-1946
2526 https://nhsd-jira.digital.nhs.uk/browse/AEA-1942
2527 https://nhsd-jira.digital.nhs.uk/browse/AEA-2178
2528 https://nhsd-jira.digital.nhs.uk/browse/AEA-2114
2529 https://nhsd-jira.digital.nhs.uk/browse/AEA-1978
2530 https://nhsd-jira.digital.nhs.uk/browse/AEA-2115
2531 https://nhsd-jira.digital.nhs.uk/browse/AEA-2122
2532 https://nhsd-jira.digital.nhs.uk/browse/AEA-2130
2533 https://nhsd-jira.digital.nhs.uk/browse/AEA-2153
2534 https://nhsd-jira.digital.nhs.uk/browse/AEA-2198
2535 https://nhsd-jira.digital.nhs.uk/browse/AEA-2090
2536 https://nhsd-jira.digital.nhs.uk/browse/AEA-2208

- AEA-2172[2537] - Not incrementing the V3 'repeatNumber.high' field

9.90.2 Prepared Release

Release Tag: v1.0.1113-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94570&view=results

9.90.3 Release Process

We would like to release this ourselves in service hours. As per the roll-back comment below, we will deploy and fix forward, so the whole EPS development team will be available to fix any issues as they arise.

9.90.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay:
    a. https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
    b. e.g. Tue 22nd Nov saw steady drip of 4 transactions per minute from 8am with a few spikes
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support? - someone with security access
    b. Ask someone in Spine if they can see data coming through from EPS in Splunk? Do APIM already have access?

9.90.4 Roll-Back

We were unable to create a build that worked to roll back to, and there was no capacity in APIM to do an A/B release, so we will fix forward.

9.91 CHG0037450: APIM Infrastructure Changes

https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3D1dd87b3d1b9f991093bfc88b274bcbc1%26sysparm_record_target%3Dtask%26sysparm_record_row%3D4%26sysparm_record_rows%3D4%26sysparm_record_list%3Dassignm

2537 https://nhsd-jira.digital.nhs.uk/browse/AEA-2172

ent_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

9.91.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

- APM-3551[2538] - nhsapp-20211214-145846 Cert needs replacing DONE
- APM-3604[2539] - prod/truststore/aos.signin.nhs.uk/crt cert needs replacing DONE

9.91.2 Prepared Release

Release Tag: **v2022.11.18.16.43**

APIGEE ONLY

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94356&view=results

9.91.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.91.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2540]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. **Ensure NHS App status is still up**

9.91.4 Roll-Back

- Fetch original secrets from prod secrets manager
    - prod/server-certs/nhs-app/api.service.nhs.uk/key/backup[2541] - https://eu-west-2.console.aws.amazon.com/secretsmanager/secret?name=prod%2Fserver-certs%2Fnhs-app%2Fapi.service.nhs.uk%2Fkey%2Fbackup&region=eu-west-2

2538 https://nhsd-jira.digital.nhs.uk/browse/APM-3551?src=confmacro
2539 https://nhsd-jira.digital.nhs.uk/browse/APM-3604?src=confmacro
2540 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2541 http://api.service.nhs.uk/key/backup

    - prod/server-certs/nhs-app/api.service.nhs.uk/crt/backup[2542] - https://eu-west-2.console.aws.amazon.com/secretsmanager/secret?name=prod%2Fserver-certs%2Fnhs-app%2Fapi.service.nhs.uk%2Fcrt%2Fbackup&region=eu-west-2

  Replace the secrets with the old values

    - prod/server-certs/nhs-app/api.service.nhs.uk/key[2543]
    - prod/server-certs/nhs-app/api.service.nhs.uk/crt[2544]

  Re-run the build pipeline of infra then release from that build tag - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94081&view=results
- Redeploy same tag

If that doesn't work

- Redeploy:
    - Release Tag: **v2022.11.08.10.49**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93029&view=results

9.92 CHG0037724 - Digital Onboarding Service

link to service now request[2545]

9.92.1 Tickets

CHG created by **Lee Richardson**. The Release implements the changes in the following tickets:

- ADZ-2570 - Error handling for JWKS file upload

9.92.2 Prepared Release

Release Tag: **v2022.11.18.16.43**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95001&view=results

9.92.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2542 http://api.service.nhs.uk/crt/backup
2543 http://api.service.nhs.uk/key
2544 http://api.service.nhs.uk/crt
2545 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2Fchange_request.do%3Fsys_id%3Dd4d5210c1b97d55493bfc88b274bcb1b

9.92.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2546]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.92.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.11.18.16.43**
    - Release Pipeline [2547]
- Redeploy *last successful non-production release* (to all envs except Production)

9.93 CHG0037795: APIM Infrastructure Changes

9.93.1 Tickets

RFC created by ***Ben Strutt***

The release contains:

- Code changes for the following completed tickets:
    - APM-3601[2548] - Improve setup process for Infra repo
    - APM-3575[2549] - Fix Hello-World release pipeline
    - APM-3545[2550] - Slack bot sends targetted documentation message based on keywords
    - APM-3619[2551] - Move support report from support-devs channel
    - APM-3061[2552] - Create ECS resources in proxygen microservice
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-3095[2553] - API producer support FY22-23
    - DOS-1917[2554] - Release To PTL

2546 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2547 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94272&view=results
2548 https://nhsd-jira.digital.nhs.uk/browse/APM-3601
2549 https://nhsd-jira.digital.nhs.uk/browse/APM-3575
2550 https://nhsd-jira.digital.nhs.uk/browse/APM-3545
2551 https://nhsd-jira.digital.nhs.uk/browse/APM-3619
2552 https://nhsd-jira.digital.nhs.uk/browse/APM-3061
2553 https://nhsd-jira.digital.nhs.uk/browse/APM-3095
2554 https://nhsd-jira.digital.nhs.uk/browse/DOS-1917

9.93.2 Release Process

Start a release pipeline with the **refs/tags/v2022.11.24.19.51**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.11.24.19.51**
- Release Pipeline[2555]

9.93.3 Post Release Checks

- Monitor Grafana Prod dashboard[2556] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2557] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.93.4 Roll-Back

Redeploy:

- Release Tag: **v2022.11.18.16.43**
- Rollback Pipeline[2558]

9.94 CHG0037877 - APIM Identity Service Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D-1%26sysparm_query%3Dtype%253dstandard%5Estd_change_producer_version%253decd835d81b6d1510d1817662164bcb8c%26sysparm_link_parent%3De230cdd91bc2b410a4cdc88b274bcbfa%26sysparm_catalog%3De0d08b13c3330100c8b837659bba8fb4%26sysparm_catalog_view%3Dcatalog_default

9.94.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

- APM-2189[2559] - Identity service errors - subject token DONE
- APM-1681[2560] - Identity service errors - callback endpoint - make external id token errors specific DONE

2555 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95130&view=results
2556 https://grafana.prod.api.platform.nhs.uk
2557 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2558 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95140&view=results
2559 https://nhsd-jira.digital.nhs.uk/browse/APM-2189?src=confmacro
2560 https://nhsd-jira.digital.nhs.uk/browse/APM-1681?src=confmacro

9.94.2 Prepared Release

Release Tag: **v1.0.1671-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95415&view=results

9.94.3 Release Process

1. ~~Ensure the APIM Infrastructure Release preceding this, has been successfully deployed~~ Not needed for this release
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.94.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2561]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.94.4 Roll-Back

- Redeploy:
  - Release Tag: **v1.0.1646-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=94195&view=results

9.95 CHG0037892 - Digital Onboarding Service

Link to the service now request[2562]

9.95.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-1896[2563] - allow APIs to be added to products

2561 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2562 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D637392b01befd1d093bfc88b274bcbe8
2563 https://nhsd-jira.digital.nhs.uk/browse/DOS-1896

9.95.2 Prepared Release

Release Tag: **v2022.11.24.19.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95426&view=results

9.95.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.95.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2564]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.95.4 Roll-Back

- Redeploy:
    - Release Tag: **v2022.11.24.19.51**
    - Release Pipeline[2565]
- Redeploy *last successful non-production release* (to all envs except Production)

9.96 CHG0037865 - APIM EPS Updates

Link to ServiceNow Change request[2566]

9.96.1 Tickets

RFC created by Michael Walters.

The release contains the following completed tickets:

2564 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2565 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95130&view=results

2566 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D299052a01ba355d05189855bd34bcbee%26sysparm_record_target%3Dtask%26sysparm_record_row%3D2%26sysparm_record_rows%3D2%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

- AEA-2826[2567] - Fix API bugs introduced by the release to Production

9.96.2 Prepared Release

Release Tag:

Link to Azure pipeline build:

9.96.3 Release Process

We would like to release this ourselves in service hours.

9.96.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.96.4 Roll-Back

Rollback link

9.97 CHG0038027 - Digital Onboarding Service

Link to the service now request[2568]

9.97.1 Tickets

CHG created by **Sophie Clayton.** The Release implements the changes in the following tickets:

- ADZ-2612[2569] - Change feedback and contact us links

2567 https://nhsd-jira.digital.nhs.uk/browse/AEA-2826

2568 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8f9440561be3551493bfc88b274bcb47%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D19%26sysparm_record_rows%3D333%26sysparm_record_list%3Dactive%253Dtrue%255EORDERBYDESCnumber

2569 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2612

- DOS-2314[2570] - SQS Queue URL needs to be single variable

9.97.2 Prepared Release

Release Tag: **v2022.11.24.19.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95812&view=results

9.97.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.97.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2571]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.97.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.11.24.19.51**
  - Release Pipeline[2572]
- Redeploy *last successful non-production release* (to all envs except Production)

9.98 CHG0038099 - NHS e-RS APIM Patient Care Release R1.8 Production Deployment

9.98.1 Content

High-level summary:

- Addition of 'Errors' section (wording only) on OAS documents (wording authored by APIM team) and version upgrades/security patches

Tickets:

2570 https://nhsd-jira.digital.nhs.uk/browse/DOS-2314
2571 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2572 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95426&view=results

- NA

Further details on APIM-PC-1.8 Release Page[2573] Release build page.

9.98.2 Prepared Release

Release Tag: 1.8.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=96854&view=results

9.98.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.98.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2574]

9.98.4 Roll-Back

- Redeploy:
    - Release Tag: 1.7.0-beta
    - **Link to Azure DevOps:** https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93444&view=results

2573 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/524899041/APIM-PC-1.8+Release+Page
2574 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.99 CHG0038100 - NHS e-RS APIM Referral Release R1.16 Production Deployment

9.99.1 Content

High-level summary:

- Addition of 'Errors' section (wording only) on OAS documents (wording authored by APIM team) and version upgrades/security patches.

Tickets:

- NA

Further details on the APIM-R-1.16 Release Page[2575] Release build page.

9.99.2 Prepared Release

Release Tag: 1.16.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=96856&view=results

9.99.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.99.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2576]

2575 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/524898129/APIM-R-1.16+Release+Page
2576 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.99.4 Roll-Back

- Redeploy:
    - Release Tag: 1.15.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=93442&view=results

9.100 CHG0038330: APIM Infrastructure Changes

9.100.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
    - APM-3610[2577] - Point setup-api and api-management-utils at prod api_registry table
    - APM-3558[2578] - Slackbot creates jira ticket when instructed to on slack thread
- Minor config changes to support the future release of the following in-flight tickets:
    - APM-3095[2579] - API producer support FY22-23

9.100.2 Release Process

Start a release pipeline with the **refs/tags/v2022.12.13.10.12**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.12.13.10.12**
- Release Pipeline[2580]

9.100.3 Post Release Checks

- Monitor Grafana Prod dashboard[2581] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2582] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

2577 https://nhsd-jira.digital.nhs.uk/browse/APM-3610
2578 https://nhsd-jira.digital.nhs.uk/browse/APM-3558
2579 https://nhsd-jira.digital.nhs.uk/browse/APM-3095
2580 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=97037&view=results
2581 https://grafana.prod.api.platform.nhs.uk
2582 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.100.4 Roll-Back

Redeploy:

- Release Tag: **v2022.11.24.19.51**
- Rollback Pipeline (see page 1732)

9.101 CHG0038358: Personal Demographics Service

9.101.1 Service Now CHG0038358[2583]

9.101.2 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:
  - APM-1616[2584] - Handle error message for missing token on PDS FHIR API

9.101.3 Release Process

Start a release pipeline with the **refs/tags/v1.2.1651-beta**, click **Review** and **Approve** the prod release.

- Release Tag: **v1.2.1651-beta**
- Release Pipeline[2585]

9.101.4 Post Release Checks

- Monitor Grafana Prod dashboard[2586] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2587] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

2583 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D302e85fc1b335d945189855bd34bcb26%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

2584 https://nhsd-jira.digital.nhs.uk/browse/APM-1616

2585 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=96887&view=results

2586 https://grafana.prod.api.platform.nhs.uk

2587 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.101.5 Roll-Back

Redeploy:

- Release Tag: **v1.2.1575-beta**
- Rollback Pipeline (see page 1733)

9.102 CHG0038470: APIM Infrastructure Changes

9.102.1 Tickets

RFC created by **Ben Davies**

The release contains:

- Code changes for the following completed tickets:
- AMB-1350[2588] - Add TLS-MA Client Certificate to PROD Secrets Manager

9.102.2 Release Process

Start a release pipeline with the **refs/tags/v2022.12.14.11.25**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.12.14.11.25**
- Release Pipeline[2589]

9.102.3 Post Release Checks

- Monitor Grafana Prod dashboard[2590] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2591] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.102.4 Roll-Back

Redeploy:

- Release Tag: **v2022.12.13.10.12**
- Rollback Pipeline (see page 1734)

2588 https://nhsd-jira.digital.nhs.uk/browse/AMB-1350
2589 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=97603&view=results
2590 https://grafana.prod.api.platform.nhs.uk
2591 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health

9.103 CHG0038571: APIM Infrastructure Changes

9.103.1 Tickets

RFC created by ***Ben Strutt***

The release contains:

- Code changes for the following completed tickets:
  - APM-3559[2592] - Pipeline creates jira ticket when cert is about to expire
  - APM-3584[2593] - Deploy keycloak proxygen prod and PTL

9.103.2 Release Process

Start a release pipeline with the **refs/tags/v2022.12.15.14.46**, click **Review** and **Approve** the prod release.

- Release Tag: **v2022.12.15.14.46**
- Release Pipeline[2594]

9.103.3 Post Release Checks

- Monitor Grafana Prod dashboard[2595] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2596] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.103.4 Roll-Back

Redeploy:

- Release Tag: **v2022.12.14.11.25**
- Rollback Pipeline (see page 1735)

9.104 CHG0038936 - Digital Onboarding Service

Service Now CHG0038936[2597]

2592 https://nhsd-jira.digital.nhs.uk/browse/APM-3559
2593 https://nhsd-jira.digital.nhs.uk/browse/APM-3584
2594 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=97940&view=results
2595 https://grafana.prod.api.platform.nhs.uk
2596 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2597 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7b5318a91bd461945189855bd34bcb00

9.104.1 Tickets

CHG created by Unknown User (gipa6) The Release implements the changes in the following tickets:

- DOS-667[2598] **- Move background email work to a lambda processing layer**
- DOS-1897[2599] **- Allow APIs to be removed from products**
- DOS-2085[2600] **- Validator: Display error if an API is removed that has products assigned (test mode only)**
- DOS-2072[2601] **- Optimise loading speed of assessment tracker**
- DOS-2229[2602] **- Summarise the roll forwards impact of the import**
- DOS-1884[2603] **- Tracker: Pre-select API filter based on assigned APIs**
- DOS-1521[2604] **- Update wording for the 'no section' validation message**
- ADZ-2633[2605] **- Link to the community from keycloak emails**
- DOS-2228[2606] **- Inform the user the environment is in test mode**
- ADZ-2599[2607] **- App latency UI improvement**
- DOS-2079[2608] **- Introduce an 'alias' of the API name for use in the interface**
- DOS-2391[2609] **- Uploading file larger than 8MB shows error and does not allow upload**
- DOS-2390[2610] **- Add Quicksight permissions to DevOps role**

9.104.2 Prepared Release

Release Tag: v2023.01.09.12.18

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=99949&view=results

9.104.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2598 https://nhsd-jira.digital.nhs.uk/browse/DOS-667
2599 https://nhsd-jira.digital.nhs.uk/browse/DOS-1897
2600 https://nhsd-jira.digital.nhs.uk/browse/DOS-2085
2601 https://nhsd-jira.digital.nhs.uk/browse/DOS-2072
2602 https://nhsd-jira.digital.nhs.uk/browse/DOS-2229
2603 https://nhsd-jira.digital.nhs.uk/browse/DOS-1884
2604 https://nhsd-jira.digital.nhs.uk/browse/DOS-1521
2605 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2633
2606 https://nhsd-jira.digital.nhs.uk/browse/DOS-2228
2607 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2599
2608 https://nhsd-jira.digital.nhs.uk/browse/DOS-2079
2609 https://nhsd-jira.digital.nhs.uk/browse/DOS-2391
2610 https://nhsd-jira.digital.nhs.uk/browse/DOS-2390

9.104.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2611]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.104.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.12.15.14.46**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=97943&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.105 CHG0038946 - Identity Service Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D58dc58291bd0259493bfc88b274bcb4f%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

Very low risk. Small changes to proxy around additional logic for PRs and mocked auth.

9.105.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-3433[2612] - Identity Service: refactor logging tests DONE

9.105.2 Prepared Release

Release Tag: **v1.0.1691-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=99707&view=results

9.105.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2611 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2612 https://nhsd-jira.digital.nhs.uk/browse/APM-3433?src=confmacro

9.105.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2613]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.105.4 Roll-Back

- Redeploy:
  - Release Tag: **v1.0.1671-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=95415&view=results
- Redeploy *last successful non-production release* (to all envs except Production) - prepared release above

9.106 CHG0039474 - APIM Infrastructure Changes

Link to ServiceNow Change Request[2614]

9.106.1 Tickets

CHG created by **Fayez Ahmed** The Release implements the changes in the following tickets:

- APM-3596[2615]

9.106.2 Prepared Release

Will need to **Approve** the prod builds

Release Tag: **v2023.01.17.15.31**

Link to Azure Pipeline Build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=101385&view=results

9.106.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2613 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2614 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=c1ebefa81be46d1093bfc88b274bcb1a

2615 https://nhsd-jira.digital.nhs.uk/browse/APM-3596

9.106.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2616]
2. Monitor Splunk APIM Dashboard[2617] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.106.4 Roll-Back

- Redeploy:
  - Release Tag: **v2022.12.15.14.46**
  - Azure:
- Redeploy *last successful non-production release* (to all envs except Production)

9.107 CHG0039266 - Digital Onboarding Service

Service Now CHG0039266[2618]

9.107.1 Tickets

CHG created by Sophie Clayton . The Release implements the changes in the following tickets:

- ADZ-2589[2619] **- Fix accessibility issues**
- ADZ-2388[2620] **- Change the "Contact us" link in the DOS application footer**
- ADZ-2643[2621] **- Password reset wording issue**

9.107.2 Prepared Release

Release Tag: v2023.01.13.16.16

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=100992&view=results

9.107.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

---

2616 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2617 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2618 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D80875c3b1bd0a51893bfc88b274bcb4a%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue
2619 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2589
2620 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2388
2621 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2643

2. Click the Release Link above and approve the manual step to deploy the release to prod

9.107.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2622]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.107.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.01.09.12.18
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=99949&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.108 CHG0039377 - NHS e-RS APIM Referral Release R1.17 Production Deployment

9.108.1 Content

High-level summary:

- Several minor bug fixes for the OAS documentation (included typos, wording improvements, Sandpit response fixes)
- Update to the R4 OperationOutcome policies to add a profile to error responses inc. 401 and 429 responses.
  Also reflected in the OAS documentation

Tickets:

- NA

Further details on the APIM-R-1.17 Release Page[2623] Release build page.

9.108.2 Prepared Release

Release Tag: 1.17.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105346&view=results

---

2622 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2623 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/576127212/APIM-R-1.17+Release+Page

9.108.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.108.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2624]

9.108.4 Roll-Back

- Redeploy:
    - Release Tag: 1.16.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=96856&view=results

9.109 CHG0039378 - NHS e-RS APIM Patient Care Release R1.9 Production Deployment

9.109.1 Content

High-level summary:

- We moved our suite of patient-care automation tests away from reliance on ERSALPHA, and now rely on our FT05 test environment instead.
- Minor bug fixes for the OAS documentation (included typos, wording improvements)
- Standard TRM version upgrades

Tickets:

- NA

Further details on APIM-PC-1.9 Release Page[2625] Release build page.

2624 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2625 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/576127222/APIM-PC-1.9+Release+Page

9.109.2 Prepared Release

Release Tag: 1.9.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105199&view=results

9.109.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.109.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2626]

9.109.4 Roll-Back

- Redeploy:
    - 1.8.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=96854&view=results

9.110 CHG0039499: APIM Infrastructure Changes

9.110.1 Tickets

RFC created by **Ben Strutt**

The release contains:

- Code changes for the following completed tickets:

2626 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- APM-3624[2627] - Separate proxygen into ptl and prod
- APM-3775[2628] - Fix and add registered developers to apigee insights pipeline
- APM-3677[2629] - Integrate jira ticket creation for report a leaver slack workflows

9.110.2 Release Process

Start a release pipeline with the **refs/tags/v2023.01.19.17.19**, click **Review** and **Approve** the prod release.

- Release Tag: **v2023.01.19.17.19**
- Release Pipeline[2630]

9.110.3 Post Release Checks

- Monitor Grafana Prod dashboard[2631] (note that some of the monitoring stack may be down during deployment).
- Monitor Splunk APIM Dashboard[2632] for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions.

9.110.4 Roll-Back

Redeploy:

- Release Tag: **v2023.01.17.15.31**
- Rollback Pipeline (see page 1742)

9.111 CHG0039788 - APIM Infrastructure Changes

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd89343be1ba4695093bfc88b274bcb46%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.111.1 Tickets

CHG created by Harry Coulter The Release implements the changes in the following tickets:

- AEA-2648
- APM-3852[2633] - Fix async writefile on slackbot DONE

2627 https://nhsd-jira.digital.nhs.uk/browse/APM-3624
2628 https://nhsd-jira.digital.nhs.uk/browse/APM-3775
2629 https://nhsd-jira.digital.nhs.uk/browse/APM-3677
2630 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=101757&view=results
2631 https://grafana.prod.api.platform.nhs.uk
2632 https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/health
2633 https://nhsd-jira.digital.nhs.uk/browse/APM-3852?src=confmacro

9.111.2 Prepared Release

Release Tag: v2023.01.23.08.57

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=102861&view=results

9.111.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.111.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2634]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.111.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.01.17.15.31**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=101385&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.112 CHG0039901 - APIM Infrastructure changes

Service now[2635]

9.112.1 Tickets

CHG created by **Hannah Ibberson.** The Release implements the changes in the following tickets:

- DOS-2537[2636] - fix ARIA navigation issues on landing page
- DOS-2392[2637] - increase max. file size to 250MB

2634 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2635 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D1d040c941b34e99093bfc88b274bcbbb
2636 https://nhsd-jira.digital.nhs.uk/browse/DOS-2537
2637 https://nhsd-jira.digital.nhs.uk/browse/DOS-2392

- DOS-2541[2638] - add label to banner input field
- DOS-2552[2639] - add alt text for DAQ validator icons
- DOS-2544[2640] - add label to product name input on edit product page
- ADZ-2577 - add "delete app" page

9.112.2 Prepared Release

Release Tag: **v2023.01.27.09.17**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=103923&view=results

9.112.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.112.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2641]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.112.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.01.23.08.57

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=102861&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.113 CHG0040020 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7ab62abc1bbc29d093bfc88b274bcbca

2638 https://nhsd-jira.digital.nhs.uk/browse/DOS-2541
2639 https://nhsd-jira.digital.nhs.uk/browse/DOS-2552
2640 https://nhsd-jira.digital.nhs.uk/browse/DOS-2544
2641 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.113.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-2535
- DOS-2583
- DOS-2582
- DOS-2549
- DOS-2545

9.113.2 Prepared Release

Release Tag: v2023.02.01.12.37

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104305&view=results

9.113.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.113.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2642]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.113.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.01.27.09.17**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=103923&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2642 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.114 CHG0040127 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da26f33551bf8a11493bfc88b274bcb94%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.114.1 Tickets

CHG created by Harry Coulter The Release implements the changes in the following tickets:

- DOS-2553
- NRLF-273

9.114.2 Prepared Release

Release Tag: **v2023.02.02.14.56**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104664&view=results

9.114.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.114.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2643]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.114.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.02.01.12.37**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104305&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2643 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.115 CHG0040159 - APIM EPS Updates

Link to ServiceNow Change request[2644]

9.115.1 Tickets

RFC created by Anthony Brown

The release contains the following completed tickets:

- https://nhsd-jira.digital.nhs.uk/browse/AEA-3023 - Automate the creation of release notes
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2650 - Check whether the X509 Certificate has been revoked
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2775 - Identify the number of authorised eRD repeats within a released prescription
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2796 - Identify the number of authorised eRD repeats within a dispense notification
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2985 - EPSAT - Ability to create eRD and repeat prescriptions from test packs
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2641 - Withdraw the last dispense notification for an eRD prescription
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2648 - Verify chain of valid CA certificates that link this certificate to the NHSD issued subCa certificate
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2976 - improve CI build so they are more stable
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2633 - Identify an eRD prescription instance within a return request
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2661 - Display information about the dispensing org. within a release rejection
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2960 - [dependabot] update nhsuk-react-components
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2976 - improve CI build so they are more stable
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2749 - Verify signatures on prescriptions that are released from Spine
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2758 - Return a prescription to Spine when a signature is invalid
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2957 - [dependabot] update lossless-json
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2661 - Display information about the dispensing org. within a release rejection
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2775 - Identify the number of authorised eRD repeats within a released prescription
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2758 - Return a prescription to Spine when a signature is invalid

2644 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=2ff655a51bbc651493bfc88b274bcb87

- https://nhsd-jira.digital.nhs.uk/browse/AEA-2520 - Representation of Managing Organisation Details in Release Response
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2758 - Return a prescription to Spine when a signature is invalid
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2837 - [SPIKE] Investigate regular failures of CI builds
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2638 - ODS code for payor organisation should be for the BSA
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2826 - Fix API bugs introduced by the release to Production
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2647- Verify whether the X509 Certificate was valid at the time the prescription was signed
- https://nhsd-jira.digital.nhs.uk/browse/AEA-2749 - Verify signatures on prescriptions that are released from Spine

9.115.2 Prepared Release

Release Tag: v1.0.1170-beta

Link to Azure pipeline build:

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104323&view=results

9.115.3 Release Process

We would like to release this ourselves in service hours.

9.115.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.115.4 Roll-Back

Rollback link

9.116 CHG0040263 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4dd2e1be1b7ca1545189855bd34bcbd6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.116.1 Tickets

CHG created by Alex Carrie on behalf of EPS. The Release implements the changes in the following tickets:

- APM-3910[2645] - Assist EPS with HA-proxy config fetching resources from container DONE

9.116.2 Prepared Release

Release Tag: **v2023.02.07.13.31**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105228&view=results

9.116.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.116.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2646]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.116.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.02.02.14.56**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104664&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2645 https://nhsd-jira.digital.nhs.uk/browse/APM-3910?src=confmacro
2646 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.117 CHG0040295 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4b8b56cf1bbce95493bfc88b274bcb7f%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.117.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-2550
- DOS-2539
- DOS-2362
- DOS-2623
- DOS-2622
- ADZ-2580
- ADZ-2543
- ADZ-2539

9.117.2 Prepared Release

Release Tag: v2023.02.07.13.31

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105613&view=results

9.117.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.117.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2647]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.117.4 Roll-Back

- Redeploy:
    -

2647 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

Release Tag: **v2023.02.01.12.37**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=104305&view=results

- Redeploy *last successful non-production release* (to all envs except Production)

9.118 CHG0040638 - Digital Onboarding Service

Link to ServiceNow Change request[2648]

9.118.1 Tickets

CHG created by **Hannah Ibberson**. The Release implements the changes in the following tickets:

- DOS-2547[2649] - Add labels to disabled radio buttons (accessibility)
- DOS-2546[2650] - Add labels to all questions (accessibility)
- DOS-2622[2651] - Remove temporary bug fix code
- DOS-2325[2652] - Fix changelog ordering
- DOS-2654[2653] - Return onboarding lead to review section after reopening it
- DOS-2551[2654] - Restyle table sort (accessibility)
- DOS-2486[2655] - Remove unused policy page (accessibility)

9.118.2 Prepared Release

Release Tag: **v2023.02.15.13.49**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=107200&view=results

9.118.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2648 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3eb0fa5d1b89ed5093bfc88b274bcbf4
2649 https://nhsd-jira.digital.nhs.uk/browse/DOS-2547
2650 https://nhsd-jira.digital.nhs.uk/browse/DOS-2546
2651 https://nhsd-jira.digital.nhs.uk/browse/DOS-2622
2652 https://nhsd-jira.digital.nhs.uk/browse/DOS-2325
2653 https://nhsd-jira.digital.nhs.uk/browse/DOS-2654
2654 https://nhsd-jira.digital.nhs.uk/browse/DOS-2551
2655 https://nhsd-jira.digital.nhs.uk/browse/DOS-2486

9.118.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2656]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.118.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.02.07.13.31

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105613&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.119 CHG0040876 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Daa0469571bc9e11093bfc88b274bcb09

9.119.1 Tickets

CHG created by Harry Coulter The Release implements the changes in the following tickets:

- APM-4022[2657] - Infrastructure release pipeline is destroying proxygen secret-squirrel kvms DONE
- APM-3693[2658] - Split DOS AWS components into separate state file DONE

9.119.2 Prepared Release

v2023.02.23.11.18

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108292&view=results

9.119.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

2656 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2657 https://nhsd-jira.digital.nhs.uk/browse/APM-4022?src=confmacro
2658 https://nhsd-jira.digital.nhs.uk/browse/APM-3693?src=confmacro

2. Click the Release Link above and approve the manual step to deploy the release to prod

9.119.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2659]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.119.4 Roll-Back

- Redeploy:
- - Release Tag: **v2023.02.15.13.49**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=107200&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.120 CHG0040978 - APIM EPS Updates

Link to ServiceNow Change request[2660] **TO DO**

9.120.1 Tickets

RFC created by Anthony Brown

See 2023-02-28[2661] for list of changes

9.120.2 Prepared Release

Release Tag: v1.0.1183-beta

Link to Azure pipeline build:

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=107233&view=results

9.120.3 Release Process

We would like to release this ourselves in service hours.

2659 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2660 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ed72aaf31b41611093bfc88b274bcb65%26sysparm_view=text_search
2661 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/580651611/2023-02-28

9.120.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.120.4 Roll-Back

Rollback link

9.121 CHG0041013 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D922433b31b85251093bfc88b274bcbbf

9.121.1 Tickets

CHG created by Harry Coulter The Release implements the changes in the following tickets:

- APM-3693[2662] - Split DOS AWS components into separate state file **DEV IN PROGRESS**

9.121.2 Prepared Release

v2023.02.23.14.30

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108350&view=results

9.121.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2662 https://nhsd-jira.digital.nhs.uk/browse/APM-3693

9.121.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2663]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.121.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.02.15.13.49**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=107200&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.122 CHG0041198 - Identity Service Updates

Link to Service Now Change[2664]

9.122.1 Tickets

RFC was created by Alex Hawdon The Release implements the changes in the following tickets:

- Code changes for the following tickets
  - APM-3924[2665] - Incorrect wording in Identity Service error message when the Content-Type header is not set correctly on POST DONE
  - APM-3898[2666] - Refresh token expiring before advertised time DONE

9.122.2 Prepared Release

Release Tag: v1.0.1702-alpha+108287

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108287&view=results

2663 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2664 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D58dc58291bd0259493bfc88b274bcb4f%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

2665 https://nhsd-jira.digital.nhs.uk/browse/APM-3924?src=confmacro

2666 https://nhsd-jira.digital.nhs.uk/browse/APM-3898?src=confmacro

9.122.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.122.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2667]
2. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.122.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1691-alpha
- Redeploy *last successful non-production release* (to all envs except production):

9.123 CHG0041210 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3ea58ef51b15a150d1817662164bcb12%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.123.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-3623[2668] - Enable Proxygen prod to be used by the "hello-world" release pipeline DONE
- APM-3123[2669] - CSOC - Onboard KeyCloak Auth and Audit Logs DONE
- DOS-556
- AMB-1457[2670] - Bugfix: access-record - fix broken master release DONE

9.123.2 Prepared Release

Release Tag: **v2023.03.01.15.50**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=109639&view=results

---

2667 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2668 https://nhsd-jira.digital.nhs.uk/browse/APM-3623?src=confmacro
2669 https://nhsd-jira.digital.nhs.uk/browse/APM-3123?src=confmacro
2670 https://nhsd-jira.digital.nhs.uk/browse/AMB-1457?src=confmacro

9.123.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.123.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2671]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.123.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.02.23.14.30**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108350&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.124 CHG0041275 - APIM Infrastructure updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=7db0c41a1bdda950d1817662164bcb0d&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.124.1 Tickets

CHG created by Daniel Brewerton-Singh The Release implements the changes in the following tickets:

- APM-3928[2672] - Certificate expiry - KeyStore api_service-20220208-000000 (internal-dev) DONE
- APM-3935[2673] - Certificate expiry - TrustStore identity-service-truststore (prod) DONE

9.124.2 Prepared Release

Release Tag: **v2023.03.01.15.50**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=109872&view=results

---

2671 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2672 https://nhsd-jira.digital.nhs.uk/browse/APM-3928?src=confmacro
2673 https://nhsd-jira.digital.nhs.uk/browse/APM-3935?src=confmacro

9.124.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.124.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2674]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.124.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.03.01.15.50**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=109639&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.125 CHG0041319 - APIM Infrastructure updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=e16644a71bd525d0c994206fe54bcbe0

9.125.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- ADZ-2659
- ADZ-2657
- ADZ-2691

9.125.2 Prepared Release

Release Tag: **v2023.03.06.09.19**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110339&view=results

2674 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.125.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.125.3.1 Post Release Checks

1. Confirm Keycloak theme has been updated by going to https://identity.prod.api.platform.nhs.uk/auth/realms/developer-identity/protocol/openid-connect/registrations?client_id=digital-onboarding-service&redirect_uri=https%3A%2F%2Fonboarding.prod.api.platform.nhs.uk&response_type=code&scope=openid%20profile and checking the text below the title is:
   You need to provide your details to create an account. For more information on why we collect your information, see our privacy policy.
2. Monitor Grafana Prod dashboard[2675]
3. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.125.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.03.01.15.50**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=109872&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.126 CHG0041490 - APIM EPS Updates

Link to ServiceNow Change request[2676]

9.126.1 Tickets

RFC created by Anthony Brown

See 14-03-23 [PROD][2677] for list of changes

2675 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2676 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7b12cb5c1b6da510c994206fe54bcb0c
2677 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/587376062/14-03-23+PROD

9.126.2 Prepared Release

Release Tag: v1.0.1249-beta

Link to Azure pipeline build:

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110682&view=results

9.126.3 Release Process

We would like to release this ourselves in service hours.

9.126.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.126.4 Roll-Back

Rollback link

9.127 CHG0041498 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3ea58ef51b15a150d1817662164bcb12%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.127.1 Tickets

CHG created Alex Hawdon . The Release implements the changes in the following tickets:

- APM-3942[2678] - Certificate expiry - TrustStore identity-service-truststore (prod) DONE

2678 https://nhsd-jira.digital.nhs.uk/browse/APM-3942?src=confmacro

9.127.2 Prepared Release

Release Tag: **v2023.03.06.09.19**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111150&view=results

9.127.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy **Apigee Deploy Prod stage only**

9.127.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2679]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.127.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.03.01.15.50**

9.128 CHG0041507 - NHS e-RS APIM Patient Care Release R1.10 Production Deployment

9.128.1 Content

High-level summary:

Standard TRM upgrades and security patches.

Tickets:

- NA

Further details on APIM-PC-1.10 Release Page[2680] Release build page.

2679 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2680 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/583833541/APIM-PC-1.10+Release+Page

9.128.2 Prepared Release

Release Tag: 1.10.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110365&view=results

9.128.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.128.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2681]

9.128.4 Roll-Back

- Redeploy:
    - Release Tag: 1.9.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105199&view=results

9.129 CHG0041499 - NHS e-RS APIM Referral Release R1.18 Production Deployment

9.129.1 Content

High-level summary:

Standard TRM upgrades and security patches.

Tickets:

2681 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- NA

Further details on the APIM-R-1.18 Release Page[2682] Release build page.

9.129.2 Prepared Release

Release Tag: 1.18.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110366&view=results

9.129.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.129.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2683]

9.129.4 Roll-Back

- Redeploy:
    - Release Tag: 1.17.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=105346&view=results

9.130 CHG0041524 - Digital Onboarding Service

Link to ServiceNow Change request[2684]

2682 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/564777450/APIM-R-1.18+Release+Page
2683 https://grafana.prod.api.platform.nhs.uk/d/4etllBk7z/api-status?orgId=1&refresh=5m
2684 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D38af48b81be52d10c994206fe54bcbcc%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.130.1 Tickets

CHG created by **Phillip Macdonald**. The Release implements the changes in the following tickets:

- DOS-2730
- DOS-2588
- DOS-2053
- DOS-1092
- DOS-1703
- DOS-2624
- DOS-2710
- DOS-2712
- DOS-2711
- DOS-2677
- ADZ-2588

9.130.2 Prepared Release

Release Tag: v2023.03.10.13.49

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111494&view=logs&j=84fd99cf-33e2-5a32-c4f0-2b3c83c1fc26

9.130.3 Release Process

1. Ensure the DOS Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.130.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2685]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.130.4 Roll-Back

- Redeploy:
    - Run DOS image rollback pipeline to roll back each environment to old version
    - Release Tag: v2023.03.10.13.49
- Redeploy *last successful non-production release* (to all envs except Production)

2685 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.131 CHG0041597 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D039da8391ba9e150d1817662164bcbe6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.131.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- NRLF-273
- APM-2675[2686] - Apigee - collate headers and send to splunk DONE
- ADZ-2706
- AMB-1438[2687] - Authenticate Proxy against GPC PFS keycloak realm (internal-dev) DONE
- APM-3713[2688] - Update shared-flow tests to only use 2nd gen mock auth DONE

9.131.2 Prepared Release

Release Tag: **v2023.03.13.15.18**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111853&view=results

9.131.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy **Apigee Deploy Prod stage only**

9.131.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2689]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2686 https://nhsd-jira.digital.nhs.uk/browse/APM-2675?src=confmacro
2687 https://nhsd-jira.digital.nhs.uk/browse/AMB-1438?src=confmacro
2688 https://nhsd-jira.digital.nhs.uk/browse/APM-3713?src=confmacro
2689 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.131.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.03.06.09.19**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111150&view=results

9.132 CHG0041622 - APIM Infrastructure Releases

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7705f8061bada550d1817662164bcba6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.132.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- NRLF-273[2690] - MVP: Build PROD **IN PROGRESS**
- APM-2675[2691] - Apigee - collate headers and send to splunk **DEV IN PROGRESS**
- AMB-1438[2692] - Authenticate Proxy against GPC PFS keycloak realm (internal-dev) **IN DEVELOPMENT**

9.132.2 Prepared Release

Release Tag: **v2023.03.14.11.29**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=112013&view=results

9.132.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy **Apigee Deploy Prod stage only**

2690 https://nhsd-jira.digital.nhs.uk/browse/NRLF-273
2691 https://nhsd-jira.digital.nhs.uk/browse/APM-2675
2692 https://nhsd-jira.digital.nhs.uk/browse/AMB-1438

9.132.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2693]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.132.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.03.06.09.19**

      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111150&view=results

9.133 CHG0042084 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D-1%26sysparm_query%3Dtype%253dstandard%5Estd_change_producer_version%253d98015d801ba4e11093bfc88b274bcbca%26sysparm_link_parent%3D8b10c51d1bc2b410a4cdc88b274bcb2a%26sysparm_catalog%3De0d08b13c3330100c8b837659bba8fb4%26sysparm_catalog_view%3Dcatalog_default

9.133.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

- https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.03.13.15.18...v2023.03.21.14.12

9.133.2 Prepared Release

Release Tag: **v2023.03.21.14.12**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=113929&view=results

9.133.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2693 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.133.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2694]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.133.4 Roll-Back

- Redeploy:
  - Release Tag: Terrafrom - v2023.03.13.15.18 | Apigee - v2023.03.14.11.29
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=111865&view=results | https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=112013&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.134 CHG0042126 - APIM EPS Updates

Link to ServiceNow Change request[2695]

9.134.1 Tickets

RFC created by Anthony Brown

See 28-03-2023 [PROD][2696] for list of changes

9.134.2 Prepared Release

Release Tag: v1.0.1249-beta

Link to Azure pipeline build:

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=113911&view=results

9.134.3 Release Process

We would like to release this ourselves in service hours.

