Conformance Requirements for the OASIS Security Assertion Markup Language (SAML) V2.0 – Errata Composite

Working Draft, 219 August 2006

Document identifier: sstc-saml-conformance-errata-2.0-wd-024


Editors:
- Prateek Mishra, Principal Identity
- Rob Philpott, RSA Security
- Jahan Moreh, Sigaba (errata document editor)
- Eve Maler, Sun Microsystems (errata composite document editor)

Contributors to the Errata:
- Nick Ragouzis, Enosis Group
- Thomas Wisniewski, Entrust
- Greg Whitehead, HP
- Heather Hinton, IBM
- Connor P. Cahill, Intel
- Scott Cantor, Internet2
- Eric Tiffany, Liberty Alliance
- Tom Scavo, NCSA/University of Illinois
- Jeff Hodges, Neustar
- Ari Kermaier, Oracle
- Prateek Mishra, Oracle
- Brian Campbell, Ping Identity
- Jim Lien, RSA Security
- Rob Philpott, RSA Security
- Jahan Moreh, Sigaba
- Emily Xu, Sun Microsystems
- David Staggs, Veteran's Health Administration

SAML V2.0 Contributors:
- Conor P. Cahill, AOL
- John Hughes, Atos Origin
- Hal Lockhart, BEA Systems
- Michael Beach, Boeing
- Rebekah Metz, Booz Allen Hamilton
- Rick Randall, Booz Allen Hamilton
- Thomas Wisniewski, Entrust
- Irving Reid, Hewlett-Packard
Abstract:
The SAML V2.0 Conformance specification provides the technical requirements for SAML V2.0 conformance and specifies the entire set of documents comprising SAML V2.0. This document, known as an "errata composite", combines corrections to reported errata with the original specification text. By design, the corrections are limited to clarifications of ambiguous or conflicting specification text. This document shows deletions from the original specification as struck-through text, and additions as blue underlined text. The "[PEnn]" designations embedded in the text refer to particular errata and their dispositions.

Status:
This errata composite document is a working draft based on the original OASIS Standard document that had been produced by the Security Services Technical Committee and approved by the OASIS membership on 1 March 2005. While the errata corrections appearing here are non-normative, they reflect the consensus of the TC about how to interpret the specification and are likely to be incorporated into any future standards-track revision of the SAML specifications.

This document includes errata corrections through revision 33 of the errata document plus decisions made at the 15 August 2006 SSTC meeting, including PE11, PE25, PE28, PE29, PE42, and PE50.

Committee members should submit comments and potential errata to the security-services@lists.oasis-open.org list. Others should submit them by following the instructions at http://www.oasis-open.org/committees/comments/form.php?wg_abbrev=security.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights web page for the Security Services TC (http://www.oasis-open.org/committees/security/ipr.php).
# Table of Contents

1 Introduction ............................................................................................................................................. 4
   1.1 Overview and Specification of SAML V2.0 ........................................................................................ 4
   1.2 Notation ............................................................................................................................................. 5
2 SAML V2.0 Profiles and Possible Implementations ............................................................................. 6
3 Conformance ............................................................................................................................................ 8
   3.1 Operational Modes ............................................................................................................................ 8
   3.2 Feature Matrix ..................................................................................................................................... 8
   3.3 Implementation of SAML-Defined Identifiers .................................................................................. 10
   3.4 Implementation of Encrypted Elements .......................................................................................... 11
   3.5 Security Models for SOAP and URI Bindings ............................................................................... 11
   3.6 [PE25]Metadata Structures ............................................................................................................ 11
   3.7 Metadata Interoperation .................................................................................................................. 11
4 XML Digital Signature and XML Encryption ...................................................................................... 13
   4.1 XML Signature Algorithms .............................................................................................................. 13
   4.2 XML Encryption Algorithms .......................................................................................................... 13
5 Use of SSL 3.0 or TLS 1.0 .................................................................................................................... 14
   5.1 SAML SOAP and URI Binding ....................................................................................................... 14
   5.2 Web SSO Profiles of SAML ............................................................................................................ 14
6 References .............................................................................................................................................. 15
1 Introduction

This normative specification describes features that are mandatory and optional for implementations claiming conformance to SAML V2.0 and also specifies the entire set of documents comprising SAML V2.0.

