



SAML V2.0 Metadata Profile for Algorithm Support Version 1.0

Working Draft 02 19 October 2010

Specification URIs:

This Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport-cd-01.html>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport-cd-01.odt>
(Authoritative)

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport-cd-01.pdf>

Previous Version:

None

Latest Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport.html>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport.odt>
(Authoritative)

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-alsupport.pdf>

Technical Committee:

OASIS Security Services TC

Chair:

Thomas Hardjono, M.I.T.

Nate Kingenstein, Internet2

Editor:

Scott Cantor, Internet2

Related Work:

This specification defines an extension for use with SAML V2.0 Metadata [SAML2Meta].

Declared XML Namespace(s):

urn:oasis:names:tc:SAML:metadata:alsupport

Abstract:

The SAML V2.0 Metadata specification [SAML2Meta] includes an element allowing entities to describe the XML Encryption [XMLEnc] algorithms they support. This specification defines metadata extension elements to enable entities to describe the XML Signature [XMLSig] algorithms they support, and a profile for using both elements to enable better algorithm agility for profiles that rely on metadata.

36 **Status**

37 This document was last revised or approved by the SSTC on the above date. The level of
38 approval is also listed above. Check the current location noted above for possible later revisions
39 of this document. This document is updated periodically on no particular schedule.

40 TC members should send comments on this specification to the TC's email list. Others
41 should send comments to the TC by using the "Send A Comment" button on the TC's
42 web page at <http://www.oasis-open.org/committees/security>.

43 For information on whether any patents have been disclosed that may be essential to
44 implementing this specification, and any offers of patent licensing terms, please refer to the IPR
45 section of the TC web page (<http://www.oasis-open.org/committees/security/ipr.php>).

46 The non-normative errata page for this specification is located at [http://www.oasis-](http://www.oasis-open.org/committees/security)
47 [open.org/committees/security](http://www.oasis-open.org/committees/security).

48 Notices

49 Copyright © OASIS Open 2010. All Rights Reserved.

50 All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual
51 Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

52 This document and translations of it may be copied and furnished to others, and derivative works that
53 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published,
54 and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice
55 and this section are included on all such copies and derivative works. However, this document itself may
56 not be modified in any way, including by removing the copyright notice or references to OASIS, except as
57 needed for the purpose of developing any document or deliverable produced by an OASIS Technical
58 Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be
59 followed) or as required to translate it into languages other than English.

60 The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors
61 or assigns.

62 This document and the information contained herein is provided on an "AS IS" basis and OASIS
63 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY
64 WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY
65 OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
66 PARTICULAR PURPOSE.

67 OASIS requests that any OASIS Party or any other party that believes it has patent claims that would
68 necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to
69 notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such
70 patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced
71 this specification.

72 OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any
73 patent claims that would necessarily be infringed by implementations of this specification by a patent
74 holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR
75 Mode of the OASIS Technical Committee that produced this specification. OASIS may include such
76 claims on its website, but disclaims any obligation to do so.

77 OASIS takes no position regarding the validity or scope of any intellectual property or other rights that
78 might be claimed to pertain to the implementation or use of the technology described in this document or
79 the extent to which any license under such rights might or might not be available; neither does it represent
80 that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to
81 rights in any document or deliverable produced by an OASIS Technical Committee can be found on the
82 OASIS website. Copies of claims of rights made available for publication and any assurances of licenses
83 to be made available, or the result of an attempt made to obtain a general license or permission for the
84 use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS
85 Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any
86 information or list of intellectual property rights will at any time be complete, or that any claims in such list
87 are, in fact, Essential Claims.

