



SAML V2.0 Holder-of-Key Assertion Profile

Working Draft 0908, 1120 January 2009

Specification URIs:

TBD

Technical Committee:

OASIS Security Services TC

Chair(s):

Hal Lockhart, BEA Systems, Inc.

Brian Campbell, Ping Identity Corporation

Editors:

Tom Scavo, National Center for Supercomputing Applications (NCSA)

Contributors:

Nate Klingenstein, Internet2

Scott Cantor, Internet2

Declared XML Namespace(s):

`urn:oasis:names:tc:SAML:2.0:profiles:holder-of-key`

Abstract:

The *SAML V2.0 Holder-of-Key Assertion Profile* describes the issuing and processing of holder-of-key SAML assertions. Specifically, we show how a SAML issuer binds X.509 data to a `<ds:KeyInfo>` element and how a relying party confirms that a `<ds:KeyInfo>` element matches given X.509 data. The binding material used by the SAML issuer and the matching data used by the relying party are obtained from an X.509 certificate.

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 should send comments to the TC by using the "Send A Comment" button on the TC's web page at <http://www.oasis-open.org/committees/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 IPR section of the TC web page (<http://www.oasis-open.org/committees/security/ipr.php>).

The non-normative errata page for this specification is located at <http://www.oasis-open.org/committees/security>.

36 Notices

37 Copyright © OASIS Open 2008–2009. All Rights Reserved.

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

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

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

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

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

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

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

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

80 **Table of Contents**

81 1 Introduction.....4
82 1.1 Notation.....4
83 1.2 Terminology.....5
84 1.3 Normative References.....5
85 1.4 Non-normative References.....5
86 1.5 Conformance.....5
87 1.5.1 SAML V2.0 Holder-of-Key Assertion Profile.....5
88 2 SAML V2.0 Holder-of-Key Assertion Profile.....7
89 2.1 Required Information.....7
90 2.2 Profile Description.....7
91 2.3 X.509 Certificate Usage.....8
92 2.4 Issuing Holder-of-Key Assertions.....8
93 2.4.1 KeyInfo Usage.....9
94 2.4.2 Example.....9
95 2.5 Processing Holder-of-Key Assertions.....11
96 2.6 Security and Privacy Considerations.....12
97 2.6.1 ASN.1 Encoding.....12
98 2.6.2 X.509 Serial Number.....12
99 Appendix A. Acknowledgments.....14
100 Appendix B. Revision History.....15
101

1 Introduction

The *SAML V2.0 Holder-of-Key Assertion Profile* describes the issuing and processing of a holder-of-key SAML assertion, that is, an assertion containing a `<saml:SubjectConfirmation>` element whose Method attribute is set to `urn:oasis:names:tc:SAML:2.0:cm:holder-of-key`. Specifically, we describe the structural characteristics of a `<ds:KeyInfo>` element with bound X.509 data and show how a relying party confirms that such a `<ds:KeyInfo>` element matches given X.509 data. The binding material used by the SAML issuer and the matching data used by the relying party are obtained from an X.509 certificate.

This profile involves a SAML issuer and a SAML relying party, each with an X.509 certificate in its possession. The SAML issuer uses its certificate to produce a holder-of-key SAML assertion. The relying party consumes the assertion, confirming the [subjectattesting entity](#) by comparing the X.509 data in the assertion with the X.509 data in its possession.

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 defined in the SAML V2.0 core specification [SAML2Core].
ds:	http://www.w3.org/2000/09/xmldsig#	This is the XML Signature namespace [XMLSig].
xs:	http://www.w3.org/2001/XMLSchema	This is the XML Schema namespace [Schema1].
xsi:	http://www.w3.org/2001/XMLSchema-instance	This is the XML Schema namespace for schema-related markup that appears in XML instances [Schema1].

This specification uses the following typographical conventions in text: `<SAMLElement>`, `<ns:ForeignElement>`, Attribute, **Datatype**, OtherCode.

131 | 1.2 Terminology

132 | In this specification, a SAML issuer is a producer of holder-of-key assertions. Similarly, a relying party is
133 | a consumer of holder-of-key assertions.