2694 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2695 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D98fa48d51bf52954c994206fe54bcbc8
2696 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/587390092/28-03-2023+PROD

9.134.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.134.4 Roll-Back

Rollback link

9.135 CHG0042198 - Digital Onboarding Content release for DAQ 11.2 Production

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df61ec85a1b752594c994206fe54bcbdd%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.135.1 Tickets

CHG created by **Scott Williams** The Release implements the changes in the following tickets:

- DOS-2750

9.135.2 Prepared Release

Link to DAQ 11.2 Production release[2697]

9.135.3 Release Process

Documentation on <How to release a DAQ to Production[2698]>

2697 https://hscic365.sharepoint.com/:x:/r/sites/OnboardingforConnectingSystems/_layouts/15/Doc.aspx?sourcedoc=%7BCEDD93A7-1CFF-40A5-9D47-C07E7454D9AA%7D&file=DAQ%2011.2%20-%20PROD%20RELEASE.xlsx&wdOrigin=TEAMS-ELECTRON.p2p.mw&_DSL=1&action=default&mobileredirect=true

2698 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.135.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.135.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2699]>

9.136 CHG0042217 - APIM Infrastructure updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D-1%26sysparm_catalog%3De0d08b13c3330100c8b837659bba8fb4%26sysparm_catalog_view%3Dcatalog_default%26sysparm_query%3Dtype%253Dstandard%255Estd_change_producer_version%253D98015d801ba4e11093bfc88b274bcbca%26sysparm_link_parent%3D8b10c51d1bc2b410a4cdc88b274bcb2a

9.136.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

- https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.03.21.14.12...v2023.03.28.12.18

9.136.2 Prepared Release

Release Tag: **v2023.03.28.12.18**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114750&view=results

9.136.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. AWS and Apigee
3. Click the Release Link above and approve the manual step to deploy the release to prod

2699 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.136.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2700]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.136.4 Roll-Back

- Replace the secrets
    - *prod/server-certs/api.service.nhs.uk/crt* with the value of *prod/server-certs/api.service.nhs.uk/crt-OLD*
    - *prod/server-certs/api.service.nhs.uk/key* with the value of *prod/server-certs/api.service.nhs.uk/key-OLD*
- Rebuild: **v2023.03.21.14.12**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=113928&view=results
- Redeploy:
    - **v2023.03.21.14.12**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114163&view=results

9.137 CHG0042220 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da1e7341a1b35a594c994206fe54bcbba%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.137.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-2754
- DOS-2760
- DOS-1438
- DOS-2769
- ADZ-2579
- ADZ-2688
- ADZ-2730

2700 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.137.2 Prepared Release

Release Tag: v2023.03.20.13.01+114599

<Link to Azure https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114599&view=results

9.137.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.137.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2701]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.137.4 Roll-Back

- Redeploy:
    - Release Tag: main+112202
    - **<Link** to Azure devops https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=112202&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.138 CHG0042299 - APIM Infrastructure updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dcd8ab13a1b716994d1817662164bcba2

9.138.1 Tickets

CHG created by Alex Carrie The Release implements the changes in the following tickets:

- https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.03.21.14.12...v2023.03.28.12.18

2701 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.138.2 Prepared Release

Release Tag: **v2023.03.28.12.18**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114750&view=results

9.138.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. AWS and Apigee
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.138.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2702]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.138.4 Roll-Back

- Replace the secrets
    - *prod/server-certs/api.service.nhs.uk/crt* with the value of *prod/server-certs/api.service.nhs.uk/crt-OLD*
    - *prod/server-certs/api.service.nhs.uk/key* with the value of *prod/server-certs/api.service.nhs.uk/key-OLD*
- Rebuild: **v2023.03.21.14.12**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=113928&view=results
- Redeploy:
    - **v2023.03.21.14.12**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114163&view=results

2702 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.139 CHG0042547 - NHS e-RS APIM Referral Release R1.19 Production Deployment

9.139.1 Content

High-level summary:

- ERSSUP-66713[2703]: APIM-R - comma truncates file name on upload
- ERSSUP-66523[2704] - APIM-R - FHIR R4 remove from documentation the reference to profiles

Tickets:

- NA

Further details on the APIM 1.19 Release Page (APIM-R)[2705]Release build page.

9.139.2 Prepared Release

Release Tag: 1.19.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=115716&view=results

9.139.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.139.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2706]

2703 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-66713
2704 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-66523
2705 https://nhsd-confluence.digital.nhs.uk/pages/viewpage.action?pageId=587389361
2706 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.139.4 Roll-Back

- Redeploy:
    - Release Tag: 1.18.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110366&view=results

9.140 CHG0042546 - NHS e-RS APIM Patient Care Release R1.11 Production Deployment

9.140.1 Content

High-level summary:

- ERSSUP-66050[2707]: Refactor APIM-R and APIM-PC manifest_template.yml files
- Minor version upgrades

Tickets:

- NA

Further details on APIM-PC-1.11 Release Page[2708] Release build page.

9.140.2 Prepared Release

Release Tag: 1.11.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116309&view=results

9.140.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

2707 https://nhsd-jira.digital.nhs.uk/browse/ERSSUP-66050
2708 https://nhsd-confluence.digital.nhs.uk/display/ERS/APIM-PC-1.11+Release+Page

9.140.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2709]

9.140.4 Roll-Back

- Redeploy:
    - Release Tag: 1.10.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=110365&view=results

9.141 CHG0042663 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5f703b1d1b426598d1817662164bcbd4

Please note that this release is Apigee-only and does not represent any changes to the underlying platform infrastructure.

9.141.1 Tickets

CHG created by Harry Coulter The Release implements the changes in the following tickets:

- APM-4085[2710] - Certificate expiry - TrustStore - *.login.nhs.uk DONE

9.141.2 Prepared Release

Release Tag: **v2023.03.28.12.18**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116296&view=results

9.141.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2709 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

2710 https://nhsd-jira.digital.nhs.uk/browse/APM-4085?src=confmacro

9.141.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2711]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.141.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.03.28.12.18
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116296&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.142 CHG0042712- Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dfc93ad4f1b46a9d8c994206fe54bcb59%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.142.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-2752
- DOS-2755
- DOS-2913
- DOS-2923

9.142.2 Prepared Release

Release Tag: <**eg. v2023.03.20.13.01+116563**>

<**Link to Azure** devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116563&view=results>

9.142.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

2711 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.142.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2712]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.142.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.03.20.13.01+114599
  - <**Link** to Azure devops https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114599&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.143 CHG0042721 - Digital Onboarding Content release for DAQ 11.3 Production

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D9ec6bd831b4629d8d1817662164bcbfb%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.143.1 Tickets

CHG created by **Andy Thompson**. The Release implements the changes in the following tickets:

- DOS-2832
- DOS-2929
- DOS-2655

9.143.2 Prepared Release

Link to DAQ 11.3 - PROD RELEASE[2713]

2712 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2713 https://hscic365.sharepoint.com/:x:/r/sites/OnboardingforConnectingSystems/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2011.3%20-%20PROD%20RELEASE.xlsx?d=w096d0a95b038405fb4e8030935e9293b&csf=1&web=1&e=e1wDdO

9.143.3 Release Process

1. Documentation on <How to release a DAQ to Production[2714]>

9.143.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.143.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2715]>

9.144 CHG0042720 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd234790f1b42e9d8c994206fe54bcb79

9.144.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

9.144.2 Prepared Release

Release Tag: **v2023.04.11.14.41**

https://dev.azure.com/NHSD-APIM/API Platform/_build/results?buildId=116586&view=results

9.144.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy **Apigee Deploy Prod/PTL stages only**

2714 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

2715 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.144.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2716]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.144.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.03.28.12.18**
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=114750&view=results

9.145 CHG0042725 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dff41028b1b8ae9d8c994206fe54bcb30%26sysparm_record_target%3Dtask%26sysparm_record_row%3D9%26sysparm_record_rows%3D9%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

Releases changes to shared flows so is outside of normal standard change.

The release changes the way we log to splunk. This will not effect traffic as it it's only changes to the post-proxy flow.

9.145.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4118[2717] - Splunk logging to include headers - log headers as json object not json string
  DONE

9.145.2 Prepared Release

Release Tag: **v2023.04.11.15.05**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116611&view=results

2716 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2717 https://nhsd-jira.digital.nhs.uk/browse/APM-4118?src=confmacro

9.145.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.145.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2718]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.145.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.04.11.14.41**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116586&view=results

9.146 CHG0042923 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D51e7e8441b96ed1cc994206fe54bcba0

9.146.1 Tickets

CHG created by Phillip Macdonald. The Release implements the changes in the following tickets:

- ADZ-2774
- DOS-2895
- DOS-2427
- DOS-2756
- DOS-2961

9.146.2 Prepared Release

Release Tag: **v2023.04.12.10.40+117048**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=117048&view=results

---

2718 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.146.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.146.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2719]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.146.4 Roll-Back

- Redeploy:
  - Release Tag: <**eg. v2023.03.20.13.01+116563**>

    <**Link to Azure** devops results, e.g. https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116563&view=results>
- Redeploy *last successful non-production release* (to all envs except Production)

9.147 CHG0043067 - Digital Onboarding Content release for DAQ 11.4 Production

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dab8ffd551b5ead9cd1817662164bcb5f

9.147.1 Tickets

CHG created by **Jess Emery** The Release implements the changes in the following tickets:

- DOS-2971: SCR use case section addition and tech conformance updates
- DOS-2798: HRDC removal of tech conformance change
- DOS-2941: BaRs updated clinical safety questions
- DOS-2652: PDS additional tech conformance question

2719 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.147.2 Prepared Release

Link to: DAQ 11.4 - PROD RELEASE.xlsx[2720]

9.147.3 Release Process

Documentation on <How to release a DAQ to Production[2721]>

9.147.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.147.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2722]>

9.148 CHG0043151 - APIM EPS Updates

Link to ServiceNow Change request[2723]

9.148.1 Tickets

RFC created by John Kitson

See 28-04-2023 [PROD][2724] for list of changes

9.148.2 Prepared Release

Release Tag: v1.0.1347-hotfix-prod-deployment+118771

2720 https://hscic365.sharepoint.com/:x:/r/sites/OnboardingforConnectingSystems/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2011.4%20-%20PROD%20RELEASE.xlsx?d=wdd5eee093a7b4415aa06b6a8248b883f&csf=1&web=1&e=HwWfJg

2721 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

2722 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

2723 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6183df2d1b126d9cc994206fe54bcbf4

2724 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/608903583/28-04-2023+PROD

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=118771&view=results

9.148.3 Release Process

We would like to release this ourselves in service hours.

9.148.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.148.4 Roll-Back

Rollback link

9.149 CHG0043556 - APIM Infrastructure Updates

CHG0043556[2725]

Releases changes to shared flows so is outside of normal standard change.

The release is to bring APIM Infrastructure up to date with the codebase; it does not contain any functional changes.

9.149.1 Tickets

- N/A - Refresh/update infra only

9.149.2 Prepared Release

Release Tag: **v2023.04.24.15.56**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119069&view=results

2725 https://nhsdigitallive.service-now.com/change_request.do?sys_id=8cbb57841ba26d90d1817662164bcb2e&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.149.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.149.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2726]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.149.4 Roll-Back

- Redeploy:
    - Release Tag:
        - AWS: v2023.03.28.12.18
        - Apigee: v2023.04.11.15.05

9.150 CHG0043595 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=246fc4dc1ba2e590c994206fe54bcbf5

9.150.1 Tickets

CHG created by Lee Richardson. The Release implements the changes in the following tickets:

ADZ-2747 - Cache or workaround large Apigee call

DOS-2784 - Add the ability to switch roles in sys admin org

DOS-2653 - Ability to upload using video file formats
DOS-3004 - Fix typo in accessibility statement
ADZ-2626 - Miss out the extra "My developer account - BETA" screen when logging into developer account, go straight to "My applications and Teams" screen instead
DOS-2755 - Update the Add product files section of the Product page
 - Added better detection of section changes for alerts and fix login error
ADZ-2705 - Determine if we will need to delete existing team apps and recreate them as regular apps.
ADZ-2718 - Emergency Developer Portal Shutdown
DOS-2963 - Additional Answer Focus shifting

2726 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.150.2 Prepared Release

Release Tag: **v2023.04.12.10.40+119231**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119231&view=results

9.150.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.150.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2727]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.150.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.04.12.10.40+117048**

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=117048&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.151 CHG0043908 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De94100e61bea2554d1817662164bcbec%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

Releases changes to shared flows so is outside of normal standard change.

https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.04.24.15.56...v2023.05.04.08.56

- Bump python version for slack notify for aws
- Log header names as lower case for apigee -

  APM-2550[2728] - Splunk logging to include headers - redacted index DONE

2727 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2728 https://nhsd-jira.digital.nhs.uk/browse/APM-2550?src=confmacro

PTL release - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119607&view=results

9.151.1 Prepared Release

Release Tag: **v2023.05.04.08.56**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119666&view=results

9.151.2 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.151.2.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2729]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.151.3 Roll-Back

- Redeploy:
  - Release Tag: **v2023.04.24.15.56**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119069&view=results

9.152 CHG0043949- APIM EPS Updates

Link to ServiceNow Change request[2730]

9.152.1 Tickets

RFC created by John Kitson

See 10-05-2023 [PROD][2731] for list of changes

2729 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2730 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=c9d2503e1b6e6110c994206fe54bcb05
2731 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/608916916/10-05-2023+PROD

9.152.2 Prepared Release

Release Tag: v1.0.1347-hotfix-prod-deployment+118771

Link to Azure pipeline build: <TBA>

9.152.3 Release Process

We would like to release this ourselves in service hours.

9.152.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.152.4 Roll-Back

Rollback link

9.153 CHG0043967 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=fc2709b61b662510c994206fe54bcb20

9.153.1 Tickets

CHG created by Lee Richardson. The Release implements the changes in the following tickets:

DOS-1721 - Add support for Question Settings to CheckBox question type

ADZ-2626 - Revert environment access link change

DOS-2784 - Add the ability to switch roles in sys admin org

DOS-2824 - Add link to developer help section on API Selection page

ADZ-2786 - Sort the list of APIs (shown when adding APIs to an application) into alphabetical order

DOS-2542 - ARIA tags and high contrast for form elements (WCAG 2.1 AA)

ADZ-2747 - Cache or workaround large Apigee call

9.153.2 Prepared Release

Release Tag: **v2023.04.12.10.40+119796**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119796&view=results

9.153.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.153.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2732]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.153.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.04.12.10.40+119231**
    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119231&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.154 CHG0044062 - NHS e-RS APIM Patient Care Release R1.12 Production Deployment

9.154.1 Content

High-level summary:

- tbc

Tickets:

- NA

2732 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

Further details on APIM-PC-1.12 Release Page[2733] Release build page.

9.154.2 Prepared Release

Release Tag: 1.12.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119968&view=results

9.154.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.154.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2734]

9.154.4 Roll-Back

- Redeploy:
    - Release Tag: 1.11.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=116309&view=results

9.155 CHG0044059 - NHS e-RS APIM Referral Release R1.20 Production Deployment

9.155.1 Content

High-level summary:

2733 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/608911144/APIM-PC-1.12+Release+Page
2734 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- tbc

Tickets:

- NA

Further details on the APIM-R-1.20 Release Page[2735] Release build page.

9.155.2 Prepared Release

Release Tag: 1.20.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119967&view=results

9.155.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

9.155.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2736]

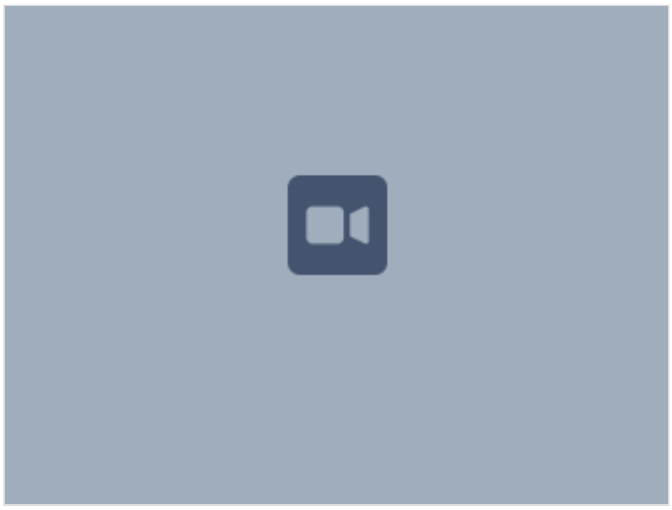
9.155.4 Roll-Back

- Redeploy:
    - Release Tag: 1.19.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=115716&view=results

Recording of the release call:

2735 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/608911168/APIM-R-1.20+Release+Page
2736 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m



9.156 CHG0044171 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D51b793241bf66dd4d1817662164bcb25%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.05.04.08.56...v2023.05.11.10.02

- APM-4242[2737] - Address AWS Security Hub alerts - S3 buckets DONE

9.156.1 Prepared Release

Release Tag: **v2023.05.11.10.02**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=120186&view=results

9.156.2 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run `terraform state mv module.account.aws_s3_bucket.kinesis_backsplash_bucket module.account.module.s3_kinesis_backsplash.aws_s3_bucket.s3` in both ptl and prod accounts to update terraform state
3. Click the Release Link above and approve the manual step to deploy the release to prod

2737 https://nhsd-jira.digital.nhs.uk/browse/APM-4242?src=confmacro

9.156.2.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2738]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.156.3 Roll-Back

- Redeploy:
  - Release Tag: **v2023.05.04.08.56**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119666&view=results

9.157 CHG0044202 - Digital Onboarding Content release for DAQ 11.5 Production

9.157.1 Tickets

CHG created by **Ernest Kissiedu** The Release implements the changes in the following tickets:

- DOS-2971[2739] - NHS Login: Callout warnings update
- DOS-2798[2740] - Typo in Declare Data Security and Information Security

9.157.2 Prepared Release

Link to: DAQ 11.5 - PROD RELEASE.xlsx[2741]

9.157.3 Release Process

Documentation on <How to release a DAQ to Production[2742]>

2738 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2739 https://nhsd-jira.digital.nhs.uk/browse/DOS-2910
2740 https://nhsd-jira.digital.nhs.uk/browse/DOS-2651
2741 https://hscic365.sharepoint.com/:x:/r/sites/OnboardingforConnectingSystems/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2011.5%20-%20PROD%20RELEASE.xlsx?d=we28feae18ce144c3ad39f85f85d1d37d&csf=1&web=1&e=btgQ6z
2742 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.157.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.157.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2743]>

9.158 CHG0044409 - APIM EPS Updates

Link to ServiceNow Change request[2744]

9.158.1 Tickets

RFC created by John Kitson

See 17-05-2023 [PROD][2745] for list of changes

9.158.2 Prepared Release

Release Tag: <TBA>

Link to Azure pipeline build: <TBA>

9.158.3 Release Process

We would like to release this ourselves in service hours.

9.158.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping

2743 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful
2744 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=d198f92a1b322550c994206fe54bcb50
2745 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/626237437/2023-17-05+PROD

2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.158.4 Roll-Back

Rollback link

9.159 CHG0044433 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D26ea62ae1bf6e150d1817662164bcb3c%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.159.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-3030 Do not remove error message when field loses focus
- DOS-2785 API Setup: View API setup and sections in Admin
- ADZ-2745 Write teams migration scripts
- ADZ-2630 Audit trail for API removal on My Applications screen

9.159.2 Prepared Release

Release Tag: v2023.05.17.13.28+120834

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=120834&view=results

9.159.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.159.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2746]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.159.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.04.12.10.40+119796
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119796&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.160 CHG0044790 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D470790dd1b832914d1817662164bcbf1%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.160.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- (n/a, BAU release)

9.160.2 Prepared Release

Release Tag: v2023.05.24.10.43[2747]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121524&view=results

9.160.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2746 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2747 https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.05.11.10.02...v2023.05.24.10.43

9.160.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2748]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.160.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.05.11.10.02**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=120186&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.161 CHG0044977 - APIM EPS Updates

Link to ServiceNow Change request[2749]

9.161.1 Tickets

RFC created by Anthony Brown

See 06-06-2023 [PROD][2750] for list of changes

9.161.2 Prepared Release

Release Tag v1.0.1382-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.161.3 Release Process

We would like to release this ourselves in service hours.

2748 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2749 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=93230b471b03e1d4c994206fe54bcba9
2750 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/631285119/06-06-2023+PROD

9.161.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.161.4 Roll-Back

Rollback link

9.162 CHG0045027 - NHS e-RS APIM Patient Care Release R1.13 Production Deployment

9.162.1 Content

High-level summary:

- Introduction of GitHub Code Owners mechanism to ensure only e-RS team approves PRs to develop.
- Fixes of OAS bugs caused by the new RapiDoc renderer
- Stack upgrades

Tickets:

- NA

Further details on APIM-PC-1.13 Release Page[2751] Release build page.

9.162.2 Prepared Release

Release Tag: 1.13.1-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123102&view=results

2751 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/626242249/APIM-PC-1.13+Release+Page

9.162.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.162.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2752]

9.162.4 Roll-Back

- Redeploy:
  - Release Tag: 1.12.0-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119968&view=results

9.163 CHG0045025 - NHS e-RS APIM Referral Release R1.21 Production Deployment

9.163.1 Content

High-level summary:

- Introduction of GitHub Code Owners mechanism to ensure only e-RS team approves PRs to develop.
- Fixes of OAS bugs caused by the new RapiDoc renderer
- Stack upgrades

Tickets:

- NA

2752 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

Further details on the APIM-R-1.21 Release Page[2753] Release build page.

9.163.2 Prepared Release

Release Tag: 1.21.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=122317&view=results

9.163.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.163.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2754]

9.163.4 Roll-Back

- Redeploy:
  - Release Tag: 1.20.0-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119967&view=results

9.164 CHG0045116 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D-1%26sysparm_query%3Dtype%253dstandard%5Estd_change_producer_version%253d98015d801ba4e11093bfc88b274bcbca%26sysparm_link_parent%3D8b10c51d1bc2b410a4cdc8

2753 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/626241657/APIM-R-1.21+Release+Page
2754 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

8b274bcb2a%26sysparm_catalog%3De0d08b13c3330100c8b837659bba8fb4%26sysparm_catalog_view%3Dcatalog_default

9.164.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- (n/a, BAU release)

9.164.2 Prepared Release

Release Tag: v2023.06.02.12.41[2755]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=122277&view=results

9.164.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.164.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2756]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.164.4 Roll-Back

- Redeploy:
  - **Release Tag:** v2023.05.24.10.43[2757]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121524&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.165 CHG0045315 - APIM EPS Updates

Link to ServiceNow Change request[2758]

2755 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2023.06.02.12.41
2756 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2757 https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.05.11.10.02...v2023.05.24.10.43
2758 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=75171d8a1b576d50c994206fe54bcb45

9.165.1 Tickets

RFC created by Anthony Brown

See 13-06-2023 [PROD][2759] for list of changes

9.165.2 Prepared Release

Release Tag v1.0.1390-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.165.3 Release Process

We would like to release this ourselves in service hours.

9.165.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.165.4 Roll-Back

Rollback link

9.166 CHG0045108 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Deea1d4f81b176d18c994206fe54bcbbc%26sysparm_view%3Dtext_search

9.166.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

2759 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/626255802/13-06-2023+PROD

- DOS-3113 ODS validation bug fix
- ADZ-2782 Provide more information to a user about our environments
- ADZ-2790 My application public key URL validation timeout handling

9.166.2 Prepared Release

Release Tag: v2023.06.07.12.44+123093

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123347&view=results (actual pipeline run for the same DOS/Infra build)

9.166.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.166.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2760]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.166.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.06.07.12.44+123090 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123090&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.167 CHG0039918 - APIM - NHS App Updates for Comms Hub:

CHG0039918[2761] - Added in March but due to issues transitioning from previous Producer - delayed.

9.167.1 Tickets

CHG0039918 created by Jason Pratt. The Release implements the changes in the following tickets:

2760 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2761 https://nhsdigitallive.service-now.com/change_request.do?sys_id=671528901b382d9093bfc88b274bcb94&sysparm_record_target=change_request&sysparm_record_row=52&sysparm_record_rows=365&sysparm_record_list=active%3Dtrue%5EORDERBYDESCstart_date

- NHSO-26287[2762] - N&M API software updates including:
  - NHSO-0000: Revert to Speccy so references are resolved.
  - Bump gitpython from 3.0.5 to 3.1.30
  - Bump json5 and react-hot-loader
  - Bump json-pointer from 0.6.1 to 0.6.2
  - cis2-scope-to-manifest
  - allow AMB ticket branches
  - Install node.js version 14, instead of 13
  - AMB-1179-adding-cis2-scope-to-manifest
  - Bump express from 4.17.1 to 4.18.2 in /sandbox
  - Bump express from 4.17.1 to 4.18.2
  - ~~NHSO-24008~~: Update node version.
  - Bump qs from 6.5.2 to 6.5.3
  - Bump decode-uri-component from 0.2.0 to 0.2.2
  - Bump loader-utils from 1.2.3 to 1.4.2
  - Bump minimatch from 3.0.4 to 3.1.2
  - Bump json-schema and jsprim
  - Bump moment from 2.29.2 to 2.29.4
  - Bump jszip and apigeetool
  - AMB-1223 replace speccy lint
  - Bump crypto-js and newman
  - ~~NHSO-28492~~: Increase campaign-id and request-id field lengths to 128 characters
  - NHSO-0000: Upgrade minimist to fix a critical vulnerability.

9.167.2 Prepared Release

Release Tag: <v1.0.243-alpha+123179 >

<https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123179&view=results>

9.167.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2762 https://nhsd-jira.digital.nhs.uk/browse/NHSO-26287

9.167.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2763]
2. Monitor APIM Dashboard for NHS App CommsHub ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.167.4 Roll-Back

- Redeploy:
  - Release Tag: <#v1.0.180-alpha+121388>
  - <**Link** to Azure devops results -[2764] Link not available yet, as it would roll back the documentation - but tested as available for Rollback.

9.168 CHG0045519 - APIM EPS Updates

Link to ServiceNow Change request[2765]

9.168.1 Tickets

RFC created by Anthony Brown

See 15-06-2023 [PROD][2766] for list of changes

9.168.2 Prepared Release

Release Tag v1.0.1393-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.168.3 Release Process

We would like to release this ourselves in service hours.

2763 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2764 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=39090&view=results
2765 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b9a29f731b5fadd0c994206fe54bcb12
2766 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/641176116/15-06-2023+PROD

9.168.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.168.4 Roll-Back

Rollback link

9.169 CHG0045620- APIM EPS Updates

Link to ServiceNow Change request[2767]

9.169.1 Tickets

RFC created by Anthony Brown

See 20-06-2023 [PROD][2768] for list of changes

9.169.2 Prepared Release

Release Tag v1.0.1394-beta[2769]

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.169.3 Release Process

We would like to release this ourselves in service hours.

2767 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=520ffb411be3e51093bfc88b274bcb37
2768 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/641180970/20-06-2023+PROD
2769 https://nhsd-jira.digital.nhs.uk/browse/AEA/fixforversion/41830

9.169.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.169.4 Roll-Back

Rollback link

9.170 CHG0045708 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D38b875251befed105189855bd34bcbf2%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.170.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4364[2770] - Renew api_service-20220629-000000 dev cert and key for KEYSTORE DONE

9.170.2 Prepared Release

Release Tag: v2023.06.20.09.58

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123826&view=results

9.170.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2770 https://nhsd-jira.digital.nhs.uk/browse/APM-4364?src=confmacro

9.170.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2771]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.170.4 Roll-Back

- Redeploy:
  - **Release Tag: v2023.06.13.06.11**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123089&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.171 CHG0045726 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=c8b4aae11ba3a1505189855bd34bcb69

9.171.1 Tickets

CHG created by Lee Richardson The Release implements the changes in the following tickets:

- nuget-package-fix - Explicitly reference some implied packages
- dependabot/pip/tests/dos-auto-assessment/cryptography-41.0.0 - Bump version
- ADZ-2804 - Link directly from home page to My Applications
- ADZ_2663 - Improve text advising people what to do from the "My developer account" screen
- DOS-2548 - Bugfix. Re-reviewing a section - "issue has occurred"
- DOS-3120 - Re-reviewing a section - "issue has occurred"
- DOS-3129 - Remove hardcoded message content
- Enable email lambda

9.171.2 Prepared Release

Release Tag: v2023.06.07.12.44+123867

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123867&view=results (actual pipeline run for the same DOS/Infra build)

2771 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.171.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.171.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2772]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.171.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.06.07.12.44+123093
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123347&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.172 CHG0045741 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D38b875251befed105189855bd34bcbf2%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.172.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- Updating prod cert/key for Covid-19 Test Results API

9.172.2 Prepared Release

Release Tag: v2023.06.20.09.58

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123826&view=results

2772 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.172.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod ***(Apigee only)***

9.172.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2773]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.172.4 Roll-Back

- Redeploy:
    - **Release Tag: v2023.06.13.06.11**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123089&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.173 CHG0045793 - APIM EPS Updates

Link to ServiceNow Change request[2774]

9.173.1 Tickets

RFC created by John Kitson

See 23-06-2023 [PROD][2775] for list of changes

9.173.2 Prepared Release

Release Tag v1.0.1395-beta[2776]

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

---

2773 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2774 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=882547821be72d505189855bd34bcbb7
2775 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/641182877/23-06-2023+PROD
2776 https://nhsd-jira.digital.nhs.uk/issues/?jql=project+%3D+AEA+AND+fixVersion+%3D+v1.0.1395-beta

9.173.3 Release Process

We would like to release this ourselves in service hours.

9.173.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.173.4 Roll-Back

Rollback link

9.174 CHG0046077 - APIM EPS Updates

Link to ServiceNow Change request[2777]

9.174.1 Tickets

RFC created by John Kitson

See 28-06-2023 [PROD][2778] for list of changes

9.174.2 Prepared Release

Release Tag v1.0.1397-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

2777 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=7c256604c3ff65506c39935c0501318f
2778 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/648285018/28-06-2023+PROD

9.174.3 Release Process

We would like to release this ourselves in service hours.

9.174.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.174.4 Roll-Back

Rollback link

9.175 CHG0046184 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D6294d064c37321906c39935c050131da%26sysparm_view%3Dtext_search%26sysparm_view%3Dtext_search%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.175.1 Tickets

CHG created by Jon Marston . The Release implements the changes in the following tickets:

- APM-3250[2779] - Handle deletion of products when there are subscribed apps in proxygen DONE
- APM-4426[2780] - Secret deployment fails on Prod proxies with no docker containers defined DONE

9.175.2 Prepared Release

Release Tag: 2023.06.29.09.41[2781]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125050&view=results

---

2779 https://nhsd-jira.digital.nhs.uk/browse/APM-3250?src=confmacro
2780 https://nhsd-jira.digital.nhs.uk/browse/APM-4426?src=confmacro
2781 https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.06.20.09.58...v2023.06.29.09.41

9.175.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod ***(Apigee only)***

9.175.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2782]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.175.4 Roll-Back

- Redeploy:
    - **Release Tag: v2023.06.20.09.58**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123826&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.176 CHG0046325 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De11f11bdc37fa5d06c39935c050131c5%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.176.1 Tickets

CHG created by Jon Marston . The Release implements the changes in the following tickets:

- APM-3424[2783] - Handle clean up of proxygen PRs on Apigee and AWS DONE
- AMB-1577

9.176.2 Prepared Release

Release Tag: v2023.07.04.08.48

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125497&view=results

---

2782 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2783 https://nhsd-jira.digital.nhs.uk/browse/APM-3424?src=confmacro

9.176.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod ***(Apigee only)***

9.176.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2784]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.176.4 Roll-Back

- Redeploy:
    - **Release Tag:** 2023.06.29.09.41[2785]
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125050&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.177 CHG0046555 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=17dd0cfac33be9146c39935c050131d9

9.177.1 Tickets

CHG created by Alexander Carrie . The Release implements the changes in the following tickets:

- APM-3424[2786] - Handle clean up of proxygen PRs on Apigee and AWS DONE

Changes to release - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.07.04.08.48...v2023.07.06.09.38

9.177.2 Prepared Release

Release Tag: v2023.07.06.09.38

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125976&view=results

---

2784 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2785 https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.06.20.09.58...v2023.06.29.09.41
2786 https://nhsd-jira.digital.nhs.uk/browse/APM-3424?src=confmacro

9.177.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.177.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2787]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.177.4 Roll-Back

- Redeploy:
  - **Release Tag: v2023.07.04.08.48**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125497&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.178 CHG0046713 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dadff815cc38435946c39935c0501313a%26sysparm_view%3Dtext_search

9.178.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-3112 Remove manual invitation page
- DOS-3082 Improved guidance content on Organisation set up flow
- DOS-2788 API Setup: Add sections to API in Admin
- DOS-2797 Re-add feedback page element
- ADZ-2663 Developer guidance (spelling correction)

9.178.2 Prepared Release

Release Tag: v2023.06.27.15.29+126131

2787 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=126131&view=results (actual pipeline run for the same DOS/Infra build)

9.178.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.178.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2788]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.178.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.06.07.12.44+123999 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123999&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.179 CHG0046042 - NHS e-RS APIM Patient Care Release R1.14 Production Deployment

9.179.1 Content

High-level summary:

- Introduction of GitHub Code Owners mechanism to ensure only e-RS team approves PRs to develop
- Fixes of OAS bugs caused by the new RapiDoc renderer
- Stack upgrades

Tickets:

- NA

Further details on APIM-PC-1.14 Release Page[2789]

2788 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2789 https://nhsd-confluence.digital.nhs.uk/display/ERS/APIM-PC-1.14+Release+Page

9.179.2 Prepared Release

Release Tag: 1.14.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125572&view=results

9.179.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.179.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2790]

9.179.4 Roll-Back

- Redeploy:
  - Release Tag: 1.13.1-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=123102&view=results

2790 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.180 CHG0046041 NHS e-RS APIM Referral Release R1.22 Deployment (Prod)

9.180.1 Content

High-level summary:

- Introduction of GitHub Code Owners mechanism to ensure only e-RS team approves PRs to develop.
  Fixes of OAS bugs caused by the new RapiDoc renderer
  Stack upgrades

Tickets:

- NA

Further details on the APIM 1.22 Release Build Page (APIM-R)[2791]

9.180.2 Prepared Release

Release Tag: 1.22.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125574&view=results

9.180.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.180.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)

2791 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/626254668/APIM-R-1.22+Release+Page

2. Monitor Grafana Prod dashboard[2792]

9.180.4 Roll-Back

- Redeploy:
    - Release Tag: 1.21-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=119967&view=results

9.181 CHG0046633 - APIM EPS Updates

Link to ServiceNow Change request[2793]

9.181.1 Tickets

RFC created by John Kitson

See 10-07-2023 [PROD][2794] for list of changes

9.181.2 Prepared Release

Release Tag v1.0.1398-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.181.3 Release Process

We would like to release this ourselves in service hours.

9.181.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems

2792 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2793 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f28f7733c3bb2d546c39935c05013168
2794 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/657326123/10-07-2023+PROD

4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.181.4 Roll-Back

Rollback link

9.182 CHG0046863 - APIM EPS Updates

Link to ServiceNow Change request[2795]

9.182.1 Tickets

RFC created by John Kitson

See 13-07-2023 [PROD][2796] for list of changes

9.182.2 Prepared Release

Release Tag v1.0.1399-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.182.3 Release Process

We would like to release this ourselves in service hours.

9.182.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

2795 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b9804078c300b1d46c39935c050131fa
2796 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/657329291/13-07-2023+PROD

9.182.4 Roll-Back

Rollback link

9.183 CHG0047223 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=e485d0d7c30cf19042453fcb050131b0

9.183.1 Tickets

CHG created by Lee Richardson The Release implements the changes in the following tickets:

- DOS-2789 - Remove sections
- ADZ-2629 - Add filtering when adding an API to an application

9.183.2 Prepared Release

Release Tag: v2023.06.27.15.29+126131

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=128412 (actual pipeline run for the same DOS/Infra build)

9.183.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.183.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2797]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.183.4 Roll-Back

- Redeploy:

2797 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

    - Release Tag: v2023.06.27.15.29+126131
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=126131&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.184 CHG0047419 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4ebed5b88394bd1000adf388beaad3c8

9.184.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

9.184.2 Prepared Release

Release Tag: v2023.07.25.09.59

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=128610&view=results

9.184.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod ***(Apigee only)***

9.184.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2798]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.184.4 Roll-Back

- Redeploy:
    - **Release Tag: v2023.07.04.08.48**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125497&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2798 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.185 CHG0047578 - Digital Onboarding Content release for DAQ 11.6 Production

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=068144198398b95000adf388beaad356

9.185.1 Tickets

CHG created by **Andy Thompson**. The Release implements the changes in the following tickets:

- ADZ-3199
- ADZ-2938

9.185.2 Prepared Release

Link to: DAQ 11.6 - PROD RELEASE.xlsx[2799]

9.185.3 Release Process

Documentation on <How to release a DAQ to Production[2800]>

9.185.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.185.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

2799 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2011.6%20-%20PROD%20RELEASE.xlsx?d=w9c7c7560260c4c0bab80e946adb3cc5f&csf=1&web=1&e=86DCII

2800 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.186 CHG0047662 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Ded95b4b98398759000adf388beaad3c9%26sysparm_view%3D

9.186.1 Tickets

CHG created by Unknown User (gipa6) The Release implements the changes in the following tickets:

- DOS-3090
- DOS-2321
- DOS-2322
- DOS-3122
- DOS-3161
- DOS-2899

9.186.2 Prepared Release

Release Tag: **v2023.07.26.13.31**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129227&view=results

9.186.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.186.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2801]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.186.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.06.27.15.29+126131
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=128412&view=results

2801 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Redeploy *last successful non-production release* (to all envs except Production)

9.187 CHG0047461 - NHS e-RS APIM Patient Care Release R1.15 Production Deployment

9.187.1 Content

High-level summary:

- Remove unused NFT deployment from APIM-PC azure pipelines

Tickets:

- NA

Further details on APIM-PC-1.15 Release Page[2802]

9.187.2 Prepared Release

Release Tag: 1.15.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129432&view=results

9.187.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.187.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2803]

2802 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/662605911/APIM-PC-1.15+Release+Page
2803 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.187.4 Roll-Back

- Redeploy:
    - Release Tag: 1.14.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125572&view=results

9.188 CHG0047463 - NHS e-RS APIM Referral Release R1.23 Deployment (Prod)

9.188.1 Content

High-level summary:

- Investigate replacement for test utils package

Tickets:

- NA

Further details on the APIM 1.23 Release Build Page (APIM-R)[2804]

9.188.2 Prepared Release

Release Tag: 1.23.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129430&view=results

9.188.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

2804 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/662605855/APIM-R+1.23+Release+Page

9.188.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2805]

9.188.4 Roll-Back

- Redeploy:
    - Release Tag: 1.22.0-beta

      Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125574&view=results

9.189 CHG0047777 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/change_request.do?sys_id=8acb708783dc39d000adf388beaad3da&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.189.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-2790
- DOS-3187
- DOS-3229

9.189.2 Prepared Release

Release Tag: **v2023.07.31.15.49+129698**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129698&view=results

9.189.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2805 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.189.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2806]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.189.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.07.26.13.31+129384
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129384&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.190 CHG0048068 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D6c6be40583a8391000adf388beaad399

9.190.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-3230

9.190.2 Prepared Release

Release Tag: **v2023.08.07.07.20+130273**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=130273&view=results

9.190.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
3. Run the maintenance on the RDS databases to update instances/IOPS
   a. if the UI doesn't allow, use CloudShell command: *aws rds modify-db-instance --db-instance-identifier [DB identifier]--apply-immediately*

2806 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.190.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2807]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.190.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.07.31.15.49+129698
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129698&view=results
- Redeploy *last successful non-production release* (to all envs except Production)
- Run the maintenance on the RDS databases to update

9.191 CHG0048062- Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D780aa4058368391000adf388beaad348%26sysparm_record_target%3Dtask%26sysparm_record_row%3D3%26sysparm_record_rows%3D3%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

9.191.1 Tickets

CHG created by **Andy Thompson**. The Release implements the changes in the following tickets:

- ADZ-3199

9.191.2 Prepared Release

Link to: DAQ 11.9 - PROD RELEASE[2808]

9.191.3 Release Process

Documentation on <How to release a DAQ to Production[2809]>

2807 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2808 https://nhs.sharepoint.com/:f:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development?csf=1&web=1&e=RwFJXF
2809 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.191.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.191.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.192 CHG0048116 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df5b5955583a8bd1000adf388beaad308%26sysparm_record_target%3Dtask%26sysparm_record_row%3D9%26sysparm_record_rows%3D9%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

APIGEE ONLY

9.192.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- Producer team changes to add target server - Elective waiting list
- Changes (only apigee prod changes will be released) - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.07.29.14.48...v2023.08.07.15.05

9.192.2 Prepared Release

Release Tag: **v2023.08.07.15.05**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=130313&view=results

9.192.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.192.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2810]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.192.4 Roll-Back

- Redeploy:
  - **v2023.07.29.14.48**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129456&view=results

9.193 CHG0048266- Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D780aa4058368391000adf388beaad348%26sysparm_record_target%3Dtask%26sysparm_record_row%3D2%26sysparm_record_rows%3D4%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYnumber

9.193.1 Tickets

CHG created by **Andy Thompson**. The Release implements the changes in the following tickets:

- ADZ-2938
- ADZ-3085

9.193.2 Prepared Release

Link to: DAQ 11.10 - PROD RELEASE[2811]

9.193.3 Release Process

Documentation on <How to release a DAQ to Production[2812]>

2810 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2811 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2011.10%20-%20PROD%20RELEASE.xlsx?d=w5f8632e8fc3843f59690cdbe579cece2&csf=1&web=1&e=4znNEg

2812 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.193.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.193.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.194 CHG0048359 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8bf75c9bc32c7d9042453fcb0501313e%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.194.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4452[2813] - Proxygen - Spec only APIs require different validation criteria DONE
- APM-4483[2814] - Implement MTLS for proxies DONE
- APM-4465[2815] - Add Proxygen function for publishing specs to UAT Bloomreach DONE

9.194.2 Prepared Release

Release Tag: **v2023.08.14.07.41**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=131851&view=results

9.194.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2813 https://nhsd-jira.digital.nhs.uk/browse/APM-4452?src=confmacro
2814 https://nhsd-jira.digital.nhs.uk/browse/APM-4483?src=confmacro
2815 https://nhsd-jira.digital.nhs.uk/browse/APM-4465?src=confmacro

9.194.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2816]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.194.4 Roll-Back

- Redeploy AWS:
  - Release Tag: **v2023.07.06.09.38**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=125976&view=results
- Redeploy Apigee:
  - Release Tag: **v2023.08.07.14.51**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=130313&view=results

9.195 CHG0048398 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=9b3c21d3c36c71d042453fcb050131a3&sysparm_stack=change_request_list.do?sysparm_query=active=true

AWS only

9.195.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4483[2817] - Implement MTLS for proxies DONE - only a small fix forward for a missing dependency once deployed - https://github.com/NHSDigital/api-management-infrastructure/pull/1133/files

9.195.2 Prepared Release

Release Tag: **v2023.08.14.14.31**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=131969&view=results

---

2816 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2817 https://nhsd-jira.digital.nhs.uk/browse/APM-4483?src=confmacro

9.195.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.195.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2818]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.195.4 Roll-Back

- Redeploy AWS:
  - Release Tag: **v2023.08.14.07.41**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=131851&view=results

9.196 CHG0048360- Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db9cb905fc3ec7d9042453fcb050131ce%26sysparm_record_target%3Dtask%26sysparm_record_row%3D3%26sysparm_record_rows%3D4%26sysparm_record_list%3Dassignment_group%253Djavascript%253AgetMyGroups%2528%2529%255Eactive%253Dtrue%255Esys_class_name%253Dchange_request%255EORsys_class_name%253Dchange_task%255EORDERBYassigned_to

9.196.1 Tickets

CHG created by **Ernie Kissiedu** . The Release implements the changes in the following tickets:

- ADZ-3195
- ADZ-3025
- ADZ-3139
- ADZ-3038
- ADZ-3170

2818 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.196.2 Prepared Release

Link to: DAQ 11.11 - PROD RELEASE[2819]

9.196.3 Release Process

Documentation on <How to release a DAQ to Production[2820]>

9.196.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.196.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.197 CHG0048455 - Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4abfaaafc3e479d042453fcb050131a0%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.197.1 Tickets

CHG created by **Sam Perera** . The Release implements the changes in the following tickets:

https://nhsd-jira.digital.nhs.uk/browse/ADZ-2869
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3248
https://nhsd-jira.digital.nhs.uk/browse/ADZ-2947
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3021
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3029
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3012
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3007

2819 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2011.11%20-%20PROD%20RELEASE.xlsx?d=w67b2a96af903408697ed40cfc0a8b503&csf=1&web=1&e=Hen3bT

2820 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.197.2 Prepared Release

Link to: DAQ 12.0 - PROD RELEASE.xlsx[2821]

9.197.3 Release Process

Documentation on <How to release a DAQ to Production[2822]>

9.197.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.197.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.198 CHG0048511 - Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D87c9dfbfc3a0fdd042453fcb05013149%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.198.1 Tickets

CHG created by **Sam Perera** . The Release implements the changes in the following tickets:
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3020
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3026
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3005
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3097
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3098
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3040
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3154
https://nhsd-jira.digital.nhs.uk/browse/ADZ-2999

2821 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2012.0%20-%20PROD%20RELEASE.xlsx?d=wd58869a8081f45e4ac2e73a2edb65cde&csf=1&web=1&e=boFoMS

2822 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.198.2 Prepared Release

Link to: https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2012.1%20-%20PROD%20RELEASE.xlsx?d=wa164415b1a404fac9bb1cb8d38cad72c&csf=1&web=1&e=dgzqR7

9.198.3 Release Process

Documentation on <How to release a DAQ to Production[2823]>

9.198.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.198.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.199 CHG0048714 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=315731b183b8bd1000adf388beaad36f&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.199.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4559[2824] - Support mtls keystores/refs per instance not env DONE
- APM-4451[2825] - Disallow temporary instances for nhsd-prod envs DONE
- Communication Manager Api MTLS set up

2823 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
2824 https://nhsd-jira.digital.nhs.uk/browse/APM-4559?src=confmacro
2825 https://nhsd-jira.digital.nhs.uk/browse/APM-4451?src=confmacro

GitHub comparison - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.08.14.14.31...v2023.08.21.15.08

9.199.2 Prepared Release

Release Tag: **v2023.08.21.15.08**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=133838&view=results

9.199.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.199.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2826]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.199.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.08.14.14.31**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=131969&view=results

9.200 CHG0048869 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dcdbda39a83f0f95000adf388beaad3ba

9.200.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-3220
- DOS-3088

2826 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.200.2 Prepared Release

Release Tag: **v2023.08.07.07.20+134767**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=134767&view=results

9.200.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.200.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2827]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.200.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.08.07.07.20+130353
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=130353&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.201 CHG0049012 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=f4bd6148c30d791042453fcb05013165&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.201.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4627[2828] - Deleting proxy with no apps subscribed errors DONE

2827 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2828 https://nhsd-jira.digital.nhs.uk/browse/APM-4627?src=confmacro

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.08.21.15.08...v2023.08.24.15.09

9.201.2 Prepared Release

Release Tag: **v2023.08.24.15.09**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=135782&view=results

9.201.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.201.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2829]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.201.4 Roll-Back

- Redeploy:
  - **v2023.08.21.15.08**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=133838&view=results

9.202 CHG0049112 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D82c3dcf08381395000adf388beaad36d

9.202.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-3276
- DOS-3255
- DOS-3203
- DOS-3249

2829 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.202.2 Prepared Release

Release Tag: **v2023.08.07.07.20+136602**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=136602&view=results

9.202.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.202.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2830]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.202.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.08.07.07.20+134767
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=134767&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.203 CHG0049183 - APIM EPS Updates

Link to ServiceNow Change request[2831]

9.203.1 Tickets

RFC created by John Kitson

See 05-09-2023 [PROD][2832] for list of changes

2830 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2831 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=06f295c1c345719042453fcb050131f1
2832 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/673997076/05-09-2023+PROD

9.203.2 Prepared Release

Release Tag v1.0.1406-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.203.3 Release Process

We would like to release this ourselves in service hours.