88 The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be
89 used only to refer to the organization and its official outputs. OASIS welcomes reference to, and
90 implementation and use of, specifications, while reserving the right to enforce its marks against
91 misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

92 **Table of Contents**

93 1 Introduction..... 5
94 1.1 Notation..... 5
95 1.2 Normative References..... 6
96 1.3 Non-Normative References..... 6
97 2 SAML V2.0 Metadata Profile for Algorithm Support..... 7
98 2.1 Required Information..... 7
99 2.2 Profile Description..... 7
100 2.3 Expression of Encryption Capabilities..... 7
101 2.4 Expression of Signature Capabilities..... 8
102 2.4.1 Element <alg:DigestMethod>..... 8
103 2.4.2 Element <alg:SigningMethod>..... 8
104 2.5 Metadata Consumers..... 9
105 2.6 Security Considerations..... 9
106 2.7 Example..... 10
107 3 Conformance..... 11
108 3.1 SAML V2.0 Metadata Profile for Algorithm Support Version 1.0..... 11
109 Appendix A. Acknowledgements..... 12
110 Appendix B. Revision History..... 13
111

1 Introduction

The SAML V2.0 Metadata specification [SAML2Meta] includes an `<md:EncryptionMethod>` element intended to communicate the XML Encryption [XMLEnc] algorithms supported for use with the key described by a containing `<md:KeyDescriptor>` element. The use of this element is not completely defined by the original specification, and there is no comparable support for communicating the XML Signature [XMLSig] algorithms supported by an entity. This profile addresses both considerations to improve algorithm agility and interoperability for deployments that make use of metadata.

There are more general standards for the description of security requirements of communicating endpoints, such as [WS-SecPol]. This specification is not intended as a replacement for such mechanisms, but is directed at systems with fewer requirements that are already designed around SAML V2.0 Metadata.

1.1 Notation

This specification uses normative text.

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [RFC2119]:

...they MUST only be used where it is actually required for interoperation or to limit behavior which has potential for causing harm (e.g., limiting retransmissions)...

These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

Listings of XML schemas appear like this.

Example code listings appear like this.

Conventional XML namespace prefixes are used throughout the listings in this specification to stand for their respective namespaces as follows, whether or not a namespace declaration is present in the example:

Prefix	XML Namespace	Comments
saml:	urn:oasis:names:tc:SAML:2.0:assertion	This is the SAML V2.0 assertion namespace [SAML2Core].
samlp:	urn:oasis:names:tc:SAML:2.0:protocol	This is the SAML V2.0 protocol namespace [SAML2Core].
md:	urn:oasis:names:tc:SAML:2.0:metadata	This is the SAML V2.0 metadata namespace [SAML2Meta].
alg:	urn:oasis:names:tc:SAML:metadata:algsupport	This is the SAML V2.0 metadata extension namespace defined by this document and its accompanying schema [AlgSup-XSD].
xenc:	http://www.w3.org/2001/04/xmlenc#	This is the XML Encryption namespace [XMLEnc].
xsd:	http://www.w3.org/2001/XMLSchema	This namespace is defined in the W3C XML Schema specification [Schema1]. In schema listings, this is the default namespace and no prefix is shown.

139 This specification uses the following typographical conventions in text: <SAML*E*lement>,
140 <ns:ForeignElement>, Attribute, **Datatype**, OtherCode.

141 1.2 Normative References

- 142 **[AlgSup-XSD]** OASIS Working Draft, *Metadata Extension Schema for SAML V2.0 Metadata*
143 *Profile for Algorithm Support Version 1.0*, June 2010. <http://docs.oasis->
144 [open.org/security/saml/Post2.0/sstc-saml-metadata-algsupport.xsd](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-metadata-algsupport.xsd)
- 145 **[RFC2119]** S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF
146 RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>
- 147 **[SAML2Core]** OASIS Standard, *Assertions and Protocols for the OASIS Security Assertion*
148 *Markup Language (SAML) V2.0*, March 2005. <http://docs.oasis->
149 [open.org/security/saml/v2.0/saml-core-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf)
- 150 **[SAML2Errata]** OASIS Approved Errata, *SAML V2.0 Errata*, October 2009. <http://docs.oasis->
151 [open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf](http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf)
- 152 **[SAML2Meta]** OASIS Standard, *Metadata for the OASIS Security Assertion Markup Language*
153 *(SAML) V2.0*, March 2005. <http://docs.oasis-open.org/security/saml/v2.0/saml->
154 [metadata-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
- 155 **[XMLEnc]** D. Eastlake et al. *XML Encryption Syntax and Processing*. World Wide Web
156 Consortium Recommendation. <http://www.w3.org/TR/2002/REC-xmlenc-core->
157 [20021210/](http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/)
- 158 **[Schema1]** H. S. Thompson et al. XML Schema Part 1: Structures. World Wide Web
159 Consortium Recommendation, May 2001. <http://www.w3.org/TR/2001/REC->
160 [xmlschema-1-20010502/](http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)
- 161 **[Schema2]** Paul V. Biron, Ashok Malhotra. *XML Schema Part 2: Datatypes*. World Wide Web
162 Consortium Recommendation, May 2001. <http://www.w3.org/TR/2001/REC->
163 [xmlschema-2-20010502/](http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/)
- 164 **[XMLSig]** D. Eastlake et al. *XML-Signature Syntax and Processing, Second Edition*. World
165 Wide Web Consortium Recommendation, June 2008.
166 <http://www.w3.org/TR/xmlsig-core/>