134 | A presenter transmits a holder-of-key assertion to the relying party. An attesting entity is a presenter who
135 | is able to satisfy the subject confirmation requirements of the holder-of-key assertion.

136 | Usually the attesting entity is the subject of the assertion (hence the terms "subject confirmation" and
137 | "confirming the subject"). In general, however, the attesting entity may not be the subject, in which case
138 | the previous phrases are misnomers. Thus the terms "attestation" and "confirming the attesting entity"
139 | are more technically correct than "subject confirmation" and "confirming the subject," respectively. We
140 | will use the term "attesting entity" exclusively in this document.

141 | 1.3 Normative References

- 142 | **[RFC2119]** S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF
143 | RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>
- 144 | **[RFC4514]** K. Zeilenga. *Lightweight Directory Access Protocol (LDAP): String*
145 | *Representation of Distinguished Names*. IETF RFC 4514, June 2006.
146 | <http://www.ietf.org/rfc/rfc4514.txt>
- 147 | **[RFC5280]** D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk. *Internet*
148 | *X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL)*
149 | *Profile*. IETF RFC 5280, May 2008. <http://www.ietf.org/rfc/rfc5280.txt>
- 150 | **[SAML2Core]** S. Cantor, J. Kemp, R. Philpott, E. Maler. *Assertions and Protocols for the*
151 | *OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS Standard,
152 | March 2005. <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- 153 | **[SAML2Prof]** J. Hughes, S. Cantor, J. Hodges, F. Hirsch, P. Mishra, R. Philpott, E. Maler.
154 | *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*.
155 | OASIS Standard, March 2005. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)
156 | [open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)
- 157 | **[Schema1]** H. S. Thompson et al. *XML Schema Part 1: Structures*. World Wide Web
158 | Consortium Recommendation, May 2001. See [http://www.w3.org/TR/2001/REC-](http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)
159 | [xmlschema-1-20010502/](http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)
- 160 | **[XMLSig]** D. Eastlake, J. Reagle, D. Solo, F. Hirsch, T. Roessler. *XML Signature Syntax*
161 | *and Processing (Second Edition)*. World Wide Web Consortium
162 | Recommendation, 10 June 2008. <http://www.w3.org/TR/xmlsig-core/>

163 | 1.4 Non-normative References

- 164 | **[RFC3820]** S. Tuecke, V. Welch, D. Engert, L. Pearlman, M. Thompson. *Internet X.509*
165 | *Public Key Infrastructure (PKI) Proxy Certificate Profile*. IETF RFC 3820, June
166 | 2004. <http://www.ietf.org/rfc/rfc3820.txt>
- 167 | **[RFC4346]** T. Dierks, E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.1*.
168 | IETF RFC 4346, April 2006. <http://www.ietf.org/rfc/rfc4346.txt>

169 | 1.5 Conformance

170 | 1.5.1 SAML V2.0 Holder-of-Key Assertion Profile

171 | Both the SAML issuer and the relying party MUST conform to section 2.3.

172 A SAML issuer MUST follow the issuing rules in section 2.4. In particular, a SAML issuer MUST produce
173 <ds:KeyInfo> elements that conform to section 2.4.1. Likewise, a relying party MUST follow the
174 processing rules in section 2.5.

175 To claim conformance to this specification, a SAML issuer implementation MUST support the
176 <ds:X509Certificate> element specified in section 2.4.1. Support for the remaining child elements
177 specified in section 2.4.1 is OPTIONAL for SAML issuers.

178 Likewise a conforming relying party implementation MUST support the <ds:X509Certificate>
179 element specified in section 2.5. Support for the remaining child elements specified in section 2.5 is
180 OPTIONAL for relying parties.

2 SAML V2.0 Holder-of-Key Assertion Profile

2.1 Required Information

Identification: urn:oasis:names:tc:SAML:2.0:profiles:holder-of-key

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

SAML Confirmation Method Identifiers: The SAML V2.0 holder-of-key confirmation method identifier (urn:oasis:names:tc:SAML:2.0:cm:holder-of-key) is associated with every <saml:SubjectConfirmation> element issued under this profile.

Description: Given below.