1.1 Overview and Specification of SAML V2.0

The SAML V2.0 standard consists of the following documents:

- This specification: Conformance Requirements for the OASIS Security Assertion Markup Language (SAML) V2.0
- Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0
  [SAMLCore]
  - SAML assertions schema [SAMLAssn-xsd]
  - SAML protocols schema [SAMLProt-xsd]
- Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAMLBind]
- Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAMLProf]
  - SAML ECP profile schema [SAMLECP-xsd]
  - SAML X.500/LDAP attribute profile schema [SAMLX500-xsd]
  - SAML DCE PAC attribute profile schema [SAMLDCE-xsd]
  - SAML XACML attribute profile schema [SAMLXAC-xsd]
- Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAMLMeta]
  - SAML metadata schema [SAMLMeta-xsd]
- Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAMLAuthnCxt]
  - SAML authentication context schema [SAMLAC-xsd]
  - SAML authentication context schema types [SAMLACTyp-xsd]
  - SAML context class schema for Internet Protocol [SAMLAC-IP]
  - SAML context class schema for Internet Protocol Password [SAMLAC-IPP]
  - SAML context class schema for Kerberos [SAMLAC-Kerb]
  - SAML context class schema for Mobile One Factor Unregistered [SAMLAC-MOFC]
  - SAML context class schema for Mobile Two Factor Unregistered [SAMLAC-MOFC]
  - SAML context class schema for Mobile One Factor Contract [SAMLAC-MOFC]
  - SAML context class schema for Mobile Two Factor Contract [SAMLAC-MOFC]
  - SAML context class schema for Password [SAMLAC-Pass]
  - SAML context class schema for Password Protected Transport [SAMLAC-PPT]
  - SAML context class schema for Previous Session [SAMLAC-Prev]
  - SAML context class schema for Public Key – X.509 [SAMLAC-X509]
  - SAML context class schema for Public Key – PGP [SAMLAC-PEP]
  - SAML context class schema for Public Key – SPKI [SAMLAC-SPKI]
  - SAML context class schema for Public Key – XML Signature [SAMLAC-XSig]
  - SAML context class schema for Smartcard [SAMLAC-Smart]
  - SAML context class schema for Smartcard PKI [SAMLAC-SmPKI]
  - SAML context class schema for Software PKI [SAMLAC-SwPKI]
The term “SAML V2.0” or “SAML2” is often used informally to refer to the standard specified by the above documents, or subsets thereof. However, the SAML V2.0 standard should be formally identified in other documents by a normative reference to this document.

Additional non-normative documents, such as a Technical Overview [SAMLTechOvw], are available to provide assistance to developers and others in understanding SAML. These documents are available at the SAML website, http://www.oasis-open.org/committees/security.

SAML V2.0 defines a number of named profiles. Each profile (other than attribute profiles) describes details of selected SAML message flows and can also be viewed as indivisible functionality that could be implemented by a software component. Implementation of a profile involves use of a binding for each message exchange included in the profile. A binding can be viewed as a specific implementation technique for achieving a message exchange.

Section 2 of this document enumerates all of the different profiles defined by [SAMLProfiles]. For each profile, the relevant SAML V2.0 message flows are listed, and for each message flow the set of possible bindings is also described. The combination of profile, message exchange and a selected binding is termed a SAML V2.0 feature.

Section 3 describes the conformance matrix for SAML V2.0. A number of different operational modes or roles are identified. The conformance matrix describes describes the feature set that must be implemented by each operational mode.

1.2 Notation

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted in this specification and all of the SAML V2.0 specifications as described in IETF RFC 2119 [RFC 2119]:

…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.
## SAML V2.0 Profiles and Possible Implementations

The following table enumerates all of the profiles defined by the SAML profiles specification [SAMLProf]. For each profile, the message protocol flows (defined in the assertions and protocols specification [SAMLCore]) found within the profile are also described. For each message flow, a list of relevant bindings (defined in the bindings specification [SAMLBind]) is given in the final column.