167 1.3 Non-Normative References

- 168 **[RFC4051]** IETF RFC 4051, *Additional XML Security Uniform Resource Identifiers*, April
169 2005. <http://www.ietf.org/rfc/rfc4051.txt>
- 170 **[WS-SecPol]** OASIS Standard, *WS-SecurityPolicy 1.3*, February 2009. <http://docs.oasis->
171 [open.org/ws-sx/ws-securitypolicy/v1.3/os/ws-securitypolicy-1.3-spec-os.pdf](http://docs.oasis-open.org/ws-sx/ws-securitypolicy/v1.3/os/ws-securitypolicy-1.3-spec-os.pdf)
- 172

2 SAML V2.0 Metadata Profile for Algorithm Support

2.1 Required Information

Identification: urn:oasis:names:tc:SAML:metadata:algsupport

Contact information: security-services-comment@lists.oasis-open.org

Description: Given below.

Updates: None.

2.2 Profile Description

One of the interoperability challenges in large-scale, and long-term, SAML deployments is the selection of XML Signature [XMLSig] and XML Encryption [XMLEnc] algorithms at runtime when communicating with peer entities. In particular, accounting for software limitations that prevent support of newer algorithms, while supporting those algorithms where possible to gradually strengthen systems, is difficult to manage without knowledge of a peer's capabilities. This profile makes use of SAML metadata to enable deployments to document their algorithm capabilities and preferences. It also allows for future expansion to address the interoperability requirements of more complex algorithms.

This profile provides guidance on the use of the `<md:EncryptionMethod>` element defined in the SAML V2.0 Metadata specification [SAML2Meta], and defines extension elements, `<alg:SigningMethod>` and `<alg:DigestMethod>`, to address comparable requirements related to XML Signature usage.

2.3 Expression of Encryption Capabilities

The SAML V2.0 Metadata specification [SAML2Meta] permits zero or more `<md:EncryptionMethod>` elements to appear inside a `<md:KeyDescriptor>` element. This profile provides guidance for the use of this element only in enclosing elements whose `use` attribute is omitted or set to "encryption".

In the common case that a `<md:KeyDescriptor>` element contains an asymmetric encryption key, an `<md:EncryptionMethod>` element SHOULD be present for each of a Block or Stream Encryption, and a Key Transport or Key Agreement algorithm. The Key Transport or Key Agreement algorithm(s) listed MUST be compatible with the associated encryption key.

If the `<md:KeyDescriptor>` element contains or identifies by reference a symmetric key (e.g., a name referring to a shared master secret or password), then an `<md:EncryptionMethod>` element SHOULD be present for a Block or Stream Encryption algorithm, and MAY be present for other algorithm types such as Symmetric Key Wrap or Key Derivation.

Per [XMLEnc], the `<md:EncryptionMethod>` element MUST contain an `Algorithm` attribute containing the identifier for the algorithm defined for use with the specification. If the algorithm permits varying key sizes, the element MAY contain an `<xenc:KeySize>` element defining a key size for the algorithm that the entity will accept. If the algorithm definition includes the specification of additional public content that the party performing encryption needs, that content MAY also be present.