Updates: Supplements the holder-of-key confirmation method described in section 3.1 of [SAML2Prof].

2.2 Profile Description

~~Suppose a SAML response issued by a SAML issuer contains one or more holder-of-key assertions (otherwise this specification is not applicable). This specification profiles a type of assertion called a holder-of-key assertion. By definition, a holder-of-key SAML assertion contains a <saml:SubjectConfirmation> element whose Method attribute is set to urn:oasis:names:tc:SAML:2.0:cm:holder-of-key. This specification describes how the SAML issuer binds selected X.509 data from an X.509 certificate to the <saml:SubjectConfirmation> element of a holder-of-key assertion. The complementary process involves a relying party who confirms that the X.509 data bound to the assertion matches the data in a given X.509 certificate.~~

~~Suppose a SAML response issued by a SAML issuer contains one or more holder-of-key assertions (otherwise this specification is not applicable). At the time the assertion is issued, the issuer possesses an X.509 certificate known to be associated with the attesting entity (who may or may not be present when the assertion is issued). The SAML issuer binds some (or all) of the X.509 data in the certificate to the holder-of-key assertion.~~

~~Subsequently, the attesting entity presents the holder-of-key assertion and an X.509 certificate to the relying party. The attesting entity proves possession of the private key corresponding to the public key bound to the certificate, the details of which are out of scope with respect to this profile. The relying party compares the X.509 data in the certificate to the X.509 data bound to the assertion, thereby confirming the attesting entity.~~

~~Precisely how the issuer comes to possess a certificate known to be associated with attesting entity and how the assertion and the certificate are presented to the relying party are all out of scope with respect to this profile. On the other hand, the issuing of the holder-of-key assertion itself and the ultimate confirmation of the attesting entity are in scope.~~

~~The complementary process involves a relying party that confirms that the X.509 data bound to the assertion matches the data in a given X.509 certificate. We assume that the relying party trusts the SAML issuer to issue holder-of-key assertions. The SAML issuer, on the other hand, may not even know the intended relying party, so there is no underlying assumption that the SAML issuer trusts the relying party.~~

~~It is assumed that both the SAML issuer and the relying party each possess an X.509 certificate that is known to be associated with the subject of the assertion. How the X.509 certificate is obtained, however, is completely out of scope.~~

221 2.3 X.509 Certificate Usage

222 There are no explicit requirements with respect to the X.509 certificate(s) possessed by the SAML issuer
223 and the relying party. If, however, ~~the~~ certificate contains a Subject Key Identifier (SKI) extension, then
224 the certificate MUST be an X.509 v3 certificate [RFC5280]. Other than that, the specific characteristics
225 of these certificates are wholly out of scope with respect to this specification. In particular, there is no
226 expectation that either the SAML issuer or the relying party trusts the issuer of the certificate, and
227 therefore all portions of the certificate, apart from the X.509 data specified in the following sections, are
228 unspecified.

229 The only exception to the above rule is the case where the `<ds:X509Data>` element specified in
230 section 2.4.1 contains a `<ds:X509SubjectName>` element or a `<ds:X509SerialIssuer>` element.
231 In these two cases, the relying party MUST trust the X.509 issuer in order to confirm the subjectattesting
232 entity. This is discussed more fully in section 2.5 below.

233 2.4 Issuing Holder-of-Key Assertions

234 Every assertion containing a holder-of-key `<saml:SubjectConfirmation>` element MUST conform
235 to [SAML2Core] (see section 2.4.1 of Core, especially section 2.4.1.3) and section 3.1 of [SAML2Prof].
236 Where this specification conflicts with the SAML V2.0 specification, the former takes precedence.

