XML Security Algorithm Cross-Reference

W3C Working Draft 26 February 2009

This version:
http://www.w3.org/TR/2009/WD-xmlsec-algorithms-20090226/
Latest version:
http://www.w3.org/TR/xmlsec-algorithms/
Editors:
Frederick Hirsch, Nokia
Thomas Roessler, W3C
Kelvin Yiu, Microsoft

Abstract

This Note summarizes XML Security algorithm URI identifiers and the specifications associated with them.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This is a First Public Working Draft of "XML Security Algorithm Cross-Reference."

This document is intended to serve as a cross-reference to various commonly used XML security specifications. It does not include any normative requirements of its own.

This document was developed by the XML Security Working Group.

Please send comments about this document to public-xmlsec-comments@w3.org (with public archive).

Publication as a Working Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents

at any time. It is inappropriate to cite this document as other than work in progress.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. The group does not expect this document to become a W3C Recommendation. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

Table of Contents

---

1 Introduction

The various XML Security specifications have defined a number of algorithms of various types, while allowing and expecting additional algorithms to be defined later. Over time, these identifiers have been defined in a number of different specifications, including XML Signature, XML Encryption, RFCs and elsewhere.

This makes it difficult for users of the XML Security specifications to know whether and where a URI for an algorithm of interest has been defined, and can lead to the use of incorrect URIs. The purpose of this Note is to collect the various known URIs at the time of its publication and indicate the specifications in which they are defined in order to avoid confusion and errors.

This note is not intended as an exhaustive list of all known related identifiers, some of

which may have been defined by other standards or specifications. Furthermore, this note is not to be taken as normative regarding the information provided; if information here conflicts with the referenced specification, the specification takes precedence in all cases.

The architecture of the XML Security specifications distinguishes between the (universally useful) identifiers for algorithms and the roles that these algorithms can take. Roles are identified through elements like `ds:SignatureMethod`, `ds:DigestMethod`, `ds:CanonicalizationMethod`, or `ds:Transform`, whereas the algorithms are identified through URIs. Explicit parameters for the respective algorithms are transmitted in child elements of the role element.

2 Namespaces

This specification uses the following XML namespace prefixes:

ds
`http://www.w3.org/2000/09/xmldsig#`
xenc
`http://www.w3.org/2001/04/xmlenc#`
dsig11
`http://www.w3.org/2009/xmldsig11#`
dsigmore
`http://www.w3.org/2001/04/xmldsig-more#`

Algorithm URIs have been coined in a variety of namespaces, and are always given in full.

3 Signature Algorithms

The algorithms listed in this section are typically used in the signature algorithm role, identified through the `ds:SignatureMethod` role element ([XMLDSIG2e], section 4.3.2). Each signature method takes an octet-stream as input, and produces a signature value (an octet-stream that is always base64 encoded, see section 4.2 of [XMLDSIG2e]).

3.1 DSA

A container for key material, `ds:DSAKeyValue`, is defined in section 4.4.2.1 of [XMLDSIG2e]. When used with `ds:RetrievalMethod`, this container type is identified through the URI `http://www.w3.org/2000/09/xmldsig#DSAKeyValue`.

DSA-SHA1
URI:
http://www.w3.org/2000/09/xmldsig#dsa-sha1
Specified in:
section 6.4.1 of [XMLDSIG2e]

Implementation of this algorithm is required in both [XMLDSIG] and [XMLDSIG2e].

3.2 RSA

This section lists variants of the RSA algorithm. A container for key material, `ds:RSAKeyValue`, is defined in section 4.4.2.2 of [XMLDSIG]. When used with `ds:RetrievalMethod`, this container type is identified through the URI `http://www.w3.org/2000/09/xmldsig#RSAKeyValue`.

RSA-MD5

URI:

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

Specified in:

section 2.3.1 of [RFC4051]

We only list the algorithm URI for RSA-MD5 for the sake of completeness. The cryptographic strength of the MD5 algorithm is sufficiently doubtful that its use is not recommended at this time.

RSA-SHA1

URI:

http://www.w3.org/2000/09/xmldsig#rsa-sha1

Specified in:

section 6.4.2 of [XMLDSIG]

Implementation of this algorithm is recommended in [XMLDSIG] and [XMLDSIG2e].

RSA-SHA256

URI:

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

Specified in:

section 2.3.2 of [RFC4051]

This algorithm is under consideration as a mandatory to implement algorithm for a future version of XML Signature [XMLDSIG11].

