



Subject-based Profiles for SAML V1.1 Assertions

Working Draft 01

18 December 2007

Specification URIs:

This Version:

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject-draft-01.html>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject-draft-01.odt>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject-draft-01.pdf>

Previous Version:

NA

Latest Version:

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject.html>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject.odt>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-saml1-profiles-assertion-subject.pdf>

Technical Committee:

OASIS Security Services TC

Chair(s):

Hal Lockhart, BEA Systems, Inc.

Brian Campbell, Ping Identity Corporation

Editor(s):

Tom Scavo, National Center for Supercomputing Applications (NCSA)

Related Work:

NA

Declared XML Namespace(s):

urn:oasis:names:tc:SAML:1.1:profiles:assertion:subject

Abstract:

This profile places constraints upon SAML V1.1 subjects and assertions so that they have properties similar to SAML V2.0 subjects and assertions.

Status:

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

TC members should send comments on this specification to the TC's email list. Others

36 should send comments to the TC by using the “Send A Comment” button on the TC’s
37 web page at <http://www.oasis-open.org/committees/security>.
38 For information on whether any patents have been disclosed that may be essential to
39 implementing this specification, and any offers of patent licensing terms, please refer to the IPR
40 section of the TC web page (<http://www.oasis-open.org/committees/security/ipr.php>).
41 The non-normative errata page for this specification is located at [http://www.oasis-
open.org/committees/security](http://www.oasis-
42 open.org/committees/security).

43 Notices

44 Copyright © OASIS Open 2007. All Rights Reserved.

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

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

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

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

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

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

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

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

87 **Table of Contents**

88 1 Introduction..... 5

89 1.1 Terminology..... 5

90 1.2 Outline..... 5

91 1.3 Normative References..... 6

92 1.4 Non-Normative References..... 6

93 2 SAML V1.1 Subject Profile..... 7

94 2.1 Required Information..... 7

95 2.2 Profile Description..... 7

96 2.3 Usage of <saml:Subject> Element..... 7

97 2.4 Example..... 7

98 2.5 Strongly Matching Subjects..... 8

99 3 SAML V1.1 Subject-based Assertion Profile..... 9

100 3.1 Required Information..... 9

101 3.2 Profile Description..... 9

102 3.3 Usage of <saml:Assertion> Element..... 9

103 3.4 Example..... 10

104 3.5 Complex type SubjectStatementType..... 11

105 4 Implementation Conformance..... 13

106 5 Acknowledgments..... 14

107 6 Revision History..... 15

108

109 1 Introduction

110 This profile places constraints upon SAML V1.1 subjects and assertions so that they have properties
111 similar to SAML V2.0 subjects and assertions.

112 1.1 Terminology

113 This specification uses normative text to describe the contents of conforming SAML subjects and
114 assertions.

115 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD
116 NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as
117 described in [RFC 2119]:

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

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

123 Listings of XML schemas appear like this.

124 Example code listings appear like this.

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

| <i>Prefix</i> | <i>XML Namespace</i> | <i>Comments</i> |
|---------------|--|---|
| saml: | urn:oasis:names:tc:SAML:1.1:assertion | This is the SAML V1.1 assertion namespace [SAMLCore]. |
| saml2: | urn:oasis:names:tc:SAML:2.0:assertion | This is the SAML V2.0 assertion namespace [SAML2Core]. |
| samlsap: | urn:oasis:names:tc:SAML:1.1:profiles:assertion:subject | This is the SAML V1.1 subject-based assertion namespace defined by this document and its accompanying schema [SAMSAP-XSD]. |
| ds: | http://www.w3.org/2000/09/xmldsig# | This is the W3C XML Signature namespace, defined in the XML-Signature Syntax and Processing specification [XMLSig] and schema [XMLSig-XSD]. |
| xs: | http://www.w3.org/2001/XMLSchema | This is the XML Schema namespace [Schema1]. This is the default namespace used throughout this document. |
| xsi: | http://www.w3.org/2001/XMLSchema-instance | This is the XML Schema namespace for schema-related markup that appears in XML instances [Schema1]. |

129 This specification uses the following typographical conventions in text: <UnqualifiedElement>,
130 <ns:QualifiedElement>, Attribute, **Datatype**, OtherKeyword.