237 Suppose a SAML issuer wishes to issue a response containing one or more holder-of-key assertions. As
238 a prerequisite, the SAML issuer MUST possess an X.509 certificate known to be associated with the
239 subjectattesting entity. The SAML issuer binds some or all of the X.509 data in the certificate to the
240 `<saml:SubjectConfirmation>` element of a SAML assertion. ~~The expected content of a holder-of-~~
241 ~~key `<saml:SubjectConfirmation>` element is specified in the next section.~~

242 Briefly, tThe SAML issuer binds a `<ds:KeyInfo>` element to the
243 `<saml:SubjectConfirmationData>` element of a SAML holder-of-key assertion. The
244 `<ds:KeyInfo>` element contains one or more of the following elements: `<ds:X509Certificate>`,
245 `<ds:X509SKI>`, `<ds:X509SubjectName>`, or `<ds:X509IssuerSerial>`. A
246 `<ds:X509Certificate>` element contains a base64 encoding of the certificate possessed by the
247 SAML issuer. A `<ds:X509SKI>` element contains the base64 encoding of the Subject Key Identifier
248 (SKI) extension (if there is one) bound to the certificate. A `<ds:X509SubjectName>` element contains
249 the subject distinguished name (DN) bound to the certificate. A `<ds:X509IssuerSerial>` element
250 contains the issuer DN and the issuer serial number bound to the certificate. In each case, the content of
251 the `<ds:KeyInfo>` element conforms to the XML Signature specification [XMLSig]. These
252 requirements are spelled out more clearly in the next section.

253 If the SAML issuer has reason to believe that the relying party trusts the certificate issuer, the SAML
254 issuer MAY include `NotBefore` or `NotOnOrAfter` XML attributes on the
255 `<saml:SubjectConfirmationData>` element. If so, the values bound to the assertion MUST be
256 consistent with the values in the certificate. In particular, the value of the `NotBefore` attribute (resp.,
257 the `NotOnOrAfter` attribute) MUST be greater than or equal to (resp., less than or equal to) the
258 `NotBefore` field (resp., the `NotOnOrAfter` field) of the certificate.

259 The `<saml:SubjectConfirmation>` element MAY contain a `<saml:NameID>` element. If it does,
260 the latter identifies an attesting entity different from the subject of the assertion. If the
261 `<saml:SubjectConfirmation>` element does not contain a `<saml:NameID>` element, then the
262 attesting entity and the subject are one and the same. Since the `<ds:KeyInfo>` element is extensible
263 [XMLSig], other fields or extensions from the X.509 certificate may be bound to the holder-of-key
264 `<saml:SubjectConfirmation>` element. These are provided as a convenience to the relying party,
265 so that the relying party need not have to decode and parse the certificate. All such extensions are out-
266 of scope with respect to this profile, however.

267 2.4.1 KeyInfo Usage

268 According to the SAML V2.0 specification, a holder-of-key ~~assertion~~~~<saml:SubjectConfirmation>~~
269 ~~element~~ MUST contain at least one <ds:KeyInfo> element within the
270 <saml:SubjectConfirmationData> element and that the <ds:KeyInfo> element MUST conform
271 to the XML Signature specification. The current specification requires that the <ds:KeyInfo> element
272 MUST conform to the *Second Edition* of the XML Signature specification [XMLSig] and further constrains
273 the content of each <ds:KeyInfo> element to contain exactly one <ds:X509Data> element. The
274 <ds:X509Data> element MUST NOT contain a <ds:X509CRL> element. Instead, the following content
275 options are specified, at least one of which MUST be satisfied:

- 276 • The <ds:X509Data> element MAY contain a <ds:X509Certificate> element. If it does, the
277 <ds:X509Certificate> element MUST contain a base64 encoding of the X.509 certificate
278 possessed by the SAML issuer.
- 279 • The <ds:X509Data> element MAY contain a <ds:X509SKI> element. If it does, the
280 <ds:X509SKI> element MUST contain the base64 encoding of the plain (i.e., *not* DER-
281 encoded) value of the Subject Key Identifier (SKI) extension (as specified in [XMLSig]) of the
282 X.509 certificate possessed by the SAML issuer. If the certificate does not contain an SKI
283 extension, the <ds:X509Data> element MUST NOT contain a <ds:X509SKI> element.
- 284 • The <ds:X509Data> element MAY contain a <ds:X509SubjectName> element. If it does, the
285 <ds:X509SubjectName> element MUST contain the subject distinguished name (DN) bound
286 to the X.509 certificate possessed by the SAML issuer.
- 287 • The <ds:X509Data> element MAY contain a <ds:X509IssuerSerial> element. If it does,
288 the <ds:X509IssuerSerial> element MUST contain the issuer DN and the issuer serial
289 number (as specified in [XMLSig]) bound to the X.509 certificate possessed by the SAML issuer.