RSA-SHA384

URI:

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

Specified in:

section 2.3.3 of [RFC4051]

RSA-SHA512

URI:

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

Specified in:

section 2.3.4 of [RFC4051]

RSA-RIPEMD160

URI:

http://www.w3.org/2001/04/xmldsig-more#rsa-ripemd160

Specified in:

section 2.3.5 of [RFC4051]

3.3 Elliptic Curve DSA

This section lists various variants of the Elliptic Curve DSA (ECDSA) algorithm. A container for key material, `dsigmore:ECDSAKeyValue`, is defined in [RFC4050]. No `ds:RetrievalMethod` type URI is defined for this container.

Work is under way to revise this container format.

ECDSA-SHA1

URI:

http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha1

Specified in:

section 2.3.6 of [RFC4051]

ECDSA-SHA224

URI:

http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha224

Specified in:

section 2.3.6 of [RFC4051]

ECDSA-SHA256

URI:

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

Specified in:

section 2.3.6 of [RFC4051]

This algorithm is under consideration as a mandatory to implement algorithm for a future version of XML Signature [XMLDSIG11].

ECDSA-SHA384

URI:

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

Specified in:

section 2.3.6 of [RFC4051]

ECDSA-SHA512

URI:

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

Specified in:

section 2.3.6 of [RFC4051]

3.4 HMAC

The following URIs have been defined for various Message Authentication Codes that use the HMAC construction [RFC2104]. All of these algorithms take an explicit truncation length parameter. A container for this parameter, `ds:HMACOutputLength`, is defined in section 6.3.1 of [XMLDSIG2e]. This container occurs as a child element of the role element.

HMAC-SHA1

URI:

http://www.w3.org/2000/09/xmldsig#hmac-sha1

Specified in:

section 6.3 of [XMLDSIG]

This algorithm is used as the default MAC algorithm in [XKMS2]. It is mandatory to implement in XML Signature [XMLDSIG], [XMLDSIG2e].

HMAC-SHA256

URI:

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

Specified in:

section 2.2.2 of [RFC4051]

This algorithm is under consideration as a recommended algorithm for a future version of XML Signature [XMLDSIG11].

HMAC-SHA384

URI:

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

Specified in:

section 2.2.2 of [RFC4051]

HMAC-SHA512

URI:

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

Specified in:

section 2.2.2 of [RFC4051]

HMAC-RIPEMD160

URI:

http://www.w3.org/2001/04/xmldsig-more#hmac-ripemd160

Specified in:

Section 2.2.3 of [RFC4051]

4 Digest Methods

The following URIs have been defined for Digest Methods. They are typically used in the `ds:DigestMethod` role in [XMLDSIG]. Note that `ds:DigestMethod` also occurs as in the context of `xenc:AgreementMethod`, as specified in the Key Agreement part of [XMLENC].

4.1 MD5

MD5

URI:

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

Specified in:

section 2.1.1 of [RFC4051]

We only list the algorithm URI for MD5 for the sake of completeness. The cryptographic strength of this algorithm is sufficiently doubtful that its use is not recommended at this time.

4.2 SHA variants

Note that URIs for the various algorithms of the Secure Hash Algorithm family have been coined in a number of name spaces and specifications, specifically [XMLDSIG] (and, in this regard identically, [XMLDSIG2e]), [XMLENC], and [RFC4051].

SHA-1

URI:

http://www.w3.org/2000/09/xmldsig#sha1

Specified in:

section 6.2.1 of [XMLDSIG2e]

SHA-1 is the only digest algorithm defined in [XMLDSIG2e], and is mandatory to implement in that specification, and in [XMLENC]. Given recent cryptographic research, however, future versions of the XML Signature specification are likely to recommend use of other, stronger digest algorithms over use of SHA-1, while maintaining SHA-1 as a mandatory to implement algorithm.

SHA-224

URI:

http://www.w3.org/2001/04/xmldsig-more#sha224

Specified in:

section 2.1.2 of [RFC4051]

SHA-256

URI:

http://www.w3.org/2001/04/xmlenc#sha256

Specified in:

section 5.7.2 of [XMLENC]

This algorithm is under consideration as a mandatory to implement algorithm for a future version of XML Signature [XMLDSIG11]. It is recommended in [XMLENC].