9.203.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.203.4 Roll-Back

Rollback link

9.204 CHG0049251 - Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd8e6608ac30d3d9042453fcb05013160%26sysparm_stack%3D%26sysparm_view%3D

9.204.1 Tickets

CHG created by **Sam Perera** . The Release implements the changes in the following tickets:
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3144
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3293
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3004
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3265
https://nhsd-jira.digital.nhs.uk/browse/ADZ-3264

9.204.2 Prepared Release

Link to: DAQ 13.0 - PROD Release[2833]

9.204.3 Release Process

Documentation on <How to release a DAQ to Production[2834]>

9.204.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.204.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.205 CHG0049159 - Booking and Referrals

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=d4a36c0d838dfd5000adf388beaad3f9

9.205.1 Tickets

CHG created by Ryan Thomas. The Release implements the changes in the following tickets:

- AMB-1672 https://nhsd-jira.digital.nhs.uk/browse/AMB-1672
- AMB-1564 https://nhsd-jira.digital.nhs.uk/browse/AMB-1564
- AMB-1537 https://nhsd-jira.digital.nhs.uk/browse/AMB-1537
- AMB-1536 https://nhsd-jira.digital.nhs.uk/browse/AMB-1536

2833 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2013.0%20-%20PROD%20RELEASE.xlsx?d=wf0ab4b41c70443ec8229f27e52d20e22&csf=1&web=1&e=ghbwgN

2834 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.205.2 Prepared Release

Release Tag: **v1.0.1023-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=135014&view=results

9.205.3 Release Process

We would like to release this ourselves in service hours.

9.205.3.1 Post Release Checks

- Check deployed version of API with _ping link: https://api.service.nhs.uk/booking-and-referral/FHIR/R4/_ping
- Check data in Grafana for prod on BaRS
- Confirm with suppliers using service that it is working as expected

9.205.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.777-alpha[2835]

9.205.5 CHG0051804 - Booking and Referrals

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=782c8d3a47463550b93f4579516d433f

9.205.5.1 Tickets

CHG created by Ryan Thomas. The Release implements the changes in the following tickets:

- AMB-1722 AMB-1722[2836] - BaRS service-callout url to fetch targets should be configurable DONE

- AMB-1529 AMB-1529[2837] - BaRS - Implement Product ID DONE

9.205.5.2 Prepared Release

Release Tag: **v1.0.1046-alpha**

2835 https://github.com/NHSDigital/booking-and-referral-fhir-api/releases/tag/v1.0.777-alpha
2836 https://nhsd-jira.digital.nhs.uk/browse/AMB-1722?src=confmacro
2837 https://nhsd-jira.digital.nhs.uk/browse/AMB-1529?src=confmacro

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=158699&view=results

9.205.5.3 Release Process

We would like to release this ourselves in service hours.

9.205.5.3.1 Post Release Checks

- Check deployed version of API with _ping link: https://api.service.nhs.uk/booking-and-referral/FHIR/R4/_ping
- Check data in Grafana for prod on BaRS
- Confirm with suppliers using service that it is working as expected

9.205.5.4 Roll-Back

- Redeploy:
    - Release Tag: **v1.0.1023-alpha**

9.206 CHG0049315 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=0e4929d2834135d000adf388beaad399&sysparm_stack=change_request_list.do?sysparm_query=active=true

9.206.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4656[2838] - Support release of handling company apps with proxygen DONE

GitHub changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.08.24.15.09...v2023.09.04.13.22

Only changes that will be released to prod are proxygen prod

9.206.2 Prepared Release

Release Tag: **v2023.09.04.13.22**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=138019&view=results

2838 https://nhsd-jira.digital.nhs.uk/browse/APM-4656?src=confmacro

9.206.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.206.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2839]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.206.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.08.24.15.09**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=135782&view=results

9.207 CHG0048743 - NHS e-RS APIM Patient Care Release R1.16 Production Deployment

9.207.1 Content

High-level summary:

- TRM version upgrades

Tickets:

- NA

Further details on APIM-PC-1.16 Release Page[2840]

9.207.2 Prepared Release

Release Tag: 1.16.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=136689&view=results

---

2839 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2840 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/662623796/APIM-PC-1.16+Release+Page

9.207.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.207.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2841]

9.207.4 Roll-Back

- Redeploy:
  - Release Tag: 1.15.0-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129432&view=results

9.208 CHG0048741 - NHS e-RS APIM Referral Release R1.24 Deployment (Prod)

9.208.1 Content

High-level summary:

- Three minor, cosmetic fixes in the OAS documentation

Tickets:

- NA

Further details on the APIM-R-1.24 Release Page[2842]

2841 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2842 https://nhsd-confluence.digital.nhs.uk/spaces/ERS/pages/662623785/APIM-R-1.24+Release+Page

9.208.2 Prepared Release

Release Tag: 1.24.0-beta

Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=136688&view=results

9.208.3 Release Process

1. **Please ensure all communication for this release go out on the platforms-apim-producer-support channel as well as the platforms-apim-consumer-support channel.** This is to ensure representatives from e-RS can monitor the outcome of the release.
2. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed (Is there a pre-requisite Infrastructure RFC?)
3. Click the Release Link above and approve the manual step to deploy the release to prod

   NB ONLY prod should be released in this deployment, all other deployments are managed seperately

9.208.3.1 Post Release Checks

1. Confirm the post-deployment automated smoke tests pass (via Azure DevOps link above - i.e. 100% pass)
2. Monitor Grafana Prod dashboard[2843]

9.208.4 Roll-Back

- Redeploy:
  - Release Tag: 1.23.0-beta

    Link to Azure DevOps: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=129430&view=results

9.209 CHG0049615- Identity Service Changes

9.209.1 Tickets

RFC was created by Jon Marston The Release implements the changes in the following tickets:

2843 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

APM-3686[2844] - Retire 1st gen mock auth - Implement the retirement - Technical Steps DONE

9.209.2 Prepared Release

Release Tag: v1.0.1734-alpha[2845]

Pipelines - Run v1.0.1734-alpha+139497 (azure.com)[2846]

9.209.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.209.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2847]
2. Verify the creation of a new entry on the dashboard for Identity Service _status
3. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.209.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1702-alpha+108287[2848]
    - Pipelines - Run v1.0.1702-alpha+108287 (azure.com)[2849]
- Redeploy *last successful non-production release* (to all envs except production)

9.210 CHG0049711 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5e50f2f08311315000adf388beaad336

9.210.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- DOS-3276

2844 https://nhsd-jira.digital.nhs.uk/browse/APM-3686?src=confmacro
2845 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1734-alpha
2846 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=139496&view=results
2847 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
2848 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108287
2849 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=108287&view=results

- DOS-2791

9.210.2 Prepared Release

Release Tag: **v2023.08.07.07.20+140586**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=140586&view=results

9.210.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.210.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2850]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.210.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.08.07.07.20+136602**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=136602&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.211 RFC CHG0049916 - APIM Infrastructure Updates

9.211.1 Tickets

CHG created by Jon Marston. The Release implements the changes in the following tickets:

- APM-3686[2851] - Retire 1st gen mock auth - Implement the retirement - Technical Steps DONE

GitHub changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.09.04.13.22...v2023.09.18.10.26

2850 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2851 https://nhsd-jira.digital.nhs.uk/browse/APM-3686?src=confmacro

Changes relating to
APM-4576[2852] - Create MTLS secured backend service for use in proxygen tests DONE apply to PTL infrastructure only.

9.211.2 Prepared Release

Release Tag: **v2023.09.18.10.26**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=141974&view=results

9.211.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.211.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2853]
2. Monitor APIM Dashboard ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.211.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.09.04.13.22**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=137728&view=results

9.212 CHG0050015 - Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd1f17e36c3d13dd042453fcb05013140

9.212.1 Tickets

CHG created by **Andy Thompson** . The Release implements the changes in the following tickets:
ADZ-3284

ADZ-3287

2852 https://nhsd-jira.digital.nhs.uk/browse/APM-4576?src=confmacro
2853 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

ADZ-3310

ADZ-3311

ADZ-3314

ADZ-3326

ADZ-3028

ADZ-3094

9.212.2 Prepared Release

Link to: DAQ 13.1 - PROD Release[2854]

9.212.3 Release Process

Documentation on <How to release a DAQ to Production[2855]>

9.212.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.212.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.213 CHG0050060 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D26b5a343839dbdd000adf388beaad3fa

9.213.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

2854 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2013.1%20-%20PROD%20RELEASE.xlsx?d=w22e6b469b8b040a98a4a2a6fe134518a&csf=1&web=1&e=w2Xsf6

2855 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

- DOS-3224
- DOS-2968
- DOS-3221

9.213.2 Prepared Release

Release Tag: **v2023.08.07.07.20+142768**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=142768&view=results

9.213.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.213.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2856]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.213.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.08.07.07.20+140586**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=140586&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.214 CHG0050127 - APIM - Add rate limiting custom attribute to app

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D21877c23c3d9f91442453fcb0501314a

9.214.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

2856 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- APM-4704[2857] - Configure ratelimiting for PDS app DONE

9.214.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow (see page 1045)

Url:

```
https://apigee-deep-rest.ptl.api.platform.nhs.uk:9004/v1/organizations/nhsd-prod/
apps/b49dbfbb-5fbf-4ed1-ae1a-42e13bc6c385/attributes/ratelimiting
```

Payload:

```
{
  "personal-demographics-prod": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 21000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": true
      "ratelimit": "21000pm"
    }
  }
}
```

9.214.3 Release Process

1. Rate limiting shared flow (see page 1045)

9.214.3.1 Post Release Checks

1. Ensure traffic still present (this comes in batches, so check for some time after and check with owners of app)

2857 https://nhsd-jira.digital.nhs.uk/browse/APM-4704?src=confmacro

9.214.4 Roll-Back

- Manually remove the 'ratelimiting' attribute from the prod app and click save.

9.215 CHG0050342 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=c290f6ddc3297d9442453fcb0501319b

9.215.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4710[2858] - Raise maxage of Strict-Transport-Security to 31536000 in PostProxySharedFlow DONE

  <--- the reason this isn't a standard change is that this is a change the the shared flows
- APM-4658[2859] - Upgrade HA Proxy for Egress Proxy DONE

GitHub changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.09.18.10.26...v2023.09.26.09.15

Ignore the scripts changes that wont release anything

9.215.2 Prepared Release

Release Tag: **v2023.09.26.09.15**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=144556&view=results

9.215.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.215.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2860]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

---

2858 https://nhsd-jira.digital.nhs.uk/browse/APM-4710?src=confmacro
2859 https://nhsd-jira.digital.nhs.uk/browse/APM-4658?src=confmacro
2860 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.215.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.09.18.10.26**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=142026&view=results

9.216 CHG0050379 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D46b0b39183ed7d9400adf388beaad308%26sysparm_stack%3D%26sysparm_view%3D

9.216.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-2919
- DOS-3177
- DOS-3104
- DOS-2530

9.216.2 Prepared Release

Release Tag: **v2023.09.21.11.20+144918**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=144918&view=results

9.216.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
   a. if it fails run the pipeline again using the same version tag

9.216.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2861]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2861 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.216.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.08.07.07.20+142768**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=142768&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.217 CHG0050564 - Digital Onboarding Content release

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D1637f843c3adb9106c39935c05013176%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.217.1 Tickets

CHG created by **Andy Thompson** . The Release implements the changes in the following tickets:

ADZ-3094

ADZ-2999

ADZ-3135

ADZ-3028

ADZ-3330

9.217.2 Prepared Release

Link to: DAQ 13.2 - PROD Release[2862]

9.217.3 Release Process

Documentation on <How to release a DAQ to Production[2863]>

9.217.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

---

2862 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2013.2%20-%20PROD%20RELEASE.xlsx?d=wee78be7459bb49b58d81fa974c1ada09&csf=1&web=1&e=rJ8fxf

2863 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.217.4 Roll-Back

Depending on nature of issue. If non manual DB required - Redeploy last successful production release
Link to https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

If manual DB roll back required - https://nhsd-confluence.digital.nhs.uk/display/DRM/KOP-003+Rollback+Release

9.218 CHG0050574 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=fff0898bc329f9106c39935c05013198

9.218.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4724[2864] - Release mTLS config for prescriptions-for-patients DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.09.26.09.15...v2023.09.27.14.02

9.218.2 Prepared Release

Release Tag: **v2023.09.27.14.02**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146017&view=results

9.218.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.218.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2865]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2864 https://nhsd-jira.digital.nhs.uk/browse/APM-4724?src=confmacro
2865 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.218.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.09.26.09.15**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=144556&view=results

9.219 CHG0050653 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db8720257c32df1506c39935c050131bc%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.219.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4430[2866] - Enforce limit on instances and mtls secrets per environment DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.09.27.14.02...v2023.10.02.14.11

9.219.2 Prepared Release

Release Tag: **v2023.10.02.14.11**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146400&view=results

9.219.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod **Terraform (AWS) Only**

9.219.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2867]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

---

2866 https://nhsd-jira.digital.nhs.uk/browse/APM-4430?src=confmacro
2867 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.219.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.09.27.14.02**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146017&view=results

9.220 CHG0050994 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=d1b33ae1c3bd3d506c39935c0501316e

9.220.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-3569[2868] - Secret dependency validation DONE
- APM-4009[2869] - Fix keycloak ResourceOp.update methods DONE
- APM-4664[2870] - Implement gate for deployment to prod DONE

Changes since apigee release tag (only includes ansible/roles/apigee-key-value-maps/vars/main/key_value_maps.yml) - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.09.27.14.02...v2023.10.10.10.32

AWS changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.10.02.14.11...v2023.10.10.10.32

9.220.2 Prepared Release

Release Tag: **v2023.10.10.10.32**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=148419&view=results

9.220.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

2868 https://nhsd-jira.digital.nhs.uk/browse/APM-3569?src=confmacro
2869 https://nhsd-jira.digital.nhs.uk/browse/APM-4009?src=confmacro
2870 https://nhsd-jira.digital.nhs.uk/browse/APM-4664?src=confmacro

9.220.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2871]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.220.4 Roll-Back

- Redeploy:
  - AWS
    - Release Tag: **v2023.10.02.14.11**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146400&view=results
  - Apigee
    - Release Tag: **v2023.09.27.14.02**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146017&view=results

9.221 CHG0051070 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3934df3947bd7190b93f4579516d433c%26sysparm_stack%3D%26sysparm_view%3D

9.221.1 Tickets

CHG created by Unknown User (adbe3) The Release implements the changes in the following tickets:

- DOS-2818
- DOS-3006
- DOS-1470
- DOS-3339
- DOS-3346
- DOS-2295

9.221.2 Prepared Release

Release Tag: **v2023.09.21.11.20+149015**

2871 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=149015&view=results

9.221.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.221.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2872]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.221.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.09.21.11.20+144918**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=144918&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.222 CHG0051201 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=bb0459aac3fd71d06c39935c050131a6

9.222.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-3569[2873] - Secret dependency validation DONE
- APM-4009[2874] - Fix keycloak ResourceOp.update methods DONE
- APM-4664[2875] - Implement gate for deployment to prod DONE

Changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.10.02.14.11...v2023.10.13.08.01

2872 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2873 https://nhsd-jira.digital.nhs.uk/browse/APM-3569?src=confmacro
2874 https://nhsd-jira.digital.nhs.uk/browse/APM-4009?src=confmacro
2875 https://nhsd-jira.digital.nhs.uk/browse/APM-4664?src=confmacro

9.222.2 Prepared Release

Release Tag: **v2023.10.13.08.01**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=149365&view=results

9.222.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.222.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2876]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.222.4 Roll-Back

- Redeploy:
  - AWS
    - Release Tag: **v2023.10.02.14.11**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146400&view=results
  - Apigee
    - Release Tag: **v2023.09.27.14.02**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=146017&view=results

9.223 CHG0051205 - APIM EPS FHIR Façade Updates

Link to ServiceNow Change request[2877]

2876 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2877 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=782b1daec335b1d06c39935c0501311e

9.223.1 Tickets

RFC created by John Kitson

See 26-10-2023 [PROD][2878] for list of changes

9.223.2 Prepared Release

Release Tag v1.0.1426-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.223.3 Release Process

We would like to release this ourselves in service hours.

9.223.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.223.4 Roll-Back

Rollback link

9.224 CHG0051432 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Ddbee3fc4c3ce71546c39935c0501314e%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

2878 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/704909252/26-10-2023+PROD

9.224.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-4646[2879] - Examples and schemas not rendering properly on spec DONE
- APM-3324[2880] - Restore apps script recreates client id and secret exactly DONE
- AMB-1746[2881] - Update UnderlyingConditions scopes DONE

Changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.10.13.08.01...v2023.10.18.11.10

9.224.2 Prepared Release

Release Tag: **v2023.10.13.08.01**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=150877&view=results

9.224.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.224.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2882]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.224.4 Roll-Back

- Redeploy:
    - AWS & Apigee
        - Release Tag: **v2023.10.13.08.01**
        - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=149365&view=results

2879 https://nhsd-jira.digital.nhs.uk/browse/APM-4646?src=confmacro
2880 https://nhsd-jira.digital.nhs.uk/browse/APM-3324?src=confmacro
2881 https://nhsd-jira.digital.nhs.uk/browse/AMB-1746?src=confmacro
2882 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.225 CHG0051704 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Deae1e7924786f910b93f4579516d4352[2883]

9.225.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

DOS-3308[2884] - Replace NHS Digital references in DOS pages & emails
DOS-3310[2885] - Allow account managers to change their DoS organisation name
DOS-3246[2886] - Page not found shown when accepting invitation
DOS-3047[2887] - Only sys admins can give sys admin role
ADZ-2732[2888] - Implement step-up MFA on necessary pages
DOS-3347[2889] - Prefix email subjects in non prod environments
pull/841[2890] Handle bad DAQ change log version strings

9.225.2 Prepared Release

Release Tag: v2023.09.21.11.20+152622

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=152622&view=results

9.225.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.225.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2891]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2883 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Deae1e7924786f910b93f4579516d4352%26sysparm_stack%3D%26sysparm_view%3D
2884 https://nhsd-jira.digital.nhs.uk/browse/DOS-3308
2885 https://nhsd-jira.digital.nhs.uk/browse/DOS-3310
2886 https://nhsd-jira.digital.nhs.uk/browse/DOS-3246
2887 https://nhsd-jira.digital.nhs.uk/browse/DOS-3047
2888 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2732
2889 https://nhsd-jira.digital.nhs.uk/browse/DOS-3347
2890 https://github.com/NHSDigital/drm-dap/pull/841
2891 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.225.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.09.21.11.20+149015
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=149015&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.226 CHG0051849 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D2b811ecfc34a3d506c39935c0501313d%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.226.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- EM-495

Changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.10.18.11.10..v2023.10.27.09.53

9.226.2 Prepared Release

Release Tag: v2023.10.27.09.53[2892]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=153246&view=results

9.226.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.226.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2893]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2892 https://github.com/NHSDigital/api-management-infrastructure/tree/v2023.10.27.09.53
2893 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.226.4 Roll-Back

- Redeploy:
    - AWS & Apigee
        - Release Tag: **v2023.10.18.11.10**
        - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=150877&view=results

9.227 CHG0051855 - APIM EPS FHIR Façade Updates

Link to ServiceNow Change request[2894]

9.227.1 Tickets

RFC created by John Kitson

See 02-11-2023 [PROD] v1.0.1443-beta[2895] for list of changes

9.227.2 Prepared Release

Release Tag v1.0.1443-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.227.3 Release Process

We would like to release this ourselves in service hours.

9.227.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:

2894 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=9f5f1a8b47063d50b93f4579516d4318
2895 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/710056835/02-11-2023+PROD+v1.0.1443-beta

a. APIM will need to access this for us - request in producer-support

9.227.4 Roll-Back

Rollback link

9.228 CHG0051994 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=10cd9b94c396bd1042453fcb050131d3

9.228.1 Tickets

CHG created by Lee Richardson The Release implements the changes in the following tickets:

ADZ-3167[2896] - API key created datetime does not use local time
ADZ-3208[2897] - Revoking own team ownership confirmation
ADZ-2804[2898] - Link directly from home page to My Applications
ADZ-3225[2899] - Company display name restrictions
ADZ-3210[2900] - Review new permissions model and ensure it meets requirement
ADZ-2794[2901] - Delete team
ADZ-2793[2902] - Edit team details
ADZ-2792[2903] - UI improvements to Team summary page
ADZ-2754[2904] - UI Improvements to My Apps and Teams
ADZ-2783[2905] - UI Improvements to manage team members
ADZ-2777[2906] - UI Improvements to Create a new team

9.228.2 Prepared Release

Release Tag: v2023.09.21.11.20+159851

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=159851&view=results

9.228.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod

---

2896 https://nhsd-jira.digital.nhs.uk/browse/ADZ-3167
2897 https://nhsd-jira.digital.nhs.uk/browse/ADZ-3208
2898 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2804
2899 https://nhsd-jira.digital.nhs.uk/browse/ADZ-3225
2900 https://nhsd-jira.digital.nhs.uk/browse/ADZ-3210
2901 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2794
2902 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2793
2903 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2792
2904 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2754
2905 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2783
2906 https://nhsd-jira.digital.nhs.uk/browse/ADZ-2777

a. if it fails run the pipeline again using the same version tag

9.228.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2907]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.228.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.09.21.11.20+152622
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=152622&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.229 CHG0052282 - Digital Onboarding Content release for DAQ 13.4 Production

Link to ServiceNow Change request[2908]

9.229.1 Tickets

CHG created by Ernest Kissiedu.
The Release implements the changes in the following tickets:

- ADZ-3373

2907 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2908 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De9117ee6835afdd000adf388beaad387%26sysparm_record_target%3Dtask%26sysparm_record_row%3D1%26sysparm_record_rows%3D1%26sysparm_record_list%3Dassigned_to%253D88db7f5d1b299550d1817662164bcb4d%255Esys_class_name%253Dsn_customerservice_case%255EORsys_class_name%253Dincident%255EORsys_class_name%253Dchange_request%255EORsys_class_name%253Dproblem%255EORsys_class_name%253Dchange_task%255EORsys_class_name%253Dincident_task%255EORsys_class_name%253Dproblem_task%255EORsys_class_name%253Dsc_task%255EORsys_class_name%253Dsn_customerservice_task%255EORsys_class_name%253Dsc_req_item%255EstateNOT%2BIN5%252C3%252C4%252C7%252C6%252C8%252C9%252C106%252C107%252C157%255Eassignment_groupDYNAMICd6435e965f510100a9ad2572f2b47744%255EORDERBYnumber

9.229.2 Prepared Release

Link to: DAQ 13.4 Production Release[2909]

9.229.3 Release Process

Documentation on <How to release a DAQ to Production[2910]>

9.229.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.229.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2911]>

9.230 CHG0052295 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D6a0f4ba68352711400adf388beaad3aa

9.230.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

- DOS-3372 API Setup roll forward fixes
- DOS-3362 Link to organisation page
- DOS-3368 Email Lambda certificate error
- DOS-3356 Correct link in approver email
- DOS-3311 Change organisation name (OBL)
- DOS-3312 Change organisation name (Admin)
- DOS-478 Lambda health checks
- DOS-3232 Lambda resourcing

2909 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2013.4%20-%20PROD%20RELEASE.xlsx?d=wc23feff3a9d342eaa47387ebad6dca07&csf=1&web=1&e=gLwBcl

2910 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

2911 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.230.2 Prepared Release

Release Tag: v2023.11.03.15.38+156148

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=156148&view=results

9.230.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.230.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2912]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.230.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.09.21.11.20+152857
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=152857&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.231 CHG0052407 - APIM EPS FHIR Façade Updates

Link to ServiceNow Change request[2913]

9.231.1 Tickets

RFC created by John Kitson

See PROD (see page 1873) for list of changes

9.231.2 Prepared Release

Release Tag v1.0.1454-beta

2912 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2913 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=c917bc038316b91400adf388beaad334

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.231.3 Release Process

We would like to release this ourselves in service hours.

9.231.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.231.4 Roll-Back

Rollback link

9.232 CHG0052605 - APIM PfP FHIR API Apigee Updates v1.0.148-beta

Link to ServiceNow Change request[2914]

9.232.1 Tickets

RFC created by John Kitson

See PROD (see page 1874) for list of changes

9.232.2 Prepared Release

Release Tag v1.0.1454-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

2914 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f90c5eecc36af11042453fcb050131a0

9.232.3 Release Process

We would like to release this ourselves in service hours.

9.232.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.232.4 Roll-Back

Rollback link

9.233 CHG0052689 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D781c5bb8c3627d1042453fcb050131d0%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.233.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- AMB-1773[2915] - Implement PUT /UnderlyingConditionList endpoint DONE
- APM-4697[2916] - Certificate expiry - KeyStore nhsapp-20221118-091351 (prod) DONE
- APM-4745[2917] - Certificate expiry - TrustStore identity-service-truststore (prod) DONE
- APM-4746[2918] - Certificate expiry - TrustStore identity-service-truststore (internal-dev) DONE
- APM-4747[2919] - Certificate expiry - TrustStore identity-service-truststore (internal-dev-sandbox) DONE

2915 https://nhsd-jira.digital.nhs.uk/browse/AMB-1773?src=confmacro
2916 https://nhsd-jira.digital.nhs.uk/browse/APM-4697?src=confmacro
2917 https://nhsd-jira.digital.nhs.uk/browse/APM-4745?src=confmacro
2918 https://nhsd-jira.digital.nhs.uk/browse/APM-4746?src=confmacro
2919 https://nhsd-jira.digital.nhs.uk/browse/APM-4747?src=confmacro

- APM-4748[2920] - Certificate expiry - TrustStore identity-service-truststore (internal-qa) DONE
- APM-4749[2921] - Certificate expiry - TrustStore identity-service-truststore (internal-qa-sandbox) DONE
- APM-4750[2922] - Certificate expiry - TrustStore identity-service-truststore (ref) DONE
- APM-4751[2923] - Certificate expiry - TrustStore identity-service-truststore (sandbox) DONE
- APM-4752[2924] - Certificate expiry - TrustStore identity-service-truststore (int) DONE
- APM-4753[2925] - Certificate expiry - TrustStore identity-service-truststore (dev) DONE

Changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.10.27.09.53...v2023.11.13.14.47

9.233.2 Prepared Release

Release Tag: **v2023.11.13.14.47**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=157962&view=results

9.233.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.233.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2926]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.233.4 Roll-Back

- Redeploy:
  - AWS & Apigee
    - Release Tag: **v2023.10.27.09.53**

2920 https://nhsd-jira.digital.nhs.uk/browse/APM-4748?src=confmacro
2921 https://nhsd-jira.digital.nhs.uk/browse/APM-4749?src=confmacro
2922 https://nhsd-jira.digital.nhs.uk/browse/APM-4750?src=confmacro
2923 https://nhsd-jira.digital.nhs.uk/browse/APM-4751?src=confmacro
2924 https://nhsd-jira.digital.nhs.uk/browse/APM-4752?src=confmacro
2925 https://nhsd-jira.digital.nhs.uk/browse/APM-4753?src=confmacro
2926 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=153246&view=results

9.234 CHG0052687 - Digital Onboarding Content release for DAQ 13.5 Production

Link to ServiceNow Change request[2927]

9.234.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- ADZ-3352
- ADZ-3368
- ADZ-3377

9.234.2 Prepared Release

Link to: DAQ 13.5 Production Release[2928]

DAQ version updated after problem with staging to DAQ 13.6 Production Release[2929]

9.234.3 Release Process

Documentation on <How to release a DAQ to Production[2930]>

9.234.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.234.4 Roll-Back

- Redeploy last successful production release

2927 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D92b9d334c3ee3d1042453fcb050131e8
2928 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2013.5%20-%20PROD%20RELEASE.xlsx?d=wb34f4a9a4e674572b1ad74f8f3d33e79&csf=1&web=1&e=1V1UBp
2929 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2013.6%20-%20PROD%20RELEASE.xlsx?d=wb7f03c3cc8c843a98c27ff1f1bcd372f&csf=1&web=1&e=FEjgnJ
2930 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

- Link to <How to roll-back a DAQ if a release is unsuccessful[2931]>

9.235 CHG0052866 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8dfb3df5c3eeb19042453fcb05013177

9.235.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4716[2932] - Certificate expiry - KeyStore intspineservices-20201125-101015 (internal-dev) DONE
- APM-4717[2933] - Certificate expiry - KeyStore veit07-20201125-101021 (internal-dev) DONE
- APM-4718[2934] - Certificate expiry - KeyStore intspineservices-20201125-101015 (internal-qa) DONE
- APM-4719[2935] - Certificate expiry - KeyStore veit07-20201125-101021 (internal-qa) DONE
- APM-4720[2936] - Certificate expiry - KeyStore intspineservices-20201125-101015 (ref) DONE
- APM-4721[2937] - Certificate expiry - KeyStore intspineservices-20201125-101015 (int) DONE
- APM-4756[2938] - Certificate expiry - KeyStore spineservices-20201215-094959 (prod) DONE
- APM-4776[2939] - Certificate expiry - KeyStore devspineservices-20201224-085211 (internal-dev) DONE
- APM-4777[2940] - Certificate expiry - KeyStore devspineservices-20201224-085211 (internal-qa) DONE
- DOS-3309[2941]
- APM-4798[2942] - Validate a path has security set when posting instances DONE

2931 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful
2932 https://nhsd-jira.digital.nhs.uk/browse/APM-4716?src=confmacro
2933 https://nhsd-jira.digital.nhs.uk/browse/APM-4717?src=confmacro
2934 https://nhsd-jira.digital.nhs.uk/browse/APM-4718?src=confmacro
2935 https://nhsd-jira.digital.nhs.uk/browse/APM-4719?src=confmacro
2936 https://nhsd-jira.digital.nhs.uk/browse/APM-4720?src=confmacro
2937 https://nhsd-jira.digital.nhs.uk/browse/APM-4721?src=confmacro
2938 https://nhsd-jira.digital.nhs.uk/browse/APM-4756?src=confmacro
2939 https://nhsd-jira.digital.nhs.uk/browse/APM-4776?src=confmacro
2940 https://nhsd-jira.digital.nhs.uk/browse/APM-4777?src=confmacro
2941 https://nhsd-jira.digital.nhs.uk/browse/AMB-1773
2942 https://nhsd-jira.digital.nhs.uk/browse/APM-4798?src=confmacro

- APM-4809[2943] - Deploying instances: expanding $ref within a the 'path' section of a spec fails
  DONE

Changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.11.13.14.47...v2023.11.17.10.23

9.235.2 Prepared Release

Release Tag: **v2023.11.17.10.23**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=159050&view=results

9.235.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.235.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2944]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.235.4 Roll-Back

- Redeploy:
  - AWS & Apigee
    - Release Tag: **v2023.11.13.14.47**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=157962&view=results

9.236 CHG0052918 - APIM EPS FHIR Façade Updates

Link to ServiceNow Change request[2945]

2943 https://nhsd-jira.digital.nhs.uk/browse/APM-4809?src=confmacro
2944 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2945 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=86e6adfac36271d042453fcb0501313f

9.236.1 Tickets

RFC created by John Kitson

See APIM EPS FHIR Façade Updates v1.0.1467-beta[2946] for list of changes

9.236.2 Prepared Release

Release Tag v1.0.1467-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.236.3 Release Process

We would like to release this ourselves in service hours.

9.236.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.236.4 Roll-Back

Rollback link

9.237 CHG0052921 - APIM PfP FHIR API Apigee Updates v1.0.150-beta

Link to ServiceNow Change request[2947]

2946 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/732529029/21-11-2023+PROD+v1.0.1467-beta
2947 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=2c38e97ec3a271d042453fcb05013152

9.237.1 Tickets

RFC created by John Kitson

See 21-11-2023 [PROD] v1.0.150-beta[2948] for list of changes

9.237.2 Prepared Release

Release Tag v1.0.150-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.237.3 Release Process

We would like to release this ourselves in service hours.

9.237.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

9.237.4 Roll-Back

Rollback link

9.238 CHG0053201 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De6924608837af51400adf388beaad3d6

9.238.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

2948 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/732529066/21-11-2023+PROD+v1.0.150-beta

- DOS-2417 Add logging for product section unauthorised postback
- DOS-3287 Delete a product
- DOS-3288 Delete an organisation
- DOS-3302 API lambda roll forward
- DOS-3373 Email cannot contain leading or trailing whitespace
- DOS-3375 Health check lambda cannot access DOS url
- DOS-3377 Change organisation redirect fix
- DOS-3386 Remove existing delete product for test mode
- DOS-3389 API setup test mode fix
- ADZ-3308 Restrict production apps to team owners
- ADZ-3217 Application name space after name issue
- ADZ-3215 Validation consistency issues with editing team name and description
- ADZ-2779 Prevent viewer-level users from creating applications for their team

9.238.2 Prepared Release

Release Tag: v2023.11.27.09.19+162194

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162194&view=results

9.238.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.238.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2949]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.238.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.16.15.32+160663
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=160663&view=results
- Redeploy *last successful non-production release* (to all envs except Productio

2949 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.239 CHG0053252 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D54440fccc3fa395442453fcb05013144

9.239.1 Tickets

CHG created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4788[2950] - Removing an instance doesn't untag a secret as in-use DONE
- APM-4791[2951] - Fix secret tagging DONE
- APM-3920[2952] - Refresh API instance Apigee components DONE
- APM-4798[2953] - Validate a path has security set when posting instances DONE

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.11.17.10.23...v2023.11.24.15.20

9.239.2 Prepared Release

Release Tag: **v2023.11.24.15.20**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=161294&view=results

9.239.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.239.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2954]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

---

2950 https://nhsd-jira.digital.nhs.uk/browse/APM-4788?src=confmacro
2951 https://nhsd-jira.digital.nhs.uk/browse/APM-4791?src=confmacro
2952 https://nhsd-jira.digital.nhs.uk/browse/APM-3920?src=confmacro
2953 https://nhsd-jira.digital.nhs.uk/browse/APM-4798?src=confmacro
2954 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.239.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.11.17.10.23**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=159050&view=results

9.240 CHG0053287 - APIM EPS FHIR Façade Updates

Link to ServiceNow Change request[2955]

9.240.1 Tickets

RFC created by John Kitson

FHIR-v1.0.1475-beta [2956]See for list of changes

9.240.2 Prepared Release

Release Tag v1.0.1475-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.240.3 Release Process

We would like to release this ourselves in service hours.

9.240.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:

2955 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f71b01c1c3ba719442453fcb05013129

2956 https://nhsd-confluence.digital.nhs.uk/spaces/APIMC/pages/734563830/28-11-2023+PROD+v1.0.1475-beta

a. APIM will need to access this for us - request in producer-support

9.240.4 Roll-Back

Rollback link

9.241 CHG0053288 - APIM PfP FHIR API Apigee Updates v1.0.152-beta

Link to ServiceNow Change request[2957]

9.241.1 Tickets

RFC created by John Kitson

See PROD (see page 1885) for list of changes

9.241.2 Prepared Release

Release Tag v1.0.152-beta

Link to Azure pipeline build: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=121745&view=results

9.241.3 Release Process

We would like to release this ourselves in service hours.

9.241.3.1 Post Release Checks

1. Check deployed version of API with _ping link: https://api.service.nhs.uk/electronic-prescriptions/_ping
2. Check data in Grafana for prod on EPS - 15 min delay: https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=All&var-proxy=electronic-prescription-service-api-prod&var-product=All&var-revision=All&var-deployed_revision=All
3. Check with IC24 for any problems
4. If there are errors, check logs in Splunk:
    a. APIM will need to access this for us - request in producer-support

2957 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=99acc58dc3fa719442453fcb05013123

9.241.4 Roll-Back

Rollback link

9.242 CHG0053370 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D54440fccc3fa395442453fcb05013144

9.242.1 Tickets

CHG created by Alexander Carrie. The Release is a re-release of the last released tag to refresh a secret.