131 1.2 Outline

132 Section 2 describes a profile that constrains SAML V1.1 subjects so that they have properties similar to
133 SAML V2.0 subjects. Section 3 describes a profile that places constraints upon SAML V1.1 assertions so

134 that they have properties similar to SAML V2.0 assertions. Finally, section 4 specifies requirements that all
135 conforming implementations must follow.

136 **1.3 Normative References**

- 137 **[RFC 2119]** S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF
138 RFC 2119, March 1997. See <http://www.ietf.org/rfc/rfc2119.txt>
- 139 **[SAML2Core]** S. Cantor et al. *Assertions and Protocols for the OASIS Security Assertion
140 Markup Language (SAML) V2.0*. OASIS Standard, March 2005. See
141 <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- 142 **[SAMLCore]** E. Maler et al. *Assertions and Protocols for the OASIS Security Assertion Markup
143 Language (SAML) V1.1*. OASIS Standard, September 2003. Document ID oasis-
144 sstc-saml-core-1.1. See [http://www.oasis-
open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf](http://www.oasis-
145 open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf)
- 146 **[SAMSAP-XSD]** *Schema for Subject-based Profiles for SAML V1.1 Assertions*. OASIS, December
147 2007. Document ID sstc-saml1-profiles-assertion-subject.xsd. See
148 http://www.oasis-open.org/committees/documents.php?wg_abbrev=security
- 149 **[Schema1]** H. S. Thompson et al. *XML Schema Part 1: Structures*. World Wide Web
150 Consortium Recommendation, May 2001. See [http://www.w3.org/TR/2001/REC-
xmlschema-1-20010502/](http://www.w3.org/TR/2001/REC-
151 xmlschema-1-20010502/)
- 152 **[XMLSig]** D. Eastlake et al. *XML-Signature Syntax and Processing*. World Wide Web
153 Consortium Recommendation, February 2002. See
154 <http://www.w3.org/TR/2002/REC-xmlsig-core-20020212/>
- 155 **[XMLSig-XSD]** *Schema for XML Signatures*. World Wide Web Consortium Recommendation,
156 February 2002. See [http://www.w3.org/TR/2002/REC-xmlsig-core-
20020212/xmlsig-core-schema.xsd](http://www.w3.org/TR/2002/REC-xmlsig-core-
157 20020212/xmlsig-core-schema.xsd)

158 **1.4 Non-Normative References**

- 159 **[MACEAttrib]** S. Cantor et al. *MACE-Dir SAML Attribute Profiles*. Internet2 MACE, April 2006.
160 See [http://middleware.internet2.edu/dir/docs/internet2-mace-dir-saml-attributes-
200604.pdf](http://middleware.internet2.edu/dir/docs/internet2-mace-dir-saml-attributes-
161 200604.pdf)
- 162 **[RFC2246]** T. Dierks and C. Allen. *The TLS Protocol Version 1.0*. IETF RFC 2246, January
163 1999. See <http://www.ietf.org/rfc/rfc2246.txt>

2 SAML V1.1 Subject Profile

This SAML V1.1 *Subject Profile* constrains SAML V1.1 subjects so that they have properties similar to SAML V2.0 subjects.

2.1 Required Information

Identification:

urn:oasis:names:tc:SAML:1.1:profiles:subject

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

Description: Given below.

Updates: N/A

Extends: N/A

2.2 Profile Description

This profile specifies a SAML V1.1 `<saml:Subject>` element that can be readily mapped to SAML V2.0.