### Table 1: Possible Implementations

<table>
<thead>
<tr>
<th>Profile</th>
<th>Message Flows</th>
<th>Binding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web SSO</td>
<td><code>&lt;AuthnRequest&gt;</code> from SP to IdP</td>
<td>HTTP redirect</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td>IdP <code>&lt;Response&gt;</code> to SP</td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td>Enhanced Client/Proxy SSO</td>
<td>ECP to SP, SP to ECP to IdP</td>
<td>PAOS</td>
</tr>
<tr>
<td></td>
<td>IdP to ECP to SP, SP to ECP</td>
<td>PAOS</td>
</tr>
<tr>
<td>Identity Provider Discovery</td>
<td>Cookie setter</td>
<td>HTTP</td>
</tr>
<tr>
<td></td>
<td>Cookie getter</td>
<td>HTTP</td>
</tr>
<tr>
<td>Single Logout</td>
<td><code>&lt;LogoutRequest&gt;</code></td>
<td>HTTP redirect</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td></td>
<td><code>&lt;LogoutResponse&gt;</code></td>
<td>HTTP redirect</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td>Name Identifier Management</td>
<td><code>&lt;ManageNameIDRequest&gt;</code></td>
<td>HTTP redirect</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td></td>
<td><code>&lt;ManageNameIDResponse&gt;</code></td>
<td>HTTP redirect</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP POST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>HTTP artifact</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SOAP</td>
</tr>
<tr>
<td>[PE28] Artifact Resolution</td>
<td><code>&lt;ArtifactResolve&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td></td>
<td><code>&lt;ArtifactResponse&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>Authentication Query</td>
<td><code>&lt;AuthNQuery&gt;</code>, <code>&lt;Response&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>Profile</td>
<td>Message Flows</td>
<td>Binding</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Attribute Query</td>
<td><code>&lt;AttributeQuery&gt;, &lt;Response&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>Authorization Decision Query</td>
<td><code>&lt;AuthzDecisionQuery&gt;, &lt;Response&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>Assertion Query/Request</td>
<td>Artifact resolution: <code>&lt;ArtifactResolve&gt;, &lt;ArtifactResponse&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td></td>
<td>Authentication query: <code>&lt;AuthnQuery&gt;, &lt;Response&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attribute query: <code>&lt;AttributeQuery&gt;, &lt;Response&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Authorization decision query: <code>&lt;AuthzDecisionQuery&gt;, &lt;Response&gt;</code></td>
<td></td>
</tr>
<tr>
<td>Request for Assertion by Identifier</td>
<td><code>&lt;AssertionIDRequest&gt;, &lt;Response&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>Name Identifier Mapping</td>
<td><code>&lt;NameIDMappingRequest&gt;, &lt;NameIDMappingResponse&gt;</code></td>
<td>SOAP</td>
</tr>
<tr>
<td>[PE28]SAML URI binding</td>
<td>GET, HTTP Response</td>
<td>HTTP</td>
</tr>
<tr>
<td>UUID attribute profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCE PAC attribute profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X.500 attribute profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XACML attribute profile</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[PE28]Metadata</td>
<td>Consumption</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Exchange</td>
<td></td>
</tr>
</tbody>
</table>
3 Conformance

This section describes the technical conformance requirements for SAML V2.0.

3.1 Operational Modes

This document uses the phrase “operational mode” to describe a role that a software component can play in conforming to SAML. The operational modes are as follows:

- IdP – Identity Provider
- IdP Lite – Identity Provider Lite
- SP – Service Provider
- SP Lite – Service Provider Lite
- ECP – Enhanced Client/Proxy
- SAML Attribute Authority
- SAML Authorization Decision Authority
- SAML Authentication Authority
- SAML Requester

3.2 Feature Matrix

The following matrices identify unique sets of conformance requirements by means of a triple taken from Table 1 with the form: profile, message(s), binding. The message component is not always included when it is obvious from context.
### Table 2: Feature Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>IdP</th>
<th>IdP Lite</th>
<th>SP</th>
<th>SP Lite</th>
<th>ECP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web SSO, &lt;AuthnRequest&gt;, HTTP redirect</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Web SSO, &lt;Response&gt;, HTTP POST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Web SSO, &lt;Response&gt;, HTTP artifact</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Artifact Resolution, SOAP</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Enhanced Client/Proxy SSO, PAOS</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
</tr>
<tr>
<td>Name Identifier Management <a href="IdP-initiated">PE11</a>, HTTP redirect-(IdP-initiated)</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>N/A</td>
</tr>
<tr>
<td>Name Identifier Management (IdP-initiated), SOAP (IdP-initiated)</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>OPTIONAL</td>
<td>MUST NOT</td>
<td>N/A</td>
</tr>
<tr>
<td>Name Identifier Management (SP-initiated), HTTP redirect-(IdP-initiated)</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>N/A</td>
</tr>
<tr>
<td>Name Identifier Management (SP-initiated), SOAP (SP-initiated)</td>
<td>MUST</td>
<td>MUST NOT</td>
<td>OPTIONAL</td>
<td>MUST NOT</td>
<td>N/A</td>
</tr>
<tr>
<td>Single Logout (IdP-initiated),—HTTP redirect</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Single Logout (IdP-initiated),—SOAP</td>
<td>MUST</td>
<td>OPTIONAL</td>
<td>MUST</td>
<td>OPTIONAL</td>
<td>N/A</td>
</tr>
<tr>
<td>Single Logout (SP-initiated),—HTTP redirect</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>N/A</td>
</tr>
<tr>
<td>Single Logout (SP-initiated),—SOAP</td>
<td>MUST</td>
<td>OPTIONAL</td>
<td>MUST</td>
<td>OPTIONAL</td>
<td>N/A</td>
</tr>
<tr>
<td>Identity Provider Discovery (cookie)</td>
<td>MUST</td>
<td>MUST</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>N/A</td>
</tr>
<tr>
<td>[PE29]Request for Assertion by Identifier</td>
<td>OPTIONAL</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>SAML URI Binding</td>
<td>OPTIONAL</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>[PE25]Metadata Structures</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>N/A</td>
</tr>
<tr>
<td>Metadata Interoperation</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>N/A</td>
</tr>
</tbody>
</table>

