

Expires: October 2001

Additional XML Digital Signature URIs

<draft-eastlake-xmldsig-uri-01.txt>

Status of This Document

Distribution of this draft is unlimited. It is intended to become an Informational RFC and will probably also be published as a W3C Note. Comments should be sent to the author or the XMLDSIG working group <w3c-ietf-xmldsig@w3.org>.

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC 2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

Abstract

A number of algorithm URIs intended for use with XML Digital Signatures [RFC 3075] are defined.

Acknowledgements

Glenn Adams, Merlin Hughs, Brian LaMachia, Joseph Reagle

Table of Contents

Status of This Document.....	1
Copyright Notice.....	1
Abstract.....	1
Acknowledgements.....	2
Table of Contents.....	2
1. Introduction.....	3
2. URIs.....	3
2.1 DigestMethod Algorithms.....	3
2.1.1 MD5.....	3
2.1.2 SHA-256.....	4
2.1.3 SHA-384.....	4
2.1.4 SHA-512.....	4
2.2 SignatureMethod Message Authentication Code Algorithms.....	5
2.2.1 HMAC-MD5.....	5
2.2.2 HMAC-SHA-256.....	6
2.2.3 HMAC-SHA-384.....	6
2.2.4 HMAC-SHA-512.....	6
2.3 SignatureMethod Public Key Signature Algorithms.....	6
2.3.1 RSA-MD5.....	7
2.3.2 RSA-SHA256.....	8
2.3.3 RSA-SHA384.....	8
2.3.4 RSA-SHA512.....	8
2.4 CanonicalizationMethod Algorithms.....	8
2.4.1 Minimal Canonicalization.....	8
2.5 Transform Algorithms.....	9
2.5.1 XPointer.....	9
3. KeyInfo Elements.....	10
3.1 PKCS #7 Bag of Certificates and CRLs.....	10
4. IANA Considerations.....	10
5. Security Considerations.....	10
References.....	11
Author's Address.....	12
Expiration and File Name.....	12
Full Copyright Statement.....	13

1. Introduction

XML Digital Signatures have been standardized by the joint IETF/W3C XMLDSIG working group. The Proposed Standard is specified in [RFC 3075]. In addition, Canonical XML, which is used by many digital signatures, has been standardized by the W3C and is documented in Informational [RFC 3076].

[RFC 3075] specifies URIs to identify algorithms. However, this protocol is likely to be raised to Draft Standard soon, which requires two independent interoperable implementations to exist. This may require algorithms in which there appears to be continued interest to be dropped from the standards track specification. This document is intended as a convenient reference list of URIs and descriptions for any dropped from the Proposed Standard due to lack of implementations plus additional suggested algorithms in which there appears to be substantial interest.

2. URIs

The sections below parallel those in Section 6 of RFC 3075 which group various algorithms by their use in XML Digital Signatures. URIs being dropped from the standard due to the transition from Proposed Standard to Draft Standard are included herein. Additional non-proprietary algorithms, particularly those based on USA Government and W3C standards, are given URIs that start with

<http://www.w3.org/2001/04/xmlldsig-more>

This does not imply any official W3C status for these algorithms. Currently, dereferencing such URIs produces, at best, a temporary placeholder document. Permission to use these URIs was tentatively given by W3C staff.

2.1 DigestMethod Algorithms

2.1.1 MD5

Identifier:

<http://www.w3.org/2001/04/xmlldsig-more#md5>

The MD5 algorithm [RFC 1321] takes no explicit parameters. An example of an MD5 DigestAlgorithm element is:

```
<DigestMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#md5"/>
```

An MD5 digest is a 128-bit string. The content of the DigestValue element shall be the base64 [RFC 2045] encoding of this bit string viewed as a 16-octet octet stream.

2.1.2 SHA-256

Identifier:
`http://www.w3.org/2001/04/xmldsig-more#sha256`

The SHA-256 algorithm [SHA-256] takes no explicit parameters. An example of a SHA-256 DigestAlgorithm element is:

```
<DigestMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"/>
```

A SHA-256 digest is a 256 bit string. The content of the DigestValue element shall be the base64 [RFC 2045] encoding of this string viewed as a 32-octet stream.

2.1.3 SHA-384

Identifier:
`http://www.w3.org/2001/04/xmldsig-more#sha384`

The SHA-384 algorithm [SHA-384] takes no explicit parameters. An example of a SHA-384 DigestAlgorithm element is:

```
<DigestMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha384"/>
```

A SHA-384 digest is a 384 bit string. The content of the DigestValue element shall be the base64 [RFC2045] encoding of this string viewed as a 48-octet stream.