2.3 Usage of `<saml:Subject>` Element

Neither SAML V1.1 nor SAML V2.0 explicitly requires a name identifier, but certain SAML V2.0 profiles (most notably the Single Logout Profile) implicitly require one, so a `<saml:Subject>` element that conforms to this profile SHOULD contain a `<saml:NameIdentifier>` element. To further align with SAML V2.0, the `NameQualifier` attribute on the `<saml:NameIdentifier>` element SHOULD be omitted unless the identifier's type definition explicitly defines its use and semantics. In particular, if the `Format` attribute on the `<saml:NameIdentifier>` element has a value specified in section 7.3 of [SAMLCore], the `NameQualifier` attribute SHOULD be omitted.

Certain deprecated features of SAML V1.1 were removed in SAML V2.0. Thus a `<saml:Subject>` that conforms to this profile MUST NOT contain a `<saml:NameIdentifier>` element with any of the following `Format` attribute values:

- urn:oasis:names:tc:SAML:1.0:assertion#emailAddress
- urn:oasis:names:tc:SAML:1.0:assertion#X509SubjectName
- urn:oasis:names:tc:SAML:1.0:assertion#WindowsDomainQualifiedName

See section 7.3 of [SAMLCore] for the URIs to be used in lieu of these deprecated values.

In SAML V1.1, a `<saml:Subject>` element contains at most one `<saml:SubjectConfirmation>` element containing one or more `<saml:ConfirmationMethod>` elements. In SAML V2.0, on the other hand, there may be multiple `<saml2:SubjectConfirmation>` elements, each with a required `Method` attribute. Therefore, a `<saml:Subject>` element that conforms to this profile MAY contain a `<saml:SubjectConfirmation>` element, but that element MUST contain one and only one `<saml:ConfirmationMethod>` element.

2.4 Example

```
<!-- SAML V1.1 Subject -->
<saml:Subject>
  <saml:NameIdentifier
    Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">
    C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
```

```

203 </saml:NameIdentifier>
204 <saml:SubjectConfirmation>
205   <saml:ConfirmationMethod>
206     urn:oasis:names:tc:SAML:1.0:cm:holder-of-key
207   </saml:ConfirmationMethod>
208   <ds:KeyInfo>
209     <ds:X509Data>
210       <!-- subject's X.509 cert -->
211       <ds:X509Certificate>
212 MIICiDCCAXACCQDE+9eiWrm62jANBgkqhkiG9w0BAQQFADBFMQswCQYDVQQGEwJV
213 UzESMBAGA1UEChMJTkNTQS1URVNUMQ0wCwYDVQQLEwRvc2VyMwEwEwYDVQDEwP
214 UC1TZXJ2aWNlMmB4XDTA2MDcxNzIwMjE0MVoXDTA2MDcxODIwMjE0MVowSzELMAkG
215 A1UEBHMCMVVMxejAQBGNVBAoTCU5DU0EtVEVTVDENMAsGA1UECXMVXN1cjEzMBcG
216 A1UEAwwQdHJzY2F2b0B1aXVjLmVkdTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkC
217 gYEA9QMe41Rl3XbWpCflbCjGK9gty6zBJmp+tsaJINM0VaBaZ3t+tSXknelyife
218 nCc2O3yaX76aq53QMxy+5wKQYe8Rzdw28Nv3a73wfjXJXoUhGkvERcscs9EfIWCc
219 g2bH0g8uSh+Fbv3lHih4lBJ5MCS2buJfsR7dlr/xsadU2RcCAwEAATANBgkqhkiG
220 9w0BAQQFAAOCAQEAdyIcMTob7TVkelfJ7+I1j0LO24U1KvbLzd2OPvcFTcv6fVHx
221 EjK0QxaZXJhreZ6+rIdiMXrEz1RdJESNMxtDW8++sVp6avoB5EXly3ez+CEAIL4g
222 cjvKZUR4dMryWshWIBHKFFul+r7urUgvWI12KbMeE9KP+kiiiiTskLcKgFzngw1J
223 selmHhTcTCrcDocn5yO2+d3dog52vSotVFDBsBuvDixO2hv679JR6Hlqjtk4GExp
224 E9iVI0wdPE038uQIJJTXlshMMLvUGVh/c0ReJbn92Vj4dI/yy6PtY/8ncYLYNkjg
225 oVN0J/ymOktn9lTlFyTiuY4OuJsZRO1+zWLy9g==
226     </ds:X509Certificate>
227   </ds:X509Data>
228   </ds:KeyInfo>
229 </saml:SubjectConfirmation>
230 </saml:Subject>