9.242.2 Prepared Release

Release Tag: **v2023.11.24.15.20**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=161294&view=results

9.242.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.242.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2958]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.242.4 Roll-Back

- Replace secret with previous version and re-release

2958 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.243 CHG0053374 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=6b696e99c3fef99442453fcb0501310b

9.243.1 Tickets

CHG created by Sophie Clayton . The Release implements the changes in the following tickets:

- ADZ-3396

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.11.24.15.20...v2023.11.28.08.28

9.243.2 Prepared Release

Release Tag: **v2023.11.24.15.20**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162291&view=results

9.243.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.243.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2959]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.243.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.11.24.15.20**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=161294&view=results

2959 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.244 CHG0053407 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da4781b1183babd5400adf388beaad336%26sysparm_stack%3D%26sysparm_view%3D

9.244.1 Tickets

CHG created by Unknown User (adbe3). The release is for **UAT only** and implements the changes from CHG0053201 - Digital Onboarding Service (see page 1881)

9.244.2 Prepared Release

Release Tag: v2023.11.27.09.19+162343

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162343&view=results

9.244.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.244.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2960]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.244.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.27.09.19+162194
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162194&view=results
- Redeploy *last successful non-production release* (to all envs except Productio

2960 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.245 CHG0053638 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df89a53dec3fe3d1842453fcb05013198

9.245.1 Tickets

CHG created by Sophie Clayton . The Release implements the changes in the following tickets:

- ADZ-3423 Lowercase emails when adding new member to team
- DOS-2543 Improve contrast for "test mode enabled" banner
- DOS-3349 Date input is not accessible
- DOS-3250 I want to save my information as I progress
- DOS-3400 ApiSetup Add API does not respect test mode
- DOS-3402 Add Api to product in Test mode failing

9.245.2 Prepared Release

Release Tag: v2023.11.27.09.19+164071

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=164071&view=results

9.245.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.245.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2961]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.245.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.27.09.19+162194
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162194&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2961 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.246 CHG0053705 - Identity Service Updates

Link to Service Now Change[2962]

9.246.1 Tickets

RFC was created by Alexander Carrie. The Release implements the changes in the following tickets:

- Code changes for the following tickets
  - APM-4847[2963] - Wrong error given when subject token invalid - reports malformed jwt in client assertion DONE

GitHub changes - https://github.com/NHSDigital/identity-service-api/compare/v1.0.1734-alpha...v1.0.1737-alpha

9.246.2 Prepared Release

Release Tag: **v1.0.1737-alpha**

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=163112&view=results

9.246.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.246.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2964]
2. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.246.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1734-alpha (build then release)
- Redeploy *last successful non-production release* (to all envs except production):

2962 https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=e0807e5fc3327d5842453fcb050131fe
2963 https://nhsd-jira.digital.nhs.uk/browse/APM-4847?src=confmacro
2964 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.247 CHG0053719 - APIM Infrastructure Updates

https://www.support.internalservices.england.nhs.uk/change_request.do?sys_id=3793e2a383baf91800adf388beaad372

9.247.1 Tickets

CHG0053719 created by Jon Marston . The Release implements the changes in the following tickets:

- APM-4703[2965] - Allow access to PTL Keycloak admin URLs from local Proxygen instances DONE
- APM-4845[2966] - JIRA tickets not created by bot for expiring certs held in AWS DONE
- APM-4856[2967] - Properly handle instances with legacy Apigee portal API features DONE

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.11.28.08.28...v2023.12.05.09.43

9.247.2 Prepared Release

Release Tag: **v2023.12.05.09.43**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=164387&view=results

9.247.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.247.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2968]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

2965 https://nhsd-jira.digital.nhs.uk/browse/APM-4703?src=confmacro
2966 https://nhsd-jira.digital.nhs.uk/browse/APM-4845?src=confmacro
2967 https://nhsd-jira.digital.nhs.uk/browse/APM-4856?src=confmacro
2968 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.247.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.11.28.08.28**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=162291&view=results

9.248 CHG0054065 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df319e631c3cbb11042453fcb0501313d%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.248.1 Tickets

CHG0054065 created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-3923[2969] - Refresh API pre-requisite AWS components DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.12.05.09.43...v2023.12.12.10.00

9.248.2 Prepared Release

Release Tag: **v2023.12.12.10.00**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=167585&view=results

9.248.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.248.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2970]

2969 https://nhsd-jira.digital.nhs.uk/browse/APM-3923?src=confmacro
2970 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.248.4 Roll-Back

- Redeploy:
    - Release Tag: **v2023.12.05.09.43**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=164387&view=results

9.249 CHG0054160 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D27200c1e838f3d1000adf388beaad374

9.249.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

- DOS-3376 Button for developers to reopen API post approval removed
- DOS-3350 Order of validation summary errors inconsistent with form
- DOS-2759 Add phone number validation
- DOS-3401 Improve DOS Splunk logging

9.249.2 Prepared Release

Release Tag: v2023.11.27.09.19+168309

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=168309&view=results

9.249.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.249.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2971]

2971 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.249.4 Roll-Back

- Redeploy:
  - Release Tag: v2023.11.27.09.19+164071
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=164071&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.250 CHG0054271 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Daf3b796ac3c3715042453fcb050131f6%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.250.1 Tickets

CHG0054065 created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4769[2972] - ECR infrastructure is not deployed by default when creating or updating an API DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.12.12.10.00...v2023.12.13.14.14

9.250.2 Prepared Release

Release Tag: **v2023.12.13.14.14**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=168389&view=results

9.250.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

2972 https://nhsd-jira.digital.nhs.uk/browse/APM-4769?src=confmacro

9.250.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2973]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.250.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.12.12.10.00**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=167585&view=results

9.251 CHG0054438 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D734953c4c3df31d042453fcb05013142

9.251.1 Tickets

CHG0054438 created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4773[2974] - Docker credentials endpoint returns HTTP 500 for APIs with no Docker repo configured DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.12.13.14.14...v2023.12.19.13.31

9.251.2 Prepared Release

Release Tag: **v2023.12.19.13.31**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=169963&view=results

9.251.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl

---

2973 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2974 https://nhsd-jira.digital.nhs.uk/browse/APM-4773?src=confmacro

4. Click the Release Link above and approve the manual step to deploy the release to prod

9.251.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2975]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.251.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.12.13.14.14**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=168389&view=results

9.252 CHG0054514 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8c533020c35f351042453fcb05013181

9.252.1 Tickets

CHG0054514 created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4869[2976] - Confluence access workflow broken DONE
- APM-4860[2977] - Allow paths prepended with $ DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.12.19.13.31...v2023.12.20.09.28

9.252.2 Prepared Release

Release Tag: **v2023.12.20.09.28**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=170309&view=results

9.252.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

2975 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2976 https://nhsd-jira.digital.nhs.uk/browse/APM-4869?src=confmacro
2977 https://nhsd-jira.digital.nhs.uk/browse/APM-4860?src=confmacro

2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.252.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2978]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.252.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.12.19.13.31**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=169963&view=results

9.253 CHG0054686 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3f751d5dc36fb11042453fcb05013167

9.253.1 Tickets

Change created by Harry Coulter . The Release implements the changes in the following tickets:

- APM-4896[2979] - Update Accurx - PDS - Production rate limiting DONE [2980]

Github comparison of changes: n/a

9.253.2 Prepared Release

9.253.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

2978 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2979 https://nhsd-jira.digital.nhs.uk/browse/APM-4896?src=confmacro
2980 https://nhsd-jira.digital.nhs.uk/browse/APM-3921

2. Retrieve custom rate limiting settings from
   APM-4896[2981] - Update Accurx - PDS - Production rate limiting DONE
3. Follow instructions for adding custom rate limiting, **ensuring to make a copy of any rate limiting settings already in place**, as described on Confluence - Rate limiting shared flow (see page 1045)

9.253.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2982]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Validate that consumer app can still access Personal Demographics Service API

9.253.4 Roll-Back

- If custom rate limiting was not previously set, remove custom rate limiting following instructions on Confluence - Rate limiting shared flow (see page 1049)
- If custom rate limiting was previously set, repeat release process, setting the custom rate limiting to what it was prior.

9.254 CHG0054703 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=fe21f55dc3eff11042453fcb0501317d

9.254.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-3921[2983] - Refresh API instance AWS components DONE
- APM-3922[2984] - Refresh API pre-requisite KeyCloak components DONE
- APM-4586[2985] - Improve secrets endpoint - make it more RESTful DONE
- PI-109

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2023.12.20.09.28...v2024.01.04.14.05

2981 https://nhsd-jira.digital.nhs.uk/browse/APM-4896?src=confmacro
2982 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2983 https://nhsd-jira.digital.nhs.uk/browse/APM-3921?src=confmacro
2984 https://nhsd-jira.digital.nhs.uk/browse/APM-3922?src=confmacro
2985 https://nhsd-jira.digital.nhs.uk/browse/APM-4586?src=confmacro

9.254.2 Prepared Release

Release Tag: **v2024.01.04.14.05**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=173779&view=results

9.254.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.254.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2986]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.254.4 Roll-Back

- Redeploy:
  - Release Tag: **v2023.12.20.09.28**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=170309&view=results

9.255 CHG0054951 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=2ba2f3b283ab315000adf388beaad32a

9.255.1 Tickets

CHG0054951 created by Fayez Ahmed . The Release implements the changes in the following tickets:

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.01.04.14.05...v2024.01.09.14.47

2986 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.255.2 Prepared Release

Release Tag: **v2024.01.09.14.47**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=175251&view=results

9.255.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.255.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2987]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.255.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.01.04.14.05**[2988]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=173779&view=results

9.256 CHG0055041 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D2e943c178367795000adf388beaad369

9.256.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

- ADZ-3392 API product visibility
- DOS-3406 Product statuses display names
- DOS-3424 Awaiting review tags disappear bug

2987 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
2988 https://github.com/NHSDigital/api-management-infrastructure/tree/v2024.01.04.14.05

- DOS-3426 Save bar overflow bug
- DOS-3413 Validation priority ordering
- DOS-3407 Add API to approved product (feature flagged)
- DOS-3206 Bulk add assessors
- DOS-3414 Error messages accessibility fixes
- ADZ-3219 Team description overflow bug
- ADZ-2168 .Net Analysers

9.256.2 Prepared Release

Release Tag: v2023.11.27.09.19+175609

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=175857&view=results

9.256.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.256.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2989]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.256.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.27.09.19+168309
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=168309&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.257 CHG0055343 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=575b246dc3bb7d5442453fcb050131a7

2989 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.257.1 Tickets

CHG0054951 created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4894[2990] - Improve efficiency of proxygen tests DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.01.09.14.47...v2024.01.15.16.29

9.257.2 Prepared Release

Release Tag: **v2024.01.15.16.29**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=177225&view=results

9.257.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.257.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2991]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.257.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.01.09.14.47**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=175251&view=results

2990 https://nhsd-jira.digital.nhs.uk/browse/APM-4894?src=confmacro
2991 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.258 CHG0055353 - Digital Onboarding Content release for DAQ 14.0 Production

Link to ServiceNow Change request[2992]

9.258.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- ADZ-3035
- ADZ-3347
- ADZ-3131

9.258.2 Prepared Release

Link to: DAQ 14.0 Production Release[2993]

updated from 13.7 to 14.0 to align with naming protocols with new APIs/Services being added to DOS

9.258.3 Release Process

Documentation on <How to release a DAQ to Production[2994]>

9.258.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.258.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[2995]>

2992 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd9b74925c377fd5442453fcb050131a8

2993 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20development/DAQ%2014.0%20-%20PROD%20RELEASE.xlsx?d=w5b99d460828c4b35a844b0cfa6891e6a&csf=1&web=1&e=u09LVf

2994 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

2995 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.259 CHG0055559 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4afd56538373399400adf388beaad31c

9.259.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

- DOS-3387 Help users find and navigate unanswered questions
- DOS-3440 API definitions do not exclude new DAQ sections
- DOS-3441 Assessment sections and answers lost
- DOS-3394 Change organisation return url causes problems
- DOS-3403 Add a reason for adding a new API to a completed product (feature flagged)
- DOS-3420 Switch between developer and account manager

9.259.2 Prepared Release

Release Tag: v2023.11.27.09.19+182515

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=182515&view=results

9.259.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.259.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2996]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.259.4 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.27.09.19+175609
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=175857&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

2996 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.260 CHG0055831- APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=a4a8175cc3cc829842453fcb050131f6

9.260.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4962[2997] - Update CA certs for ptl and prod DONE

Github comparison of changes: https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.01.15.16.29...v2024.01.26.13.46

9.260.2 Prepared Release

Release Tag: **2024.01.26.13.46**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=180801&view=results

9.260.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.260.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2998]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.260.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.01.15.16.29**

2997 https://nhsd-jira.digital.nhs.uk/browse/APM-4962?src=confmacro
2998 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=177225&view=results

9.261 CHG0056020 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D1e45090a8300429800adf388beaad3d6

9.261.1 Tickets

CHG created by Unknown User (adbe3). The Release implements the changes in the following tickets:

- DOS-3442 Last active account manager can be switched to a developer
- DOS-3421 Switch between Assessor and Approver
- DOS-3443 Answers lost after roll forward fix
- DOS-3439 Edit user, change assessment tracker view 404
- DOS-3422 Switch between OBL and Sys Admin
- DOS-3447 Apigee API filtering doesn't work with other custom attributes
- DOS-3404 Update an API section on a Completed product (feature flagged)

9.261.2 Prepared Release

Release Tag: v2023.11.27.09.19+182298

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=182298&view=results

9.261.3 Release Process

1. Click the Release Link above and approve the manual step to deploy the release to prod
    a. if it fails run the pipeline again using the same version tag

9.261.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[2999]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.261.4 Roll-Back

- Redeploy:

2999 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

    - Release Tag: v2023.11.27.09.19+179573
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=179573&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.262 CHG0056746 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da6d0dc5ac3d0029042453fcb050131b7%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.262.1 Tickets

DOS-3435 session fixes[3000]
DOS-3454 Ensure question statuses are set correctly when rolling forward[3001]
DOS-3409 Only use lambda roll forward when test mode disabled[3002]
DOS-3412 nhsuk hint text fix [3003]

9.262.2 Prepared Release

Release Tag: v2024.02.05.08.14+186001

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=186001&view=results

9.262.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.262.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3004]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3000 https://nhsd-jira.digital.nhs.uk/browse/DOS-3435
3001 https://nhsd-jira.digital.nhs.uk/browse/DOS-3454
3002 https://nhsd-jira.digital.nhs.uk/browse/DOS-3409
3003 https://nhsd-jira.digital.nhs.uk/browse/DOS-3412
3004 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.262.4 Roll-Back

- ### 9.262.5 Release Process

    a. Click the Release Link above and approve the manual step to deploy the release to prod
        i. if it fails run the pipeline again using the same version tag

9.262.5.1 Post Release Checks

a. Monitor Grafana Prod dashboard[3005]
b. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.262.6 Roll-Back

- Redeploy:
    - Release Tag: v2023.11.27.09.19+182298
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=182298&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.263 CHG0056988- APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=2848b1c4c324061442453fcb050131eb

9.263.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-5102[3006] - Investigate egress haproxy capacity doubling warnings DONE
- APM-4459[3007] - Mark PR instances as temporary for Hello World DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.01.26.13.46...v2024.02.19.10.21

3005 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3006 https://nhsd-jira.digital.nhs.uk/browse/APM-5102?src=confmacro
3007 https://nhsd-jira.digital.nhs.uk/browse/APM-4459?src=confmacro

9.263.2 Prepared Release

Release Tag: **v2024.02.19.10.21**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=187443&view=results

9.263.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.263.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3008]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.263.4 Roll-Back

- The current tag is in a broken state. We would endevour to fix forward
- For reference though -
  Release Tag: **2024.01.26.13.46**
  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=180801&view=results

9.264 CHG0057120 - Digital Onboarding Content release for DAQ 14.1 Production

Link to ServiceNow Change request[3009]

9.264.1 Tickets

CHG created by Ernest Kissiedu
The release implements the changes in the following tickets:

3008 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3009 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3eb5ac20c328825442453fcb05013104%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

- ~~ADZ-3446~~[3010]
- ADZ-3454

9.264.2 Prepared Release

Link to: DAQ 14.1 Production Release[3011]

9.264.3 Release Process

Documentation on <How to release a DAQ to Production[3012]>

9.264.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.264.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3013]>

9.265 CHG0057534 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=6dee154783e4ce5400adf388beaad3e2

9.265.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- Add new version of UserRoleService shared flow. This is a new verison, not an update to the old one so standard change is suitable even though we're in the world of shared flows - AEA-3708

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.02.26.11.30...v2024.02.27.15.28

3010 https://nhsd-jira.digital.nhs.uk/browse/ADZ-3035

3011 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2014.1%20-%20PROD%20RELEASE.xlsx?d=w0bca7f40e0da4f18bdd1eda321a950e4&csf=1&web=1&e=MPu05V

3012 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3013 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.265.2 Prepared Release

Release Tag: **v2024.02.27.15.28**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=190268&view=results

9.265.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod
5. Check shared flow is deployed to prod in Apigee

9.265.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3014]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.265.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.02.26.11.30**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=189962&view=results

9.266 CHG0057559 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=212fd11b83ecc29400adf388beaad335

9.266.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- AEA-3787

3014 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.02.27.15.28...v2024.02.29.09.12

9.266.2 Prepared Release

Release Tag: **v2024.02.27.15.28**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=190268&view=results

9.266.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.266.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3015]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.266.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.02.27.15.28**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=190268&view=results

9.267 CHG0057636 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=2f179f2b832c82d400adf388beaad3c5

9.267.1 Tickets

Change created by Sophie Clayton . The Release implements the changes in the following tickets:

- AEA-3787 Add target identity options to Proxygen

3015 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- NPA-1868 Add target servers for validated-relationships-service-api

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.02.27.15.28...v2024.03.01.11.43

9.267.2 Prepared Release

Release Tag: **v2024.03.01.11.43**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=191051&view=results

9.267.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod
5. Check shared flow is deployed to prod in Apigee

9.267.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3016]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.267.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.02.27.15.28**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=190268&view=results

9.268 CHG0057776 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=c887540d83f8ca1800adf388beaad3d4

3016 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.268.1 Tickets

Change created by Sophie Clayton . The Release implements the changes in the following tickets:

- Add address quality service api secrets
- PI-243 Add SDS FHIR to Apigee egress
- APM-4925[3017] Update to internal-dev.api.service.nhs.uk[3018] cert

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.01.11.43...v2024.03.05.14.53

9.268.2 Prepared Release

Release Tag: **v2024.03.05.14.53**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192042&view=results

9.268.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod
5. Check shared flow is deployed to prod in Apigee

9.268.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3019]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.268.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.03.01.11.43**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=191051&view=results

3017 https://nhsd-jira.digital.nhs.uk/browse/APM-4925
3018 http://internal-dev.api.service.nhs.uk
3019 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.269 CHG0057788 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dbe4bff0dc3b08a1042453fcb05013123%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.269.1 Tickets

https://nhsd-jira.digital.nhs.uk/browse/DOS-3355 - Securely handle special characters
https://nhsd-jira.digital.nhs.uk/browse/DOS-3457 - Update confirmation message to tell users that service owners should be informed
https://nhsd-jira.digital.nhs.uk/browse/DOS-2761 - Prevent the Add/Remove API process from being 'cancelled' by the user
https://nhsd-jira.digital.nhs.uk/browse/DOS-3459 - Add Apigee to DOS health check

9.269.2 Prepared Release

Release Tag: v2024.02.05.08.14+192248

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192248&view=results

9.269.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.269.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3020]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.269.4 Roll-Back

- Redeploy:
  -

    Release Tag: v2024.02.05.08.14+186001

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=186001&view=results

3020 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.270 CHG0057834 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b0e9451d8334c25800adf388beaad367

9.270.1 Tickets

Change created by Sophie Clayton . The Release implements the changes in the following tickets:

- Update address quality service api secret

This is a re-release of the previously released tag

9.270.2 Prepared Release

Release Tag: **v2024.03.05.14.53**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192338&view=results

9.270.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod
5. Check shared flow is deployed to prod in Apigee

9.270.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3021]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.270.4 Roll-Back

- Redeploy:

3021 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Release Tag: **v2024.03.05.14.53**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192042&view=results

9.271 CHG0057875 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=866b01e1c338825042453fcb050131f4

9.271.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4908[3022] - Secret tags hitting tag limits DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.05.14.53...v2024.03.06.17.06

9.271.2 Prepared Release

Release Tag: **v2024.03.06.17.06**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192392&view=results

9.271.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.271.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3023]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3022 https://nhsd-jira.digital.nhs.uk/browse/APM-4908?src=confmacro
3023 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.271.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.03.05.14.53**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192042&view=results

9.272 CHG0057999 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=8c192f35c3bc8e5042453fcb050131bf

9.272.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-5190[3024] - Release address-quality-service target servers DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.06.17.06...v2024.03.08.13.51

9.272.2 Prepared Release

Release Tag: **v2024.03.08.13.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=193020&view=results

9.272.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.272.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3025]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

---

3024 https://nhsd-jira.digital.nhs.uk/browse/APM-5190?src=confmacro
3025 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.272.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.03.06.17.06**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192392&view=results

9.273 CHG0058074 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=49d913728330825000adf388beaad3b5

9.273.1 Tickets

Change created by Sophie Clayton . The Release implements the changes in the following tickets:

- APM-4952[3026] - Certificate expiry - AWS ptl and prod cert {account}/truststore/github.com/crt DONE

Re-releasing same tag

9.273.2 Prepared Release

Release Tag: **v2024.03.08.13.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=193652&view=results

9.273.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.273.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3027]

3026 https://nhsd-jira.digital.nhs.uk/browse/APM-4952?src=confmacro
3027 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.273.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.03.08.13.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=193020&view=results

9.274 CHG0058130 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=82bee49783b88a5000adf388beaad3f0

9.274.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4832[3028] - Fix proxygen API status codes DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.08.13.51...v2024.03.12.15.46

9.274.2 Prepared Release

Release Tag: **v2024.03.12.15.46**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194018&view=results

9.274.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.274.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3029]

3028 https://nhsd-jira.digital.nhs.uk/browse/APM-4832?src=confmacro

3029 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.274.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.03.08.13.51**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=193652&view=results

9.275 CHG0058183 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D04bad1a3c3788ed042453fcb0501311c%26sysparm_stack%3Dchange_request_list.do%253Fsysparm_query%253Dactive%253Dtrue

9.275.1 Tickets

Change created by Vikas Nautiyal . The Release implements the changes in the following tickets:

- APM-4993[3030] Certificate expiry - AWS prod/ptl cert {account}/truststore/am.nhsref-1.auth-ptl.cis2.spineservices.nhs.uk/crt[3031]

9.275.2 Prepared Release

Release Tag: **v2024.03.12.15.46**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194382&view=results

9.275.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

3030 https://nhsd-jira.digital.nhs.uk/browse/APM-4993
3031 http://am.nhsref-1.auth-ptl.cis2.spineservices.nhs.uk/crt

9.275.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3032]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.275.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.03.12.15.46**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194018&view=results

9.276 CHG0058247 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=b29c4a33fb3c0610cfc6feceaeefdc40

9.276.1 Tickets

Change created by Aleksandra Bak The Release implements the changes in the following tickets:

- [APM-4799] Hello world fix open access security option - CDT JIRA (digital.nhs.uk)[3033]

Comparing v2024.03.12.15.46...v2024.03.14.11.49 · NHSDigital/api-management-infrastructure (github.com)[3034]

9.276.2 Prepared Release

Release Tag: **v2024.03.14.11.49**

Pipelines - Run v2024.03.14.11.49+194712 (azure.com)[3035]

9.276.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

---

3032 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3033 https://nhsd-jira.digital.nhs.uk/browse/APM-4799
3034 https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.12.15.46...v2024.03.14.11.49
3035 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194712&view=results

9.276.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3036]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.276.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.03.12.15.46**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194018&view=results

9.277 CHG0058407 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=da7cd349c3490e106c39935c05013143

9.277.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4852[3037] - SPIKE - Cors errors for 'try this api' DONE
- Will also release several truststores:
  - APM-4987[3038] - Certificate expiry - AWS prod/ptl cert {account}/truststore/am.nhsint.auth-ptl.cis2.spineservices.nhs.uk/crt DONE
  - APM-4989[3039] - Certificate expiry - AWS prod/ptl cert {account}/truststore/am.nhsdep.auth-ptl.cis2.spineservices.nhs.uk/crt DONE
  - APM-4991[3040] - Certificate expiry - AWS prod/ptl cert {account}/truststore/am.nhsdev.auth-ptl.cis2.spineservices.nhs.uk/crt DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.14.11.49...v2024.03.18.15.50

3036 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3037 https://nhsd-jira.digital.nhs.uk/browse/APM-4852?src=confmacro
3038 https://nhsd-jira.digital.nhs.uk/browse/APM-4987?src=confmacro
3039 https://nhsd-jira.digital.nhs.uk/browse/APM-4989?src=confmacro
3040 https://nhsd-jira.digital.nhs.uk/browse/APM-4991?src=confmacro

9.277.2 Prepared Release

Release Tag: **v2024.03.14.11.49**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=196040&view=results

9.277.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.277.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3041]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.277.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.03.14.11.49**

Pipelines - Run v2024.03.14.11.49+194712 (azure.com)[3042]

9.278 CHG0058536 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Daa2d9df1c3818e500317da3c0501317a%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.278.1 Tickets

https://nhsd-jira.digital.nhs.uk/browse/DOS-3398 - Updated filter and column heading on Organisation table
https://nhsd-jira.digital.nhs.uk/browse/DOS-3477 - Fix my apps e2e tests
https://nhsd-jira.digital.nhs.uk/browse/DOS-3211 - show-additional-information-files
https://nhsd-jira.digital.nhs.uk/browse/DOS-3475 - DAQ import causes all APIs to become Available
https://nhsd-jira.digital.nhs.uk/browse/DOS-3371 - Email lambda failure alerting

---

3041 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3042 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=194712&view=results

9.278.2 Prepared Release

Release Tag: v2024.03.11.08.06+196563

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=196563&view=results

9.278.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.278.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3043]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.278.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.02.05.08.14+192248

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=192248&view=results
- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.279 CHG0058624 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=53d73e9ec3c94a906c39935c05013125

9.279.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-4792[3044] - SPIKE: Test our implementation of rate limiting is working as expected DONE

3043 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3044 https://nhsd-jira.digital.nhs.uk/browse/APM-4792?src=confmacro

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.18.15.50...v2024.03.22.10.34

9.279.2 Prepared Release

Release Tag: **v2024.03.22.10.34**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=197306&view=results

9.279.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.279.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3045]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.279.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.03.18.15.50**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=196040&view=results

9.280 CHG0058712 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D90926e9fc3cd4ad00317da3c050131e0

9.280.1 Tickets

Change created by Vini Akshara Kannan . The Release implements the changes in the following tickets:

3045 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- APM-5049[3046] Certificate expiry - AWS prod/ptl cert {account}/truststore/ am.nhsidentity.spineservices.nhs.uk/crt[3047]
- APM-5031[3048] - Certificate expiry - AWS prod cert prod/client/ apigee.x26.api.direct.data.digital.nhs.uk/crt DONE

This change will be releasing the last released tag to pick up new certs (truststore and keystore mentioned above)

9.280.2 Prepared Release

Release Tag: **v2024.03.22.10.34**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=197997&view=results

9.280.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.280.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3049]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.280.4 Roll-Back

- Replace the secrets
- Rerun the build
- Run a new release pipeline of the build tag.

9.281 CHG0058728 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4f35c3dfc34dcad00317da3c050131f3

3046 https://nhsd-jira.digital.nhs.uk/browse/APM-5049
3047 http://am.nhsidentity.spineservices.nhs.uk/crt
3048 https://nhsd-jira.digital.nhs.uk/browse/APM-5031?src=confmacro
3049 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.281.1 Tickets

Change created by Vini Akshara Kannan . The Release implements the changes in the following tickets:

- APM-5076[3050] - Certificate expiry - int.api.service.nhs.uk DONE
- Also releasing the KVM for nhs-website-content-api

This change will also be releasing the new certs (truststore mentioned above)

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.22.10.34...v2024.03.22.16.51

9.281.2 Prepared Release

Release Tag: v2024.03.22.16.51[3051]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=198040&view=results

9.281.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.281.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3052]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.281.4 Roll-Back

- Redeploy:

  Release Tag: **v2024.03.22.10.34**

  https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=197997&view=results

3050 https://nhsd-jira.digital.nhs.uk/browse/APM-5076?src=confmacro
3051 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.03.22.16.51
3052 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.282 CHG0058779 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=fc632befc3cd8ed06c39935c05013195

9.282.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-5211[3053] - Target identity errors on api key paths if cis2 enabled DONE
- APM-5206[3054] - Investigate proxygen sandbox 401s DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.22.16.51...v2024.03.26.10.08

9.282.2 Prepared Release

Release Tag: **v2024.03.26.10.08**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=198341&view=results

9.282.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.282.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3055]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.282.4 Roll-Back

- Redeploy AWS:

3053 https://nhsd-jira.digital.nhs.uk/browse/APM-5211?src=confmacro
3054 https://nhsd-jira.digital.nhs.uk/browse/APM-5206?src=confmacro
3055 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

    - Release Tag: **v2024.03.22.10.34**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=197997&view=results
- Redeploy Apigee:
    - Release Tag: **v2024.03.22.16.51**
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=198040&view=results

9.283 CHG0058794 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3bf6c033c30106140317da3c05013167

9.283.1 Tickets

Change created by Vini Akshara Kannan . The Release implements the changes in the following tickets:

- APM-5074[3056] - Certificate expiry - AWS prod cert prod/server-certs/api.service.nhs.uk/crt
  DONE

9.283.2 Prepared Release

Release Tag: We will be releasing the **last released tag**

Will run the release on the day of schedule, so as to not block other releases of infrastructure till the time.

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200015&view=results

Comment:

Latest tag: v2024.04.02.09.53

With the first release on the last released tag, there was a error saying resource in use. Hence, a new build with latest tag (changes detailed below) was required to resolve the issue here[3057] and there were no changes in Apigee.

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.27.12.03...v2024.04.02.09.53

- mns.multicast-notifications.ref.national.nhs.uk crt and key references - had to retrieve the deleted secret
- ignore references to the portal keystore in the apigee-keystores-refs-targetservers and apigee-truststores pipeline tasks

---

3056 https://nhsd-jira.digital.nhs.uk/browse/APM-5074?src=confmacro

3057 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200005&view=logs&j=81749736-3d82-5e51-d1aa-cf02de26f6bf&t=1786db97-2dcc-56ea-0f66-1bcd7696ce40&l=98

9.283.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.283.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3058]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.283.4 Roll-Back

- Replace the secrets with the old crt and key
- Run a new build, on the last released tag
- Release on the last released tag

9.284 CHG0058838 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=673d7c80c359c6146c39935c050131ea

9.284.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- Fix poetry dependancy issue
- Release prod config for patient care aggregator reporting

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.03.26.10.08...v2024.03.27.12.03

9.284.2 Prepared Release

Release Tag: **v2024.03.27.12.03**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=198676&view=results

3058 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.284.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.284.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3059]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.284.4 Roll-Back

- Redeploy:
  - Release Tag: **v2024.03.26.10.08**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=198341&view=results

9.285 CHG0058988- Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D35591c02c3d10a946c39935c05013106%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.285.1 Tickets

https://nhsd-jira.digital.nhs.uk/browse/DOS-3430 - Make telephone number optional on user registration forms
https://nhsd-jira.digital.nhs.uk/browse/DOS-3284 - Content sections and links in supporting information not displayed correctly
https://nhsd-jira.digital.nhs.uk/browse/DOS-3492 - Delete product errors
https://nhsd-jira.digital.nhs.uk/browse/DOS-3429 - Inform user that the user already exists when adding Dev/AccMan
https://nhsd-jira.digital.nhs.uk/browse/DOS-3478 - Localstack / Lambda documentation updates
https://nhsd-jira.digital.nhs.uk/browse/DOS-3481 - Uploaded Date format issue - Maintain visibility of attachments after first review

---

3059 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.285.2 Prepared Release

Release Tag: #v2024.03.11.08.06+200583

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200583&view=results

9.285.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.285.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3060]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.285.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.03.11.08.06+196563

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=196563&view=results
- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.286 CHG0059059 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=e05c85eec3d106d46c39935c050131d8

9.286.1 Tickets

Change created by Alexander Carrie. The Release implements the changes in the following tickets:

- APM-5195[3061] - Add rate Limiting Custom Attributes in *NHS Login - Demographics - Prod* DONE

3060 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3061 https://nhsd-jira.digital.nhs.uk/browse/APM-5195?src=confmacro

9.286.2 Prepared Release

The release will involve adding the below value as a custom attribute of ratelimiting to the app 'NHS Login - Demographic - Prod'. App id - 8001a5a9-289b-4be0-ad6a-9d19e8aec104

https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/8001a5a9-289b-4be0-ad6a-9d19e8aec104

```
{"personal-demographics-prod": {"quota": {"enabled": true, "interval": 1, "limit":
3000, "timeunit": "minute"}, "spikeArrest": {"enabled": true, "ratelimit": "230ps"}}}
```

9.286.3 Release Process

1. Log on to apigee and go to link - https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/8001a5a9-289b-4be0-ad6a-9d19e8aec104
2. Edit app and add the above custom attribute to app
3. Monitor traffic from the app to check no increase in errors
4. Report back to consumer and pds

9.286.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3062]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.286.4 Roll-Back

- Remove attribute

9.287 CHG0059171 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db386567fc3190e186c39935c05013113%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3062 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.287.1 Tickets

Change created by Aleksandra Bak The Release implements the changes in the following tickets:

- APM-5170[3063] - Proxygen spec only - schemas and examples are ordered alphabetically DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.04.02.09.53...v2024.04.08.12.54

9.287.2 Prepared Release

Release Tag: v2024.04.08.12.54[3064]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=201730&view=results

9.287.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.287.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3065]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.287.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.04.02.09.53**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200015&view=results

3063 https://nhsd-jira.digital.nhs.uk/browse/APM-5170?src=confmacro
3064 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.04.08.12.54
3065 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.288 CHG0059319 - Digital Onboarding Content release for DAQ 14.4 Production

Link to ServiceNow Change request[3066]

9.288.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- ADZ-3456
- ADZ-3487

9.288.2 Prepared Release

Link to: DAQ 14.4[3067]Production Release

9.288.3 Release Process

Documentation on <How to release a DAQ to Production[3068]>

9.288.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.288.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3069]>

3066 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D142a5d6cc3a102986c39935c050131cb
3067 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2014.4%20-%20PROD%20RELEASE.xlsx?d=w154a4d81732049408a5c77d7f50499b8&csf=1&web=1&e=xK0qGJ
3068 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3069 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.289 CHG0059463 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=cfcb8ec6c3ad021c0317da3c05013150&sysparm_view=&sysparm_domain=null&sysparm_domain_scope=null

9.289.1 Tickets

Change created by Vikas Nautiyal . The Release implements the changes in the following tickets:

- APM-5238[3070] - Certificate expiry - AWS prod cert prod/client/bars-proxy.spineservices.nhs.uk/crt DONE

9.289.2 Prepared Release

The release will involve adding the below certificate in Production environment.

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=206405&view=results

AWS prod cert prod/client/bars-proxy.spineservices.nhs.uk/crt[3071]

Release Tag: v2024.04.22.11.29

9.289.3 Release Process

1. Log on to apigee and go to link - https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/8001a5a9-289b-4be0-ad6a-9d19e8aec104
2. Edit app and add the above custom attribute to app
3. Monitor traffic from the app to check no increase in errors
4. Report back to consumer and pds

9.289.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3072]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3070 https://nhsd-jira.digital.nhs.uk/browse/APM-5238?src=confmacro
3071 http://bars-proxy.spineservices.nhs.uk/crt
3072 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.289.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.04.08.12.54**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=201728&view=results

9.290 CHG0059608 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ca2cacf6c3a14e106c39935c05013198

9.290.1 Tickets

Change created by Sophie Clayton . The Release implements the changes in the following tickets:

- APM-3408[3073] - Correct naming of proxies, products & portal apis DONE
- APM-5068[3074] - Fix proxygen flakey tests DONE - only change outside of tests was increase max attempts when deleting a security group through proxygen

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.04.08.12.54...v2024.04.16.08.15

9.290.2 Prepared Release

Release Tag: v2024.04.16.08.15

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=204675&view=results

9.290.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

3073 https://nhsd-jira.digital.nhs.uk/browse/APM-3408?src=confmacro
3074 https://nhsd-jira.digital.nhs.uk/browse/APM-5068?src=confmacro

9.290.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3075]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.290.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.04.08.12.54**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=204065&view=results

9.291 CHG0059853- Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D70f26c30c37dc6d00317da3c050131b1%26sysparm_stack%3D%26sysparm_view%3D

9.291.1 Tickets

https://nhsd-jira.digital.nhs.uk/browse/DOS-3464 Add audit log entry when API is https://nhsd-jira.digital.nhs.uk/browse/added/removed from product
https://nhsd-jira.digital.nhs.uk/browse/DOS-3520 Health check lambda adhoc alerting errors
https://nhsd-jira.digital.nhs.uk/browse/DOS-3428 Inform user that the user already exists when adding Approver/Assessor
https://nhsd-jira.digital.nhs.uk/browse/DOS-3420 Spacing around page titles aligns to NHS Design System
https://nhsd-jira.digital.nhs.uk/browse/DOS-3423 Inform user that the user already exists when adding SysAdmin/OBL
https://nhsd-jira.digital.nhs.uk/browse/DOS-3476 Unanswered questions not working correctly
https://nhsd-jira.digital.nhs.uk/browse/DOS-3479 Error handling on checkboxes should follow GDS pattern
https://nhsd-jira.digital.nhs.uk/browse/DOS-3508 Delete product bug
https://nhsd-jira.digital.nhs.uk/browse/DOS-2768 Invalid protocol in links results in an error rather than a validation error
https://nhsd-jira.digital.nhs.uk/browse/DOS-3446 Validator: Upload crashes if incorrect date format on the Summary tab
https://nhsd-jira.digital.nhs.uk/browse/DOS-3521 Create a new preview page that will list sections by stages
Add database view for ApiDefinition sections

9.291.2 Prepared Release

Release Tag: #v2024.03.11.08.06+206382

3075 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=206382&view=results

9.291.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.291.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3076]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.291.4 Roll-Back

- Redeploy:
  - Release Tag: #v2024.03.11.08.06+200583

    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200583&view=results
- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.292 CHG0059908 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=30d79dc1c3b5ced00317da3c050131ee

9.292.1 Tickets

Re-releasing last tag to revert the BARS prod certs

9.292.2 Prepared Release

Release Tag: **v2024.04.22.11.29**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=206684&view=results

3076 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.292.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.292.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3077]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.292.4 Roll-Back

- Add back previous secrets, rebuild and redeploy same tag

9.293 CHG0060017 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D1e699be1c3bd42546c39935c0501316f%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.293.1 Tickets

Change created by Vini Akshara Kannan

Releasing merged PR https://github.com/NHSDigital/api-management-infrastructure/pull/1300

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.04.22.11.29...v2024.04.26.10.44

9.293.2 Prepared Release

Release Tag: **v2024.04.26.10.44**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=207390&view=results

3077 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.293.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.293.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3078]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.293.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.04.22.11.29[3079]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=207371&view=results

9.294 CHG0060198 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=d2f211d7c335ce940317da3c05013179

9.294.1 Tickets

Change created by Alexander Carrie

- APM-5185[3080] - Proxygen-cli user access with new keycloak client DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.04.22.11.29...v2024.04.29.13.26

9.294.2 Prepared Release

Release Tag: **v2024.04.29.13.26**

3078 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3079 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.04.22.11.29
3080 https://nhsd-jira.digital.nhs.uk/browse/APM-5185?src=confmacro

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=208049&view=results

9.294.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.294.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3081]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.294.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.04.22.11.29[3082]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=207371&view=results

9.295 CHG0060298 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dc2fe2633c37546186c39935c050131d0

9.295.1 Tickets

Change created by Fayez Ahmed

- APM-5256[3083] - Verify API Key - Writing the Shared flow

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.04.22.11.29...v2024.05.02.15.40

3081 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3082 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.04.22.11.29
3083 https://nhsd-jira.digital.nhs.uk/browse/APM-5256

9.295.2 Prepared Release

Release Tag: v2024.05.02.15.40[3084]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=210537&view=results

9.295.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.295.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3085]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.295.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.04.22.11.29[3086]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=207371&view=results

9.296 CHG0060407 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df71f805dc3c24a586c39935c05013158%26sysparm_stack%3D%26sysparm_view%3D

9.296.1 Tickets

Change created by Vini Akshara Kannan

- Update secret for prod/app-credentials/booking-and-referral/prod/GITHUB_TOKEN

3084 https://github.com/NHSDigital/api-management-infrastructure/tree/v2024.05.02.15.40
3085 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3086 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.04.22.11.29

9.296.2 Prepared Release

Release Tag: v2024.05.02.15.40[3087]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=210791&view=results

9.296.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.296.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3088]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.296.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.05.02.15.40[3089]
  - Replace the secrets
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=210537&view=results

9.297 CHG0060650 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=ab2d178fc306021c6c39935c05013187

9.297.1 Tickets

Change created by Sophie Clayton

3087 https://github.com/NHSDigital/api-management-infrastructure/tree/v2024.05.02.15.40
3088 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3089 https://github.com/NHSDigital/api-management-infrastructure/tree/v2024.05.02.15.40

- Adding new EPS target server as per request: https://nhsdigital-platforms.slack.com/archives/C04APUN97LG/p1715586195144499

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.05.02.15.40...v2024.05.13.07.44#

9.297.2 Prepared Release

Release Tag: v2024.05.13.07.44[3090]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=212170&view=results

9.297.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.297.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3091]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.297.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.05.02.15.40[3092]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=210791&view=results
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=210537&view=results

9.298 CHG0060675 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=f3562b1bc3c2461c6c39935c0501313b

3090 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.05.13.07.44
3091 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3092 https://github.com/NHSDigital/api-management-infrastructure/tree/v2024.05.02.15.40

9.298.1 Tickets

Change created by Alexander Carrie

- APM-5258[3093] - Verify API Key - Proxygen update api key flows to use shared flow DONE - Proxygen changes and involves udpating the EnhancedVerifyApiKey shared flow but **putting through as a standard change as this shared flow is not in use by anyone**

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.05.13.07.44...v2024.05.13.14.23

9.298.2 Prepared Release

Release Tag: **v2024.05.13.14.23**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=212284&view=results

9.298.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.298.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3094]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.298.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.05.13.07.44[3095]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=212170&view=results

3093 https://nhsd-jira.digital.nhs.uk/browse/APM-5258?src=confmacro
3094 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3095 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.05.13.07.44

9.299 CHG0060761 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=14c76933c386425c6c39935c0501310d

9.299.1 Tickets

Change created by Sophie Clayton

- APM-5356[3096]: allowing SDS/SCR access new G2 signed LDAP certs stored in secrets manager

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.05.13.14.23...v2024.05.15.10.49

9.299.2 Prepared Release

Release Tag: **v2024.05.13.14.23**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=213071&view=results

9.299.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.299.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3097]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.299.4 Roll-Back

- Redeploy:
    - Release Tag: **v2024.05.13.14.23**

3096 https://nhsd-jira.digital.nhs.uk/browse/APM-5356
3097 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=212284&view=results

9.300 CHG0060857 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=9cc5ee18c35a42106c39935c0501316f

9.300.1 Tickets

Change created by Sophie Clayton

- APM-5238[3098] - Certificate expiry - AWS prod cert prod/client/bars-proxy.spineservices.nhs.uk/crt DONE

  Renew cert and key for prod/client/bars-proxy.spineservices.nhs.uk/crt|key

There are no changes to the terraform for production, but will deploy the latest tag too to prevent it from going stale

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.05.15.10.49...v2024.05.20.15.01

9.300.2 Prepared Release

Release Tag: **v2024.05.20.15.01**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=214971&view=results

9.300.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt and key in prod AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.300.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3099]

3098 https://nhsd-jira.digital.nhs.uk/browse/APM-5238?src=confmacro

3099 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details for BaRs have been updated in the Apigee TLS Keystore

9.300.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.05.13.14.23
    - Replace the secrets with the old values
    - Run Infra build
    - Run infra release

9.301 CHG0060881 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=3e625bd4c39ac2106c39935c050131a7

9.301.1 Tickets

Change created by Sophie Clayton

- APM-5236[3100] - Certificate expiry - AWS prod cert prod/client/prod.apigee.client.ers.nhs.uk/crt
  DONE
  Renew cert and key for prod/client/prod.apigee.client.ers.nhs.uk/crt|key

This is a re-release of a previous tag

9.301.2 Prepared Release

Release Tag: **v2024.05.20.15.01**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=215287&view=results

9.301.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt and key in prod AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

3100 https://nhsd-jira.digital.nhs.uk/browse/APM-5236?src=confmacro

9.301.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3101]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details for e-RS have been updated in the Apigee TLS Keystore
4. Call e-RS /_status endpoint and ensure successful response

9.301.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.05.20.15.01
  - Replace the secrets with the old values
  - Run Infra build
  - Run infra release

9.302 CHG0061730 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Def6ae576c3e68e940317da3c0501312a

9.302.1 Tickets

Change created by Aleksandra Bak

- APM-4934[3102] - Add capability for specifying header name for custom attributes DONE
- APM-5266[3103] - Pass user id (nhs number for nhs login or uuid for cis2) to backend using Proxygen DONE
- APM-5346[3104] - Certificate expiry - AWS ptl cert ptl/server-certs/dev.api.service.nhs.uk/crt DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.05.20.15.01...v2024.06.06.07.46

3101 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3102 https://nhsd-jira.digital.nhs.uk/browse/APM-4934?src=confmacro
3103 https://nhsd-jira.digital.nhs.uk/browse/APM-5266?src=confmacro
3104 https://nhsd-jira.digital.nhs.uk/browse/APM-5346?src=confmacro

9.302.2 Prepared Release

Release Tag: v2024.06.06.07.46[3105]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=219528&view=results

9.302.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.302.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3106]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.302.4 Roll-Back

- Redeploy:
  - Release Tag:v2024.05.20.15.01[3107]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=214860&view=results

9.303 CHG0061790 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=7e4553b6c32a06106c39935c05013103

9.303.1 Tickets

Change created by Sophie Clayton

3105 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.06.07.46
3106 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3107 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.05.20.15.01

- APM-5346[3108] - Certificate expiry - AWS ptl cert ptl/server-certs/dev.api.service.nhs.uk/crt DONE
  - we're rolling back the ptl cert for the identity service prod trust store while we look into a cert issue.