290 Use of the <ds:X509Certificate> element or the <ds:X509IssuerSerial> element is most
291 restrictive since each implies that the exact same certificate is used by both the SAML issuer and the
292 relying party. Use of the <ds:X509SKI> element or the <ds:X509SubjectName> element is less
293 restrictive since each permits a different certificate to be used by the relying party provided the certificate
294 contains the same key or DN (resp.) in the certificate used by the SAML issuer.

295 Use of the <ds:X509SubjectName> element or the <ds:X509IssuerSerial> element is warranted
296 in those situations where the relying party trusts the issuer of the X.509 certificate. The SAML issuer
297 SHOULD NOT bind either of these elements to the <ds:X509Data> element unless it knows that such
298 a trust relationship exists.

299 Note that the format of the DN contained in the <ds:X509SubjectName> element or the
300 <ds:X509IssuerSerial> element is specified in [XMLSig]. In accordance with that specification, it is
301 RECOMMENDED that the DN conform to [RFC4514] in all cases.

302 Since the <ds:KeyInfo> element is extensible [XMLSig], other fields or extensions from the X.509
303 certificate may be bound to the holder-of-key assertion. These are provided as a convenience to the
304 relying party, so that the relying party need not have to decode and parse the certificate. All such
305 extensions are out of scope with respect to this profile, however.

306 2.4.2 Example

307 Here is an example of a holder-of-key <saml:SubjectConfirmation> element illustrating three of
308 the content options specified in section 2.4:

```
309 <saml:SubjectConfirmation  
310   xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"  
311   Method="urn:oasis:names:tc:SAML:2.0:cm:holder-of-key">
```

```

312 <saml:SubjectConfirmationData
313   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
314   xsi:type="saml:KeyInfoConfirmationDataType">
315   <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
316     <ds:X509Data>
317
318       <!-- a base64 encoding of an X.509 certificate -->
319       <ds:X509Certificate>
320 MIIDuDCCAqACCQCJZK8wF0xVXjANBqkqhkiG9w0BAQQFADCBnTELMaKGA1UEBhMCQlIxZzARBgNV
321 BAgTC1NvbWUtU3RhdGUxEjAQBgNVBACTCVNvbWUtQ2l0eTESMBAGA1UEChMjR1NvQyAyMDA4MRIw
322 EAYDVQQLLEw1HU29DIDIwMDgxZzFzAVBgNVBAMTDkpvYW5hIFRyaW5kYWRLMSQwIgyJKoZihvcNAQkB
323 FhVzb211LWFkZHZlc3NAaG9zdC5vcmcwHhcNMDgwNjE2MTcyMTQzWhcNMDkwNjE2MTcyMTQzWjCB
324 nTELMaKGA1UEBhMCQlIxZzARBgNVBAGTC1NvbWUtU3RhdGUxEjAQBgNVBACTCVNvbWUtQ2l0eTES
325 MBAGA1UEChMjR1NvQyAyMDA4MRIwEAYDVQQLLEw1HU29DIDIwMDgxZzFzAVBgNVBAMTDkpvYW5hIFRy
326 aW5kYWRLMSQwIgyJKoZihvcNAQkBFhVzb211LWFkZHZlc3NAaG9zdC5vcmcwgwEiMA0GCSqGSIb3
327 DQEBAQUAA4IBDwAwggEKAoIBAQDIDVKdO2CCVYA0TspOPmcSNnivjQq7jCacrgRPawKi3/pTuvnW
328 3c2XCpyT2s6Sks3Eg5T4HIXta5E+lOpN8VbTunVdSrac54r2uK8x+8AqX7M0wQw+98iGw9E2an5q
329 xRZfqqE1T5jWL/a/G1/e2TG1mp521W3k1nNtF8rYH39JpwBSZMeW7uHOSzOKT/pVvqPTgG7vUQT6
330 BiRh7PfwsLrLOMubbeQ6Z2m3Vnsv20E1FbPzswzh4X1gXj9bnyI2UsuoisW9Y4p4byjL3GJ/hxp
331 mjRjXs+aIpzi0V3MH+jVJ98eomhlUFLaE83xycC8lns+FcCSQZ8RsbnaLZrtC8r7AgMBAAEwDQYJ
332 KoZihvcNAQEEBQADggEBACwnWSEpWq5aE7QBdDNNXyok34RIonYi9690yw7i+JU7R/QdE42GERJS
333 DVKBN959ELLJf5d0vybGv08QWbZVQ7eBGn9xaZ7MhSnb1YNDXs9vuv1V2Dy32q1J5nCSzqpJDyln
334 lVfWe9UQMCJOO6ibUtWlhiDQ49kmMabgyYfX28qB6oRdVL+mDI/XTt+mkCgk4Rs78n4kbX6qnRlj
335 dE/YnibP1A7iMh8pQkv49J6sP9SeUmQ2zxKct3tSRzzypWc8JjOZGuBYGQH19Xm7WEs4CXs7iZJW
336 E32frMATavMcTM/gnDtCc8tZAx12PSLOF1954vapfMjBhg3VTI6QRW//wPE=
337   </ds:X509Certificate>
338
339   <!-- the above X.509 certificate does not contain a
340   Subject Key Identifier extension so the SAML issuer
341   must not include a <ds:X509SKI> element -->
342
343   <!-- the subject DN (in RFC 5414 format) bound to the
344   above X.509 certificate -->
345   <ds:X509SubjectName>emailAddress=some-address@host.org,CN=Joana
346   Trindade,OU=GSoc 2008,O=GSoc 2008,L=Some-City,ST=Some-
347   State,C=BR</ds:X509SubjectName>
348
349   <!-- the issuer DN (in RFC 5414 format) and the issuer serial
350   number (in decimal) bound to the above X.509 certificate -->
351   <ds:X509IssuerSerial>
352     <ds:X509IssuerName>emailAddress=some-address@host.org,CN=Joana
353     Trindade,OU=GSoc 2008,O=GSoc 2008,L=Some-City,ST=Some-
354     State,C=BR</ds:X509IssuerName>
355     <ds:X509SerialNumber>9900230501951362398</ds:X509SerialNumber>
356   </ds:X509IssuerSerial>
357
358   </ds:X509Data>
359 </ds:KeyInfo>
360 </saml:SubjectConfirmationData>
361 </saml:SubjectConfirmation>