The following table summarizes operational modes that extend the IdP or SP modes defined above. These are to be understood as a combination of an IdP or SP mode from the table above with the corresponding extended feature set below.
### Table 3: Extended IdP, SP

<table>
<thead>
<tr>
<th>Feature</th>
<th>IdP Extended</th>
<th>SP Extended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity Provider proxy (Section 3.4.1.5 [SAMLCore])</td>
<td>MUST</td>
<td>MUST</td>
</tr>
<tr>
<td>Name identifier mapping, SOAP</td>
<td>MUST</td>
<td>MUST</td>
</tr>
</tbody>
</table>

The following table summarizes conformance requirements for SAML authorities and requesters.

### Table 4: SAML Authority and Requester Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>SAML Authentication Authority</th>
<th>SAML Attribute Authority</th>
<th>SAML Authorization Decision Authority</th>
<th>SAML Requester</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication Query, SOAP</td>
<td>MUST</td>
<td>[PE42]OPTIONALN/A</td>
<td>OPTIONALN/A</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>Attribute Query, SOAP</td>
<td>OPTIONALN/A</td>
<td>MUST</td>
<td>OPTIONALN/A</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>Authorization Decision Query, SOAP</td>
<td>OPTIONALN/A</td>
<td>OPTIONALN/A</td>
<td>MUST</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>Request for Assertion by Identifier, SOAP</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>SAML URI Binding</td>
<td>MUST</td>
<td>MUST</td>
<td>MUST</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>[PE25]Metadata Structures</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>Metadata Interoperation</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
</tr>
</tbody>
</table>

### 3.3 Implementation of SAML-Defined Identifiers

All relevant operational modes MUST implement the following SAML-defined identifiers:
- All Attribute Name Format identifiers defined in Section 8.2 of [SAMLCore]
- All Name Identifier Format identifiers defined in Section 8.3 of [SAMLCore]

Conforming SAML implementations MUST permit the use of all identifier constants described in Sections 8.2 and 8.3 when producing and consuming SAML messages. SAML message producers MUST be able to create messages and SAML message consumers MUST be able to process messages with any of the constants defined in these sections.

Sections 8.3.7 (persistent name identifiers) and 8.3.8 (transient name identifiers) define normative processing rules for the producer of such identifiers. All normative processing rules in Sections 8.3.7 and 8.3.8 MUST be supported by conforming implementations. The remaining identifiers in Sections 8.2 and 8.3 specify no normative processing rules. Hence, generation and consumption of these identifiers is meaningful only when the generating and consuming parties have externally-defined agreement on the semantic interpretation of the identifiers.

**Note:** In this context, "process" means that the implementation must successfully parse and handle the identifier without failing or returning an error. How the implementation
deals with the identifier once it is processed at this level is out of scope for this

specification.

A SAML implementation may provide the facilities described above through direct

implementation support for the identifiers or through the use of supported programming

interfaces. Interfaces provided for this purpose must allow the SAML implementation to

be programmatically extended to handle all identifiers in Sections 8.2 and 8.3 that are not

natively handled by the implementation.