Re-releasing v2024.06.06.07.46

9.303.2 Prepared Release

Release Tag: v2024.06.06.07.46[3109]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=219672&view=results

9.303.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.303.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3110]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.303.4 Roll-Back

- This is to rollback a ptl cert change to the prod identity service trust store in CHG0061730 - APIM Infrastructure Updates (see page 1951) only while an issue with the cert is resolved

9.304 CHG0062110 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6956b6cdc3ba82906c39935c050131dc

3108 https://nhsd-jira.digital.nhs.uk/browse/APM-5346?src=confmacro
3109 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.06.07.46
3110 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.304.1 Tickets

Change created by Sophie Clayton

- APM-5265[3111] - Pass full id token to backend using Proxygen DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.06.06.16.16...v2024.06.12.08.33

9.304.2 Prepared Release

Release Tag: v2024.06.12.08.33[3112]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221805&view=results

9.304.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.304.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3113]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.304.4 Roll-Back

- Redeploy:
  - Release Tag:v2024.06.06.16.16[3114]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=220478&view=results

3111 https://nhsd-jira.digital.nhs.uk/browse/APM-5265?src=confmacro
3112 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.12.08.33
3113 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3114 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.06.16.16

9.305 CHG0061960 - Identity Service Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7c181c64c3f2c2500317da3c0501314e%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.305.1 Tickets

Change created by Fayez Ahmed

- APM-5380[3115] - Update identity service token variables DONE

9.305.2 Prepared Release

Release Tag: v1.0.1740-alpha[3116]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221043&view=results

9.305.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.305.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3117]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.305.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1737-alpha[3118]
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=163112&view=results
- Redeploy *last successful non-production release* (to all envs except Production) - prepared release above

---

3115 https://nhsd-jira.digital.nhs.uk/browse/APM-5380?src=confmacro
3116 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1740-alpha
3117 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3118 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=163110

9.306 CHG0062021 - Identity Service Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=e4e431f0c3b24e506c39935c05013149

9.306.1 Tickets

Change created by Sophie Clayton

- APM-5380[3119] - Update identity service token variables DONE - Update to specify the identity service token variables for token exchange using the subject token

https://github.com/NHSDigital/identity-service-api/compare/v1.0.1740-alpha...v1.0.1742-alpha

9.306.2 Prepared Release

Release Tag: v1.0.1742-alpha[3120]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221043&view=results

9.306.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.306.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3121]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.306.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1740-alpha[3122]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221043&view=results

3119 https://nhsd-jira.digital.nhs.uk/browse/APM-5380?src=confmacro
3120 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1742-alpha
3121 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3122 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221041&view=results

- Redeploy *last successful non-production release* (to all envs except Production) - prepared release above

9.307 CHG0062247- Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D301e6f5ec336c6100317da3c050131f0%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.307.1 Tickets

https://nhsd-jira.digital.nhs.uk/browse/DOS-3577 - dotnet test is failing in git workflow on main branch
https://nhsd-jira.digital.nhs.uk/browse/DOS-3538 - Read only questions page alterations
https://nhsd-jira.digital.nhs.uk/browse/DOS-3045 - Validator: Give validation error when response type is missing
https://nhsd-jira.digital.nhs.uk/browse/DOS-3379 - Supplier email address searchable in DOS
https://nhsd-jira.digital.nhs.uk/browse/DOS-3009 - Validator: Remove email link validation warning
https://nhsd-jira.digital.nhs.uk/browse/DOS-1359 - Answer history items added for unreviewed questions when saving a section in review

9.307.2 Prepared Release

Release Tag: v2024.03.11.08.06+223037

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=223037&view=results

9.307.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.307.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3123]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.307.4 Roll-Back

- Redeploy:
    - Release Tag: #v2024.03.11.08.06+206382

3123 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=200583&view=results

- Redeploy *last successful non-production release* (to all envs except Production) **<ONLY APPLICABLE TO PRODUCER PIPELINES>**

9.308 CHG0062306 - Identity Service Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D03e8c572c3b24e100317da3c050131f2%26sysparm_stack%3Dchange_request_list.do%3Fsysparm_query%3Dactive%3Dtrue

9.308.1 Tickets

Change created by Fayez Ahmed

- APM-5267[3124] - Pass nhs login (identity proofing level) to backend using Proxygen DONE

9.308.2 Prepared Release

Release Tag: v1.0.1744[3125]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=223283&view=results

9.308.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.308.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3126]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.308.4 Roll-Back

- Redeploy:

3124 https://nhsd-jira.digital.nhs.uk/browse/APM-5267?src=confmacro
3125 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1744-alpha
3126 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

    - Release Tag: v1.0.1742-alpha[3127]
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221453&view=results
- Redeploy *last successful non-production release* (to all envs except Production) - prepared release above

9.309 CHG0062331 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dc4a305073bfa825076462a4a85e45a54

9.309.1 Tickets

Change created by Aleksandra Bak

- APM-5268[3128] - Pass cis2 (acr) to backend using Proxygen DONE
- https://github.com/NHSDigital/api-management-infrastructure/pull/1326/files

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.06.12.08.33...v2024.06.18.14.56

9.309.2 Prepared Release

Release Tag:v2024.06.18.14.56[3129]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=223487&view=results

9.309.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.309.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3130]

3127 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1742-alpha
3128 https://nhsd-jira.digital.nhs.uk/browse/APM-5268?src=confmacro
3129 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.18.14.56
3130 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.309.4 Roll-Back

- Redeploy:
  - Release Tag:v2024.06.12.08.33[3131]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=221805&view=results

9.310 CHG0062472 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=e42da4773bf64a1076462a4a85e45a66

9.310.1 Tickets

Change created by Vini Akshara Kannan

- APM-4693[3132]
- APM-5267[3133]

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.06.18.14.56...v2024.06.24.13.37

9.310.2 Prepared Release

Release Tag: v2024.06.24.13.37[3134]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=225020&view=results

9.310.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl

3131 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.12.08.33
3132 https://nhsd-jira.digital.nhs.uk/browse/APM-4693
3133 https://nhsd-jira.digital.nhs.uk/browse/APM-5267
3134 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.24.13.37

4. Click the Release Link above and approve the manual step to deploy the release to prod

9.310.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3135]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.310.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.06.18.14.56[3136]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=223487&view=results

9.311 CHG0063443 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=0621ee13c35b8e906c39935c05013145

9.311.1 Tickets

Change created by Sophie Clayton

- APM-5487[3137] - Certificate expiry - AWS prod cert prod/client/client-cert-prod-ci-apigee-client-cert/crt|key DONE

  Renew cert and key for Prescription for Patients
- APM-5460[3138] - Certificate expiry - AWS prod cert prod/truststore/identity.ptl.api.platform.nhs.uk/crt DONE

  Renew prod/truststore/identity.ptl.api.platform.nhs.uk/crt
- APM-5267[3139] - Pass nhs login (identity proofing level) to backend using Proxygen DONE - this was mentioned in a previous release but the Terraform Apply step wasn't run, so adding it here

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.06.18.14.56...v2024.06.27.09.29 (Splunk changes have already been released)

3135 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3136 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.18.14.56
3137 https://nhsd-jira.digital.nhs.uk/browse/APM-5487?src=confmacro
3138 https://nhsd-jira.digital.nhs.uk/browse/APM-5460?src=confmacro
3139 https://nhsd-jira.digital.nhs.uk/browse/APM-5267?src=confmacro

9.311.2 Prepared Release

Release Tag: v2024.06.27.09.29

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=231374&view=results

9.311.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt and key in prod AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.311.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3140]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.311.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.06.18.14.56 (terraform), v2024.06.24.13.37 (apigee)
  - Replace the secrets with the old values
  - Run Infra build
  - Run infra release

9.312 CHG0064048 - Digital Onboarding Content release for DAQ 15.1 Production

Link to ServiceNow Change request[3141]

3140 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3141 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D90eeaa3f3b2f8a9076462a4a85e45a6c

9.312.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3703
- DOS-3702
- DOS-3704

9.312.2 Prepared Release

Link to: DAQ 15.1[3142]Production Release

9.312.3 Release Process

Documentation on <How to release a DAQ to Production[3143]>

9.312.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.312.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3144]>

9.313 CHG0064121 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D883258983bbb42d076462a4a85e45a86%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3142 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/_layouts/15/Doc.aspx?sourcedoc=%7BAC8776C4-CE4C-489C-9CAF-84C6E2B401ED%7D&file=DAQ%2015.1%20-%20PROD%20RELEASE.xlsx&action=default&mobileredirect=true
3143 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3144 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.313.1 Tickets

Change created by Fayez Ahmed

- General update to ensure Infra doesn't get stale

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.06.27.09.29...v2024.07.30.10.08

9.313.2 Prepared Release

Release Tag: v2024.07.30.10.08[3145]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=235895&view=results

9.313.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.313.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3146]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.313.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.06.27.09.29[3147]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=231374&view=results

3145 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.07.30.10.08
3146 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3147 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.12.08.33

9.314 CHG0064142 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D0b9d6c983b73c2d076462a4a85e45af8%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.314.1 Tickets

Change created by Fayez Ahmed

- Update all DigiCerts due to security vulnerability. Notice was issued by DigiCert.

https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.07.30.10.08

9.314.2 Prepared Release

Release Tag: v2024.07.30.10.08[3148]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=236013&view=results

9.314.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.314.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3149]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.314.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.07.30.10.08 [3150]

3148 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.07.30.10.08
3149 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3150 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.06.12.08.33

- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=235895&view=results

9.315 CHG0064418 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=cb61f4e2c3334e986c39935c05013132

9.315.1 Tickets

Change created by Sophie Clayton

- APM-5516[3151] - Prod proxygen GET docker-token returns 500 DONE Bug fix for hosted containers with proxygen
- PI-543 USE_CPM external flag
- AEA-4290/AEA-4283 proxygen template wording updates

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.07.30.10.08...v2024.08.06.12.16

9.315.2 Prepared Release

Release Tag: v2024.08.06.12.16[3152]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=238220&view=results

9.315.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.315.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3153]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3151 https://nhsd-jira.digital.nhs.uk/browse/APM-5516?src=confmacro
3152 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.08.06.12.16
3153 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.315.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.06.27.09.29[3154]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=236013&view=results

9.316 CHG0064624 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D010027903b0c1ad476462a4a85e45abe

9.316.1 Tickets

Change created by Vini Akshara Kannan

- APM-5503[3155] Certificate expiry - TrustStore identity-service-truststore (prod)
- PR: https://github.com/NHSDigital/api-management-infrastructure/pull/1347

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.08.06.12.16...v2024.08.12.09.20

9.316.2 Prepared Release

Release Tag: v2024.08.12.09.20[3156]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=240022&view=results

9.316.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

3154 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/ v2024.07.30.10.08
3155 https://nhsd-jira.digital.nhs.uk/browse/APM-5503
3156 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.08.12.09.20

9.316.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3157]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.316.4 Roll-Back

- Redeploy:
  - Replace the cert secret
  - Build the Release Tag: v2024.08.06.12.16[3158] (last released tag) / v2024.08.12.09.20[3159] (PR alone)
  - Release the tag

9.317 CHG0065516 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f555920c3ba41ed076462a4a85e45aed

9.317.1 Tickets

Change created by Sophie Clayton

- APM-5594[3160] - Fix API Infrastructure Pipeline Poetry issue DONE - Infra dependency updates and new AMI ECS agent images from packer pipeline run

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.08.12.09.20...v2024.09.04.09.57

9.317.2 Prepared Release

Release Tag: v2024.09.04.09.57[3161]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=244412&view=results

3157 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3158 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.08.06.12.16
3159 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.08.12.09.20
3160 https://nhsd-jira.digital.nhs.uk/browse/APM-5594?src=confmacro
3161 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.09.04.09.57

9.317.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.317.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3162]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.317.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.08.12.09.20[3163] (last released tag)
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=240022&view=results

9.318 CHG0065725 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3F[…]26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull[3164]

9.318.1 Tickets

CHG created by Dhivakar Balasubramanian . The Release implements the changes in the following tickets:

- PR https://github.com/NHSDigital/api-management-infrastructure/pull/1351
- PR https://github.com/NHSDigital/api-management-infrastructure/pull/1356

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.09.04.09.57...v2024.09.09.15.27

3162 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3163 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.08.12.09.20
3164 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D391d5cb5c3ec16d46c39935c050131b9%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.318.2 Prepared Release

Release Tag: **v2024.09.09.15.27**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=245534&view=results

9.318.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.318.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3165]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.318.4 Roll-Back

Redeploy:

- **v2024.09.04.09.57**
- https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=244412

9.319 CHG0065727 - APIM - Add rate limiting custom attribute to app

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df497e4b9c36456d46c39935c0501313d%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.319.1 Tickets

CHG created by Vini Akshara Kannan . The Release implements the changes in the following Slack Workflow request:

- https://nhsdigital-platforms.slack.com/archives/C04APUN97LG/p1725547460826119

3165 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.319.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow (see page 1045)

Url:

```
https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/nhsd-prod/
apps/2183123e-c681-4de4-ac27-0973b028fcbd/attributes/ratelimiting
```

Payload:

```
{
  "identity-service-prod": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 60000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": false
    }
  }
}
```

9.319.3 Release Process

1. Rate limiting shared flow (see page 1045)

9.319.3.1 Post Release Checks

1. Ensure traffic still present (this comes in batches, so check for some time after and check with owners of app)

9.319.4 Roll-Back

- Manually remove the 'ratelimiting' attribute from the prod app and click save.

9.320 CHG0065769 - Identity Service Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=811505ce3b641a1076462a4a85e45a55

9.320.1 Tickets

Change created by Sophie Clayton

- APM-5428[3166] - Release Identity Service and Support e-RS to adds new scope (AAL2/1) to the product DONE

  / APM-5429[3167] - Enable and implement level 2 support for ID tokens DONE

9.320.2 Prepared Release

Release Tag: v1.0.1821-alpha[3168]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=243215&view=results

9.320.3 Release Process

1. Ensure the Identity Service release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.320.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3169]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.320.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.1744[3170]

3166 https://nhsd-jira.digital.nhs.uk/browse/APM-5428?src=confmacro
3167 https://nhsd-jira.digital.nhs.uk/browse/APM-5429?src=confmacro
3168 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1821-alpha
3169 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3170 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1744-alpha

    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=223283&view=results
- Redeploy *last successful non-production release* (to all envs except Production) - prepared release above

9.321 CHG0065961 - NHS App Custom Rate Limit Update

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=b5ce9c3fc320de580317da3c050131b6

9.321.1 Tickets

CHG created by Fayez Ahmed. The Release implements the changes in the following Slack Workflow request:

- https://nhsdigital-platforms.slack.com/archives/C04APUN97LG/p1726495470780619

9.321.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

Url:

```
https://apigee-deep-rest.prod.api.platform.nhs.uk[3171]:9004/v1/organizations/nhsd-prod/apps/d0dcfbfb-34eb-48fa-9f0f-f87e98ac51ad/attributes/ratelimiting
```

Payload:

```
{
  "identity-service-prod": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 1200,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": false
    }
  }
}
```

3171 http://rest.prod.api.platform.nhs.uk

9.321.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.321.3.1 Post Release Checks

1. Ensure traffic still present (this comes in batches, so check for some time after and check with owners of app)

9.321.4 Roll-Back

- Manually remove the 'ratelimiting' attribute from the prod app and click save.

9.322 CHG0066136 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=dd3c232c3bb016d076462a4a85e45a94

9.322.1 Tickets

Change created by Sophie Clayton

- APM-5679[3172] - GP Reg hosted container sandbox issues DONE - Proxygen hosted containers deployment fix
- APM-5717[3173] - CORS error when using NHS website "Try" functionality DONE - Proxygen CORS policy fix
- GP connect Apigee configuration
- Communications Manager target server

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.09.09.15.27...v2024.09.19.12.00

9.322.2 Prepared Release

Release Tag: v2024.09.19.12.00[3174]

3172 https://nhsd-jira.digital.nhs.uk/browse/APM-5679?src=confmacro
3173 https://nhsd-jira.digital.nhs.uk/browse/APM-5717?src=confmacro
3174 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.09.19.12.00

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=247308&view=results

9.322.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.322.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3175]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.322.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.09.09.15.27[3176] (last released tag)
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=245534&view=results

9.323 CHG0066257 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Df5a96042c3f85a106c39935c05013191

9.323.1 Tickets

Change created by Vikas Nautiyal

- APM-5689[3177] - Log exporter ECS task pending after latest release DONE - Log exporter ECS task pending after latest release

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.09.19.12.00...v2024.09.23.13.20

3175 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3176 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.09.09.15.27
3177 https://nhsd-jira.digital.nhs.uk/browse/APM-5689?src=confmacro

9.323.2 Prepared Release

Release Tag: v2024.09.23.13.20

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=247886&view=results

9.323.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.323.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3178]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.323.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.09.19.12.00[3179] (last released tag)
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=247308&view=results

9.324 CHG0066319 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=3af6c95e3b70de1076462a4a85e45a3b

9.324.1 Tickets

Change created by Fayez Ahmed

- APM-5673[3180] - Certificate expiry - AWS ptl cert ptl/truststore/aos.signin.nhs.uk/crt DONE [3181]

3178 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3179 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/ v2024.07.30.10.08
3180 https://nhsd-jira.digital.nhs.uk/browse/APM-5673?src=confmacro
3181 https://nhsd-jira.digital.nhs.uk/browse/APM-5487

- APM-5674[3182] - Certificate expiry - AWS prod cert prod/truststore/aos.signin.nhs.uk/crt DONE

9.324.2 Prepared Release

Release Tag: v2024.09.24.14.24

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=248229&view=results

9.324.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.324.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3183]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.324.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.09.23.13.20 (terraform), v2024.09.19.12.00 (apigee)
    - Replace the secrets with the old values
    - Run Infra build
    - Run infra release

9.325 CHG0066347 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D2d559d26c3b016500317da3c0501318a%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3182 https://nhsd-jira.digital.nhs.uk/browse/APM-5674?src=confmacro
3183 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.325.1 Tickets

AMP-3039 DOS landing page to show additional developer hub links for internal users
DOS-3766 Remove failing unit test Link_IsHttpsAndReturns200

DOS-3618 Remove unused message from message factory

9.325.2 Prepared Release

Release Tag: v2024.03.11.08.06+248211

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=248211&view=results

9.325.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.325.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3184]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.325.4 Roll-Back

Normal rollback procedure should not apply as previous release is stale. We would need to fix forwards.

9.326 CHG0066582 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D6ee3f78cc385de900317da3c050131dc%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.326.1 Tickets

AMP-3039 DOS landing page to show additional developer hub links for internal users
DOS-3766 Remove failing unit test Link_IsHttpsAndReturns200

DOS-3618 Remove unused message from message factory

3184 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.326.2 Prepared Release

Release Tag: v2024.03.11.08.06+248211

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=248211&view=results

9.326.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.326.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3185]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.326.4 Roll-Back

Normal rollback procedure should not apply as previous release is stale. We would need to fix forwards.

9.327 CHG0066743 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D207492743b41dad076462a4a85e45a4a%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.327.1 Tickets

AMP-3077 - Developer hub urls and title change

9.327.2 Prepared Release

v2024.03.11.08.06+249798

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=249798&view=results

3185 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.327.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.327.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3186]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.327.4 Roll-Back

Release previous image

v2024.03.11.08.06+249543

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=249543&view=results

9.328 CHG0066814 - Immunisation FHIR API deployment to PROD

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D0088e701c349d6146c39935c0501315a%26sysparm_stack%3D%26sysparm_view%3D

9.328.1 Tickets

CHG created by NareshKumar Joshi . The Release implements the changes in the following tickets:

- AMB-2259[3187] - Immunisation FHIR API Production Deployment DONE

9.328.2 Prepared Release

Release Tag: build_pipeline/v1.0.1639-alpha+249842[3188]

3186 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3187 https://nhsd-jira.digital.nhs.uk/browse/AMB-2259?src=confmacro
3188 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=249842

9.328.3 Release Process

1. **Pipeline Configuration**
   Azure pipelines for automatic deployment process.
   Ensure the pipelines are set to trigger on merges to the main/master branch.
2. **Build Pipeline**
   Trigger: Merge to main/master branch.
   Actions:
   Gather configuration from the GitHub repository.
   Build an artefact containing the necessary deployment files.
   Outcome: Successful build artefact creation.
3. **Release Pipeline**
   Trigger: Successful completion of the Build pipeline.
   Actions:
   Use the artefact from the Build pipeline.
   Deploy the configuration to Apigee and AWS.
   Deploy the hosted backend or sandbox if applicable.
   Outcome: Configuration deployed to Apigee and AWS.

9.328.3.1 Post Release Checks

1. **Smoke Testing**
   Conduct smoke tests using mock data to ensure the deployment is functioning correctly.
   Verify that all endpoints are responding as expected and there are no critical issues.
2. **Private Beta Phase**
   Upon successful smoke testing, transition the APIs to a private beta phase.
   Limit access to the RAV app within a specified region.
   Monitor performance and gather feedback from the RAV app users.

9.328.4 Roll-Back

- This is the initial deployment to PROD. If it fails, we do not need to roll it back since there will be no previous version to roll back to, and there will be no user impact.

9.329 CHG0066905 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dac7c6b563b01925476462a4a85e45a13%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D2%26sysparm_record_rows%3D23667%26sysparm_record_list%3DORDERBYDESCnumber

9.329.1 Tickets

Change created by Vikas Nautiyal

APM-5675[3189] - Certificate expiry - KeyStore nhsapp-20231114-122925 (prod) DONE

9.329.2 Prepared Release

Release Tag: v2024.10.09.13.49[3190]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=251076&view=results

9.329.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.329.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3191]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.329.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.10.09.13.49[3192] (terraform), v2024.10.09.13.49[3193] (apigee)
    - Replace the secrets with the old values
    - Run Infra build
    - Run infra release

3189 https://nhsd-jira.digital.nhs.uk/browse/APM-5675?src=confmacro
3190 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.10.09.13.49
3191 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3192 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.10.09.13.49
3193 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.10.09.13.49

9.330 CHG0067046 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/com.glideapp.servicecatalog_category_view.do%3Fsysparm_cartless%3Dtrue%26sysparm_catalog%3De0d08b13c3330100c8b837659bba8fb4%26sysparm_catalog_view%3Dcatalog_default%26sysparm_ck%3D32f66ac3c3411a940317da3c05013162d84cfaad8e51d38a4f3f6e7ec0360116d8a6055a%26sysparm_parent%3D8b10c51d1bc2b410a4cdc88b274bcb2a%26v%3D1

9.330.1 Tickets

Change created by Vini Akshara Kannan

- APM-5064[3194]

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.09.24.14.24...v2024.10.09.13.49

9.330.2 Prepared Release

Release Tag: v2024.10.09.13.49

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=251098&view=results

9.330.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.330.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3195]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3194 https://nhsd-jira.digital.nhs.uk/browse/APM-5064
3195 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.330.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.09.24.14.24
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=248229&view=results

9.331 CHG0067443 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dbc3dee65c3991a506c39935c05013105%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.331.1 Tickets

Merge of feature_gating branch into main, contains the following tickets/PR's.

feature_gating-sonar_fixes
dos-3812-prevent-dupe-product-names-in-dos2
DOS-3731-Pass_ProductID_delete_DOS_2_0
dos-3722-obl-can-reopen-approved-application-section-dos2
dos-3611-managing-audit-history-of-a-section-page
dos-3806-remove-landing-page-link
package-updates-october-24
main-ff-october-08
dos-3777-update-go-live-with-wording
DOS-3764-Upd_curr_AssTracker_DOS_2_0
dos-3795-remove-connection-agreement-placeholder
dos-3789-missing-breadcrumb
dos-3726-show-API-status-on-SAA-Overview-page
main-ff-into-feature_gating
dos-3788-cancel-button-showing-404
DOS-3727-Approve_API_DOS_2_0
dos-3692-ui-tests-1
dos-3721-lock-down-pseudo-api-once-approved
dos-3783-question-settings-section-merge-bug
dos-main-ff
dos-3674-merge-common-and-api-specific-questions
DOS-3717-Conn_agmnt_msg_before_snow
dos-3714-remove-api-from-dos2.0-products
dos-3682-allow-suppliers-to-answer-but-not-submit-assurance-questions-when-stage-gate1-closed
dos-3688-when-obl-create-dev-org-products-need-to-splits-between-legacy-and-singleAssessment-flows
dos-3728-path-breadcrumb-mismatch-in-an-existing-dos-1.0-product-overview-page
dos-3689-products-splits-between-flows-adding-a-product-to-existing-org
dos-3680-eligibility-gate
dos-3600-add-additional-api-in-single-assessment-api-dos-product
dos-3698-add-additional-api-in-existing-dos-system
dos-3675-product-flow-split-on-signup

DOS-3595-read-only-questions-for-dos-2.0-apis
DOS-3708-Change_Productname_CancelButton_fix
DOS-3594-3642-OBL_Supplier_SAAOverviewPage
dos-3616-add-additional-documents-product-page
dos-3690-set-api-flow-in-ApiDefinition-entity
dos-3641-products-and-org-tables-to-display-new-and-old-flow-links
DOS-3560-Delete_Product
dos-3677-show-use-case-section-in-check-your-eligibility-stage
DOS-3647-Save_Close_Cancel_Redirection
dos-3676-add-org-link-to-saaoverview-page
DOS-3619-Breadcrumb_BackLink_APIOverviewPage
dos-3620-handle-url-access-of-wrong-flow
DOS-3543-Holding_API_Assessment_stage_page
dos-3559-ability-to-edit-product-name
dos-3544-add-section-type-definition-for-use-case-sections
dos-3555-allow-section-submission-from-new-overview-page
dos-3542-overview-page-for-single-api-assessment-flow
dos-3499-create-gate

9.331.2 Prepared Release

v2024.03.11.08.06+249798

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=252646&view=results

9.331.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.331.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3196]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.331.4 Roll-Back

Release previous image

v2024.03.11.08.06+249543

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=249798&view=results

3196 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.332 CHG0067490- Digital Onboarding Content release for DAQ 16.0 Production

Link to ServiceNow Change request[3197]

9.332.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3822

9.332.2 Prepared Release

Link to: DAQ 16.0[3198] Production Release

9.332.3 Release Process

Documentation on <How to release a DAQ to Production[3199]>

9.332.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.332.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3200]>

3197 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D29c6f3a1c3511e500317da3c0501312c%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3198 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2016.0%20-%20PROD%20RELEASE.xlsx?d=wbfe8482b3d024ef7a7205552d477c2e4&csf=1&web=1&e=qcsrfG

3199 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3200 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.333 CHG0067527 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=5d2a6c0ac31d52900317da3c050131db

9.333.1 Tickets

Change created by Vini Akshara Kannan

- APM-5807[3201]

PR: https://github.com/NHSDigital/api-management-infrastructure/pull/1385/files

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.09.13.49...v2024.10.18.13.45

9.333.2 Prepared Release

Release Tag: v2024.10.18.13.45[3202]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253113&view=results

9.333.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.333.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3203]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

3201 https://nhsd-jira.digital.nhs.uk/browse/APM-5807
3202 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.10.18.13.45
3203 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.333.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.10.09.13.49[3204]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=251076&view=results

9.334 CHG0067588 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5343c4cffb11da10cfc6feceaeefdc88%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.334.1 Tickets

Change created by Vikas Nautiyal

- APM-5785[3205] - Deploy build agent infra to Prod AWS Account WITHDRAWN

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.18.13.45...v2024.10.21.11.20

9.334.2 Prepared Release

Release Tag: v2024.10.21.11.20

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253127&view=results

9.334.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

---

3204 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.10.09.13.49
3205 https://nhsd-jira.digital.nhs.uk/browse/APM-5785?src=confmacro

9.334.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3206]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.334.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.10.18.13.45[3207]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=252909&view=results

9.335 CHG0067640- APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=6d86789ffb9d5250cfc6feceaeefdcec&sysparm_view=&sysparm_domain=null&sysparm_domain_scope=null

9.335.1 Tickets

Change created by Vikas Nautiyal

- APM-5785[3208] - Deploy build agent infra to Prod AWS Account WITHDRAWN

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.18.13.45...v2024.10.22.10.11

9.335.2 Prepared Release

Release Tag: v2024.10.22.10.11[3209]

3206 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3207 https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.18.13.45...v2024.10.21.11.20
3208 https://nhsd-jira.digital.nhs.uk/browse/APM-5785?src=confmacro
3209 https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.21.11.20...v2024.10.22.10.11

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253327&view=results

9.335.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.335.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3210]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.335.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.10.18.13.45[3211]
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=252909&view=results

9.336 CHG0067720- APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D066f1e273b559e1067201da985e45afc%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.336.1 Tickets

Change created by Vikas Nautiyal

- APM-5785[3212] - Deploy build agent infra to Prod AWS Account WITHDRAWN

3210 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3211 https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.18.13.45...v2024.10.21.11.20
3212 https://nhsd-jira.digital.nhs.uk/browse/APM-5785?src=confmacro

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.22.10.11...v2024.10.23.12.19

9.336.2 Prepared Release

Release Tag:

v2024.10.23.12.19

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253871&view=results

9.336.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.336.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3213]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.336.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.10.22.10.11
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253327&view=results

9.337 CHG0067945 - Digital Onboarding Content release for DAQ 16.1 Production

Link to ServiceNow Change request[3214]

3213 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3214 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Deab17c61fb2d5a10cfc6feceaeefdcbb%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.337.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3832

9.337.2 Prepared Release

Link to: DAQ 16.1[3215] Production Release

9.337.3 Release Process

Documentation on <How to release a DAQ to Production[3216]>

9.337.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.337.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3217]>

9.338 CHG0068017 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D3f8396f53b21569067201da985e45a44%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.338.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

3215 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2016.1%20-%20PROD%20RELEASE.xlsx?d=w9e71796e9ede46f69dc8da512e4e733e&csf=1&web=1&e=Y3jqWL

3216 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3217 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

- dos-3800-error-when-removing-api-from-dos2.0-product
- dos-3821-error-when-delete-dos2.0-product
- dos-3824-readonly-questions-for-saa
- dos-3810-DOS_2_0_Product_correct_status

9.338.2 Prepared Release

Release Tag: v2024.03.11.08.06+256386

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=256386&view=results

9.338.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.338.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3218]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.338.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.03.11.08.06+252646 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=252646&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.339 CHG0068037 - Digital Onboarding Content release for DAQ 17.0 Production

Link to ServiceNow Change request[3219]

3218 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3219 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D7a7a72b1fbad1650cfc6feceaeefdccd

9.339.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3837

9.339.2 Prepared Release

Link to: DAQ 17.0[3220] Production Release

9.339.3 Release Process

Documentation on <How to release a DAQ to Production[3221]>

9.339.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.339.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3222]>

9.340 CHG0068157- APIM Infrastructure Updates

https://nhsdigital-platforms.slack.com/archives/D06MKK86AQN/p1730722995673449?thread_ts=1730721986.905169&cid=D06MKK86AQN

3220 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2017.0-%20PROD%20RELEASE.xlsx?d=we6a29d0deb2b448381efb0822e82ac11&csf=1&web=1&e=2qqfLW
3221 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3222 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.340.1 Tickets

Change created by Dhivakar Balasubramanian

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.10.22.10.11...v2024.11.04.12.07

9.340.2 Prepared Release

Release Tag:

v2024.11.04.12.07

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=257781&view=results

9.340.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl
4. Click the Release Link above and approve the manual step to deploy the release to prod

9.340.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3223]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.340.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.10.22.10.11
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=253327&view=results

3223 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.341 CHG0068402 - Digital Onboarding Content release for DAQ 18.0 Production

Link to ServiceNow Change request[3224]

9.341.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3760

9.341.2 Prepared Release

Link to: DAQ 18.0[3225] Production Release

9.341.3 Release Process

Documentation on <How to release a DAQ to Production[3226]>

9.341.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.341.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3227]>

3224 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db7f3617c3bf5561467201da985e45a48
3225 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2018.0%20-%20PROD%20RELEASE.xlsx?d=w997e89cfd5a34f22ac595f3f8a2e65d7&csf=1&web=1&e=mGMegc
3226 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3227 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.342 CHG0068438- APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D0114f8313bb99a1467201da985e45a7a

9.342.1 Tickets

Change created by Vikas Nautiyal

APM-5779[3228] - Azure pipeline notification setup for build agents DONE Set up an Azure Pipeline notification to alert us via a Slack message when no build agents are available. This ensures that our team is immediately informed of potential delays in the build process.

APM-5787[3229] - Update VEIT07 to new URL DONE Increase DNS cache timeouts for HA proxy`

APM-5834[3230] - Add the Immunization Accounts to Assume Role in Build Agents Role DONE Add the Immunization Accounts to Assume Role in Build Agents Role

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.11.04.12.07...v2024.11.11.13.35

9.342.2 Prepared Release

Release Tag:

v2024.11.11.13.35

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=259669&view=results

9.342.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to ptl
3. Validate that there are no graphical errors when using the proxygen cli, targeting ptl

3228 https://nhsd-jira.digital.nhs.uk/browse/APM-5779?src=confmacro
3229 https://nhsd-jira.digital.nhs.uk/browse/APM-5787?src=confmacro
3230 https://nhsd-jira.digital.nhs.uk/browse/APM-5834?src=confmacro

4. Click the Release Link above and approve the manual step to deploy the release to prod

9.342.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3231]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.342.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.11.04.12.07
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=257781&view=results

9.343 CHG0068545 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=0424e6523b35925467201da985e45a34

9.343.1 Tickets

Change created by Fayez Ahmed

APM-5799[3232] - Certificate expiry - AWS prod cert prod/client/prod.nhse.wlmdsapi.necsapps.net/crt
DONE

9.343.2 Prepared Release

Release Tag: v2024.11.11.13.35[3233]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=261495&view=results

9.343.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values

3231 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3232 https://nhsd-jira.digital.nhs.uk/browse/APM-5799?src=confmacro
3233 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.11.11.13.35

3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.343.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3234]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.343.4 Roll-Back

- Redeploy:
  - Release Tag: (terraform), (apigee)
  - Replace the secrets with the old values
  - Run Infra build
  - Run infra release

9.344 CHG0068584 - Digital Onboarding Content release for DAQ 18.1 Production

Link to ServiceNow Change request[3235]

9.344.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3849

9.344.2 Prepared Release

Link to: DAQ 18.1[3236] Production Release

3234 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3235 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd0de1716fb751698ecbef3baaeefdc8e
3236 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2018.1%20-%20PROD%20RELEASE.xlsx?d=w3485177d02c8495083e9f75c837c8e59&csf=1&web=1&e=shJKkJ

9.344.3 Release Process

Documentation on <How to release a DAQ to Production[3237]>

9.344.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.344.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3238]>

9.345 CHG0068626 - Multicast Notifications Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=00be0cb63bbdd65467201da985e45ad6

9.345.1 Tickets

CHG created by Alexander Carrie. The Release implements the changes in the following tickets:

- Dependabot updates and tidy ups

https://github.com/NHSDigital/multicast-notification-service/compare/v1.0.205-alpha...v1.0.246-alpha

9.345.2 Prepared Release

Release Tag: **v1.0.246-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=260633&view=results

9.345.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

3237 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3238 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.345.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3239]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.345.4 Roll-Back

- Redeploy:
    - Build from scratch with tag **v.1.0.205-alpha - this will trigger release**
- Redeploy *last successful non-production release* (to all envs except Production)

9.346 CHG0068898 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D9d48d3e0fb4e1e10cfc6feceaeefdc5c%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.346.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- dos-3846-product-table-incorrect-section-count-bug
- dos-3840-add-google-tm-vars-for-user-type
- dos-3796-remove-unused-email-templates
- dos-3813-add-audit-history-for-obl-to-view-when-reviewing-a-section
- DOS-3692-UI_test_for_APIOverview_AuditHistory
- dos-3826-attach-file-on-section-review-tweak

9.346.2 Prepared Release

Release Tag: v2024.03.11.08.06+262311

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=262311&view=results

9.346.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

3239 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.346.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3240]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.346.4 Roll-Back

- Redeploy:
  Release Tag: v2024.03.11.08.06+256386
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=256386&view=results

9.347 CHG0069094 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dd8362d5afbc21e50ecbef3baaeefdc1f

9.347.1 Tickets

Change created by Vini Akshara Kannan

APM-5818[3241]
https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.11.11.13.35...v2024.11.25.17.03

9.347.2 Prepared Release

Release Tag: v2024.11.11.13.35[3242]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=263629&view=results

9.347.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above

3240 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3241 https://nhsd-jira.digital.nhs.uk/browse/APM-5818
3242 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.11.11.13.35

4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.347.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3243]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.347.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.11.11.13.35
  - Run Infra release
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=261495&view=results

9.348 CHG0069202 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=e66107b2fb025a90ecbef3baaeefdc10

9.348.1 Tickets

Change created by Sophie Clayton

Reverting changes from APM-5785[3244] - Deploy build agent infra to Prod AWS Account WITHDRAWN
https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.11.25.17.03...v2024.11.27.13.41

9.348.2 Prepared Release

Release Tag: v2024.11.27.13.41[3245]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=263629&view=results

---

3243 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3244 https://nhsd-jira.digital.nhs.uk/browse/APM-5785?src=confmacro
3245 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.11.27.13.41

Had to run the pipeline a second time due to issue with destroying a non empty bucket. Once fully emptied, ran again for a fresh terraform plan and it could be destroyed: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=264203&view=results

9.348.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.348.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3246]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.348.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.11.25.17.03
  - Run Infra release
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=263629&view=results

9.349 CHG0069390 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/change_request.do?sys_id=bfebd4683bda1a9067201da985e45aea&sysparm_view=&sysparm_domain=null&sysparm_domain_scope=null

9.349.1 Tickets

Change created by Vikas Nautiyal

Change related to remove invalid character # in prooxygen
https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.11.27.13.41...v2024.12.02.11.10

3246 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.349.2 Prepared Release

Release Tag: v2024.12.02.11.10

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=265265&view=results

9.349.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.349.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3247]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.349.4 Roll-Back

- Redeploy:
  - Release Tag: v2024.11.27.13.41[3248]
  - Run Infra releasehttps://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=263629&view=results

9.350 CHG0069773 - Digital Onboarding Content release for DAQ 18.2 Production

Link to ServiceNow Change request[3249]

3247 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3248 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.11.27.13.41
3249 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da0f5229b3bda9a1467201da985e45a72%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.350.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3855
- DOS-3918
- DOS-3920

9.350.2 Prepared Release

Link to: DAQ 18.2[3250] Production Release

9.350.3 Release Process

Documentation on <How to release a DAQ to Production[3251]>

9.350.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.350.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3252]>

9.351 CHG0070092 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dfb3a9fe9fb6a1290ecbef3baaeefdc69%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3250 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2018.2%20-%20PROD%20RELEASE.xlsx?d=w133ebf2278ea49d1978d5444912c976b&csf=1&web=1&e=sjoD5X
3251 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3252 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.351.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- dos-3848-create-widget-pages
- dos-3778-UI-change-to-section-buttons
- dos-3901-developer-hub-mvp-work-external
- dos-3865-application-dormant-status-change
- dos-3904-Duplicate_product_defect
- dos-3838-increase-change-req-max-character
- dos-3891-API_creation_whitespace
- dos-3831-Section_Status_Assessment_Tracker
- dos-3687-my-applications-page-caching-unit-testing
- dos-3845-re-instate-dos1-ui-tests-part2
- dos-3604-re-instate-dos1-ui-tests-part1

9.351.2 Prepared Release

Release Tag: v2024.12.16.17.22+270623

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=270623&view=results

9.351.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.351.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3253]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.351.4 Roll-Back

- Redeploy:
  Release Tag: v2024.03.11.08.06+262311
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=262311&view=results

3253 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.352 CHG0070107 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=fcf4a329fb625290ecbef3baaeefdca1

9.352.1 Tickets

Change created by Sophie Clayton

No Prod infrastructure changes - last release before change freeze

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.12.02.11.10...v2024.12.17.11.26

(note build agent changes are PTL only)

9.352.2 Prepared Release

Release Tag: v2024.12.17.11.26[3254]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=270848&view=results

9.352.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.352.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3255]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.352.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.12.02.11.10

3254 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.12.17.11.26
3255 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Run Infra release: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=265315&view=results

9.353 CHG0070342 - NHS App Custom Rate Limit Update

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=40161137fb721a10cfc6feceaeefdca6

9.353.1 Tickets

CHG created by Sophie Clayton . The Release implements the changes in the following rate limiting request:

- https://nhsdigitallive.service-now.com/nav_to.do?uri=sc_task.do?sys_id=d2da0d7ffb3ad610cfc6feceaeefdc93

9.353.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

Url:

```
https://apigee-deep-rest.prod.api.platform.nhs.uk[3256]:9004/v1/organizations/nhsd-prod/apps/d0dcfbfb-34eb-48fa-9f0f-f87e98ac51ad/attributes/ratelimiting/identity-service-prod
```

Payload:

```
{
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 3500,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": false
    }
}
```

3256 http://rest.prod.api.platform.nhs.uk

9.353.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.353.3.1 Post Release Checks

1. Ensure traffic still present (this comes in batches, so check for some time after and check with owners of app)

9.353.4 Roll-Back

- https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/d0dcfbfb-34eb-48fa-9f0f-f87e98ac51ad
- Click edit
- Update the ratelimiting custom attribute to the old value: {"identity-service-prod": {"quota": {"enabled": true, "interval": 1, "limit": 1200, "timeunit": "minute"}, "spikeArrest": {"enabled": false}}}
- This will revert the app to it's last working state

9.354 CHG0070584 - Multicast Notifications Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=5bf864cafbc7d610cfc6feceaeefdcbb

9.354.1 Tickets

CHG created by Alexander Carrie. The Release implements the changes in the following tickets:

- EM-1514
- EM-1575

https://github.com/NHSDigital/multicast-notification-service/compare/v1.0.246-alpha...v1.0.268-alpha

9.354.2 Prepared Release

Release Tag: **v1.0.268-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=279590&view=results

9.354.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.354.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3257]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.354.4 Roll-Back

- Redeploy:
  - **v1.0.246-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=260633&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.355 CHG0072360 - PDS Custom Rate Limit Update for Anima

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=71bdd19ec3fb12506c39935c05013190

9.355.1 Tickets

CHG created by Sophie Clayton . The Release implements the changes in the following rate limiting request:

- https://nhsdigitallive.service-now.com/nav_to.do?uri=sc_task.do?sys_id=f6ff2047c3af5a100317da3c050131ac

9.355.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

First check no rate limiting attribute is current set on the app:

```
curl https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/nhsd-prod/apps/4a9ac11a-1c90-4764-853b-ede26c700607/attributes/ratelimiting -H 'Authorization: bearer <your_apigee_access_token>' | jq
```

Should return:

3257 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

```
{
  "detail": "app 4a9ac11a-1c90-4764-853b-ede26c700607 has no attribute with name
ratelimiting"
}
```

Add PDS rate limiting attribute (TBC with demo 101):

```
curl -X PUT https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/
nhsd-prod/apps/4a9ac11a-1c90-4764-853b-ede26c700607/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' -H 'content-type: application/
json' --data '
{
  "personal-demographics-prod": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 12000,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": true,
      "ratelimit": "36000pm"
    }
  }
}
'
```

9.355.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.355.3.1 Post Release Checks

1. Ensure the rate limiting custom attribute has been added to the app.
2. Monitor Grafana Prod dashboard[3258]
3. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
4. Verify with Demo 101 team

3258 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.355.4 Roll-Back

- https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/d4d83595-64a7-4a86-a2d1-6348df504dcc
- Click edit
- Remove the ratelimiting custom attribute
- Save

9.356 CHG0070601 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D961f054afb0f5a10ecbef3baaeefdcb7%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.356.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- dos-3964-and-dos-3966-assessment-tracker-speed
- DOS-3470-Re-order-DAQ-change-log
- dos-3433-My-applications-page-caching
- dos-3908-allow-to-delete-organisation-with-gateway-approved-products
- DOS-3526-Invite_email_content

9.356.2 Prepared Release

Release Tag: v2024.12.16.17.22+279677

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=279677&view=results

9.356.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.356.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3259]

[3259] https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.356.4 Roll-Back

- Redeploy:
  Release Tag: v2024.12.16.17.22+270623
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=270623&view=results

9.357 CHG0070517- Immunisation FHIR API deployment to PROD

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D948019703b47565067201da985e45a80%26sysparm_stack%3D%26sysparm_view%3D

9.357.1 Tickets

CHG created by Saptarshi Mandal . The Release implements the changes in the following tickets:

https://nhsd-jira.digital.nhs.uk/browse/AMB-2235 - Blue/Green Deployment
https://nhsd-jira.digital.nhs.uk/browse/AMB-2239 - Immutable Backup
https://nhsd-jira.digital.nhs.uk/browse/AMB-2334 - Tech Debt - Overly Permissive Security Group
https://nhsd-jira.digital.nhs.uk/browse/AMB-2335 - Tech Debt - Dockerfile Root User in Use
https://nhsd-jira.digital.nhs.uk/browse/AMB-2337 - Tech Debt - Terraform Missing Security Best Practices
https://nhsd-jira.digital.nhs.uk/browse/AMB-2340 - Tech Debt - HTTP Header Poisoning for Ping Endpoint in Apigee

9.357.2 Prepared Release

Release Tag: build_pipeline/v1.0.1639-alpha+249842[3260]

9.357.3 Release Process

1. **Pipeline Configuration**
   Azure pipelines for automatic deployment process.
   Ensure the pipelines are set to trigger on merges to the main/master branch.
2. **Build Pipeline**
   Trigger: Merge to main/master branch.
   Actions:
   Gather configuration from the GitHub repository.