If multiple `<md:EncryptionMethod>` elements identifying algorithms of the same general type are present, they MUST be listed in order of preference by the entity.

210 2.4 Expression of Signature Capabilities

211 This profile defines a pair of extension elements for the expression of an entity's capability to verify digests
212 and signatures with particular algorithms. While not strictly meant as an expression of policy, it is a natural
213 assumption that a peer stating support for particular algorithms requires their use.

214 An entity SHOULD include one or more `<alg:DigestMethod>` and `<alg:SigningMethod>` elements
215 in its metadata by means of the `<md:Extensions>` element in its `<md:EntityDescriptor>` element,
216 and/or in its roles (elements whose type is based on `md:RoleDescriptorType`).

217 If a signature algorithm permits varying key sizes, the `<alg:SigningMethod>` element MAY contain
218 `MinKeySize` and/or `MaxKeySize` attributes bounding the key size for the algorithm that the entity
219 supports. If the algorithm definition includes the specification of additional public content that the party
220 creating a signature or digest needs, that content MAY also be present.

221 If multiple elements of the same type are present, they MUST be listed in order of preference by the entity.

222 2.4.1 Element `<alg:DigestMethod>`

223 The `<alg:DigestMethod>` element describes a Message Digest algorithm. It contains the following
224 attribute:

225 `Algorithm` [Required]

226 Identifies the algorithm by means of the URL defined for its use with the XML Signature specification
227 [XMLSig].

228 This element also permits the use of arbitrary elements defined in any namespace.

229 The schema for the `<alg:DigestMethod>` element, and its corresponding `alg:DigestMethodType`
230 complex type, is as follows:

```
231 <element name="DigestMethod" type="alg:DigestMethodType"/>  
232 <complexType name="DigestMethodType">  
233   <sequence>  
234     <any namespace="##any" processContents="lax" minOccurs="0"  
235     maxOccurs="unbounded"/>  
236   </sequence>  
237   <attribute name="Algorithm" type="anyURI" use="required"/>  
238 </complexType>
```

239 2.4.2 Element `<alg:SigningMethod>`

240 The `<alg:SigningMethod>` element describes a Signature or Message Authentication Code algorithm.
241 It contains the following attributes:

242 `Algorithm` [Required]

243 Identifies the algorithm by means of the URL defined for its use with the XML Signature specification
244 [XMLSig].

245 `MinKeySize` [Optional]

246 The smallest key size, in bits, that the entity supports in conjunction with the algorithm. If omitted, no
247 minimum is implied.

248 `MaxKeySize` [Optional]

249 The largest key size, in bits, that the entity supports in conjunction with the algorithm. If omitted, no
250 maximum is implied.

251 This element also permits the use of arbitrary elements defined in any namespace.

252 The schema for the `<alg:SigningMethod>` element, and its corresponding `alg:SigningMethodType`
253 complex type, is as follows:

```
254 <element name="SigningMethod" type="alg:SigningMethodType"/>
255 <complexType name="SigningMethodType">
256   <sequence>
257     <any namespace="##any" processContents="lax" minOccurs="0"
258     maxOccurs="unbounded"/>
259   </sequence>
260   <attribute name="Algorithm" type="anyURI" use="required"/>
261   <attribute name="MinKeySize" type="positiveInteger"/>
262   <attribute name="MaxKeySize" type="positiveInteger"/>
263 </complexType>
```

264 2.5 Metadata Consumers

265 A consumer of metadata that wishes to perform XML Signature or XML Encryption operations with
266 knowledge of the peer entity (this is not always true of signatures) MUST consult the peer's metadata to
267 determine the intersection of the algorithms, key sizes, and other parameters as defined by particular
268 algorithms that it supports and that the peer entity supports.

269 The elements describing this support in metadata SHOULD be consulted in order, and the metadata
270 consumer SHOULD select the first algorithm encountered that it supports for use with a particular entity
271 (subject to local policy).