2.1.4 SHA-512

Identifier:
`http://www.w3.org/2001/04/xmldsig-more#sha512`

The SHA-512 algorithm [SHA-512] takes an no explicit parameters. An example of a SHA-512 DigestAlgorithm element is:

```
<DigestMethod
  Algorithm="http://www.w3.org/2001/04/xmlldsig-more#sha512"/>
```

A SHA-512 digest is a 512 bit string. The content of the DigestValue element shall be the base64 [RFC2045] encoding of this string viewed as a 64-octet stream.

2.2 SignatureMethod Message Authentication Code Algorithms

Some text in this section is duplicated from RFC 3075 for the convenience of the reader.

2.2.1 HMAC-MD5

Identifier:
`http://www.w3.org/2001/04/xmlldsig-more#hmac-md5`

The HMAC algorithm [RFC 2104] takes the truncation length in bits as a parameter; if the parameter is not specified then all the bits of the hash are output. An example of an HMAC-MD5 SignatureMethod element is as follows:

```
<SignatureMethod
  Algorithm="http://www.w3.org/2001/04/xmlldsig-more#hmac-md5">
  <HMACOutputLength>112</HMACOutputLength>
</SignatureMethod>
```

The output of the HMAC algorithm is ultimately the output (possibly truncated) of the chosen digest algorithm. This value shall be base64 [RFC 2405] encoded in the same straightforward fashion as the output of the digest algorithms. Example: the SignatureValue element for the HMAC-MD5 digest

9294727A 3638BB1C 13F48EF8 158BFC9D

from the test vectors in [RFC 2104] would be

```
<SignatureValue>kpRyejY4uxwT9I74FYv8nQ==</SignatureValue>
```

Schema Definition:

```
<simpleType name="HMACOutputLengthType">
  <restriction base="integer"/>
</simpleType>
```

DTD:

```
<!ELEMENT HMACOutputLength (#PCDATA)>
```

The Schema Definition and DTD immediately above are copied from RFC 3075.

Although some cryptographic suspicions have recently been cast on MD5 for use in signatures such as RSA-MD5 below, this does not effect use of MD5 in HMAC.

2.2.2 HMAC-SHA-256

Identifier:

<http://www.w3.org/2001/04/xmlldsig-more#hmac-sha256>

SHA-256 [SHA-256] can also be used in HMAC as described in section 2.2.1 above for HMAC-MD5.

2.2.3 HMAC-SHA-384

Identifier:

<http://www.w3.org/2001/04/xmlldsig-more#hmac-sha384>

SHA-384 [SHA-384] can also be used in HMAC as described in section 2.2.1 above for HMAC-MD5.

2.2.4 HMAC-SHA-512

Identifier:

<http://www.w3.org/2001/04/xmlldsig-more#hmac-sha512>

SHA-512 [SHA-512] can also be used in HMAC as describe in section 2.2.1 above for HMAC-MD5.

2.3 SignatureMethod Public Key Signature Algorithms

2.3.1 RSA-MD5

Identifier:

`http://www.w3.org/2001/04/xmldsig-more#rsa-md5`

This implies the PKCS#1 v1.5 padding algorithm described in [RFC 2437].

An example of use is

```
<SignatureMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-md5"/>
```

The SignatureValue content for an RSA-MD5 signature is the base64 [RFC 2405] encoding of the octet string computed as per [RFC 2437], section 8.1.1.

Signature generation for the RSASSA-PKCS1-v1_5 signature scheme. As specified in the EMSA-PKCS1-V1_5-ENCODE function in [RFC 2437, section 9.2.1], the value input to the signature function MUST contain a pre-pended algorithm object identifier for the hash function, but the availability of an ASN.1 parser and recognition of OIDs is not required of a signature verifier. The PKCS#1 v1.5 representation appears as:

```
CRYPT (PAD (ASN.1 (OID, DIGEST (data))))
```

Note that the padded ASN.1 will be of the following form:

```
01 | FF* | 00 | prefix | hash
```

where "|" is concatenation, "01", "FF", and "00" are fixed octets of the corresponding hexadecimal value, "hash" is the MD5 digest of the data, and "prefix" is the ASN.1 BER MD5 algorithm designator prefix required in PKCS #1 [RFC 2437], that is,

```
hex 30 20 30 0c 06 08 2a 86 48 86 f7 0d 02 05 05 00 04 10
```

This prefix is included to make it easier to use standard cryptographic libraries. The FF octet MUST be repeated the maximum number of times such that the value of the quantity being CRYPTed is one octet shorter than the RSA modulus.

Due to increases in computer processor power and advances in cryptography, use of RSA-MD5 is NOT RECOMMENDED.

2.3.2 RSA-SHA256

Identifier:

`http://www.w3.org/2001/04/xmldsig-more#rsa-sha256`

An example of use is

```
<SignatureMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"
/>
```

[I think the SHA-256/384/512 RSA signature algorithms should use PKCS#1 v2, i.e., OAEP.]