```

231 2.5 Strongly Matching Subjects

232 In general, the notion of **strongly matches** defined in section 3.4.4 of [SAMLCore] is overly restrictive, for
233 at least two reasons: 1) a `<saml:NameIdentifier>` element with no `Format` attribute is semantically
234 equivalent to a `<saml:NameIdentifier>` element with `Format` equal to
235 "urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified", and 2) a `<saml:SubjectConfirmation>`
236 element with confirmation method "urn:oasis:names:tc:SAML:1.0:cm:holder-of-key" must have a
237 `<ds:KeyInfo>` element, but two distinct `<ds:KeyInfo>` elements can refer to the same key, so two
238 distinct `<saml:SubjectConfirmation>` elements can be semantically equivalent.

239 For these reasons, this profile adopts the following alternate definition (which aligns with SAML V2.0):

240 [insert working definition of strongly matches here]

241 Like the definition of strongly matches in [SAMLCore], the above relation is not symmetric since S1
242 strongly matches S2 does not imply that S2 strongly matches S1. In other words, the order of operands
243 S1,S2 matters.

244 3 SAML V1.1 Subject-based Assertion Profile

245 This *SAML V1.1 Subject-based Assertion Profile* places constraints upon SAML V1.1 assertions so that
246 they have properties similar to SAML V2.0 assertions.

247 In SAML V1.1, each statement contains a `<saml:Subject>` element, but in SAML V2.0, there is one
248 `<saml2:Subject>` element per assertion. Thus, in SAML V2.0, every statement necessarily applies to
249 the same subject. To achieve an equivalent semantic in SAML V1.1, this profile places suitable
250 restrictions on multi-statement assertions.

251 See section 2 of the SAML V1.1 Assertions and Protocols specification [SAMLCore] for general
252 requirements regarding SAML assertions. Where this profile conflicts with [SAMLCore], the former takes
253 precedence.

254 3.1 Required Information

255 Identification:

256 `urn:oasis:names:tc:SAML:1.1:profiles:assertion:subject`

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

258 **Description:** Given below.

259 **Updates:** N/A

260 **Extends:** N/A

261 3.2 Profile Description

262 This profile places the following constraints upon conforming assertions:

- 263 • Deprecated elements must not be used.
- 264 • Each statement of the assertion must have a `<saml:Subject>` element.
- 265 • Each `<saml:Subject>` element must satisfy the SAML Subject Profile described in section 2.
266 Moreover, each pair of `<saml:Subject>` elements must **very strongly match**, a notion made
267 precise in the next section.

268 Such an assertion is called a *subject-based assertion*.

269 3.3 Usage of `<saml:Assertion>` Element

270 An assertion that conforms to this profile MUST satisfy the following general requirements:

- 271 • The assertion MUST NOT contain a `<saml:AuthorityBinding>` element.
- 272 • Every statement in the assertion MUST have a type derived from abstract type
273 **saml:SubjectStatementAbstractType** [SAMLCore].
- 274 • The `<saml:Subject>` element of each statement MUST satisfy the SAML Subject Profile
275 described in section 2.
- 276 • If the `<saml:Assertion>` element contains more than one statement, each pair of
277 `<saml:Subject>` elements MUST **very strongly match**, which we now define. Let S1 and S2 be
278 two `<saml:Subject>` elements. S1 *very strongly matches* S2 if S1 strongly matches S2 and S2
279 strongly matches S1. Note that this definition depends on the notion of *strongly matches* defined in
280 the previous section.