3.4 Implementation of Encrypted Elements

All relevant operational modes MUST be able to process or generate the following encrypted elements in

any context where they are required to process or generate the corresponding unencrypted elements, namely

<saml:NameID>, <saml:Assertion>, or <saml:Attribute>:

• <saml:EncryptedID>
• <saml:EncryptedAssertion>
• <saml:EncryptedAttribute>

3.5 Security Models for SOAP and URI Bindings

The following security models are mandatory to implement for all profiles implemented using the SOAP

binding as well as for the SAML URI binding. SAML authorities and requesters MUST implement the

following authentication methods:

• No client or server authentication.
• HTTP basic authentication [RFC 2617] with and without SSL 3.0 or TLS 1.0 (see Section 3 below).
  The SAML requester MUST preemptively send the authorization header with the initial request.
• HTTP over SSL 3.0 or TLS 1.0 server authentication with server-side certificate.
• HTTP over SSL 3.0 or TLS 1.0 mutual authentication with both server-side and a client-side
  certificate.

If a SAML authority uses SSL 3.0 or TLS 1.0, it MUST use a server-side certificate.

3.6 [PE25]Metadata Structures

Implementations claiming conformance to SAML V2.0 may declare each operational mode's conformance

to SAML V2.0 Metadata [SAMLMeta] through election of the Metadata Structures option.

With respect to each operational mode, such conformance entails the following:

• Implementing SAML metadata according to the extensible SAML V2.0 Metadata format in all cases
  where an interoperating peer has the option, as stated in SAML V2.0 specifications, of depending on
  the existence of SAML V2.0 Metadata. Electing the Metadata Structures option has the effect of
  requiring that such metadata be available to the interoperating peer. The Metadata Interoperation
  feature, described below, provides a means of satisfying this requirement.
• Referencing, consuming, and adhering to the SAML metadata, according to [SAMLMeta], of an
  interoperating peer when the known metadata relevant to that peer and the particular operation, and
  the current exchange, has expired or is no longer valid in cache, provided the metadata is available
  and is not prohibited by policy or the particular operation and that specific exchange.

3.7 Metadata Interoperation

Election of the Metadata Interoperation option requires the implementation to offer, in addition to any other
mechanism, the well-known location publication and resolution mechanism described in the SAML metadata specification [SAMLMeta].
4 XML Digital Signature and XML Encryption

SAML V2.0 uses XML Signature [XMLSig] to implement XML signing and encryption functionality for integrity, and source authentication. SAML V2.0 uses XML Encryption [XMLEnc] to implement confidentiality, including encrypted identifiers, encrypted assertions, and encrypted attributes. [PE50] The algorithms listed below as being required for SAML V2.0 conformance are based on the mandated algorithms in the W3C recommendations for XML Signature and for XML Encryption, but modified by the SSTC to ensure interoperability of conformant SAML implementations. While the SAML-defined set of algorithms is a minimal set for conformance, additional algorithms supported by XML Signature and XML Encryption MAY be used. Note, however, that the use of non-mandated algorithms may introduce interoperability issues if those algorithms are not widely implemented. As additional algorithms become mandated for use in XML Signature and XML Encryption, the set required for SAML conformance may be extended.

4.1 XML Signature Algorithms

XML Signature mandates use of the following algorithms in Section 6.1; therefore they MUST be implemented by compliant SAML V2.0 implementations:

- Digest: SHA1
- MAC: HMAC-SHA1
- XML Canonicalization:CanonicalXML (Without comments),
- Transform: Enveloped Signature

In addition, to enable interoperability, the following MUST be implemented by compliant SAML V2.0 implementations:

- Signature: RSAwithSHA1 (recommended in XML Signature but needed for interoperability)

Although XML Signature mandates the DSAwithSHA1 signature algorithm, it is not required by SAML V2.0, but is RECOMMENDED.

4.2 XML Encryption Algorithms

XML Encryption mandates use of the following algorithms in Sections 5.2.1 and 5.2.2; therefore they MUST be implemented by compliant SAML V2.0 implementations:

- Block Encryption: TRIPLE DES, AES-128, AES-256.
- Key Transport: RSA-v1.5, RSA-OAEP
5 Use of SSL 3.0 or TLS 1.0

In any SAML V2.0 use of SSL 3.0 [SSL3] or TLS 1.0 [RFC 2246], servers MUST authenticate to clients using a X.509 v3 certificate. The client MUST establish server identity based on contents of the certificate (typically through examination of the certificate’s subject DN field). [PPE50]The set of algorithms required for SAML V2.0 conformance is equivalent to that defined in SAML V1.0 and SAML V1.1. These mandated algorithms were chosen by the SSTC because of their wide implementation support in the industry. While the algorithms defined below are the minimal set for SAML conformance, additional algorithms supported by SSL 3.0 and TLS 1.0 MAY be used.