2.3.3 RSA-SHA384

Identifier:

`http://www.w3.org/2001/04/xmldsig-more#rsa-sha384`

An example of use is

```
<SignatureMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha384"
/>
```

2.3.4 RSA-SHA512

Identifier:

`http://www.w3.org/2001/04/xmldsig-more#rsa-sha512`

An example of use is

```
<SignatureMethod
  Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha512"
/>
```

2.4 CanonicalizationMethod Algorithms

2.4.1 Minimal Canonicalization

At this time two independent interoperable implementations of Minimal Canonicalization have not been announced. Therefore, when XML

Digital Signature is advanced from Proposed Standard to Draft Standard, it must be dropped from the standard track documents. However, there is still interest and indicates of possible future use for Minimal Canonicalization. For its definition, see [RFC 3075], Section 6.5.1.

For reference, it's identifier remains:

<http://www.w3.org/2000/09/xmlsig#minimal>

2.5 Transform Algorithms

Note that all CanonicalizationMethod algorithms listed can also be used as Transform algorithms.

2.5.1 XPointer

Identifier:

<http://www.w3.org/2001/04/xmlsig-more/xptr>

This transform algorithm takes an [XPointer] as an explicit parameter. An example of use is [RFC 3092]:

```
<Transform
  Algorithm="http://www.w3.org/2001/04/xmlsig-more/xptr">
  <XPointer
    xmlns="http://www.w3.org/2001/04/xmlsig-more/xptr">
    xpointer(id("foo")) xmlns(bar=urn:baz)
      xpointer(//bar:Zab[@Id="foo"])
  </XPointer>
</Transform>
```

Schema Definition:

```
<element name="XPointer" type="string"/>
```

DTD:

```
<!ELEMENT XPointer (#PCDATA)>
```

Input to this transform is an octet stream (which is then parsed into XML).

Output from this transform is a node set; the results of the XPointer are processed as defined in the XMLDSIG specification [RFC 3075] for a same-document XPointer.

3. KeyInfo Elements

3.1 PKCS #7 Bag of Certificates and CRLs

A PKCS #7 [RFC 2315] "signedData" can also be used as a bag of certificates and/or certificate revocation lists. The PKCS7signedData element is defined to accomodate such structures within KeyInfo. The binary PKCS #7 structure is base64 encoded. Any signer information present is ignored. The following is an example, elliding the base64 data [RFC 3092]:

```
<foo:PKCS7signedData
  xmlns:foo="http://www.w3.org/2001/04/xmlldsig-more">
  ...
</foo:PKCS7signedData>
```

4. IANA Considerations

None. (so far)

5. Security Considerations

Due to computer speed and cryptographic advances, the use of MD5 as a DigestMethod or in the RSA-MD5 SignatureMethod is NOT RECOMMENDED. The cryptographic advances concerned do not effect the security of HMAC-MD5; however, there is little reason not to go for one of the SHA series of algorithms.

References

- [RFC 1321] - "The MD5 Message-Digest Algorithm", R. Rivest, April 1992.
- [RFC 2104] - "HMAC: Keyed-Hashing for Message Authentication", H. Krawczyk, M. Bellare, R. Canetti, February 1997.
- [RFC 2405] - "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed, N. Borenstein, November 1996.
- [RFC 2437] - "PKCS #1: RSA Cryptography Specifications Version 2.0", B. Kaliski, J. Staddon, October 1998.
- [RFC 2315] - "PKCS #7: Cryptographic Message Syntax Version 1.5", B. Kaliski, March 1998.
- [RFC 3075] - "XML-Signature Syntax and Processing", D. Eastlake, J. Reagle, D. Solo, March 2001. <<http://www.w3.org/TR/2000/CR-xmlsig-core-20001031>>
- [RFC 3076] - "Canonical XML Version 1.0", J. Boyer, March 2001. <<http://www.w3.org/TR/2001/REC-xml-c14n-20010315>>
- [RFC 3092] - "Etymology of 'Foo'", D. Eastlake 3rd, C. Manros, E. Raymond, 1 April 2001.
- [SHA-256] -
- [SHA-384] -
- [SHA-512] -
- [XPointer] - "XML Pointer Language (XPointer) Version 1.0", W3C working draft, Steve DeRose, Eve Maler, Ron Daniel Jr., January 2001. <<http://www.w3.org/TR/2001/WD-xptr-20010108>>

Author's Address

Donald E. Eastlake 3rd
Motorola
155 Beaver Street
Milford, MA 01757 USA

Telephone: +1-508-634-2066 (h)
 +1-508-261-5434 (w)
FAX: +1-508-261-4447 (w)
EMail: Donald.Eastlake@motorola.com

Expiration and File Name

This draft expires October 2001.

Its file name is draft-eastlake-xmlldsig-uri-01.txt.

Full Copyright Statement

Copyright (C) The Internet Society (2001). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