3260 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=249842

Build an artefact containing the necessary deployment files.
Outcome: Successful build artefact creation.

3. **Release Pipeline**
   Trigger: Successful completion of the Build pipeline.
   Actions:
   Use the artefact from the Build pipeline.
   Deploy the configuration to Apigee and AWS.
   Deploy the hosted backend or sandbox if applicable.
   Outcome: Configuration deployed to Apigee and AWS.

9.357.3.1 Post Release Checks

1. **Smoke Testing**
   Conduct smoke tests using mock data to ensure the deployment is functioning correctly.
   Verify that all endpoints are responding as expected and there are no critical issues.
2. **Private Beta Phase**
   Upon successful smoke testing, transition the APIs to a private beta phase.
   Limit access to the RAV app within a specified region.
   Monitor performance and gather feedback from the RAV app users.

9.357.4 Roll-Back

- If needed deploy the build_pipeline/v1.0.1639-alpha+249842[3261] version to roll back.

9.358 CHG0070675 - Digital Onboarding Content release for DAQ 19.0 Production

Link to ServiceNow Change request[3262]

9.358.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3978
- DOS-3746

3261 https://dev.azure.com/NHSD-APIM/c1016748-466d-4949-8c46-7681cfd98859/_build/results?buildId=249842
3262 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De754eeeefb0f5650ecbef3baaeefdc32

9.358.2 Prepared Release

Link to: DAQ 19.0[3263] Production Release

9.358.3 Release Process

Documentation on <How to release a DAQ to Production[3264]>

9.358.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.358.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3265]>

9.359 CHG0070705 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=9cd3efeafbcf9650cfc6feceaeefdc12

9.359.1 Tickets

Change created by Sophie Clayton

- Apigee KVM update https://nhsdigitallive.service-now.com/nav_to.do?uri=sc_task.do?sys_id=d1e3cb2efb4bd650ecbef3baaeefdc6a

No Prod infrastructure changes

https://github.com/NHSDigital/api-management-infrastructure/compare/v2024.12.17.11.26...v2025.01.15.14.42

(note build agent changes are PTL only, AWS SSO are local only)

3263 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2019.0%20-%20PROD%20RELEASE.xlsx?d=w19ea040ccf4f43649b50704e1852f78b&csf=1&web=1&e=nvAaDx

3264 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3265 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.359.2 Prepared Release

Release Tag: v2025.01.15.14.42[3266]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=280877&view=results

9.359.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.359.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3267]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.359.4 Roll-Back

- Redeploy:
    - Release Tag: v2024.12.17.11.26[3268]
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=270848&view=results

9.360 CHG0071016 - NHS App Custom Rate Limit Update

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=09e11ff4fb175ed0cfc6feceaeefdc58

9.360.1 Tickets

CHG created by Fayez Ahmed. The Release implements the changes in the following rate limiting request:

- https://nhsdigitallive.service-now.com/nav_to.do?uri=sc_task.do?sys_id=c82f3081fb8f1690cfc6feceaeefdc87

3266 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/ v2025.01.15.14.42
3267 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3268 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2024.12.17.11.26

9.360.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

Url:

```
https://apigee-deep-rest.prod.api.platform.nhs.uk[3269]:9004/v1/organizations/nhsd-prod/apps/d0dcfbfb-34eb-48fa-9f0f-f87e98ac51ad/attributes/ratelimiting/personal-demographics-prod
```

Payload:

```
{
  "quota": {
    "enabled": true,
    "interval": 1,
    "limit": 3500,
    "timeunit": "minute"
  },
  "spikeArrest": {
    "enabled": false
  }
}
```

9.360.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.360.3.1 Post Release Checks

1. Ensure traffic still present (this comes in batches, so check for some time after and check with owners of app)

3269 http://rest.prod.api.platform.nhs.uk

9.360.4 Roll-Back

- https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/d0dcfbfb-34eb-48fa-9f0f-f87e98ac51ad
- Click edit
- Update the ratelimiting custom attribute to the old value: {"identity-service-prod": {"quota": {"enabled": true, "interval": 1, "limit": 3500, "timeunit": "minute"}, "spikeArrest": {"enabled": false}}}
- This will revert the app to it's last working state

9.361 CHG0071101 - Digital Onboarding Content release for DAQ 19.1 Production

Link to ServiceNow Change request[3270]

9.361.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3993
- DOS-3953
- DOS-3992

9.361.2 Prepared Release

Link to: DAQ 19.1[3271] Production Release

9.361.3 Release Process

Documentation on <How to release a DAQ to Production[3272]>

9.361.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

3270 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D599760e1fb5f1254ecbef3baaeefdc23
3271 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2019.1%20-%20PROD%20RELEASE.xlsx?d=w74639700a85f4d029a3528ade71bb0be&csf=1&web=1&e=2eTGmr
3272 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

9.361.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3273]>

9.362 CHG0071606 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=8b99ea0dfb2796d0ecbef3baaeefdc60

9.362.1 Tickets

Change created by Sophie Clayton

- APM-6039[3274] - Proxygen hosted container healthcheck endpoint not set DONE

(Other changes are PTL only)

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.01.15.14.42...v2025.02.04.13.57

(note build agent changes are PTL only, AWS SSO are local only)

9.362.2 Prepared Release

Release Tag: v2025.02.04.13.57[3275]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=289316&view=results

9.362.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

3273 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

3274 https://nhsd-jira.digital.nhs.uk/browse/APM-6039?src=confmacro

3275 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.02.04.13.57

9.362.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3276]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.362.4 Roll-Back

- Redeploy:
    - Release Tag: v2025.01.15.14.42[3277]
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=280877&view=results

9.363 CHG0071920 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=6882d65bc3e7d2500317da3c05013193

9.363.1 Tickets

Change created by Fayez Ahmed & Sophie Clayton

- APM-5900[3278] - Certificate expiry - {prod|ptl}/truststore/github.com/crt DONE
- APM-5934[3279] - Certificate expiry - auth.login.nhs.uk DONE
- allow getting proxy details in proxygen deployed api (https://nhsdigitallive.service-now.com/nav_to.do?uri=sc_task.do?sys_id=6bec67f33b671e5076462a4a85e45acd)

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.02.04.13.57...v2025.02.14.17.12

Note: most code changes are for PTL services.

9.363.2 Prepared Release

Release Tag: v2025.02.14.17.12[3280]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=294917&view=results

---

3276 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3277 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/ v2025.01.15.14.42
3278 https://nhsd-jira.digital.nhs.uk/browse/APM-5900?src=confmacro
3279 https://nhsd-jira.digital.nhs.uk/browse/APM-5934?src=confmacro
3280 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.02.14.17.12

9.363.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above (external Apigee and prod terraform)
5. Approve the manual step to deploy the release to prod

9.363.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3281]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.363.4 Roll-Back

- Redeploy:
  - Release Tag: v2025.02.04.13.57
  - Replace the secrets with the old values
  - Run Infra build
  - Run infra release

9.364 CHG0071923 - Digital Onboarding Content release for DAQ 20.0 Production

Link to ServiceNow Change request[3282]

9.364.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-4021
- ~~DOS-3879~~

3281 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3282 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db094dadb3b63525076462a4a85e45aea

- DOS-4008

9.364.2 Prepared Release

Link to: DAQ 20.0[3283] Production Release

9.364.3 Release Process

Documentation on <How to release a DAQ to Production[3284]>

9.364.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.364.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3285]>

9.365 CHG0071979 - Digital Onboarding Content release for DAQ 21.0 Production

Link to ServiceNow Change request[3286]

9.365.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-4003
- DOS-4027

3283 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2020.0%20-%20PROD%20RELEASE.xlsx?d=wcda287bc24c54215a3e29316d34c52dc&csf=1&web=1&e=ND4p8P
3284 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production
3285 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful
3286 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da99533efc3271e500317da3c050131de

9.365.2 Prepared Release

Link to: DAQ 21.0[3287] Production Release

9.365.3 Release Process

Documentation on <How to release a DAQ to Production[3288]>

9.365.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.365.4 Roll-Back

- Redeploy last successful production release
    - Link to <How to roll-back a DAQ if a release is unsuccessful[3289]>

9.366 CHG0072339 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D14179456c337de106c39935c05013107%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.366.1 Tickets

CHG created by Dhivakar Balasubramanian The Release implements the changes in the following tickets:

- DOS-3975-remove-fontawesome-dependency
- DOS-3957-edit-widget-ability
- DOS-3957-edit-widget-ability-hotfix
- DOS-3957-edit-widget-ability-hotfix2
- DOS-3962-question-settings-with-checkbox-fix
- DOS-3998-base-for-new-assessment-tracker-page

3287 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2021.0%20-%20PROD%20RELEASE.xlsx?d=we6bc73f39e614486849ac8d4a5240cc0&csf=1&web=1&e=rkUDNR

3288 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

3289 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful

9.366.2 Prepared Release

Release Tag: v2024.12.16.17.22+279677

Link to Azure:

9.366.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.366.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3290]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.366.4 Roll-Back

- Redeploy:
  Release Tag: v2024.12.16.17.22+279677
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=279677&view=results

9.367 CHG0072472 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=2496a8efc3b35a506c39935c0501312a

9.367.1 Tickets

Change created by Sophie Clayton

- APM-6094[3291] - proxygen /spec call not returning all API specs DONE
- APM-6039[3292] - Proxygen hosted container healthcheck endpoint not set DONE

3290 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3291 https://nhsd-jira.digital.nhs.uk/browse/APM-6094?src=confmacro
3292 https://nhsd-jira.digital.nhs.uk/browse/APM-6039?src=confmacro

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.02.14.17.12...v2025.02.24.10.07

(note build agent changes are PTL only)

9.367.2 Prepared Release

Release Tag: v2025.02.24.10.07[3293]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=298302&view=results

9.367.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.367.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3294]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.367.4 Roll-Back

- Redeploy:
  - Release Tag: v2025.02.14.17.12[3295]
    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=294917&view=results

9.368 CHG0072682 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dfb950090c34466906c39935c050131af

3293 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.02.24.10.07
3294 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3295 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/ v2025.02.14.17.12

9.368.1 Tickets

Change created by ViniAkshara Kannan

- CS0988162[3296]

https://github.com/NHSDigital/api-management-infrastructure/pull/1441/files

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.02.24.10.07...v2025.02.26.16.44

9.368.2 Prepared Release

Release Tag: v2025.02.26.16.44[3297]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=299866&view=results

9.368.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.368.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3298]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.368.4 Roll-Back

- Redeploy:
    - Release Tag: v2025.02.24.10.07[3299]

3296 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/sn_customerservice_case.do%3Fsys_id%3D8a71e7c0c38cead00317da3c0501317c%26sysparm_stack%3D%26sysparm_view%3Dcase%26sysparm_view_forced%3Dtrue

3297 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.02.26.16.44

3298 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

3299 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.02.24.10.07

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=298302&view=results

9.369 CHG0073010 - Digital Onboarding Content release for DAQ 22.0 Production

Link to ServiceNow Change request[3300]

9.369.1 Tickets

CHG created by Andrew Thompson
The Release implements the changes in the following tickets:

- DOS-3897
- DOS-4055
- DOS-4052
- DOS-4031
- DOS-4067
- DOS-3879

9.369.2 Prepared Release

Link to: DAQ 22.0[3301] Production Release

9.369.3 Release Process

Documentation on <How to release a DAQ to Production[3302]>

9.369.3.1 Post Release Checks

1. Confirm the Digital Onboarding Service Changelog has updated accurately
2. Confirm your content changes have been applied successfully

9.369.4 Roll-Back

- Redeploy last successful production release

3300 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da2be28e63b0c62d476462a4a85e45ab1%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3301 https://nhs.sharepoint.com/:x:/r/sites/msteams_b5f2aa/Shared%20Documents/General/6.%20Implementation/DOS/DAQs/DAQ%20archive/PROD/DAQ%2022.0%20-%20PROD%20RELEASE.xlsx?d=wc4e0175f07a74cf8ac10f1c111caae75&csf=1&web=1&e=q6B95u

3302 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production

- Link to <How to roll-back a DAQ if a release is unsuccessful[3303]>

9.370 CHG0073067 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D145715fac30822146c39935c05013161%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.370.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- dos-4074-use-section-alias-in-assessment-tracker-header[3304]
- dos-4072-assessment-tracker-filtering-section-and-state[3305]
- dos-4070-assessment-tracker-toggle-pagination[3306]
- dos-4062-assessment-tracker-column-sorting[3307]
- dos-4051-pagination-on_assessment-tracker[3308]
- dos-3860-show-sections-of-section-filter-only[3309]
- dos-3990-assessment-tracker-cvs-download[3310]
- dos-4039-Allow-freetext-filtering-assessment-tracker-filters[3311]
- dos-4047-new-at-filter-bugs[3312]

9.370.2 Prepared Release

Release Tag: v2024.12.16.17.22+304065

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=304065&view=results

9.370.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed

---

3303 https://nhsd-confluence.digital.nhs.uk/display/DRM/5.+How+to+release+a+DAQ+to+Production#id-5.HowtoreleaseaDAQtoProduction-9.Howtoroll-backaDAQifareleaseisunsuccessful
3304 https://nhsd-jira.digital.nhs.uk/browse/DOS-4074
3305 https://nhsd-jira.digital.nhs.uk/browse/DOS-4072
3306 https://nhsd-jira.digital.nhs.uk/browse/DOS-4070
3307 https://nhsd-jira.digital.nhs.uk/browse/DOS-4062
3308 https://nhsd-jira.digital.nhs.uk/browse/DOS-4051
3309 https://nhsd-jira.digital.nhs.uk/browse/DOS-3860
3310 https://nhsd-jira.digital.nhs.uk/browse/DOS-3990
3311 https://nhsd-jira.digital.nhs.uk/browse/DOS-4039
3312 https://nhsd-jira.digital.nhs.uk/browse/DOS-4047

2. Click the Release Link above and approve the manual step to deploy the release to prod

9.370.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3313]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.370.4 Roll-Back

- Redeploy:
  Release Tag: v2024.12.16.17.22+298407
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=298407&view=results

9.371 CHG0073262 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dfa9166503bd8aa1076462a4a85e45aa1%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.371.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

- https://nhsd-jira.digital.nhs.uk/browse/DOS-4081
- https://nhsd-jira.digital.nhs.uk/browse/DOS-4078
- https://nhsd-jira.digital.nhs.uk/browse/DOS-4075

9.371.2 Prepared Release

Release Tag: v2025.03.11.11.48+305915

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=305915&view=results

9.371.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

---

3313 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.371.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3314]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.371.4 Roll-Back

- Redeploy:
  Release Tag: v2024.12.16.17.22+304065
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=304065&view=results

9.372 CHG0073383 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8a35b0c1c3d42a580317da3c05013187

9.372.1 Tickets

Change created by ViniAkshara Kannan

- APM-6003[3315]
- APM-6005[3316]
- APM-6007[3317]
- APM-6008[3318]
- APM-6046[3319]

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.02.26.16.44...v2025.03.12.12.26

9.372.2 Prepared Release

Release Tag: v2025.03.12.12.26

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=306433&view=results

---

3314 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3315 https://nhsd-jira.digital.nhs.uk/browse/APM-6003
3316 https://nhsd-jira.digital.nhs.uk/browse/APM-6005
3317 https://nhsd-jira.digital.nhs.uk/browse/APM-6007
3318 https://nhsd-jira.digital.nhs.uk/browse/APM-6008
3319 https://nhsd-jira.digital.nhs.uk/browse/APM-6046

9.372.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.372.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3320]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.372.4 Roll-Back

- Redeploy:
    - Release Tag: v2025.02.26.16.44
      https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=299866

9.373 CHG0073656 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Deffdb3fa3bd8a6d076462a4a85e45abb%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.373.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

DOS-4086

DOS-4085

DOS-4083

DOS-4082

3320 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.373.2 Prepared Release

Release Tag: v2025.03.11.11.48+309188

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=309188&view=results

9.373.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.373.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3321]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.373.4 Roll-Back

- Redeploy:
  Release Tag: v2025.03.11.11.48+305915
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=305915&view=results

9.374 CHG0073902 - APIM monitoring-service-discovery Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db226bdb4c3ec6e100317da3c05013181%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D3%26sysparm_record_rows%3D765%26sysparm_record_list%3Dactive%253Dtrue%255Eshort_description%253E%253DapiM%255EORDERBYshort_description

9.374.1 Tickets

RFC was created by Vikas Nautiyal The Release implements the changes in the following tickets:

- Code changes for the following tickets

3321 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- APM-6124[3322] - monitoring-service-discovery repo remediations DONE

9.374.2 Prepared Release

Release Tag: v1.0.62-alpha

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312628&view=results

9.374.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.374.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3323]
2. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.374.4 Roll-Back

- Redeploy:
  - Release Tag:

9.374.4.1 Version v1.0.31-alpha

- Redeploy *last successful non-production release* (to all envs except production):

9.375 CHG0073957 - APIM Splunk Monitor Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=d2a35b34c32862500317da3c0501314f

9.375.1 Tickets

RFC was created by Sophie Clayton The Release implements the changes in the following tickets:

- Code changes for the following tickets

3322 https://nhsd-jira.digital.nhs.uk/browse/APM-6124?src=confmacro
3323 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

- APM-6127[3324] - splunk-monitor repo remediations DONE - resolves critical/high dependabot alerts for the Splunk Monitor Proxy

https://github.com/NHSDigital/splunk-monitor/compare/v1.0.36-alpha...v1.0.78-alpha

9.375.2 Prepared Release

Release Tag: v1.0.78-alpha

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312596&view=results

9.375.3 Release Process

1. Rerun the manual approval stage, then approve the prod release.

9.375.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3325]
    a. https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=prod&var-proxy=splunk-monitor-prod&var-product=All&var-revision=All&var-deployed_revision=All
2. Monitor APIM Dashboard for Splunk Monitor ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
    a. https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_all_sh_all_viz/search?q=search%20index%3D%22requests_apm_ptl%22%20%22proxy.suffix%22!%3D%22%2F_ping%22%20%22proxy.suffix%22!%3D%22%2F_status%22%20%22proxy.name%22%3D%22splunk-monitor-*%22&display.page.search.mode=verbose&dispatch.sample_ratio=1&earliest=1742515200&latest=1742601600&sid=1742977892.85407_D8960A32-7940-4CEA-9CC4-EB886C90239D
3. Call the /status endpoint directly to ensure a successful response

9.375.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.36-alpha
- Redeploy *last successful non-production release* (to all envs except production):

3324 https://nhsd-jira.digital.nhs.uk/browse/APM-6127?src=confmacro
3325 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.376 CHG0073626 - Immunisation FHIR API deployment to PROD

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D4b47f6fec31466d80317da3c0501314e%26sysparm_view%3Dtext_search

9.376.1 Tickets

CHG created by Saptarshi Mandal . The Release implements the changes in the following tickets:

AMB-2234	Software Up To Date
AMB-2289	Upgrade test suite - recordprocessor
AMB-2353	SEARCH: location is always obfuscated even if we have the values from v5
AMB-2355	ST: Convert FHIR JSON to flat JSON - Plugin the Flat JSON converter into IMMS API code - PART 1
AMB-2288	Upgrade test suite - filenameprocessor
AMB-2359	Recordforwarder tests Part - 2
AMB-2361	Recordforwarder Endpoint tests Part - 2
AMB-2062	Unify all the error responses across the board for Batch
AMB-2287	Unify all the error responses across the board for API
AMB-2315	Recordforwarder tests Part - 1
AMB-2342	Tech Debt - Remove Detailed Error Messages
AMB-2344	Recordforwarder Endpoint tests Part - 1
AMB-2382	Audit table/ file movement refactor

9.376.2 Prepared Release

Release Tag: **v1.0.2361-alpha+312042**

9.376.3 Release Process

1. **Pipeline Configuration**
   Azure pipelines for automatic deployment process.
   Ensure the pipelines are set to trigger on merges to the main/master branch.
2. **Build Pipeline**
   Trigger: Merge to main/master branch.
   Actions:
   Gather configuration from the GitHub repository.
   Build an artefact containing the necessary deployment files.
   Outcome: Successful build artefact creation.
3. **Release Pipeline**
   Trigger: Successful completion of the Build pipeline.
   Actions:

Use the artefact from the Build pipeline.
Deploy the configuration to Apigee and AWS.
Deploy the hosted backend or sandbox if applicable.
Outcome: Configuration deployed to Apigee and AWS.

9.376.3.1 Post Release Checks

1. **Smoke Testing**
   Conduct smoke tests using mock data to ensure the deployment is functioning correctly.
   Verify that all endpoints are responding as expected and there are no critical issues.
2. **Private Beta Phase**
   Upon successful smoke testing, transition the APIs to a private beta phase.
   Limit access to the RAV app within a specified region.
   Monitor performance and gather feedback from the RAV app users.

9.376.4 Roll-Back

- If needed deploy the **build_pipeline/v1.0.2108-alpha+281436v1.0.2108-alpha+281440** version to roll back.

9.377 CHG0073996 - Identity Service Updates

Link to Service Now Change[3326]

9.377.1 Tickets

RFC was created by ViniAkshara Kannan The Release implements the changes in the following tickets:

- Code changes for the following tickets
  - APM-6126[3327] - identity-service-api repo remediations DONE

9.377.2 Prepared Release

Release Tag: v1.0.1870-alpha[3328]

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312681

3326 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Da17f93cd3bece25076462a4a85e45ad2
3327 https://nhsd-jira.digital.nhs.uk/browse/APM-6126?src=confmacro
3328 https://github.com/NHSDigital/identity-service-api/releases/tag/v1.0.1870-alpha

9.377.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.377.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3329]
2. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.377.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.1821-alpha
- Redeploy *last successful non-production release* (to all envs except production):

9.378 CHG0074029 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D2801dcdd3b28665076462a4a85e45af7%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.378.1 Tickets

CHG created by Dhivakar Balasubramanian The Release implements the changes in the following tickets:

DOS-4105

DOS-4091

9.378.2 Prepared Release

Release Tag: v2025.03.11.11.48+312855

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312855&view=results

3329 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.378.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.378.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3330]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.378.4 Roll-Back

- Redeploy:
  Release Tag: v2025.03.11.11.48+309188
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=309188&view=results

9.379 CHG0074642 - APIM Sync Wrap Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=72daa6d9c378ead00317da3c0501311c

9.379.1 Tickets

Change created by Sophie Clayton. The Release implements the changes in the following tickets:

- APM-6136[3331] - sync-wrap repo remediations DONE - resolves critical/high dependabot alerts for the Sync wrap Proxy

https://github.com/NHSDigital/sync-wrap/compare/v1.0.126-alpha...v1.0.147-alpha

9.379.2 Prepared Release

Release Tag: v1.0.147-alpha

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=318024&view=results

3330 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3331 https://nhsd-jira.digital.nhs.uk/browse/APM-6136?src=confmacro

9.379.3 Release Process

1. Rerun the manual approval stage, then approve the prod release.

9.379.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3332]
    a. https://grafana.prod.api.platform.nhs.uk/d/api-details/api-details?orgId=1&var-env=prod&var-proxy=sync-wrap-prod&var-product=All&var-revision=All&var-deployed_revision=All
2. Monitor APIM Dashboard for Sync Wrap ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
    a. https://nhsdigital.splunkcloud.com/en-GB/app/nhsd_apm_public_all_sh_all_viz/health?form.index=summary_apm_prod_open&form.proxy=sync-wrap-prod&form.app=*&form.env=prod
3. Call the /_status endpoint directly to ensure a successful response

9.379.4 Roll-Back

- Redeploy:
    - Release Tag: v1.0.126-alpha
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=85523&view=results
- Redeploy *last successful non-production release* (to all envs except production):

9.380 CHG0074860 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=2b7ca8133bf46e1076462a4a85e45a23

9.380.1 Tickets

Change created by Sophie Clayton & Srikanth Thummala

- Added new practice for GPC
- APM-6128[3333] - api-management-infrastructure repo remediations - critical and some high DONE
  - dependabot updates

3332 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m
3333 https://nhsd-jira.digital.nhs.uk/browse/APM-6128?src=confmacro

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.03.12.12.26...v2025.04.10.09.25

9.380.2 Prepared Release

Release Tag: v2025.04.10.09.25[3334]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=320899&view=results

9.380.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Run APIM Infrastructure Build with release tag above
3. Run APIM Infrastructure Release with release tag above
4. Approve the manual step to deploy the release to prod

9.380.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3335]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.380.4 Roll-Back

- Redeploy:
  - Release Tag: v2025.03.12.12.26[3336]
    https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=306433&view=results

9.381 CHG0075022 - PDS Custom Rate Limit Update for IPlato

CHG0075022 | Change Request | NHS Digital (live)[3337]

3334 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.04.10.09.25
3335 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now
3336 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.03.12.12.26
3337 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Db1e0840cc3cdae106c39935c050131a4%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull%26sysparm_record_row%3D1%26sysparm_record_rows%3D1%26sysparm_record_list%3Dactive%253dtrue%255erequested_byDYNAMIC90d1921e5f510100a9ad2572f2b477fe%255eORsys_created_by%253djavascript%253ags.getUserID()%255eORDERBYDESCnumber

9.381.1 Tickets

CHG created by Srikanth Thummala . The Release implements the changes in the following rate limiting request:

INC0293321 | Incident | NHS Digital (live)[3338]

9.381.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

First check no rate limiting attribute is current set on the app:

```
curl https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/nhsd-
prod/apps/0331f48c-a9bc-49e9-a27b-c131d1d5966e/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' | jq
```

Should return:

```
{
  "detail": "app 0331f48c-a9bc-49e9-a27b-c131d1d5966e has no attribute with name
ratelimiting"
}
```

Add PDS rate limiting attribute (confirmed with demo 101):

```
curl -X PUT https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/
nhsd-prod/apps/0331f48c-a9bc-49e9-a27b-c131d1d5966e/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' -H 'content-type: application/
json' --data '
{
  "personal-demographics-prod": {
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 900,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": true,
      "ratelimit": "2700pm"
    }
```

3338 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/incident.do%3Fsysparm_tiny%3DqX25JfTQCzcI8qZW6PFBWbNzaWNu0xqC%26sys_id%3D8b5b0451c368e6500317da3c050131be%26sysparm_record_row%3D1

```
  }
}
'
```

9.381.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.381.3.1 Post Release Checks

1. Ensure the rate limiting custom attribute has been added to the app.
2. Monitor Grafana Prod dashboard[3339]
3. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
4. Verify with Demo 101 team

9.381.4 Roll-Back

- https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/0331f48c-a9bc-49e9-a27b-c131d1d5966e
- Click edit
- Remove the ratelimiting custom attribute
- Save

9.382 CHG0075379 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dbdc402f3c3892a506c39935c05013122%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

9.382.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

3339 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

dos-3931-modify-dockerfile.lambdas[3340]
dos-3931-lower-dnsdk-dotnet8-version[3341]
dos-3931-update-dnsdk-dotnet8[3342]
dos-3931-update-lambda-nuget-packages[3343]
dos-3925-cis2-login[3344]
DOS-3931-Upgrade_2_DotNet_8[3345]
DOS-3980-Update_nhs_frontend_version[3346]
dos-4112-link-to-new-assessment-tracker[3347]
dos-3994-pagenotfound[3348]
dos-4105-allow-relative-links-to-other-widgets[3349]

dos-4126-green-nhsuk-panel-fix[3350]

9.382.2 Prepared Release

Release Tag: v2025.04.17.10.43+327816

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=327816&view=results

9.382.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.382.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3351]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.382.4 Roll-Back

- Redeploy:
  Release Tag: v2025.03.11.11.48+312855

3340 https://nhsd-jira.digital.nhs.uk/browse/DOS-3931
3341 https://nhsd-jira.digital.nhs.uk/browse/DOS-3931
3342 https://nhsd-jira.digital.nhs.uk/browse/DOS-3931
3343 https://nhsd-jira.digital.nhs.uk/browse/DOS-3931
3344 https://nhsd-jira.digital.nhs.uk/browse/DOS-3925
3345 https://nhsd-jira.digital.nhs.uk/browse/DOS-3931
3346 https://nhsd-jira.digital.nhs.uk/browse/DOS-3980
3347 https://nhsd-jira.digital.nhs.uk/browse/DOS-4112
3348 https://nhsd-jira.digital.nhs.uk/browse/DOS-3994
3349 https://nhsd-jira.digital.nhs.uk/browse/DOS-4105
3350 https://nhsd-jira.digital.nhs.uk/browse/DOS-4126
3351 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312855&view=results

9.383 CHG0075648 - Mock JWKS Updates

Link to Service Now Change[3352]

9.383.1 Tickets

RFC was created by ViniAkshara Kannan The Release implements the changes in the following tickets:

- Code changes for the following tickets
  - APM-6143[3353] - mock-jwks repo remediations DONE

9.383.2 Prepared Release

Release Tag: v1.0.30-alpha[3354]

Pipelines Run: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=312681[3355]

9.383.3 Release Process

1. Rerun prod stage, review the pipeline and approve.

9.383.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3356]
2. Monitor APIM Dashboard for Identity Service ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.383.4 Roll-Back

- Redeploy:
  - Release Tag: v1.0.0
- Redeploy *last successful non-production release* (to all envs except production):

3352 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5c6446923b9d6a1876462a4a85e45a86%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3353 https://nhsd-jira.digital.nhs.uk/browse/APM-6143?src=confmacro

3354 https://github.com/NHSDigital/mock-jwks/releases/tag/v1.0.30-alpha

3355 https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=328156&view=results

3356 https://grafana.prod.api.platform.nhs.uk/d/4etIlBk7z/api-status?orgId=1&refresh=5m

9.384 CHG0075754 (N) & CHG0075915 (Er) - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Ddd0d847ac39d2ed00317da3c050131f6

The initial change wasn't successful as there was another target server (for Patient Care Aggregator Reporting) referring to the same keystore (cert), hence we have rolled back to the last successfully released tag in PROD v2025.04.10.09.25.

Since the Certificate expires on 12th May '25 (Monday), we planned to do a Emergency Change (details below) to separate out the references and release a new cert provided by the Patient Care Aggregator team without affecting the other service (i.e, no changes to Patient Care Aggregator Reporting).

New Emergency Change: CHG0075915[3357] **(9th May '25)**

- SCTASK0251652[3358]: New Secrets for the new Certificates
- SCTASK0251664[3359]: PR - https://github.com/NHSDigital/api-management-infrastructure/pull/1466/files

CHG0075918[3360] (TBD on 12th May '25): To rotate certs for the other service (Patient Care Aggregator Reporting) which is expiring on 12th May '25

9.384.1 Tickets

Change created by ViniAkshara Kannan

APM-6167[3361] - Certificate expiry - AWS prod cert prod/client/prod.patient-care-aggregator.com/crt
DONE

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.04.10.09.25...v2025.05.09.09.38

Emergency Change: https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.04.10.09.25...v2025.05.09.14.42

9.384.2 Prepared Release

Release Tag:

3357 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D8893e6d3c31dea140317da3c050131dd%26sysparm_view%3D%26sysparm_domain%3Dnull%26sysparm_domain_scope%3Dnull

3358 https://nhsdigitallive.service-now.com/sc_task.do?sysparm_tiny=qkWn5D6gvh0AFbxfCvmsvOD8SRf1BuYC&

3359 https://nhsdigitallive.service-now.com/sc_task.do?sysparm_tiny=c6Xn5VanAoDcn9Wa0HuqvwsYFzLlJN56&

3360 https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De2feee1bc3912e140317da3c050131d1%26sysparm_stack%3D%26sysparm_view%3D

3361 https://nhsd-jira.digital.nhs.uk/browse/APM-6167?src=confmacro

- Normal Change: v2025.05.09.09.38[3362]
- Emergency Change: v2025.05.09.14.42[3363]

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=332184&view=results

Emergency Change: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=332287&view=results

9.384.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.384.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3364]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.384.4 Roll-Back

- Redeploy:
  - Release Tag: v2025.04.10.09.25 (terraform), (apigee)
  - Replace the secrets with the old values
  - Run Infra build
  - Run infra release

9.385 CHG0075918 (E) - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3De2feee1bc3912e140317da3c050131d1%26sysparm_stack%3D%26sysparm_view%3D

3362 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.05.09.09.38
3363 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.05.09.14.42
3364 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.385.1 Tickets

Change created by ViniAkshara Kannan

APM-6167[3365] - Certificate expiry - AWS prod cert prod/client/prod.patient-care-aggregator.com/crt
DONE

Emergency Change: https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.05.09.14.42...v2025.05.12.12.32

9.385.2 Prepared Release

Release Tag: v2025.05.12.12.32[3366]

Emergency Change: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=333244&view=results

9.385.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.385.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3367]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

9.385.4 Roll-Back

- Redeploy:
  - Release Tag: v2025.05.09.14.42 (terraform), (apigee)
  - Replace the secrets with the old values
  - Run Infra build

3365 https://nhsd-jira.digital.nhs.uk/browse/APM-6167?src=confmacro
3366 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.05.12.12.32
3367 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

- Run infra release

9.386 CHG0076180 - Identity Custom Rate Limit Update for IPlato

CHG0076180 | Change Request | NHS Digital (live)[3368]

9.386.1 Tickets

CHG created by Valswyn Alfonso . The Release implements the changes in the following rate limiting request:

INC0307696[3369] | Incident | NHS Digital (live)[3370]

9.386.2 Prepared Release

Add rate limiting attribute to shared flow

Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

First check no rate limiting attribute is current set on the app:

```
curl https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/nhsd-
prod/apps/0331f48c-a9bc-49e9-a27b-c131d1d5966e/attributes/ratelimiting -H
'Authorization: bearer <your_apigee_access_token>' | jq
```

Should return:

```
{
   "personal-demographics-prod": {
      "quota": {"enabled": true, "interval": 1, "limit": 900, "timeunit": "minute"},
      "spikeArrest": {"enabled": true, "ratelimit": "2700pm"},
   }
}
```

Keep a backup of the above

Add rate limiting attribute for identity service:

```
curl -X PUT https://apigee-deep-rest.prod.api.platform.nhs.uk:9004/v1/organizations/
nhsd-prod/apps/0331f48c-a9bc-49e9-a27b-c131d1d5966e/attributes/ratelimiting/identity-
service-prod -H 'Authorization: bearer <your_apigee_access_token>' -H 'content-type:
application/json' --data '
```

3368 https://nhsdigitallive.service-now.com/nav_to.do?uri=change_request.do?sys_id=f99e011d3b216a5076462a4a85e45a66

3369 https://nhsdigitallive.service-now.com/incident.do?sysparm_tiny=a7UK7DkMtpMacEvCg6Mu7xKjETtyFfNU&sys_id=24ef5b3fc39d62540317da3c050131b8&sysparm_record_row=1

3370 https://nhsdigitallive.service-now.com/nav_to.do?uri=incident.do?sys_id=24ef5b3fc39d62540317da3c050131b8

```
{
    "quota": {
      "enabled": true,
      "interval": 1,
      "limit": 900,
      "timeunit": "minute"
    },
    "spikeArrest": {
      "enabled": true,
      "ratelimit": "2700pm"
    }
}
```

9.386.3 Release Process

1. Rate limiting shared flow#Addingcustomratelimitingforanindividualapplication (see page 1045)

9.386.3.1 Post Release Checks

1. Ensure the rate limiting custom attribute has been added to the app.
2. Monitor Grafana Prod dashboard[3371]
3. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
4. Verify with iPlato team

9.386.4 Roll-Back

- https://nhs-digital-prod.apigee.com/organizations/nhsd-prod/apps/details/0331f48c-a9bc-49e9-a27b-c131d1d5966e
- Click edit
- Replace with values from backup
- Save

9.387 CHG0076190 - Multicast Notifications Service

https://nhsdigitallive.service-now.com/nav_to.do?uri=task.do?sys_id=d9bb59513b296a5076462a4a85e45a09

3371 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.387.1 Tickets

CHG created by Alexander Carrie. The Release implements the changes in the following tickets:

- EM-1578
- EM-982
- EM-1579

https://github.com/NHSDigital/multicast-notification-service/compare/v1.0.268-alpha...v1.0.292-alpha

9.387.2 Prepared Release

Release Tag: **v1.0.292-alpha**

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=334677&view=results

9.387.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.387.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3372]
2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.387.4 Roll-Back

- Redeploy:
  - **v1.0.268-alpha**
  - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=279590&view=results
- Redeploy *last successful non-production release* (to all envs except Production)

9.388 CHG0076569 - Digital Onboarding Service

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D5e20a5e7c36522506c39935c050131ad

3372 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.388.1 Tickets

CHG created by Nigel Gossage The Release implements the changes in the following tickets:

DOS-4115-Code_cleanup_for_old_AT
DOS-4127-Cannot_add_Assessor_Approver
dos-4135-update-dos-guidance-link
dos-4134-add-shared-section-feature-toggle
dos-4061-create-ui-tests-for-assessment-tracker
dos-4166-target-specific-version-of-nhsuk-frontend

9.388.2 Prepared Release

Release Tag: v2025.04.17.10.43+338212

Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=338212&view=results

9.388.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.388.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3373]
2. Monitor APIM Dashboard for ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.388.4 Roll-Back

- Redeploy:
  Release Tag: v2025.04.17.10.43+327816
  Link to Azure: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=327816&view=results

3373 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.389 CHG0076652 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3D499bfa73c36926506c39935c050131e8%26sysparm_stack%3D%26sysparm_view%3D

9.389.1 Tickets

Change created by ViniAkshara Kannan

APM-6162[3374] - Certificate expiry - AWS prod cert prod/client/health-research-data-catalogue.api.service.nhs.uk/crt DEV IN PROGRESS

https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.05.12.12.32...v2025.05.21.14.26

9.389.2 Prepared Release

Release Tag: v2025.05.21.14.26

TBD: https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=333244&view=results

9.389.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Update crt in prod and ptl AWS with new values
3. Run APIM Infrastructure Build with release tag above
4. Run APIM Infrastructure Release with release tag above
5. Approve the manual step to deploy the release to prod

9.389.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3375]
2. Monitor APIM Dashboards ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions
3. Check new cert details have been updated in the Apigee TLS Keystore

3374 https://nhsd-jira.digital.nhs.uk/browse/APM-6162?src=confmacro
3375 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

9.389.4 Roll-Back

- Redeploy:
    - Release Tag: v2025.05.12.12.32[3376] (terraform), (apigee)
    - Replace the secrets with the old values
    - Run Infra build
    - Run infra release

9.390 CHG0076937 - APIM Infrastructure Updates

https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/change_request.do%3Fsys_id%3Dbc0442073bfdae1067201da985e45a96%26sysparm_record_target%3Dchange_request%26sysparm_record_row%3D1%26sysparm_record_rows%3D10%26sysparm_record_list%3Dactive%253Dtrue%255Eassigned_to.name%253E%253Dvikas%255EORDERBYassigned_to

9.390.1 Tickets

CHG created by Alex Carrie. The Release implements the changes in the following tickets:

- APM-6193[3377] - Shared flow to support Single ASID model DONE

Github comparison of changes - https://github.com/NHSDigital/api-management-infrastructure/compare/v2025.05.12.12.32...v2025.06.02.11.06

9.390.2 Prepared Release

Release Tag: v2025.06.02.11.06

https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=342829&view=results

9.390.3 Release Process

1. Ensure the APIM Infrastructure Release preceding this, has been successfully deployed
2. Click the Release Link above and approve the manual step to deploy the release to prod

9.390.3.1 Post Release Checks

1. Monitor Grafana Prod dashboard[3378]

3376 https://github.com/NHSDigital/api-management-infrastructure/releases/tag/v2025.05.12.12.32
3377 https://nhsd-jira.digital.nhs.uk/browse/APM-6193?src=confmacro
3378 https://grafana.prod.api.platform.nhs.uk/d/api-status/api-status?orgId=1&refresh=10s&from=now-1m&to=now

2. Monitor APIM Dashboard for XXXXXXX ~15mins post-release to verify traffic and error levels are consistent with pre-release conditions

9.390.4 Roll-Back

- Redeploy:
    - v2025.05.12.12.32
    - https://dev.azure.com/NHSD-APIM/API%20Platform/_build/results?buildId=133838&view=results

10 Help and support

10.1 Supporting your API delivery

10.1.1 Self-service support

Before reaching out for help, first explore the API producer zone (see page 119) as this will provide the answer to many common queries, helping you deliver your API.