```

362 | A relying party can confirm the [subject attesting entity](#) by the matching the available X.509 data to any of
363 | the above child elements of the <ds:X509Data> element.

364 2.5 Processing Holder-of-Key Assertions

365 ~~A relying party wishing to confirm the subject of a holder-of-key assertion MUST possess an X.509~~
366 ~~certificate known to be associated with the subject of the assertion. The attesting entity presents a holder-~~
367 ~~of-key assertion and an X.509 certificate to the relying party. The attesting entity MUST prove~~
368 ~~possession of the private key corresponding to the public key bound to the certificate, the details of~~
369 ~~which are out of scope with respect to this profile. The relying party confirms the attesting entity's~~
370 ~~subject of the assertion by comparing the X.509 data in the certificate to the X.509 data bound to the assertion.~~
371 ~~If the X.509 data in the certificate matches the X.509 data bound to the assertion, the subject attesting~~
372 ~~entity is said to be confirmed.~~

373 Regardless of the protocol used, any assertions relied upon MUST be valid according to the processing
374 rules specified in [SAML2Core]. In particular, the relying party MUST verify the signature (if any) on each
375 assertion containing a holder-of-key <saml:SubjectConfirmation> element. Any assertion that is
376 not valid, or whose subject confirmation requirements cannot be met, SHOULD be discarded and
377 SHOULD NOT be used to establish a security context for the subject.

