

## WS-Trust 1.3 Interoperability Profile: SAML 2.0 Token Profile

### Working Draft 01 07 January 2008

#### Specification URIs:

<urn:mace:switch.ch:doc:wst:profile:saml-200801>

#### This Version:

<http://switch.ch/grid/support/documents/wst-saml-wd01.pdf>

#### Previous Version:

No previous version

#### Latest Version:

<http://switch.ch/grid/support/documents/>

#### Editors:

Chad La Joie, SWITCH

#### Related Work:

This document relates to the WS-Trust Interoperability Profile [WST-Interop] and the Web Services Security: SAML Token Profile [WSS-SAML].

#### Abstract:

This profile provides the semantics for the use of a SAML 2.0 security token within messages that comply with the WS-Trust Interoperability Profile.

#### Status:

Working Draft

24 **Table of Contents**

25 1 Introduction.....3

26 1.1 Notation.....3

27 1.2 Normative References.....3

28 2 SAML 2 Token Processing.....5

29 2.1 <saml2:Assertion> Usage.....5

30 2.2 SAML 2 Token Claim Processing.....5

31 2.3 SAML 2 Token Creation.....5

32 3 Examples (informative).....6

33 3.1 Basic Issuance.....6

34 3.2 Delegate Issuance.....7

35 3.3 Renewal.....9

36 3.4 Cancellation.....12

37 3.5 Validation.....13

38

# 1 Introduction

This profile provides the semantics for the use of a SAML 2.0 security token within messages that comply with the WS-Trust Interoperability Profile.

## 1.1 Notation

This specification uses normative text to define an extension to the SAML V2.0 metadata specification.

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 [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.

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
ds:	http://www.w3.org/2000/09/xmldsig#	This is the XML Signature namespace [XMLSig].
saml2:	urn:oasis:names:tc:SAML:2.0:assertion	This is the SAML V2.0 assertion namespace defined in [SAML2].
wsse:	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd	This is the WS-Security extension 1.0 namespace defined in [WSS].
wst:	http://docs.oasis-open.org/ws-trust/200512	This is the WS-Trust 1.3 namespace defined in [WS-Trust].
wsu:	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd	This is the WS-Security utility namespace defined in [WSS].
xenc:	http://www.w3.org/2001/04/xmlenc#	This is the XML Encryption namespace defined in [XMLEnc].

58

This specification uses the following typographical conventions in text: <WSTrustElement>, <ns:ForeignElement>, Attribute, **Datatype**, OtherKeyword.

## 1.2 Normative References

[RFC 2119] S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF RFC 2119, March 1997. See <http://www.ietf.org/rfc/rfc2119.txt>.

64	<b>[SAML2]</b>	S. Cantor. <i>Assertions and Protocol for OASIS Security Assertion Markup Language (SAML) V2.0</i> . OASIS, 15 March 2006. See <a href="http://docs.oasis-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</a> .
65		
66		
67	<b>[WSI-BSP]</b>	M. McIntosh, et al. <i>Basic Security Profile Version 1.0</i> . Web Services Interoperability Organization, 3 March 2007. See <a href="http://www.wsi.org/Profiles/BasicSecurityProfile-1.0.html">http://www.wsi.org/Profiles/BasicSecurityProfile-1.0.html</a> .
68		
69		
70	<b>[WSS]</b>	A. Nadalin, et al. <i>Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)</i> . OASIS, 1 February 2004. See <a href="http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf">http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf</a> .
71		
72		
73		
74	<b>[WSS-SAML]</b>	R. Monzillo, et al. <i>Web Services Security: SAML Token Profile 1.1</i> . OASIS, 1 November 2006. See <a href="http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-errata-os-SAMLSAMLTokenProfile.pdf">http://docs.oasis-open.org/wss/v1.1/wss-v1.1-spec-errata-os-SAMLSAMLTokenProfile.pdf</a> .
75		
76		
77	<b>[WS-Trust]</b>	A. Nadalin, et al. <i>WS-Trust 1.3</i> . OASIS, 19 March 2007. See <a href="http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.pdf">http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.pdf</a> .
78		
79	<b>[WST-Interop]</b>	C. La Joie. <i>WS-Trust Interoperability Profile</i> . See
80	<b>[XMLEnc]</b>	D. Eastlake, et al. <i>XML Encryption Syntax and Processing</i> . World Wide Web Consortium, 10 December 2002. See <a href="http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/">http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/</a> .
81		
82		
83	<b>[XMLSig]</b>	D. Eastlake, et al. <i>XML-Signature Syntax and Processing</i> , World Wide Web Consortium, February 2002. See <a href="http://www.w3.org/TR/xmlsig-core/">http://www.w3.org/TR/xmlsig-core/</a> .
84		

## 85 2 SAML 2 Token Processing

### 86 Token Identification:

87 <http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.1#SAMLV2.0>

### 88 2.1 <saml2:Assertion> Usage

89 In order to improve the chance of interoperability when using the WSS: SAML Token Profile 1.1 [WSS-  
90 SAML] defined <saml2:Assertion> such tokens MUST also conform to the WS-I Basic Security  
91 Profile 1.0 [WSI-BSP], section 15. Additionally, this profile adds the following restrictions:

- 92 ● The assertion MUST contain a <saml2:Subject>.
- 93 ● The assertion SHOULD identify any intended recipient of the token through either (or both):
  - 94 ○ A <saml2:AudienceRestriction>
  - 95 ○ A holder of key subject confirmation method
- 96 ● The assertion SHOULD exactly one <saml2:AuthnStatement>.