This is a wiki, and we encourage you to help us make it as useful as possible, so please, please updated it if there's something missing.

10.1.2 Support via Slack

We always have two people on first line support to help API producers.

Contact them via NHS Digital Slack **#platforms-apim-producer-support** channel at https://nhsdigital-platforms.slack.com/archives/G016JRWN6AY[3379].

We rotate who is on support every 2 weeks, and other team members work on API platform improvements, often based on feedback we get from you.

To give feedback, when in the channel, click the 'Give us your feedback' short cut via the lightening bolt in the message window. This will send you a link to a feedback form which takes 1-2 minutes to complete and is invaluable to improving and maintaining our process.

For access, see Get access to Slack and support channels (see page 151).

10.1.3 Service notifications via Slack

We issue service notifications (issues, outages, new features) via Slack using the **#platforms-apim-producer-announcements** channel at https://nhsdigital-platforms.slack.com/archives/G01BNHAC1E2.

For access, see Get access to Slack and support channels (see page 151).

10.1.4 Feature requests

If you want us to add a feature to the API platform, contact us via Slack as per above.

10.1.5 Architecture support

If you have design queries or need some architecture guidance on APIs, contact us via Slack as per above.

3379 https://app.slack.com/client/TRK2Y2T1U/C016JRWN6AY

If you don't have an architect within your own team or pillar, you can make a request for architectural resource by filling in the following form https://hscic365.sharepoint.com/architecture/Pages/Architecture-Request-Forms.aspx[3380]

10.1.6 FHIR API payload design support from IOPS

If you need help designing your FHIR API payload, you could ask the IOPS team to help - see Is my API a FHIR API? Who are IOPS (see page 288)?.

10.2 Live service support for your API

The general support model is:

1. You support your own API
2. We support you (just like a platform team should!)

As per above, you support your own API in production.

If there's an issue you can't fix, or an issue with the API platform, we'll step in.

10.2.1 Service level agreements (SLAs)

We are a platinum service so we do have full 24x7 support.

You'll need to decide for yourselves what level of support you'll provide for your API. See Service levels for APIs (see page 288) for more details.

10.3 Support for API consumers

By default, we provide first line support via our Developer Hub help & support[3381] page. The developer community[3382] also provides alternative way to post queries, and is a good source of information for frequently asked questions and common issues.

Optionally, you can publish an email address to support your API consumers directly. If you do, include it at the end of the Onboarding section of your API specification.

Either way, don't forget to tell us where to forward any email questions we can't handle because we use your email address as second line support. Let us know if you use different email addresses, for example for onboarding and technical support questions.

3380 https://gbr01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fhscic365.sharepoint.com%2Farchitecture%2FPages%2FArchitecture-Request-Forms.aspx&data=05%7C01%7Cdaniela.simonato1%40nhs.net%7C09d26b15aa494ed73bd608db7e2dde87%7C37c354b285b047f5b22207b48d774ee3%7C0%7C0%7C638242507688095465%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=iuYTSOVmAJxaCbKu37%2F%2FLRcgjFpHEC1YjKWWaA8WQP8%3D&reserved=0

3381 https://digital.nhs.uk/developer/help-and-support

3382 https://developer.community.nhs.uk/

11 API Producer Workflow Requests

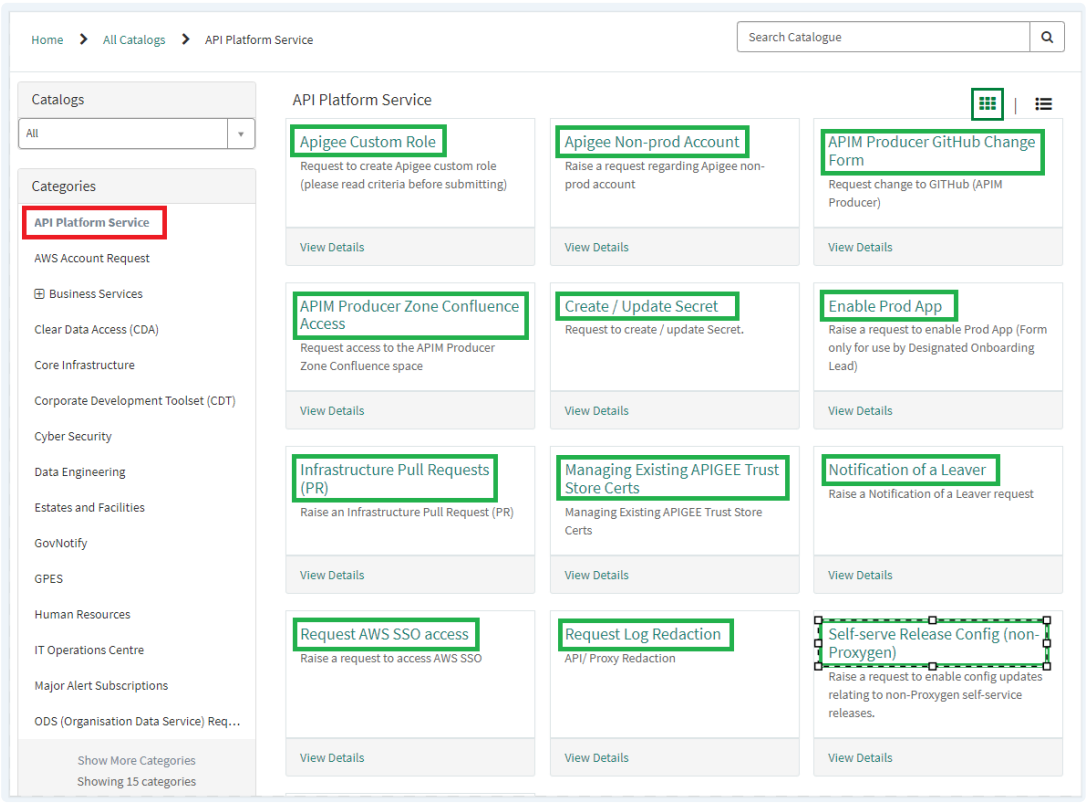
APIM are making changes to SLACK workflows within Producer and Consumer support channels.

All SLACK workflows should be logged via ServiceNOW from 16th October 2024.

We will be making changes to the workflows to point users to raise these requests via the appropriate ServiceNOW forms which are available under the Service catalogue :

API Platform Service: https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_category&catalog_id=-1.

https://www.support.internalservices.england.nhs.uk/nhs_digital?id=sc_category&sys_id=7496f0c81b1c9610772fa756b04bcbf7&catalog_id=-1&spa=1



The following workflows will be actioned via ServiceNOW from **16th October 2024**. Existing Slack Workflows will be retired/repointed to ServiceNOW forms

- **APIM Producer Zone Confluence Access**
- **APIM Producer GitHub Change Form**
- **Create / Update Secret**
- **Infrastructure Pull Requests (PR)**
- **Update/Add custom attributes**
- **Request Ratelimiting change**

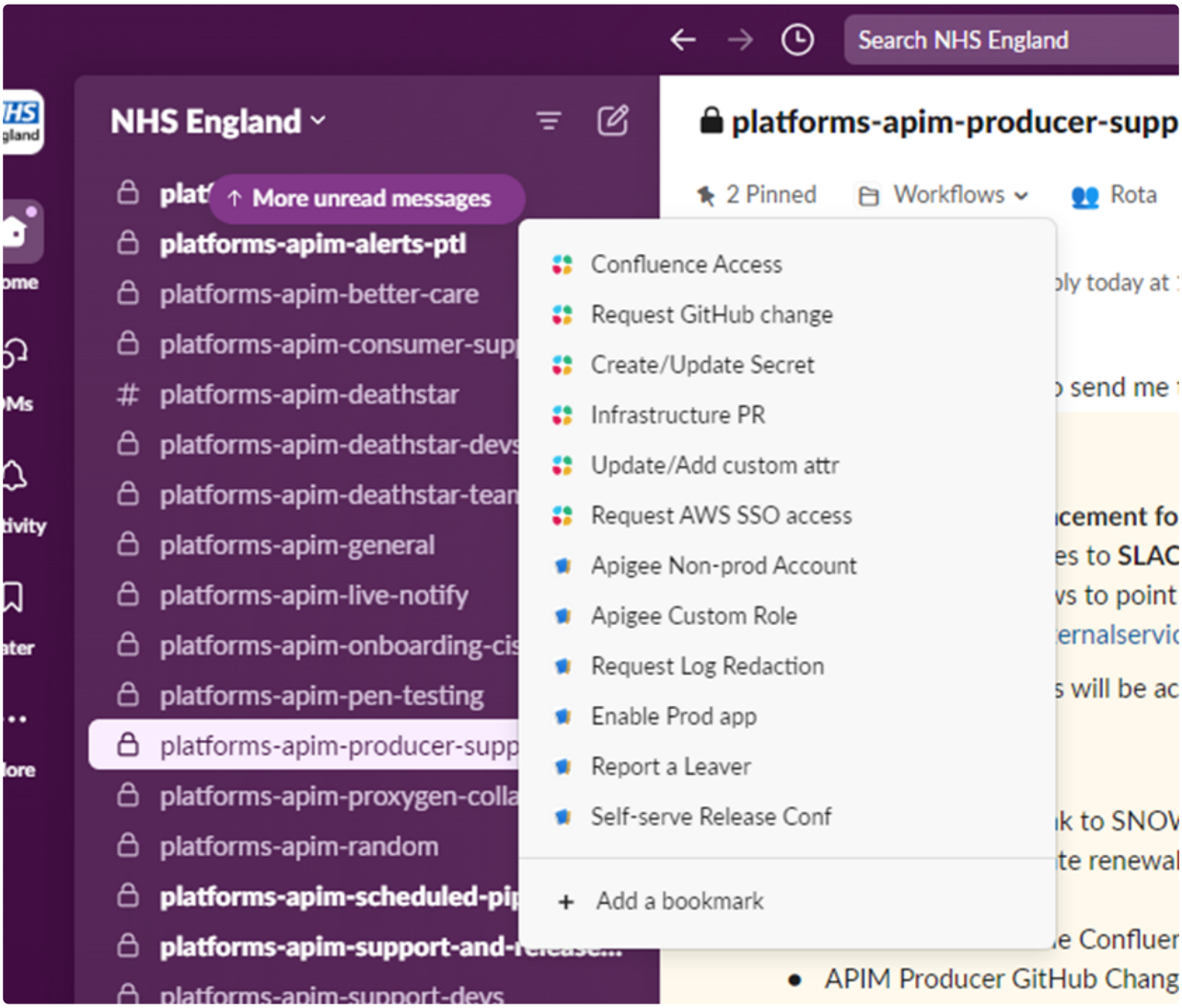
- **Request AWS SSO access**
- **Apigee Non-prod Account**
- **Apigee Custom Role**
- **Enable Prod App**
- **Self-serve Release Config (non-Proxygen)**
- **Remaining certificate management workflows**

The APIM Producer Support Slack channel will continue to provide users with Technical support which cannot be addressed through the self-serve documentation available on confluence.

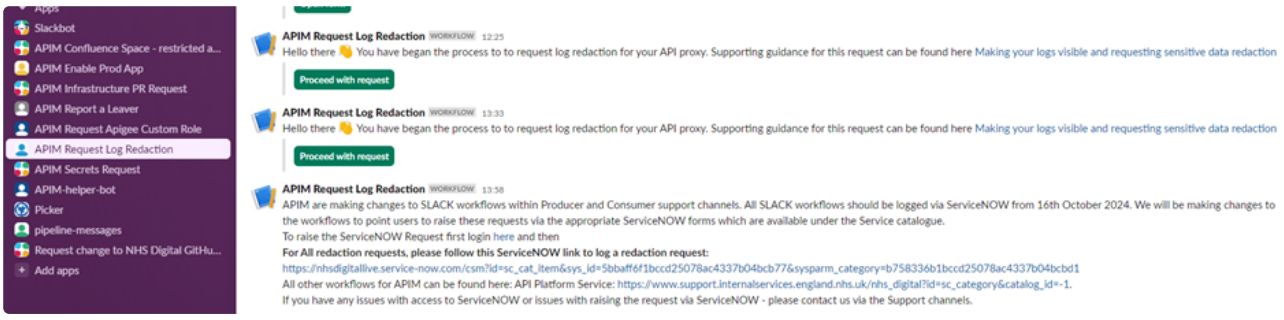
11.1 Redaction Request New Process using Service Now

For Producer

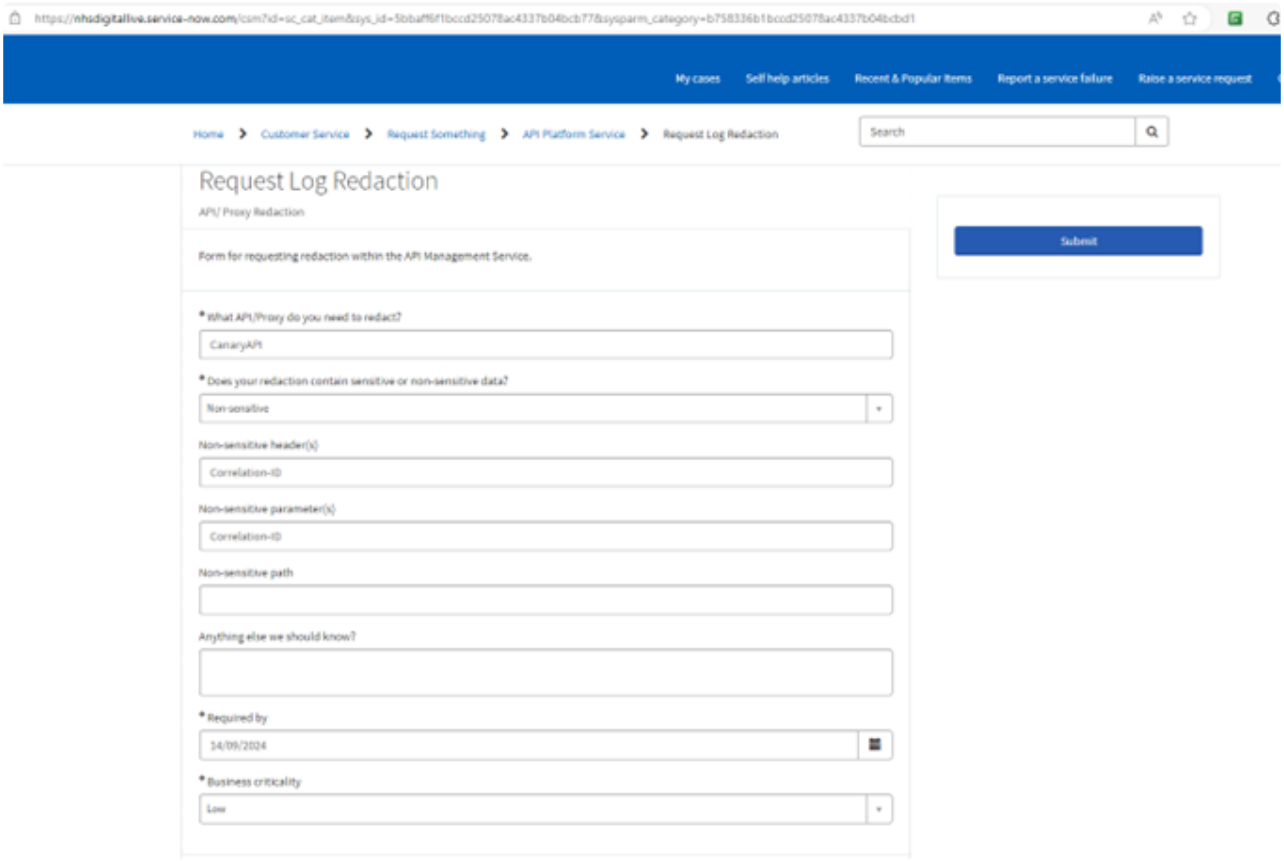
Step1:- Go to Slack Channel platform-apim-producer-support channel and run Request Log Redaction Workflow.



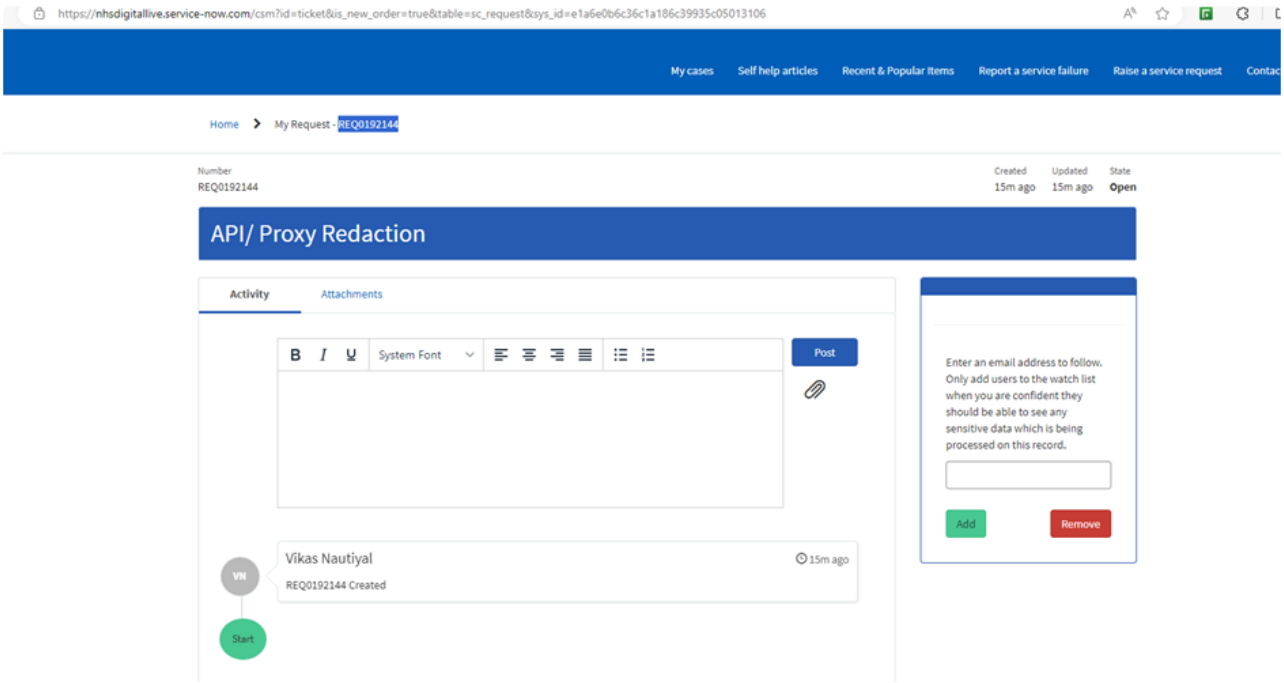
Step2:- It will create a message in the APP below with the new Service now link. Please follow the instructions.



Step3:- Once coming to the service now form Please fill the below as per below to test the functionality and click on Submit Button

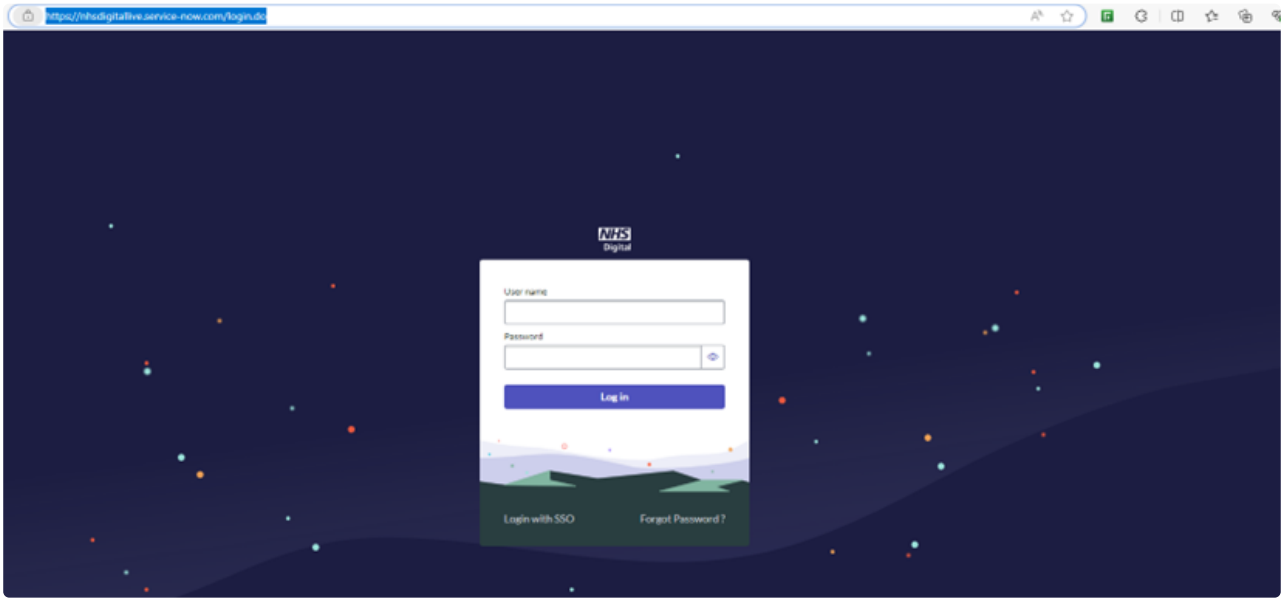


After submit you will see below Screen and submitted request will automatically assigned to ITO API Platform (DeathStar Team)

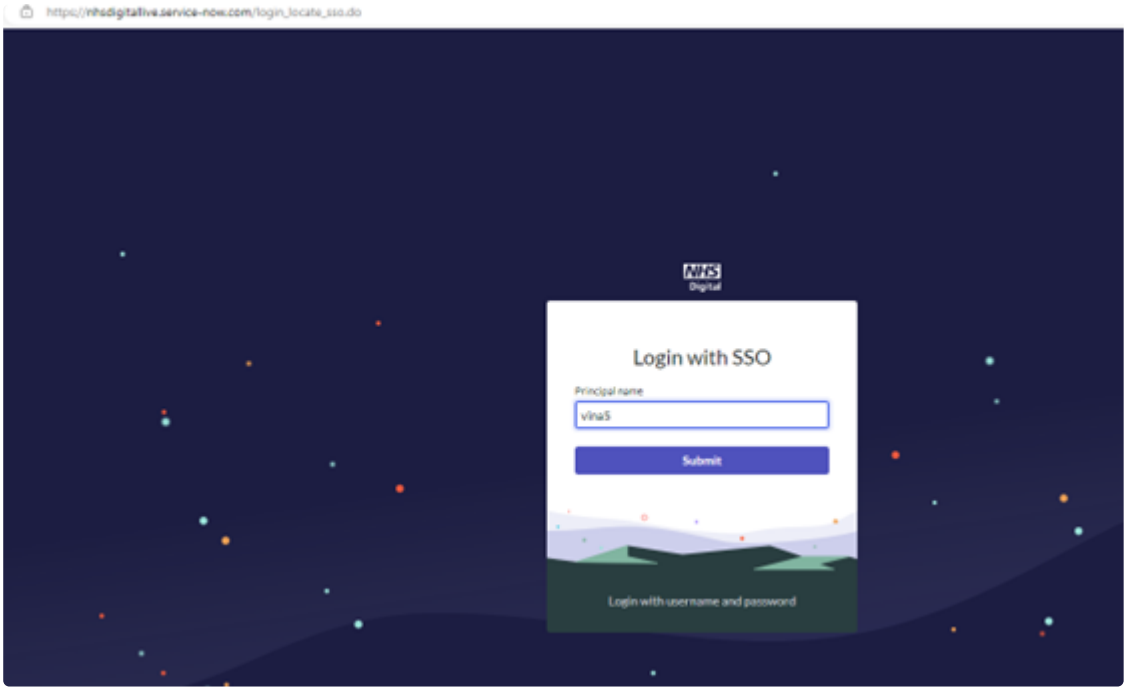


For APIM-Deathstar Team

Step4:- Login into https://nhsdigitallive.service-now.com/login.do and click Login with SSO

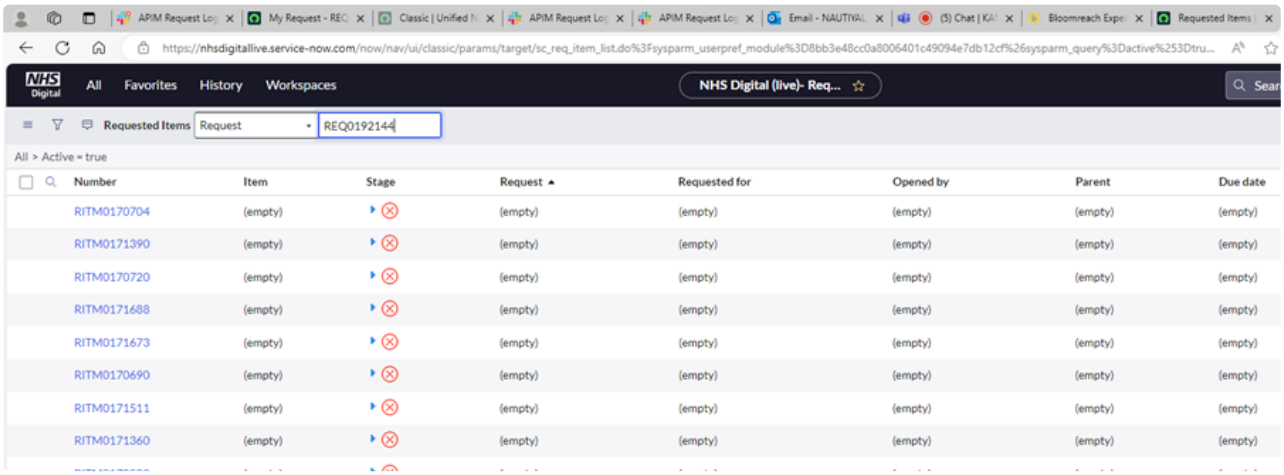


Step5:- Pass your ShortCode and submit

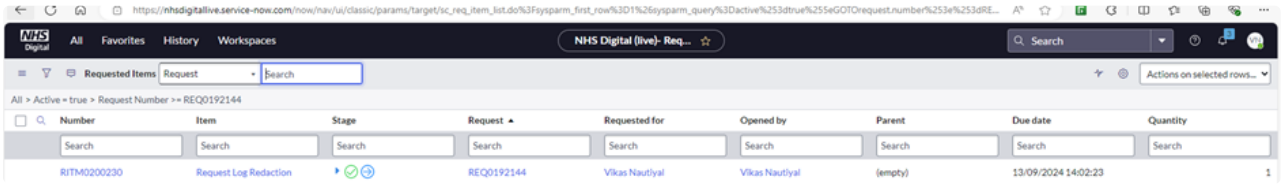


Step6:- go to this link and search for Request Created in Step3

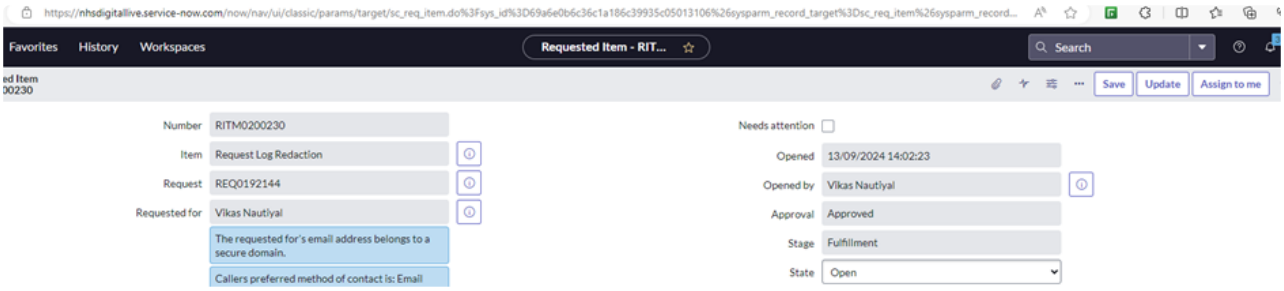
https://nhsdigitallive.service-now.com/now/nav/ui/classic/params/target/sc_req_item_list.do%3Fsysparm_userpref_module%3D8bb3e48cc0a8006401c49094e7db12cf%26sysparm_query%3Dactive%253Dtrue%255EEQ



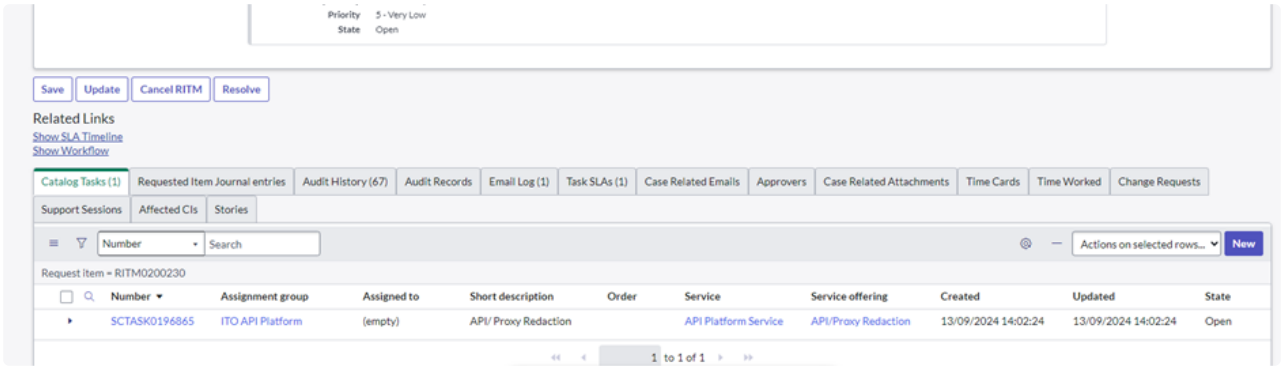
Once search is completed you will find the entry for your request click on RITM



Step7:- Click on Assign To me or the person you want to Assign from ITO API Platform

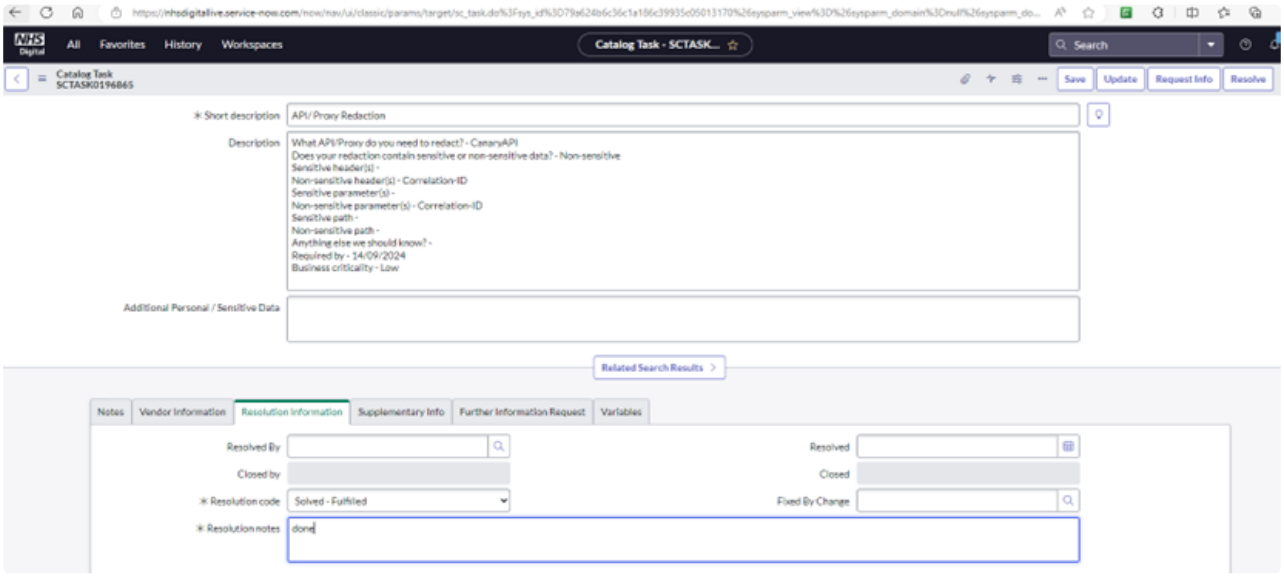


Scroll down in the page and click on Task

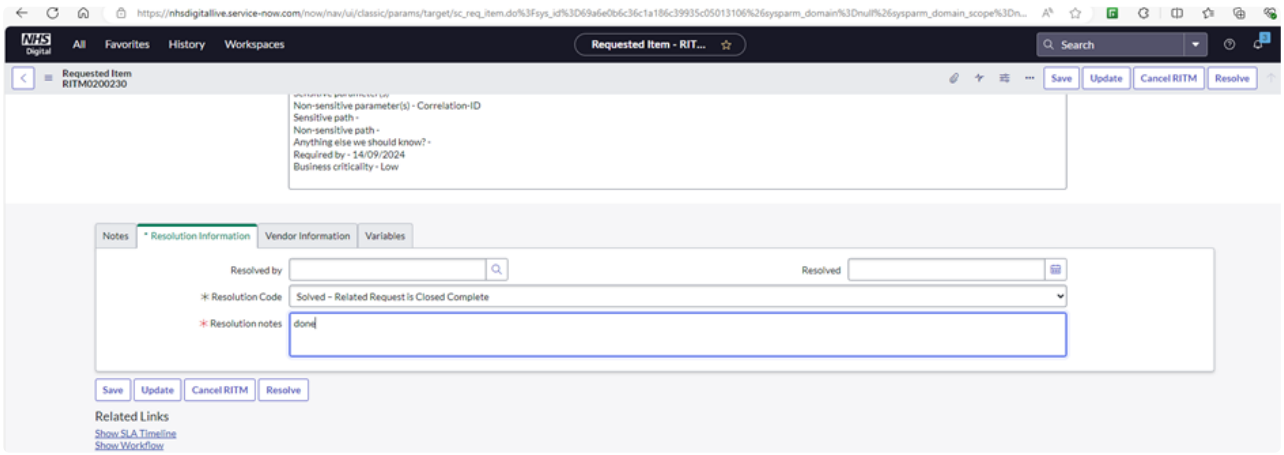


First Assign the task to Yourself

Once the task is completed add Resolution Code and Resolution Notes and click Resolve.



Come back to RITM (Requested Item) fill resolution code and resolution notes and click on Resolve. The entire process of Redaction Request is completed.