SHA-384

URI:

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

Specified in:

section 2.1.3 of [RFC4051]

SHA-512

URI:

http://www.w3.org/2001/04/xmlenc#sha512

Specified in:

section 5.7.3 of [XMLENC]

4.3 RIPEMD-160

RIPEMD-160

URI:

http://www.w3.org/2001/04/xmlenc#ripemd160

Specified in:

section 5.7.4 of [XMLENC]

5 Symmetric Key Encryption Algorithms

The following URIs have been defined for symmetric key encryption algorithms. They typically appear in the `xenc:EncryptionMethod` role.

5.1 Block Encryption Algorithms DES

Triple DES (CBC mode)

URI:

http://www.w3.org/2001/04/xmlenc#tripledes-cbc

Specified in:

section 5.2.1 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

AES-128 (CBC mode)

URI:

http://www.w3.org/2001/04/xmlenc#aes128-cbc

Specified in:

section 5.2.2 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

AES-192 (CBC mode)

URI:

http://www.w3.org/2001/04/xmlenc#aes192-cbc

Specified in:

section 5.2.2 of [XMLENC]

AES-256 (CBC mode)

URI:

http://www.w3.org/2001/04/xmlenc#aes256-cbc

Specified in:

section 5.2.2 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

6 Key Transport Algorithms

The following URIs have been defined for key transport algorithms.

RSA-v1.5

URI:

http://www.w3.org/2001/04/xmlenc#rsa-1_5

Specified in:

section 5.4.1 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

RSA-OAEP

URI:
http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p
Specified in:
section 5.4.2 of [XMLENC]

7 Key Agreement Algorithm URIs

The following URIs have been defined for key agreement algorithms.

Diffie Hellman
URI:
http://www.w3.org/2001/04/xmlenc#dh
Specified in:
section 5.5.1 of [XMLENC]

While this is the only key agreement algorithm defined in [XMLENC], it is optional to implement.

A container for key material for this key agreement algorithm, `xenc:DHKeyValue`, is defined in section 5.5.2 of [XMLENC]. When used with `ds:RetrievalMethod`, this container type is identified through the URI `http://www.w3.org/2001/04/xmlenc#dh`.

Elliptic Key Diffie-Hellman Key Agreement (Ephemeral-Static Mode)
URI:
http://www.w3.org/2009/xmlenc11#ECDH-ES
Specified in:
section 5.5.4 of [XMLENC11]

This algorithm is under consideration as a mandatory to implement algorithm for a future version of XML Encryption. [XMLENC11].

8 Symmetric Key Wrap Algorithm URIs

The following URIs have been defined for symmetric key wrap algorithms.

CMS Triple-DES Key Wrap
URI:
http://www.w3.org/2001/04/xmlenc#kw-tripledes
Specified in:
section 5.6.2 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

AES Key Wrap 128
URI:
http://www.w3.org/2001/04/xmlenc#kw-aes128
Specified in:
section 5.6.3 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

AES Key Wrap 192

URI:

http://www.w3.org/2001/04/xmlenc#kw-aes192

Specified in:

section 5.6.3 of [XMLENC]

AES Key Wrap 256

URI:

http://www.w3.org/2001/04/xmlenc#kw-aes256

Specified in:

section 5.6.3 of [XMLENC]

This algorithm is mandatory to implement in [XMLENC].

9 Canonicalization Algorithms

Canonicalization algorithms are used in [XMLDSIG]; they are typically used in the `ds:CanonicalizationMethod` and `ds:Transform` roles.

9.1 Inclusive Canonicalization

Canonical XML 1.0 [C14N1.0] without comments is mandatory to implement in both XML Signature [XMLDSIG] and XML Signature Second Edition [XMLDSIG2e]. XML Signature Second Edition recommends use of Canonical XML 1.1 [C14N1.1] over use of Canonical XML 1.0 when inclusive canonicalization is desired, to address known issues with Canonical XML 1.0.

The canonicalization methods listed in this section accept a node-set or octet-stream as input, and produce an octet-stream as output.

Canonical XML 1.0 (omits comments)

URI:

http://www.w3.org/TR/2001/REC-xml-c14n-20010315

Specified in:

section 6.5.1 of [XMLDSIG2e]

This algorithm is mandatory to implement in [XMLDSIG] and [XMLDSIG2e].