5.1 SAML SOAP and URI Binding

TLS-capable implementations MUST implement the TLS_RSA_WITH_3DES_EDE_CBC_SHA cipher suite and MAY implement the TLS_RSA_AES_128_CBC_SHA cipher suite [AES].

FIPS TLS-capable implementations MUST implement the corresponding TLS_RSA_FIPS_WITH_3DES_EDE_CBC_SHA cipher suite and MAY implement the corresponding TLS_RSA_FIPS_AES_128_CBC_SHA cipher suite [AES].

SSL-capable implementations MUST implement the SSL_RSA_WITH_3DES_EDE_CBC_SHA cipher suite.

FIPS SSL-capable implementations MUST implement the FIPS cipher suite corresponding to the SSL RSA_WITH_3DES_EDE_CBC_SHA cipher suite.

5.2 Web SSO Profiles of SAML

SSL-capable implementations of the Web SSO profile of SAML MUST implement the SSL_RSA_WITH_3DES_EDE_CBC_SHA cipher suite. TLS-capable implementations MUST implement the TLS_RSA_WITH_3DES_EDE_CBC_SHA cipher suite.
6 References


Appendix A. Acknowledgements

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

- Conor Cahill, AOL
- John Hughes, Atos Origin
- Hal Lockhart, BEA Systems
- Mike Beach, Boeing
- Rebekah Metz, Booz Allen Hamilton
- Rick Randall, Booz Allen Hamilton
- Ronald Jacobson, Computer Associates
- Gavenraj Sodhi, Computer Associates
- Thomas Wisniewski, Entrust
- Carolina Canales-Valenzuela, Ericsson
- Dana Kaufman, Forum Systems
- Irving Reid, Hewlett-Packard
- Guy Denton, IBM
- Heather Hinton, IBM
- Maryann Hondo, IBM
- Michael McIntosh, IBM
- Anthony Nadalin, IBM
- Nick Ragouzis, Individual
- Scott Cantor, Internet2
- Bob Morgan, Internet2
- Peter Davis, Neustar
- Jeff Hodges, Neustar
- Frederick Hirsch, Nokia
- Senthil Sengodan, Nokia
- Abbie Barbir, Nortel Networks
- Scott Kiester, Novell
- Cameron Morris, Novell
- Paul Madsen, NTT
- Steve Anderson, OpenNetwork
- Ari Kermoiaier, Oracle
- Vamsi Motukuru, Oracle
- Darren Platt, Ping Identity
- Prateek Mishra, Principal Identity
- Jim Lien, RSA Security
- John Linn, RSA Security
- Rob Philpott, RSA Security
- Dipak Chopra, SAP
- Jahan Moreh, Sigaba
- Bhavna Bhatnagar, Sun Microsystems
- Eve Maler, Sun Microsystems
The editors also would like to acknowledge the following former SSTC members for their contributions to this or previous versions of the OASIS Security Assertions Markup Language Standard:

- Stephen Farrell, Baltimore Technologies
- David Orchard, BEA Systems
- Krishna Sankar, Cisco Systems
- Zahid Ahmed, CommerceOne
- Tim Alsop, CyberSafe Limited
- Carlisle Adams, Entrust
- Tim Moses, Entrust
- Nigel Edwards, Hewlett-Packard
- Joe Pato, Hewlett-Packard
- Bob Blakley, IBM
- Marlena Erdos, IBM
- Marc Chanliau, Netegrity
- Chris McLaren, Netegrity
- Lynne Rosenthal, NIST
- Mark Skall, NIST
- Charles Knouse, Oblix
- Simon Godik, Overxeer
- Charles Norwood, SAIC
- Evan Prodromou, Securant
- Robert Griffin, RSA Security (former editor)
- Sai Allararpu, Sun Microsystems
- Gary Ellison, Sun Microsystems
- Chris Ferris, Sun Microsystems
- Mike Myers, Traceroute Security
- Phillip Hallam-Baker, VeriSign (former editor)
- James Vanderbeek, Vodafone
- Mark O'Neill, Vordel
- Tony Palmer, Vordel

Finally, the editors wish to acknowledge the following people for their contributions of material used as input to the OASIS Security Assertions Markup Language specifications:

- Thomas Gross, IBM
- Birgit Pfitzmann, IBM
Appendix B. Notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS’s procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.

Copyright © OASIS Open 2005. 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 does not need to be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document 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 OASIS or its successors or assigns.

This document and the information contained herein is provided on an “AS IS” basis and OASIS 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.