281 An assertion is **valid** according to this profile if and only if it satisfies the above requirements.


```

347         urn:oasis:names:tc:SAML:1.0:cm:holder-of-key
348     </saml:ConfirmationMethod>
349     <ds:KeyInfo>
350         <ds:X509Data>
351             <!-- subject's X.509 cert -->
352             <ds:X509Certificate>
353 MIICiDCCAXACCQDE+9eiWrm62jANBgkqhkiG9w0BAQQFADBFMQswCQYDVQQGEwJV
354 UzESMBAGA1UEChMJKNTQs1URVNUMQ0wCwYDVQQLEwRvc2VyMRMwEQYDVQQDEwpT
355 UC1TZXJ2aWNlMB4XDTA2MDcxNzIwMjE0MVVoXDTA2MDcxODIwMjE0MVVowSzELMAkG
356 A1UEBHMCMVVMxEjAQBgNVBAoTCU5DU0EtVEVTVDENMAsGA1UECxEVXN1cjEzMBcG
357 A1UEAwQdHjZy2F2b0B1aXVjLmVkdTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkC
358 gYEA9QMe4lRl3XbWPCflbCjGK9gty6zBJmp+tsaJINM0VaBaZ3t+tSXknelYife
359 nCc2O3yaX76aq53QMxy+5wKQYe8Rzdw28Nv3a73wfjXJXoUhGkveRcscs9EfiWcC
360 g2bHog8uSh+Fbv3lHih4lBJ5MCS2buJfsR7dlr/xsadU2RcCAwEAATANBgkqhkiG
361 9w0BAQQFAAOCAQEAdyIcMTob7TVkelfJ7+I1j0LO24U1KvbLzd2OPvcFTcV6fVHx
362 Ejk0QxaZXJhreZ6+rIdiMXrEz1RdJESNMxtDW8++sVp6avoB5EX1y3ez+CEAIL4g
363 cJvKZUR4dMryWshWIBHKFFul+r7urUgvWI12KbMeE9KP+kiiiiTskLcKgFzngw1J
364 selmHhTcTcCrcDocn5yO2+d3dog52vSotVFDBsBuvDixO2hv679JR6Hlqjtk4GExp
365 E9iVI0wdPE038uQIJJTXlshMMLvUGVh/c0ReJBn92Vj4dI/yy6PtY/8ncYLYNkjg
366 oVN0J/ymOktn9lTlFyTiuY4OuJsZR01+zWLy9g==
367         </ds:X509Certificate>
368     </ds:X509Data>
369 </ds:KeyInfo>
370 </saml:SubjectConfirmation>
371 </saml:Subject>
372 <saml:Attribute
373     AttributeNamespace="urn:mace:shibboleth:1.0:attributeNamespace:uri"
374     AttributeName="urn:mace:dir:attribute-def:eduPersonPrincipalName">
375     <saml:AttributeValue Scope="uiuc.edu">
376         trscavo
377     </saml:AttributeValue>
378 </saml:Attribute>
379 <saml:Attribute
380     AttributeNamespace="urn:mace:shibboleth:1.0:attributeNamespace:uri"
381     AttributeName="urn:mace:dir:attribute-def:givenName">
382     <saml:AttributeValue xsi:type="xs:string">
383         Tom
384     </saml:AttributeValue>
385 </saml:Attribute>
386 <saml:Attribute
387     AttributeNamespace="urn:mace:shibboleth:1.0:attributeNamespace:uri"
388     AttributeName="urn:mace:dir:attribute-def:sn">
389     <saml:AttributeValue xsi:type="xs:string">
390         Scavo
391     </saml:AttributeValue>
392 </saml:Attribute>
393 <saml:Attribute
394     AttributeNamespace="urn:mace:shibboleth:1.0:attributeNamespace:uri"
395     AttributeName="urn:mace:dir:attribute-def:mail">
396     <saml:AttributeValue xsi:type="xs:string">
397         trscavo@gmail.com
398     </saml:AttributeValue>
399 </saml:Attribute>
400 </saml:AttributeStatement>
401 <ds:Signature>...</ds:Signature>
402 </saml:Assertion>

```

403 The attributes in the above example conform to the MACE-Dir Attribute Profile for SAML 1.x [MACEAttrib]
404 and are for illustration purposes only.