272 With respect to use of XML Signature, the presence of any `<alg:DigestMethod>` and
273 `<alg:SigningMethod>` elements at the level of a role element MUST take precedence over any such
274 elements at the level of an `<md:EntityDescriptor>` element, and the two sets are not combined if
275 both are present.

276 In the absence of an element describing support for a particular algorithm type (e.g., no
277 `<alg:DigestMethod>` elements), the metadata consumer is free to select any algorithm that it
278 supports. The absence of metadata therefore implies no information, rather than lack of support.

279 2.6 Security Considerations

280 The use of metadata as a means of "negotiating" the algorithms to use exposes both parties to attacks
281 traditionally associated with such mechanisms, such as step-down attacks in which the metadata is
282 compromised to influence the selection of a weaker algorithm than the parties might otherwise support.

283 The exchange and verification of metadata should always be subject to appropriate security controls to
284 mitigate this threat, and entities should always be prepared to reject the use of algorithms that they deem
285 insufficiently secure.

286 **2.7 Example**

287 The example presented shows a partial metadata instance for a service provider that supports (as a
288 relying party) a number of newer/stronger signature and digest algorithms defined in [RFC4051]. It also
289 specifies support for encryption via two AES variants using an RSA key as a transport.

```
290 <EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"  
291   xmlns:ds="http://www.w3.org/2000/09/xmldsig#"  
292   xmlns:alg="urn:oasis:names:tc:SAML:metadata:algsupport"  
293   entityID="https://serviceprovider.example.com/SAML">  
294   <Extensions>  
295     <alg:DigestMethod  
296       Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha384"/>  
297     <alg:DigestMethod  
298       Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"/>  
299     <alg:SigningMethod MinKeySize="256" MaxKeySize="511"  
300       Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256"/>  
301     <alg:SigningMethod MinKeySize="2048" MaxKeySize="4096"  
302       Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>  
303   </Extensions>  
304   <SPSSODescriptor  
305     protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">  
306     <KeyDescriptor>  
307       <ds:KeyInfo>...RSA key elided...</ds:KeyInfo>  
308       <EncryptionMethod  
309         Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>  
310       <EncryptionMethod  
311         Algorithm="http://www.w3.org/2001/04/xmlenc#aes256-cbc"/>  
312       <EncryptionMethod  
313         Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p"/>  
314     </KeyDescriptor>  
315     ...  
316   </SPSSODescriptor>  
317   ...  
318 </EntityDescriptor>
```

319 **3 Conformance**

320 **3.1 SAML V2.0 Metadata Profile for Algorithm Support Version 1.0**

321 A metadata producer conforms to this profile if it has the ability to produce metadata in accordance with
322 sections 2.3 and 2.4.

323 A metadata consumer conforms to this profile if it can consume extended metadata produced in
324 accordance with sections 2.3 and 2.4 and conforms to the normative statements in section 2.5.

325 **Appendix A. Acknowledgements**

326 The editors would like to acknowledge the contributions of the OASIS Security Services Technical
327 Committee, whose voting members at the time of publication were:

- 328 • Rob Philpott, EMC Corporation
- 329 • John Bradley, Individual
- 330 • Scott Cantor, Internet2
- 331 • Nate Klingenstein, Internet2
- 332 • Thomas Hardjono, M.I.T.
- 333 • Anthony Nadalin, Microsoft Corporation
- 334 • Thinh Nguyenphu, Nokia Siemens Networks Gmb
- 335 • Phil Hunt, Oracle Corporation
- 336 • Ari Kermaier, Oracle Corporation
- 337 • Hal Lockhart, Oracle Corporation
- 338 • Emily Xu, Oracle Corporation
- 339 • Anil Saldhana, Red Hat
- 340 • David Staggs, Veterans Health Administration

341 **Appendix B. Revision History**

- 342 ● Working Draft 01, first working draft.
- 343 ● Committee Draft 01, CD edits.
- 344 ● Working Draft 02, fix example, add processContents="lax" to wildcards in schema.