### 97 2.2 SAML 2 Token Claim Processing

98 When a <saml2:Assertion> token is used as a claim within a request the following steps should be  
99 taken to validate the claim:

- 100 ● If the assertion contains `NotBefore` and `NotOnOrAfter` conditions the token service MUST  
101 verify that the message was not issued or before or on or after, respectively, the given time within  
102 an allowed clock skew.
- 103 ● If the assertion contains a <saml2:AudienceRestriction> the token service MUST verify  
104 that it is listed as an intended audience.
- 105 ● If the assertion is signed then the signature MUST be validated as described by the SAML 2  
106 specification.
- 107 ● If the assertion employs a holder-of-key confirmation method the token service must verify the  
108 client is in possession of the private key as described in [WST-Interop] section 9.

### 109 2.3 SAML 2 Token Creation

110 When a token service issues a SAML 2 token it MUST produce a token with the following criteria:

- 111 ● The assertion MUST contain a <saml2:Subject>.
- 112 ● The assertion MUST contain exactly one <saml2:AuthnStatement>. The  
113 <saml2:AuthnContext> SHOULD reflect the type of claim during the request.
- 114 ● The assertion MUST contain `NotBefore` and `NotOnOrAfter` conditions.
- 115 ● The assertion MUST identify the intended recipient(s) of the assertion within a  
116 <saml2:AudienceRestriction>
- 117 ● If the underlying transport does not provide confidentiality the name ID and any attributes  
118 SHOULD be encrypted with the recipient's key.

- 119       ● If the underlying transport does not provide integrity the assertion SHOULD be signed.

### 120 **2.3.1 Extension Parameters**

121 The following request extensions parameters MAY be used to convey additional information to the token  
122 service.

123 `/wst:RequestSecurityToken/<wst:Participants>`

124       This extension parameter may be used, by the requester, to indicate intended recipients of the  
125       assertion. If this parameter is used the token service MAY choose to release only the amount of data  
126       common to all participants or MAY choose to encrypt the data for each recipients if the recipients'  
127       keys are known. Either way, the token service MUST list all participants within the assertion's  
128       audience restriction.

### 129 **2.4 Token Renewal**

130 When a SAML token is renewed the assertion's `IssueInstant` as well as `NotBefore` and  
131 `NotOnOrAfter` data should be updated to reflect the instance of renewal. The contained authentication  
132 statement should remain unmodified however other statements MAY be updated. For example, if the  
133 assertion contains an attribute statement new attributes may be added or old attributes may be removed.

### 134 **2.5 Delegation Support**

135 SAML assertions MAY be delegated.

136 If a created token is meant to be delegated the following additional creation requirements MUST be met.

- 137       ● A `<saml2:SubjectConfirmation>` that contains a `<saml2:NameID>` identifying the intended  
138       recipient of the delegate token as give in the `<wst:DelegateTo>` request parameter. Additional  
139       subject confirmation data MAY be included.

140 It is recommended that a holder-of-key confirmation method be used on all delegate tokens.

### 141 3 Example (informative)

142 This example shows a SAML security token that employs the holder-of-key subject confirmation method.

143 Namespace declarations and base64-encoded content are not shown in the examples for the sake of  
144 brevity.

145

```
146 <saml:Assertion IssueInstant="1970-01-01T00:00:00" ID="99999" Version="2.0">
147   <saml:Issuer>urn:example.org:idp</saml:Issuer>
148   <saml:Subject>
149     <saml:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-
150 format:emailAddress">
151       jdoe@example.org
152     </saml:NameID>
153     <saml:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:holder-
154 of-key">
155       <saml:SubjectConfirmationData
156 xsi:type="saml:KeyInfoConfirmationDataType">
157         <ds:KeyInfo>
158           <ds:X509Data>
159             <ds:X509Certificate>Base64-encoded X.509 cert</ds:X509Certificate>
160           </ds:X509Data>
161         </ds:KeyInfo>
162       </saml:SubjectConfirmationData>
163     </saml:SubjectConfirmation>
164   </saml:Subject>
165   <saml:AuthnStatement AuthnInstant="1970-01-01T00:00:00">
166     <saml:AuthnContext>
167       <saml:AuthnContextClassRef>
168 urn:oasis:names:tc:SAML:2.0:ac:classes>PasswordProtectedTransport
169       </saml:AuthnContextClassRef>
170     </saml:AuthnContext>
171   </saml:AuthnStatement>
172   <saml:AttributeStatement>
173     <saml:Attribute Name="Attribute1">
174       <saml:AttributeValue>value1</saml:AttributeValue>
175     </saml:Attribute>
176     <saml:Attribute Name="Attribute2">
177       <saml:AttributeValue>value1</saml:AttributeValue>
178     </saml:Attribute>
179   </saml:AttributeStatement>
180
181   <saml:Conditions NotBefore="1970-01-01T00:00:00" NotOnOrAfter="1970-01-
182 01T01:00:00" />
183 </saml:Assertion>
184
```