405 3.5 Complex type SubjectStatementType

406 Recall that a SAML V1.1 assertion contains at least one statement. SAML V2.0, on the other hand,
407 permits empty assertions, that is, subject-based assertions with no statements. To duplicate this capability
408 in SAML V1.1, we define a trivial extension of **saml:SubjectStatementAbstractType**:

```
409 <complexType name="SubjectStatementType">
410 <complexContent>
411 <extension base="saml:SubjectStatementAbstractType"/>
412 </complexContent>
413 </complexType>
```

414 The following example illustrates a `<saml:Assertion>` containing a `<saml:SubjectStatement>` of
415 type **saml:SubjectStatementType**.

```
416 <saml:Assertion
417   xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
418   xmlns:samlsoap="urn:oasis:names:tc:SAML:1.1:profiles:assertion:subject"
419   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
420   AssertionID="cT S T-vKMwidT8 Pzkke8UkC68."
421   IssueInstant="2006-07-17T20:31:41Z"
422   Issuer="https://idp.example.org/saml"
423   MajorVersion="1" MinorVersion="1">
424   <saml:Conditions
425     NotBefore="2006-07-17T20:31:41Z"
426     NotOnOrAfter="2006-07-18T20:21:41Z">
427   </saml:Conditions>
428   <saml:SubjectStatement
429     xsi:type="samlsoap:SubjectStatementType">
430     <saml:Subject>
431       <saml:NameIdentifier
432         Format="urn:oasis:names:tc:SAML:1.1:nameid-
433 format:X509SubjectName">
434         C=US, O=NCSA-TEST, OU=User, CN=trscavo@uiuc.edu
435       </saml:NameIdentifier>
436     </saml:Subject>
437   </saml:SubjectStatement>
438 </saml:Assertion>
```

439 Note that the above `<saml:SubjectStatement>` element has no content apart from a
440 `<saml:Subject>` element.

441 **4 Implementation Conformance**

442 An implementation of this specification shall be ...

443 **5 Acknowledgments**

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

- 446
- 447 • Hal Lockhart, BEA Systems, Inc.
- 448 • Steve Anderson, BMC Software
- 449 • Rob Philpott, EMC Corporation
- 450 • Anthony Nadalin, IBM
- 451 • Scott Cantor, Internet2
- 452 • Bob Morgan, Internet2
- 453 • Tom Scavo, National Center for Supercomputing Applications (NCSA)
- 454 • Jeff Hodges, NeuStar, Inc.
- 455 • Frederick Hirsch, Nokia Corporation
- 456 • Abbie Barbir, Nortel
- 457 • Paul Madsen, NTT Corporation
- 458 • Ari Kermaier, Oracle Corporation
- 459 • Prateek Mishra, Oracle Corporation
- 460 • Brian Campbell, Ping Identity Corporation
- 461 • Eve Maler, Sun Microsystems
- 462 • Emily Xu, Sun Microsystems
- 463 • David Staggs, Veteran's Health Administration
- 464 • Charles Knouse, Hewlett-Packard
- 465 • Kent Spaulding, Tripod Technology Group, Inc.
- 466 • Anil Saldhana, Red Hat
- 467 • Eric Tiffany, Liberty Alliance Project

468

6 Revision History

| <i>Document ID</i> | <i>Date</i> | <i>Committer</i> | <i>Comment</i> |
|--|-------------|------------------|----------------|
| sstc-saml1-profiles-assertion-subject-draft-01 | 17 Dec 2007 | T. Scavo | Initial draft |

469

•