Canonical XML 1.0 with Comments

URI:

http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments

Specified in:

section 6.5.1 of [XMLDSIG2e]

Canonical XML 1.1 (omits comments)

URI:

http://www.w3.org/2006/12/xml-c14n11

Specified in:

section 6.5.2 of [XMLDSIG2e]

This algorithm is mandatory to implement in [XMLDSIG2e]. Its use is recommended over Canonical XML 1.0.

Canonical XML 1.1 with Comments

URI:

http://www.w3.org/2006/12/xml-c14n11#WithComments

Specified in:

section 6.5.2 of [XMLDSIG2e]

9.2 Exclusive Canonicalization

Exclusive Canonicalization XML 1.0 (omits comments)

URI:

http://www.w3.org/2001/10/xml-exc-c14n#

Specified in:

section 4 of [EXC-C14N1.0]

Exclusive Canonicalization XML 1.0 with Comments

URI:

http://www.w3.org/2001/10/xml-exc-c14n#WithComments

Specified in:

section 4 of [EXC-C14N1.0]

10 Transform Algorithms

This section lists algorithms that typically occur in the `ds:Transform` role. `ds:Transform` is defined in detail in the XML Signature Reference Processing Model ([XMLDSIG2e], section 4.3.3.2). This processing model is, in turn, applied both to signed material, and to key material referenced through `ds:RetrievalMethod` ([XMLDSIG2e], section 4.4.3).

The `ds:Transform` role element is also used by the optional `xenc:Transforms` feature which is specified in the context of `xenc:CipherReference` in XML Encryption ([XMLENC], section 3.3.1).

Transform algorithms can take an octet-stream or a node-set as input, and can produce either an octet-stream or a node-set as output.

Base64 decoding transform

URI:

http://www.w3.org/2000/09/xmldsig#base64

Specified in:

section 6.6.2 of [XMLDSIG2e]

Input:

octet-stream, node-set

Output:

octet-stream

Implementation is required in [XMLDSIG2e] and [XMLENC].

XPath Filtering

URI:

http://www.w3.org/TR/1999/REC-xpath-19991116

Specified in:

section 6.6.3 of [XMLDSIG2e]

Input:

octet-stream, node-set

Output:

node-set

XML-Signature XPath Filter 2.0

URI:

http://www.w3.org/2002/06/xmldsig-filter2

Specified in:

[FILTER2]

Input:

octet-stream, node-set

Output:

node-set

Enveloped Signature Transform

URI:

http://www.w3.org/2000/09/xmldsig#enveloped-signature

Specified in:

section 6.6.4 of [XMLDSIG2e]

Input:

node-set (same-document)

Output:

node-set

This transform is required in [XMLDSIG], [XMLDSIG2e].

XSLT Transform

URI:

http://www.w3.org/TR/1999/REC-xslt-19991116

Specified in:

section 6.6.5 of [XMLDSIG2e]

Input:

octet-stream

Output:

octet-stream

Decryption Transform (XML mode)

URI:

http://www.w3.org/2002/07/decrypt#XML

Specified in:

[DecryptTransform]

Input:

node-set

Output:

node-set

Decryption Transform (binary mode)

URI:

http://www.w3.org/2002/07/decrypt#Binary

Specified in:

[DecryptTransform]

Input:

node-set

Output:

octet-stream

11 Retrieval method type identifiers

The `ds:RetrievalMethod` element permits referencing key material that is stored outside a `ds:KeyInfo` element. The type of the material that results from retrieval of the URI reference (and possible transform processing) can be identified using the `Type` attribute.

Note: `ds:RetrievalMethod` may be deprecated in future versions of XML Signature, and is rarely used in practice.

The following `Type` values identify an XML element or document with the given element as its root:

http://www.w3.org/2000/09/xmldsig#DSAKeyValue
`ds:DSAKeyValue`, see section 4.4.2.1 of [XMLDSIG2e].
http://www.w3.org/2000/09/xmldsig#RSAKeyValue
`ds:RSAKeyValue`, see section 4.4.2.2 of [XMLDSIG2e].
http://www.w3.org/2000/09/xmldsig#X509Data
`ds:X509Data`, see section 4.4.4 of [XMLDSIG2e].
http://www.w3.org/2000/09/xmldsig#PGPData
`ds:PGPData`, see section 4.4.5 of [XMLDSIG2e].
http://www.w3.org/2000/09/xmldsig#SPKIData
`ds:SPKIData`, see section 4.4.6 of [XMLDSIG2e].
http://www.w3.org/2000/09/xmldsig#MgmtData
`ds:MgmtData`, see section 4.4.7 of [XMLDSIG2e].
http://www.w3.org/2001/04/xmldsig-more#KeyValue
`ds:KeyValue`, see section 4.4.2 of [XMLDSIG2e].
http://www.w3.org/2001/04/xmldsig-more#RetrievalMethod
`ds:RetrievalMethod`, see secion 4.4.3 of [XMLDSIG2e].
http://www.w3.org/2001/04/xmldsig-more#KeyName
`ds:KeyName`, see section 4.4.1 of [XMLDSIG2e].
http://www.w3.org/2001/04/xmldsig-more#PKCS7signedData
`dsigmore:PKCS7signedData`, see section 3.1 of [RFC4051].

The following `Type` values identify the type of raw binary data:

http://www.w3.org/2001/04/xmldsig-more#rawX509CRL
http://www.w3.org/2001/04/xmldsig-more#rawPGPKeyPacket
http://www.w3.org/2001/04/xmldsig-more#rawSPKISexp
http://www.w3.org/2001/04/xmldsig-more#rawPKCS7signedData
http://www.w3.org/2000/09/xmldsig#rawX509Certificate

12 References

C14N1.0
Canonical XML 1.0, John Boyer. W3C Recommendation 15 March 2001,

http://www.w3.org/TR/2001/REC-xml-c14n-20010315.

C14N1.1

Canonical XML 1.1, John Boyer, Glenn Marcy. W3C Recommendation 2 May 2008, http://www.w3.org/TR/2008/REC-xml-c14n11-20080502/.

DecryptTransform

Decryption Transform for XML Signature, Merlin Hughes, Takeshi Imamura, Hiroshi Maruyama. W3C Recommendation 10 December 2008, http://www.w3.org/TR/2002/REC-xmlenc-decrypt-20021210.

EXC-C14N1.0

Exclusive XML Canonicalization Version 1.0, John Boyer, Donald E. Eastlake 3rd, Joseph Reagle. W3C Recommendation 18 July 2002, http://www.w3.org/TR/2002/REC-xml-exc-c14n-20020718/.

FILTER2

XML-Signature XPath Filter 2.0, J. Boyer, M. Hughes, J. Reagle. W3C Recommendation 8 November 2002. http://www.w3.org/TR/2002/REC-xmldsig-filter2-20021108/.

RFC2104

HMAC: Keyed-Hashing for Message Authentication, H. Krawczyk, M. Bellare, R. Canetti, RFC 2104, February 1997. http://www.ietf.org/rfc/rfc2104.txt.

RFC4050

Using the Elliptic Curve Signature Algorithm (ECDSA) for XML Digital Signatures. S. Blake-Wilson, G. Karlinger, T. Kobayashi, Y. Wang. IETF 4050, April 2005. http://www.ietf.org/rfc/rfc4050.txt.

RFC4051

Additional XML Security Uniform Resource Identifiers (URIs) , D. Eastlake, RFC 4051, April 2005, http://www.ietf.org/rfc/rfc4051.txt.

XKMS2

XML Key Management Specification (XKMS 2.0) Version 2.0, P Hallam-Baker, S. Mysore, W3C Recommendation 28 June 2005, http://www.w3.org/TR/2005/REC-xkms2-20050628/.

XMLDSIG

XML-Signature Syntax and Processing , D. Eastlake, J. R., D. Solo, M. Bartel, J. Boyer , B. Fox , E. Simon. W3C Recommendation, 12 February 2002, http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/.

XMLDSIG11

XML Signature Syntax and Processing Version 1.1. W3C Working Draft 26 February 2009, http://www.w3.org/TR/2009/WD-xmldsig-core1-20090226/.

XMLDSIG2e

XML Signature Syntax and Processing (Second Edition), W3C Recommendation 10 June 2008 http://www.w3.org/TR/2008/REC-xmldsig-core-20080610/

XMLENC

XML Encryption Syntax and Processing , D. Eastlake, J. Reagle, W3C Recommendation 10 December 2002, http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/.

XMLENC11

XML Encryption Syntax and Processing Version 1.1, W3C Working Draft 26 February 2009, http://www.w3.org/TR/2009/WD-xmlenc-core1-20090226/.