12 API Producer Team - Key contacts:

Last updated 15 May 2025

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-59	Address Finder API - OS Places API Spec	mike.jones25@nhs.net[3383]						
AMR-353	Address Quality Service	milo.cowell1@nhs.net[3384] alex.kolesnikov1@nhs.net[3385]						
AMR-4	Ambulance Data Submission	paul.rees-walker@nhs.net[3386]						
AMR-378	Central Metadata Hub API - To enable automated updates to and consumption from the Single Point of Reference	sonia.khan5@nhs.net[3387] ananya.banerjee6@nhs.net[3388]						
AMR-375	Clinical Prescription Tracking Service Fhir API	john.kitson1@nhs.net[3389]						

3383 mailto:mike.jones25@nhs.net
3384 mailto:milo.cowell1@nhs.net
3385 mailto:alex.kolesnikov1@nhs.net
3386 mailto:paul.rees-walker@nhs.net
3387 mailto:sonia.khan5@nhs.net
3388 mailto:ananya.banerjee6@nhs.net
3389 mailto:john.kitson1@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-351	Connecting Party Manager	warran.mavrodaris2@nhs.net[3390] rowan.gill1@nhs.net[3391]						
AMR-360	Custom Prescription Status Update - Non-FHIR standard linked to AMR-358	john.kitson1@nhs.net[3392]						
AMR-346	Digital Genomics Test Service API	ravi.natarajan2@nhs.net[3393]						
AMR-374	Dispensing FHIR API (Proxygen) - Enables the use of the Electronic Prescription Service (EPS) for FHIR Dispensing Suppliers	john.kitson1@nhs.net[3394]						
AMR-108	EeRS (Eyecare e-Referral Service)	tony.marsh1@nhs.net[3395]						
AMR-341	Elective Waiting List	jordan.azam1@nhs.net[3396]						

3390 mailto:warran.mavrodaris2@nhs.net
3391 mailto:rowan.gill1@nhs.net
3392 mailto:john.kitson1@nhs.net
3393 mailto:ravi.natarajan2@nhs.net
3394 mailto:john.kitson1@nhs.net
3395 mailto:tony.marsh1@nhs.net
3396 mailto:jordan.azam1@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-15	Electronic Prescription Service (EPS) FHIR facade [dispense]	john.kitson1@nhs.net[3397]						
AMR-12	Electronic Prescription Service (EPS) FHIR facade [prescribe]	john.kitson1@nhs.net[3398]						
AMR-383	Eligibility Signposting API (Vaccinations)	athena.arvaniti@nhs.net[3399] john.hadfield1@nhs.net[3400]						
AMR-16	e-Referral Service (eRS)	tony.marsh1@nhs.net aubyn.crawford1@nhs.net adam.oldfield1@nhs.net[3401]						
AMR-278	e-Referrals Service Patient Care API	tony.marsh1@nhs.net[3402] nila.surti3@nhs.net[3403]						
AMR-380	Find the Right Service Ingress API	aileen.tracey1@nhs.net[3404] eugenia.hewitt1@nhs.net[3405]						

3397 mailto:john.kitson1@nhs.net
3398 mailto:john.kitson1@nhs.net
3399 mailto:athena.arvaniti@nhs.net
3400 mailto:john.hadfield1@nhs.net
3401 mailto:tony.marsh1@nhs.net
3402 mailto:tony.marsh1@nhs.net
3403 mailto:nila.surti3@nhs.net
3404 mailto:aileen.tracey1@nhs.net
3405 mailto:eugenia.hewitt1@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-389	Find the Right Service Search – Integrated Search API	katherine.odoherty1@nhs.net[3406] j.pearce1@nhs.net[3407] oliver.lewis@nhs.net[3408]						
AMR-371	Genomic Data Access and Management API	graham.ritchie@genomicsengland.co.uk[3409] christopher.boustred@genomicsengland.co.uk[3410]						
AMR-338	Genomics order communications API	ravi.natarajan2@nhs.net[3411]						
AMR-128	GP Connect - Access Record [STU3]	steven.hunt2@nhs.net[3412]						
AMR-114	GP Connect Patient Facing Services (PFS) - Access Record API	ramsey.baker@nhs.net[3413]						

3406 mailto:katherine.odoherty1@nhs.net
3407 mailto:j.pearce1@nhs.net
3408 mailto:oliver.lewis@nhs.net
3409 mailto:graham.ritchie@genomicsengland.co.uk
3410 mailto:christopher.boustred@genomicsengland.co.uk
3411 mailto:ravi.natarajan2@nhs.net
3412 mailto:steven.hunt2@nhs.net
3413 mailto:ramsey.baker@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-335	GP Connect Patient Facing Services (PFS) - Prescriptions API	ramsey.baker@nhs.net[3414]						
AMR-334	GP Connect Patient Facing Services (PFS) - User Permissions	ramsey.baker@nhs.net[3415]						
AMR-342	GP connect update record	ramsey.baker@nhs.net[3416]						
AMR-390	GP Connect Access Record Structured API	colin.mackenzie5@nhs.net[3417] steven.hunt2@nhs.net[3418] ramsey.baker@nhs.net[3419]						
AMR-57	GP Registrations Management Information REST API	amy.hayer2@nhs.net[3420]						
AMR-116	Health Research Data Catalogue (Collibra HRD UK Integration)	ketan.italiya@nhs.net[3421] laura.sato@nhs.net[3422]						

3414 mailto:ramsey.baker@nhs.net
3415 mailto:ramsey.baker@nhs.net
3416 mailto:ramsey.baker@nhs.net
3417 mailto:colin.mackenzie5@nhs.net
3418 mailto:steven.hunt2@nhs.net
3419 mailto:ramsey.baker@nhs.net
3420 mailto:amy.hayer2@nhs.net
3421 mailto:ketan.italiya@nhs.net
3422 mailto:laura.sato@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-376	Healthcare Worker API - management and access to Healthcare Worker data	catherine.ejiogu1@nhs.net[3423] aubyn.crawford1@nhs.net[3424]						
AMR-105	Immunisation FHIR API	matthew.pattison1@nhs.net[3425] david.moore52@nhs.net[3426]						
AMR-62	Immunisation History	matthew.pattison1@nhs.net[3427] james.spirit@nhs.net[3428]						
AMR-340	Multicast Notificaton Service	d.gorgees@nhs.net[3429] alexander.carrie1@nhs.net[3430]						
AMR-359	National Document Repository	danielle.anderson18@nhs.net[3431] fran.gillert1@nhs.net[3432] noga.sasson1@nhs.net[3433]						

3423 mailto:catherine.ejiogu1@nhs.net
3424 mailto:aubyn.crawford1@nhs.net
3425 mailto:matthew.pattison1@nhs.net
3426 mailto:david.moore52@nhs.net
3427 mailto:matthew.pattison1@nhs.net
3428 mailto:james.spirit@nhs.net
3429 mailto:d.gorgees@nhs.net
3430 mailto:alexander.carrie1@nhs.net
3431 mailto:danielle.anderson18@nhs.net
3432 mailto:fran.gillert1@nhs.net
3433 mailto:noga.sasson1@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-387	National Elective Waiting List Information and Contact Service API (NEWLICS)	daniel.lynagh4@nhs.net[3434] john.arnfield@nhs.net[3435]						
AMR-326	National Record Locator Cloud	elizabeth.williams96@nhs.net matt.dean3@nhs.net[3436]						
AMR-18	NHS App (notifications)	thomas.pedley1@nhs.net[3437]						
AMR-65	NHS Booking and Referral	carl.death1@nhs.net[3438] leigh.goslin1@nhs.net[3439]						
AMR-384	NHS Cyber Reporting API	jim.mcdonald@nhs.net[3440] jean.whitehead1@nhs.net[3441]						
AMR-332	NHS Notify (previously Communications Manager)	amina.abdurrahman1@nhs.net[3442] laura.mountain@nhs.net[3443] tom.droza2@nhs.net[3444]						

3434 mailto:daniel.lynagh4@nhs.net
3435 mailto:john.arnfield@nhs.net
3436 mailto:elizabeth.williams96@nhs.net
3437 mailto:thomas.pedley1@nhs.net
3438 mailto:carl.death1@nhs.net
3439 mailto:leigh.goslin1@nhs.net
3440 mailto:jim.mcdonald@nhs.net
3441 mailto:jean.whitehead1@nhs.net
3442 mailto:amina.abdurrahman1@nhs.net
3443 mailto:laura.mountain@nhs.net
3444 mailto:tom.droza2@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-386	NHS Notify Supplier API	michael.houston4@nhs.net[3445] ross.buggins@nhs.net[3446] karen.craddock1@nhs.net[3447]						
AMR-327	NHS Website Content API	igba.ujege1@nhs.net[3448] angela.carabasu@nhs.net[3449] hannah.aina@nhs.net[3450]						
AMR-9	Organisation Data Terminology API- [ODS Strategic]	kelly.ball8@nhs.net[3451] Sophie.obrien6@nhs.net[3452] phil.nott@nhs.net[3453]						
AMR-227	Patient care aggregator API (Wayfinder)	des.kelleher2@nhs.net aubyn.crawford1@nhs.net[3454]						
AMR-355	Patient Care Aggregator Reporting	sarah.wraith@nhs.net des.kelleher2@nhs.net[3455]						

3445 mailto:michael.houston4@nhs.net
3446 mailto:ross.buggins@nhs.net
3447 mailto:karen.craddock1@nhs.net
3448 mailto:igba.ujege1@nhs.net
3449 mailto:angela.carabasu@nhs.net
3450 mailto:hannah.aina@nhs.net
3451 mailto:kelly.ball8@nhs.net
3452 mailto:Sophie.obrien6@nhs.net
3453 mailto:phil.nott@nhs.net
3454 mailto:des.kelleher2@nhs.net
3455 mailto:sarah.wraith@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-372	Patient Data Manager (Initially) to allow Patient Care Aggregator to enrich appointments returned from PEPs	fraser.stewart5@nhs.net[3456] alan.parker8@nhs.net[3457]						
AMR-356	Patient Flags (Reasonable Adjustments)	nina.tombs1@nhs.net[3458] david.fletcher17@nhs.net[3459]						
AMR-13	Personal Demographic Service (PDS) [application restricted access]	claudiu.dumitrascu1@nhs.net[3460]						
AMR-2	Personal Demographic Service (PDS) [healthcare worker access]	sarah.wraith@nhs.net kitty.shen@nhs.net[3461]						
AMR-14	Personal Demographic Service (PDS) [patient access]	sarah.wraith@nhs.net kitty.shen@nhs.net[3462]						

3456 mailto:fraser.stewart5@nhs.net
3457 mailto:alan.parker8@nhs.net
3458 mailto:nina.tombs1@nhs.net
3459 mailto:david.fletcher17@nhs.net
3460 mailto:claudiu.dumitrascu1@nhs.net
3461 mailto:sarah.wraith@nhs.net
3462 mailto:sarah.wraith@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-388	Personal Health Profile (PHP) API	steven.walton6@nhs.net[3463] john.galloway2@nhs.net[3464]						
AMR-373	Prescribing FHIR API (Proxygen) - API to access the Electronic Prescription Service (EPS)	john.kitson1@nhs.net[3465]						
AMR-358	Prescription Status Update	john.kitson1@nhs.net[3466]						
AMR-344	Prescriptions for patients	john.kitson1@nhs.net[3467]						
AMR-363	Register with a GP Surgery API	josh.lilly1@nhs.net[3468] james.clark56@nhs.net[3469]						
AMR-381	Related Person Network	shabequl.islam1@nhs.net[3470] kat.hobdell@nhs.net[3471] aleksandra.bak2@nhs.net[3472]						

3463 mailto:steven.walton6@nhs.net
3464 mailto:john.galloway2@nhs.net
3465 mailto:john.kitson1@nhs.net
3466 mailto:john.kitson1@nhs.net
3467 mailto:john.kitson1@nhs.net
3468 mailto:josh.lilly1@nhs.net
3469 mailto:james.clark56@nhs.net
3470 mailto:shabequl.islam1@nhs.net
3471 mailto:kat.hobdell@nhs.net
3472 mailto:aleksandra.bak2@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-250	Service Search (Directory of Health Care Services) API	conrad.easton@nhs.net [3473]						
AMR-43	Signing Service (remote)	leannegarland@nhs.net allison.loose1@nhs.net[3474]						
AMR-7	Spine Directory Service (SDS) [GPIT Futures]	owain.williams5@nhs.net rowan.gill1@nhs.net[3475] steven.walton6@nhs.net[3476]						
AMR-347	Structured Summary Care Record	nina.tombs1@nhs.net[3477] david.fletcher17@nhs.net[3478]						
AMR-8	Summary Care Record (SCR) [GP IT Futures]	shelley.johnston@nhs.net[3479] robert.jordan2@nhs.net[3480]						

3473 mailto:conrad.easton@nhs.net
3474 mailto:leannegarland@nhs.net
3475 mailto:owain.williams5@nhs.net
3476 mailto:steven.walton6@nhs.net
3477 mailto:nina.tombs1@nhs.net
3478 mailto:david.fletcher17@nhs.net
3479 mailto:shelley.johnston@nhs.net
3480 mailto:robert.jordan2@nhs.net

API Ref	API Name	API Contact	Date of Ownership confirmed	SLA	OOO support	Service Now link	Related service	Ave vols 7 Days
AMR-385	UEC EPS-DoS API	katherine.odoherty1@nhs.net[3481] j.pearce1@nhs.net[3482] matas.kaucikas1@nhs.net[3483]						
AMR-348	Validated Relationships	jonny.ayling1@nhs.net[3484] ellis.allan2@nhs.net[3485]						

3481 mailto:katherine.odoherty1@nhs.net
3482 mailto:j.pearce1@nhs.net
3483 mailto:matas.kaucikas1@nhs.net
3484 mailto:jonny.ayling1@nhs.net
3485 mailto:ellis.allan2@nhs.net

13 Release Management

Now that we are in production, we are subject to NHSD change management processes. This document outlines the components it covers and the approach we are taking, as well as listing the change requests we have raised for when

13.1 Components

The components of the APIM Tribe eco system that could get changed or have new updates released to are:

- Apigee - API Proxy
- Apigee - Shared flows
- Apigee - Apps
- OAS files published on Bloomreach (no RFC needed)
- APIM - Authentication Service
- Infrastructure
- Infrastructure Key Value Maps
- APIM AWS Hosted Container
    - Incl. Sync Wrap service
- Digital Onboarding Service
- Developer Identity (Keycloak)

13.2 Approach and when is an RFC needed

Releases are split into two types:

- Zero down-time
- Potential impact during release

At this time, both API Proxy changes and most APIM AWS Hosted container changes have been shown to be zero-down time. However, some proxy changes also need an infrastructure change - and this means it needs an RFC and to be out of hours:

- Tuesday and Thursday evening, commencing at 11pm

✅ Spec updates (in the OAS file) do not need an RFC - once the change reaches INT, it is auto-deployed to Production Bloomreach and is visible on NHS Digital API Catalogue

Status	Component	RFC Needed?	In Hours?	Notes
Non-live	API Proxy	❌	✅	
Non-live	APIM AWS Hosted Container	❌	❓	Examine Diffs to confirm
Non-live	Infrastructure	✅	❌	
In Use	Apigee App	❌	❓	See below for detailed notes
In Use	API Proxy	✅	❓	For high volume - out of hours only, in case other dependencies cause disruption
In Use	APIM AWS Hosted Container	✅	❓	
In Use	Infrastructure	✅	❌	

Impact and scenarios:

- Updates to a **shared flow** do cause a small outage for proxies that use the shared flow
    - 11th Jan 2023 - agreed with Steven Walton that these would **not** be a standard change. With the move to more in-hours deployments, this would still need to be out of hours
- AWS is fully approved, but there (currently) no production usage. As it is approved, AWS infrastructure changes do require an RFC.
- API Proxy deployments **can** require a Target Server:
    - This requires an infrastructure deploy, if the deploy is **add only**, then it is safe to do this in-hours

    - The Diff functionality of the pipelines provides a quality gate for the implementer to check and evidence that it is safe
- Feature toggles:
    - Depending on the feature toggle, it might not require an RFC - it will need to be assessed on each change (e.g. enabling an unused feature is generally a safe option).
    - Normally, to ensure that the feature toggle remains in the correct state after the next release, it **does require** an infrastructure code change and an associated release.
    - Under emergency conditions, the change can be made manually, and if no RFC is needed could be in hours

13.2.1 Making changes to Apigee Product and API Consumer Apps

Once a service is fully live then:

- No RFC (where a JIRA ticket is logged from the API Producer), in hours changes can be done if:
    - Simple enable / disable of a Product(s), or changing of custom attributes
    - If there is no production traffic on the App - and the change is tied to a specific AppID (i.e. no risk of applying the changes to the wrong App)
    - Where there is flowing traffic - then automated changes to custom attributes (e.g. using a build agent to update Rate limiting shared flow (see page 1039)) where an AppID is required
- If the situation is different to those listed:
    - For low volume Apps, then - as long as the API producer is aware then out of hours with no RFC is suitable
    - Higher volume apps might require an out of hours standard RFC
    - Ultimately, where the change is limited to a single App, it is the API Producer to determine risk and appropriate Governance
        - It might transpire that in-hours is the lower risk scenario

IMPORTANT: Is there a risk that manual changes or automated changes (e.g. wrong AppID copied into the automated change script) could cause an outage? If these can be identified, an approach can be agreed

13.3 How to raise an RFC

Thing to do	resource
Raise an RFC for approval in ServiceNow	How to raise a release RFC change request in ServiceNow (see page 2080)
Change the date of an RFC	How to raise a release RFC change request in ServiceNow (see page 2088)

Thing to do	resource
Ensure you have a release plan entered in to the release schedule	Release Schedule (see page 1390) there is a handy checklist that helps here: Raising RFCs - Cheatsheet (see page 2091)
Is it a Spine related release?	Check how Spine releases are done: How Spine times releases[3486]
Attend a release co-ordination session to help the person releasing your change to understand it	Speak to Steven Walton to join this

13.4 How to raise a release RFC change request in ServiceNow

This document will detail how to raise a change request RFC ticket in ServiceNow for a release.

Before You Begin
You will need:

- A Release No. for the version of the release you intend to deploy, this would normally correspond to the azure pipelines build no, e.g. `#v1.2.1043-beta+36888`
- A Release Plan, that covers dependencies, out-of-band update process, like KVM changes, and impacted services that may need redeploys
- A preferred Release Date and Time, which should be 11pm on either a Tuesday or Thursday

These items will make it easier to complete and submit the RFC
Generally an RFC takes about 2 full business days to go through all approvals (as a guideline)
When you have an RFC ready - you will need to join the Release schedule:

- Apigee/platform releases - see Release Schedule (see page 1390)
- Spine releases - see When are our releases[3487]

13.4.1 The RFC Ticket Process

A release RFC consists of a type of ticket, a change request, raised in ServiceNow. This change request ticket passes through various states, such as review and implementation. Before the deployment of a release into production, the proposed changes detailed in the Change Request must be approved by various governing

3486 https://nhsd-confluence.digital.nhs.uk/display/SPINE/When+are+our+releases
3487 https://nhsd-confluence.digital.nhs.uk/spaces/SPINE/pages/105570711/When+are+our+releases

boards. Once approved, the deployment can be performed at the designated time by a member of the On-Call team.

13.4.2 How to raise a release RFC in ServiceNow

This guide will cover how to raise a change request ticket in ServiceNow. The official guidance for raising and completing an RFC can be found here: NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[3488]

There are guides for ServiceNow on the intranet. For more information see ServiceNow Change Management User Guide v2.01[3489]

(General NHSDigital Sharepoint ServiceNow Training and Support[3490])

13.4.2.1 Login to ServiceNow

- Log in to ServiceNow[3491] using your shortcode

13.4.2.2 Creating a new Change Request

Using the menu bar on the left, scroll to find 'change' OR use the menu search bar to search for 'change'

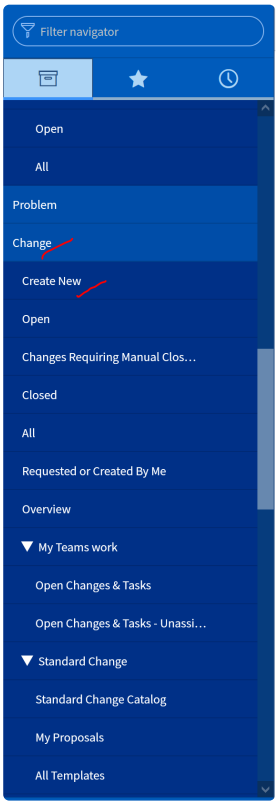
Click on 'Create New'

3488 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

3489 https://hscic365.sharepoint.com/:w:/r/sites/ServiceNowImplementation/_layouts/15/Doc.aspx?sourcedoc=%7B10F3A1C4-3E03-4DC3-B6C1-3FC88927B9D6%7D&file=ServiceNow%20Change%20Management%20User%20Guide%20v2.01.docx&action=default&mobileredirect=true

3490 https://hscic365.sharepoint.com/sites/ServiceNowImplementation/SitePages/ServiceNow-Training-and-Support-Portal.aspx

3491 https://nhsdigitallive.service-now.com/nav_to.do?uri=%2F$pa_dashboards_overview.do



13.4.2.3 Selecting the type of Change Request

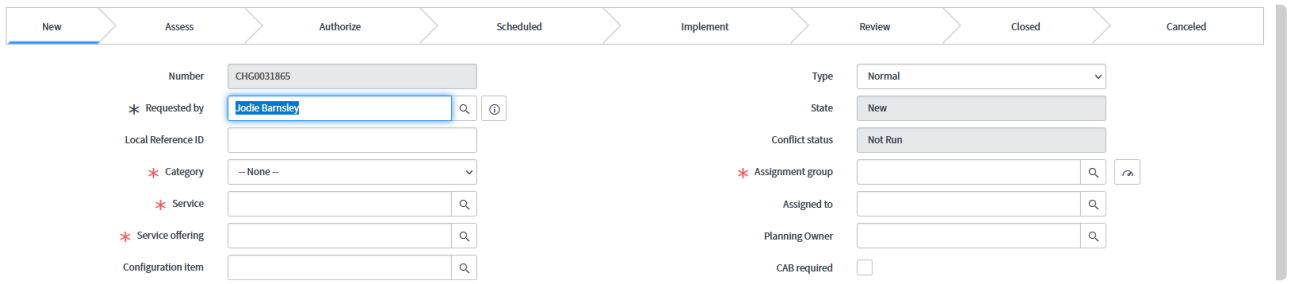
You will be presented with 3 choices:

- **Create Normal Change:** 'Normal changes' are a general purpose change type that requires one or more approvals.
- **Standard**: Select from available preapproved change templates. These changes do not require approval.
    - See Standard Changes (see page 2093) guidance for raising a new standard change template, or for raising a change using an existing template, KOP-049 Raising an RFC using a Standard Change Template[3492] .
- **Create Emergency Change:** Emergency changes are a change type that must be implemented as soon as possible.

13.4.2.4 Filling in the Request

In the top section of the form complete:

3492 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template



13.4.2.5 Select a Category

Usually 'Platforms'

13.4.2.6 Select the Service

Usually 'API Platform Service'

13.4.2.7 Select the Service offering

Usually 'API Platform'

13.4.2.8 Select an Assignment group

Usually 'ITO API Platform'

If possible, assign to the person on 2nd line support for the planned day of release. You can find this by looking at On-call rota (Out of hours - 1800 - 0800)[3493].

13.4.2.9 Configuration Item

Use the search tool to select the more relevant item for your changes. (Typically this will be API Platform/ API/DOS, generally an areas that fits your change step and has a support group of ITO ITOC Techops)

(Only required if marked as mandatory / not always required)

13.4.2.10 Select Risk level

Options: High, Moderate, Low

3493 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/153365587/On-call+rota+Out+of+hours+-+1800+-+0800

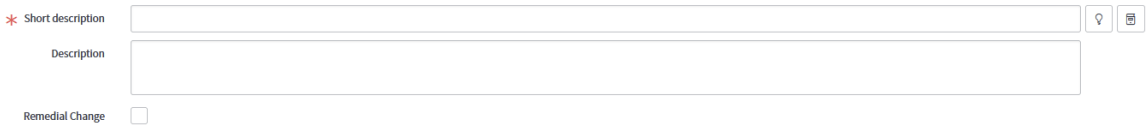
13.4.2.11 Select Impact level

Options: 1 - High, 2 - Moderate, 3 - Low

See NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[3494] for the impact level criteria

13.4.2.12 Retrospective Change

Tick this box if this change is being raised after the change has been implemented / started. This will allow a retrospective date to be entered in Scheduled start / end fields



13.4.2.13 Provide a short description (title) for the Change Request

A title should be provided in the 'Short description' field

As we are trying move to templated Standard Change Request tickets, we need to demonstrate to Service Management that we can perform regular deployments consistently. We need to gather evidence of this. So ideally we want common change request tickets to follow a consistent naming pattern. The naming pattern for your Change Request should be:

> *APIM <service-name> Updates*

Examples include:

- APIM Identity Service Updates
- APIM PDS Updates
- APIM Ambulance Analytics Updates
- APIM Utils Updates

13.4.2.14 Enter a Description

Write a description of the changes being made, also should include a link to the ticket for an in-depth view of the change in 'Description'.

Example Description:

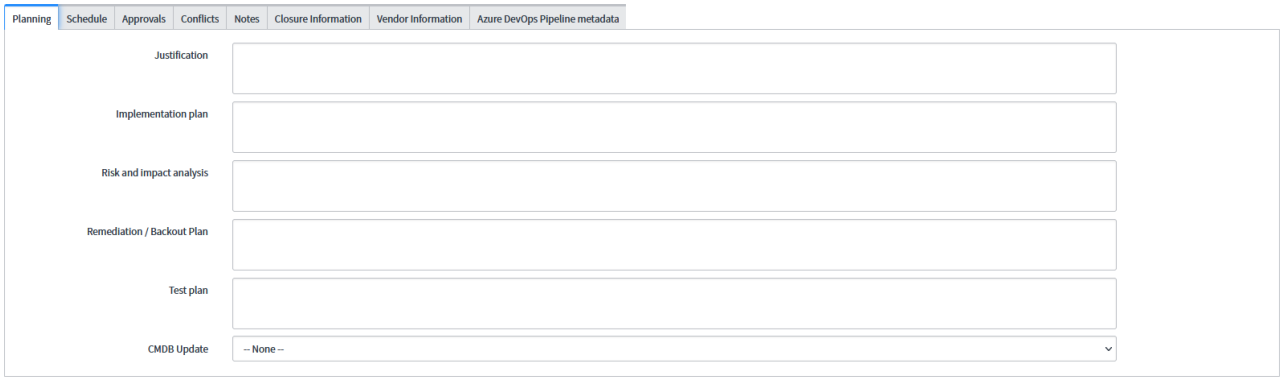
3494 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

```
This release covers the following tickets:

- JIRA-001
- JIRA-002
- JIRA-003

All production releases have a corresponding release plan: [insert link from https://
nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans]

All production releases follow our documented release KOPs, found here: https://nhsd-
confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
and here: https://nhsd-confluence.digital.nhs.uk/display/APM/
KOP-019+Release+process+-+Individual+API

Rollbacks in case of issues during the release process are detailed here: https://
nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
```

13.4.2.15 Remedial Change

Tick this box if change has been raised to resolve an issue in an earlier failed change

13.4.2.16 Planning Tab



Justification

Enter a justification for the release - for example: a small description, reference to epic name, details of provided fix/feature etc.

Implementation plan

The standard implementation plans can be found here, ensure you have read KOP-018[3495] and KOP-019[3496]. Make reference to these in this section.

3495 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
3496 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API

Example:

> "We will follow our standard process for making changes on the night:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
>
> and use this KOP for actually making the update:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API"

Risk and impact analysis

- Enter any proposed risks that your change might bring if issues/errors were to arise.
- The impact of the risk if they were to occur.
- The impact of the change as a whole, normally based on things such as chance of service outage.
    - Example: "Impact is very low - no change to any item that's currently in use."
- What have you done to mitigate the chance of this risk happening? I.e. testing throughout development.
    - Example: "Full automated test coverage for this change and it has been manually tested in PTL environments."

Further information and templated can be found here NHSE Digital Operations & Service Management - CPI Enablement - RFC Completion and Quality Guidance.docx[3497]

Remediation/ backout plan

The standard backout plan can be found here, ensure you have read KOP-016[3498] and **KOP-017**[3499]. Make reference to these in this section.

Example:

> "We will follow our KOP for rolling back:
>
> https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release"

Test plan

How can it be tested to check the release has correctly processed. i.e. Check Grafana dashboard to check service live? Look for your changes in live? ping and status endpoints, smoke test endpoints

Example:

> "Login to Apigee portal and check to see that the 'Assign message policy - remove external scopes' exists.
>
> Check Grafana dashboards to make sure all services are still running as expected.
>
> Check Splunk logs lo ensure we can still see transactions being processed successfully."

CMDB update

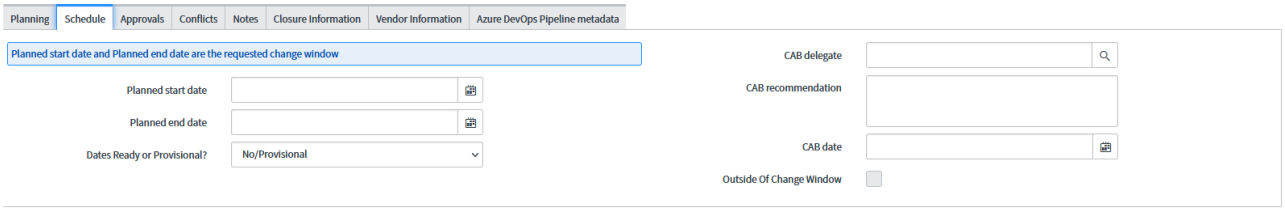
Usually 'Not Required'

3497 https://nhs.sharepoint.com/:w:/r/sites/X26_liveservices/RelMgmnt/_layouts/15/Doc.aspx?sourcedoc=%7BB6A5C623-258E-406E-9DAE-BA9B4EFDE988%7D&file=NHSE%20Digital%20Operations%20&%20Service%20Management%20-%20CPI%20Enablement%20-%20RFC%20Completion%20and%20Quality%20Guidance.docx=&action=default&mobileredirect=true

3498 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config

3499 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release

13.4.2.17 Schedule Tab

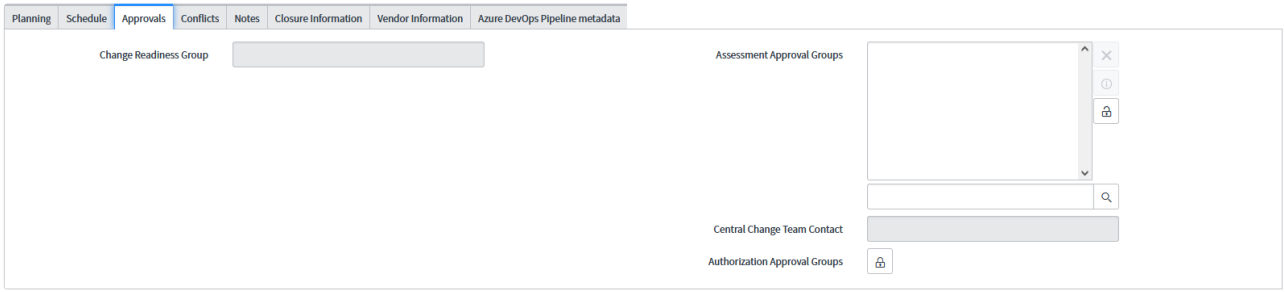


The 'planned start date' (and time) of the release is usually 23:00 on the day of release specified.

For the 'planned end date' (and time), specify 00:00 of the day after release. (An one hour window from start to end)

'Dates ready or provisional?' If you are ready to release your code (just awaiting approval) please select Yes/Ready, otherwise select No/Provisional.

13.4.2.18 Approvals Tab



13.4.2.19 Change readiness group

This field is a non-editable field. It will be auto-populated post change submission with change readiness group of the selected service offering / change readiness group of selected service.

13.4.2.20 Assessment approval group

This is an optional field. It will be auto-populated with assessment approval groups of the selected service offerings / service , if present. This field can also be used by the change team to add additional groups for assessment approvals.

13.4.2.21 Azure DevOps pipelines metadata tab

You may choose to include Azure DevOps Pipeline metadata here (as well as in the Confluence Release Plan).

This (currently) is not required.

13.4.2.22 Next steps

Submit to save and submit the change request.

13.4.2.23 Approvals

You will be emailed as approvals are actioned and these can be viewed on the change request in ServiceNow.

13.4.3 Changing the date of release

Once an RFC is approved, APIM cannot change the release date. The Change Management team can do a change, contact them:

- Use the group Teams channel for Change Management - CentralChangeTeams
- or shared mailbox - liveservices.central-changemgmt@nhs.net[3500]

13.5 In-hours Releases

Due to changing personnel within the team and a changing API landscape on the platform, we're exploring ways of releasing much more in-hours. Which we all agree, across the tribe and the API producer squads we've spoken to, is a better approach. However, it brings into focus some challenges that we need to overcome:

defining what can/can't be released in-hours;

communicating out to the producer announcements channel; and

letting everyone on the rota know what's what?

who will make the releases?

do we need to keep the release plans in Confluence?

Initial workshop held 21 Feb 2023

Agenda:

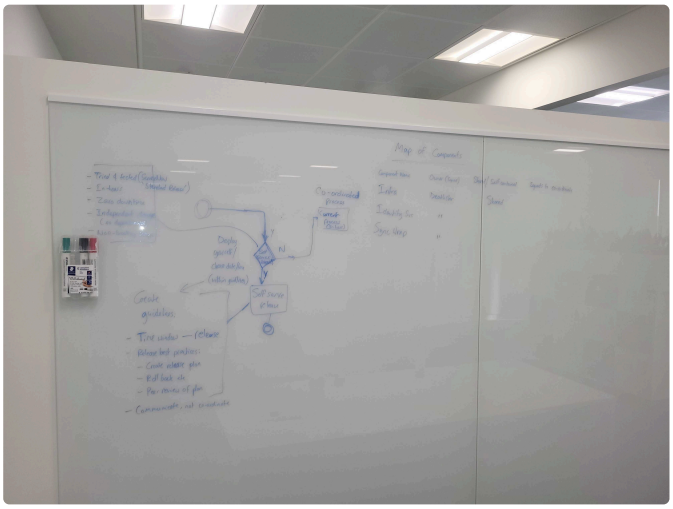
1. Criteria for releasing during the day;
2. Who actually makes the releases?
3. Permissions required to release;
4. Release scheduling – am/pm slots?
5. Release plans to make sure rollback etc. is thought about? Reviewed by a peer (inside / outside the squad?);
6. External review of release plan?

3500 mailto:liveservices.central-changemgmt@nhs.net

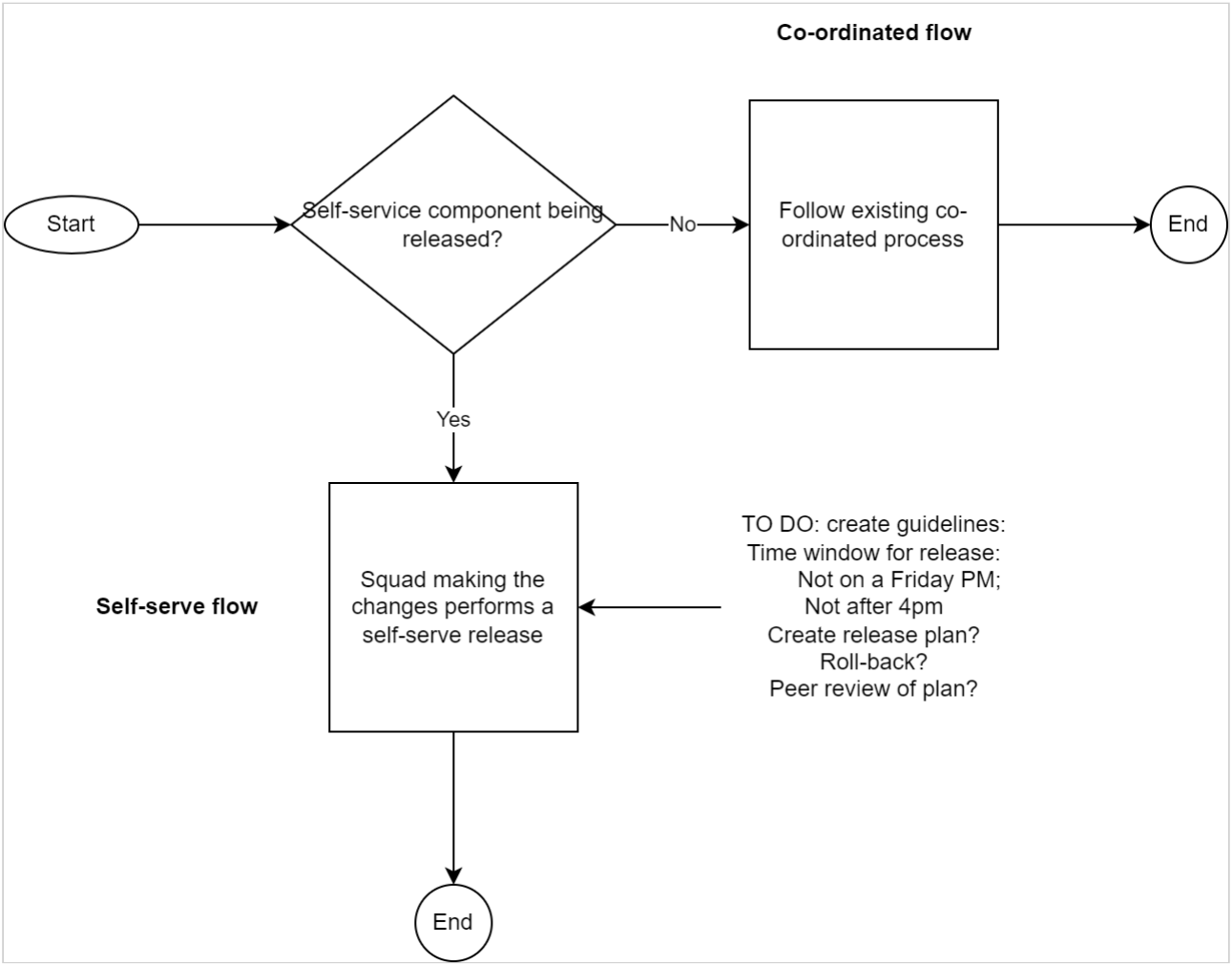
7. Notification process – is the way we broadcast releases ok or not? How do we make sure people on out of hours support know that a release has happened? And
8. AoB.

Notes

1. Criteria for releasing during the day;
    a. Criteria for not releasing during the day – default is to release during the day and only use an out of hours release in special circumstances - TO DO: draft criteria for an out of hours release
2. Who actually makes the releases?
    a. General agreement on the principle that it's best to have the squad doing the work make the release as often as possible. However, there are two scenarios that we must consider:
        i. Individual component that doesn't affect any services outside the ownership of the releasing squad;
        ii. Shared component that affects multiple squads
    b. TO DO: Map out components: who owns them; who can make the release; who needs to be informed
3. Permissions required to release;
    a. Enable Producer teams to decide who can press the button; locked behind AD groups – current process in a KoP to request access via ResolveIT
4. Release scheduling – am/pm slots?
    a. Fully self-contained release…no need for a co-ordination call
    b. Shared components – still have a co-ordination call
    c. Matrix - Two tier process:
    d. Use ServiceNow to pull out what releases are happening when
    e. Retain the Tuesday/Thursday schedule for now to gradually move to in-hours releases
5. Release plans to make sure rollback etc. is thought about? Reviewed by a peer (inside / outside the squad?);
6. External review of release plan?
7. Notification process – is the way we broadcast releases ok or not? How do we make sure people on out of hours support know that a release has happened? And
    a.
8. AoB.



Component Name	Owning Squad	Shared / Self-contained	Qualify for in-hours?	Squads to co-ordinate with	Squads to notify
Infrastructure Service	Deathstar	Shared			
Identity Service	Deathstar	Shared			
Sync-wrap	Deathstar	Shared			
Keycloak					



13.6 Raising RFCs - Cheatsheet

Page status

***update 10/03/21**: Ben Davies has also put some important stuff on* How to raise a release RFC change request in ServiceNow (see page 2080) *- we need to de-dupe*

13.6.1 When is an RFC needed

It is required when you change...

1. a production system
2. a system that has had any live external user, ever

3. a system other API Management squads are using

13.6.2 High level checklist

- [ ] Pipelines checked to ensure build is successful, including all environments
- [ ] Are you running an APIM, or a Spine release plan - it affects all of the steps/links below, the rest of this process will refer to APIM releases
- [ ] Change summary written up, and this checklist has been copied in to a new, nested page in Release Plans[3501]
- [ ] KOPs (Key Operating Procedures[3502]) used wherever possible - the standard implementation plans are KOP-018[3503] and KOP-019[3504]; standard rollback plans are KOP-016[3505] and KOP-017[3506]
- [ ] RFC raised in ServiceNow - see How to raise a release RFC change request in ServiceNow (see page 2080)
- [ ] Release has been scheduled within the Release Schedule (see page 1390) page
- [ ] The dev/tech lead has attended the "release co-ordination" session (organised by Steven Walton , usually these are Tues/Thurs 12pm) to explain what needs to be done
- [ ] Ensure RFC has been fully signed off - and marked off as signed off on Release Plans[3507]

13.6.3 Good advice

- KOPs 16/17/18/19 are frequently relevant
- Deployments should be brought through and grouped according to how you would want them rolled back, i.e. if one bit fails, do you want some of the change to remain, or does it *all* need to roll back?
    - Nesting them within the confluence instructions is an excellent approach
    - Include the "RFC codes" at every point
- RFC sign off can be slowed by Solutions Assurance if you have not signed off each part of the development with your squads resident SA person, get them to agree it, THEN submit it
- Your confluence instructions need to be explicit for the out of hours support - if they do not understand it, it WILL be rolled all the way back, so tell them what to do in all known eventualities
    - consider if you want anything to happen if part of the change does not go as intended

13.6.4 How to fill in an RFC in Service

- How to raise a release RFC change request in ServiceNow (see page 2080)

3501 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans
3502 https://nhsd-confluence.digital.nhs.uk/display/APM/Key+Operating+Procedures
3503 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-018+Releasing+changes+to+Production
3504 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-019+Release+process+-+Individual+API
3505 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-016+Back-up+and+recover+Apigee+config
3506 https://nhsd-confluence.digital.nhs.uk/display/APM/KOP-017+Rolling+back+to+a+stable+release
3507 https://nhsd-confluence.digital.nhs.uk/display/APM/Release+Plans

13.7 Standard Changes

NHSD have a fast-track RFC process known as 'standard change' that allows you to create pre-approved RFCs for specific components. There are criteria around what type of components you can create standard changes for, namely:

- they must be for a specific component; and
- you must have made 3 successful full RFC releases for that component without introducing any service incidents.

13.7.1 Raising a Standard Change

To raise a Standard Change follow the instructions using KOP-049 Raising an RFC using a Standard Change Template[3508].

13.7.2 How to create Standard Change Templates

- On ServiceNow find one of the changes that you wish to make into a Standard that has completed successfully. The change must be a known, repeatable method. It must be low risk and have low impact.
- Using the left hand menu open the 'Service Catalog'
- Use the search box in the top right to search for 'Template Management'
- Open the result titled 'Propose a new Standard Change Template'
- Complete the form (Template description, Category, Sample Change Requests, Change Request Values). Reminder this is a template so be as generic as possible so it can be used as the basis for each new occurrence of that change.
- Once completed save and click 'Request Approval'
- The Change Team will add on the usual change approvers and ask them if they are happy to approve the change to be a Standard.
- Once approved the change template will be available to use via the Standard Change Catalog.
- To raise a standard change from the template, open the template.
- Make changes, add planned start and end date etc.
- Submit
- The change has now been created with no approvals needed.
- When the change has completed then enter the actual date and times of the start and finish of the change along with whether it was successful or not and move along to the next stage (I haven't completed these but think it is Resolution?).
- If a change is unsuccessful then the change ceases to be an approved Standard Change and a Normal change with full approval will be required each time.

Source for reference: https://hscic365.sharepoint.com/:w:/r/sites/ServiceNowImplementation/_layouts/15/Doc.aspx?sourcedoc=%7BC2578153-85E0-41E7-

3508 https://nhsd-confluence.digital.nhs.uk/spaces/APM/pages/333290977/KOP-049+Raising+an+RFC+using+a+Standard+Change+Template

A9EB-85C7FBBC931B%7D&file=QRG021%20Create%20a%20Standard%20Change%20and%20Change%20Template.docx&action=default&mobileredirect=true

13.7.2.1 APIM Standard Changes

These are the components we want to raise standard changes for:

Component	Description	ServiceNow Standard Change ID	RFC to make into Standard	Linked RFCs	Standard Change ID
Infrastructure service	The infrastructure service is a repository of infrastructure related configurations items. It includes things like: • The Terraform/Docker configuration that drives our AWS infrastructure; and • The configuration for our pipelines (which are used to deploy our changes to various environments). The types of change we'll be making as 'standard changes' will be low risk, low impact, routine updates to configuration, such as: • Adding/updating Key Value Maps (KVMs); and • Updating automated test scripts.	CHG0031155	Cherwell ID: 30028795	Cherwell IDs: 30028450 30028333 30027682	Cherwell ID: 274

Component	Description	ServiceNow Standard Change ID	RFC to make into Standard	Linked RFCs	Standard Change ID
Identity Service	The identity service (or more correctly the authorisation service) is a repository that contains all things authorisation related. It contains the code for configuring Apigee to work in the way we want it to work. The types of change* we'll be making as 'standard changes' will be low risk, low impact, routine updates to configuration, such as: • Configuration change to the workflows (or just 'flows' to use Apigee's term) in Apigee; and • Updates to rate limiting functionality * Please note many changes we make in Apigee will only have a material impact when an API is updated to use the new configuration. One exception to this is where an existing shared flow is updated - in that case the update will immediately be inherited by any APIs that have implemented the shared flow. We will not make changes to shared flows via the standard change process	CHG0031154	Cherwell ID: 30028033	Cherwell IDs: 30027804 30027574 30026941	Cherwell ID: 286
Digital Onboarding Service			30028763	30028609 30028313 30028203	

Component	Description	ServiceNow Standard Change ID	RFC to make into Standard	Linked RFCs	Standard Change ID
PDS FHIR API			30028231	30027613 30027655 30027613	
Immunisation History API			30028396	30028297 30027987 30027285	
Covid 19 Test Results API			30028855	30028351 30027382 30027001	
Ambulance Data Submissions API			30025432	30024419	

13.7.3 Triage for Standard Changes

With the introduction of standard changes, we have a decision to make with each deployment (for a service that has a standard change template) over whether it meets the standard change criteria or not. We'll triage that as part of our standard process for delivering working software by discussing it as a group, either at the

elaboration stage (if enough information is known at that time) or at the ready for release stage and we'll confirm that understanding as part of the release co-ordination process for managing deployments. Where a work item constitutes a standard change we will add a label to it of 'Std Change', so that everyone can see from the work item itself that we've thought about it and made that decision.