378 If the <ds:X509Data> element contains multiple child elements, the relying party may choose to
379 confirm the [subject attesting entity](#) based on any one of them. Specifically, the relying party MUST
380 confirm that the certificate matches the content of the <ds:X509Data> element as follows:

- 381 • If the <ds:X509Data> element contains a <ds:X509Certificate> element, and the relying
382 party chooses to confirm the [subject attesting entity](#) based on this element, the relying party
383 MUST ensure that the certificate bound to the assertion matches the X.509 certificate in its
384 possession. Matching is done by comparing the base64-decoded certificates, or the hash values
385 of the base64-decoded certificates, byte-for-byte.
- 386 • If the <ds:X509Data> element contains a <ds:X509SKI> element, and the relying party
387 chooses to confirm the [subject attesting entity](#) based on this element, the relying party MUST
388 ensure that the value bound to the assertion matches the Subject Key Identifier (SKI) extension
389 bound to the X.509 certificate. Matching is done by comparing the base64-decoded SKI values
390 byte-for-byte. If the X.509 certificate does not contain an SKI extension, the [subject attesting](#)
391 [entity](#) is not confirmed and the relying party SHOULD disregard the assertion.
- 392 • If the <ds:X509Data> element contains a <ds:X509SubjectName> element, and the relying
393 party chooses to confirm the [subject attesting entity](#) based on this element, the relying party
394 MUST ensure that the subject distinguished name (DN) bound to the assertion matches the DN
395 bound to the X.509 certificate. If, however, the relying party does not trust the certificate issuer to
396 issue such a DN, the [subject attesting entity](#) is not confirmed and the relying party SHOULD
397 disregard the assertion.
- 398 • If the <ds:X509Data> element contains a <ds:X509IssuerSerial> element, and the relying
399 party chooses to confirm the [subject attesting entity](#) based on this element, the relying party
400 MUST ensure that the issuer DN and issuer serial number bound to the assertion match the
401 issuer DN and the issuer serial number (resp.) bound to the X.509 certificate. If the relying party
402 does not trust the certificate issuer to issue X.509 certificates, however, the [subject attesting](#)
403 [entity](#) is not confirmed and the relying party SHOULD disregard the assertion.

404 In the case of a <ds:X509Certificate> element or a <ds:X509SKI> element, the matching process
405 is relatively straightforward. If the <ds:X509Data> element contains a <ds:X509SubjectName>
406 element or a <ds:X509IssuerSerial> element, however, and the relying party chooses to confirm
407 the [subject attesting entity](#) based on one of these elements, the relying party MUST trust the issuer of the
408 X.509 certificate before the [subject attesting entity](#) can be considered confirmed. If such a trust
409 relationship between the relying party and the certificate issuer does not exist, the relying party SHOULD
410 disregard the assertion.

411 If the `<saml:SubjectConfirmationData>` element includes `NotBefore` or `NotOnOrAfter`
412 attributes, and the relying party trusts the issuer of the X.509 certificate, the relying party MUST confirm
413 that the current time is greater than or equal to (resp., less than or equal to) the value of the `NotBefore`
414 (resp., the `NotOnOrAfter`) attribute. If this requirement is not met, the `subjectattesting entity` is not
415 confirmed and the relying party SHOULD disregard the assertion.

416 2.6 Security and Privacy Considerations

417 This profile assumes that both the SAML issuer and the relying party have access to an X.509 certificate.
418 For those deployments that wish to avoid or do not require an X.509-based public key infrastructure
419 (PKI), this may seem unnecessarily restrictive. In fact, the use of X.509 certificates is typical and
420 provides a number of advantages. First, observe that the SSL/TLS protocol [RFC4346] requires the use
421 of X.509 certificates. Second, and most importantly, since there is no presumption of an underlying trust
422 model for X.509 certificates, the full range of possible content for the `<ds:KeyInfo>` element is
423 avoided. Those deployments that are in fact based on such a trust model, or wish to avoid X.509
424 certificates altogether, may choose to profile additional child elements such as `<ds:KeyName>` or
425 `<ds:KeyValue>`.

426 Deployments that rely on holder-of-key SAML assertions will no doubt impose their own requirements on
427 the X.509 certificates used to obtain those assertions. For example, some deployments will require the
428 certificate to be an X.509 end-entity certificate [RFC5280] issued by a trusted X.509 certification
429 authority (CA) or a certificate based on a trusted X.509 end-entity certificate (such as an X.509 proxy
430 certificate [RFC3820]). This specification imposes no such restrictions, however.

431 2.6.1 ASN.1 Encoding

432 For compatibility with the XML Signature specification [XMLSig], this profile intentionally avoids any
433 discussion of the ASN.1 encoding of the X.509 certificate possessed by the SAML issuer and the relying
434 party. Indeed, in the case of the `<ds:X509Certificate>` element, the ASN.1 encoding of the
435 certificate doesn't matter. In this case, the SAML issuer simply base64-encodes the ASN.1-encoded
436 certificate in its possession and binds it to the `<ds:X509Certificate>` element. Later the relying
437 party base64-decodes the content of the `<ds:X509Certificate>` element and compares the resulting
438 certificate (byte-for-byte) with the ASN.1-encoded certificate in its possession.

439 In the case of the `<ds:X509SKI>`, `<ds:X509SubjectName>`, or `<ds:X509IssuerSerial>` elements,
440 however, the ASN.1 encoding of the certificates *does* matter. To produce these elements, the SAML
441 issuer must ASN.1-decode the certificate in its possession and parse the ASN.1 to obtain the X.509 data
442 to be bound to the assertion. Likewise the relying party must ASN.1-decode the certificate in its
443 possession, parsing the ASN.1 to obtain the required X.509 data, which it compares to the X.509 data
444 bound to the assertion.

445 The basic problem is that the ASN.1 encoding of an X.509 certificate is not `guaranteedspecified`. While
446 it is true that an X.509 certificate is often DER-encoded, a robust implementation must be prepared to
447 handle other ASN.1 encodings besides DER, mainly BER and CER. Consequently it is anticipated that
448 deployments will prefer the `<ds:X509Certificate>` element for maximum interoperability. In fact,
449 this preference is reflected in the conformance requirements of this profile (section 1.5).

450 2.6.2 X.509 Serial Number

451 Note that some CAs use large random numbers as serial numbers to prevent sequence guessing.
452 However, not all XML libraries are capable of dealing with large integers in the
453 `<ds:X509IssuerSerial>` element. The problem is that the `<ds:X509SerialNumber>` child
454 element of the `<ds:X509IssuerSerial>` element is typed as an arbitrary integer in [XMLSig] yet

455 conforming implementations are required to support only 18 decimal digits. Thus the
456 <ds:X509IssuerSerial> element should be used with care.

457 **Appendix A. Acknowledgments**

458 The editor would like to acknowledge the contributions of the OASIS Security Services (SAML) Technical
459 Committee, whose voting members at the time of publication were:

- 460 • TBD

461 The editor would also like to acknowledge the following contributors:

- 462 • Joana M. F. da Trindade, Universidade Federal do Rio Grande do Sul (Brazil)
- 463 • The members of the IETF PKIX Working Group
- 464 • Peter Sylvester, EdelWeb (France)
- 465 • [Brett Beaumont, SSC, New Zealand Government](#)

Appendix B. Revision History

Document ID	Date	Committer	Comment
sstc-saml2-holder-of-key-draft-01	7 Aug 2008	T. Scavo	Initial draft
sstc-saml2-holder-of-key-draft-02	14 Aug 2008	T. Scavo	Remove all refs to <code>samlp:</code>
sstc-saml2-holder-of-key-draft-03	7 Sep 2008	T. Scavo	Remove proof of possession requirement
sstc-saml2-holder-of-key-draft-04	6 Oct 2008	T. Scavo	Response to comments
sstc-saml2-holder-of-key-draft-05	20 Oct 2008	T. Scavo	Updated KeyInfo Usage rules
sstc-saml2-holder-of-key-draft-06	13 Nov 2008	T. Scavo	Dropped DER-encoding requirement
sstc-saml2-holder-of-key-draft-07	7 Dec 2008	T. Scavo	Added NotBefore/NotOnOrAfter attributes
sstc-saml2-holder-of-key-draft-08	11 Jan 2009	T. Scavo	Relaxed the X.509 v3 requirement
sstc-saml2-holder-of-key-draft-09	20 Jan 2009	T. Scavo	Response to comments