



SAML V2.0 Holder-of-Key Assertion Profile

Working Draft **0807**, 11 January 2009~~December 2008~~

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 ~~n-standard~~ X.509 ~~v3~~ 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 notice
43 and this section are included on all such copies and derivative works. However, this document itself may
44 not be modified in any way, including by removing the copyright notice or references to OASIS, except as
45 needed for the purpose of developing any document or deliverable produced by an OASIS Technical
46 Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be
47 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 A
54 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, to
57 notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such
58 patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced
59 this specification.

60 OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any
61 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 represent
68 that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to
69 rights in any document or deliverable produced by an OASIS Technical Committee can be found on the
70 OASIS website. Copies of claims of rights made available for publication and any assurances of licenses
71 to be made available, or the result of an attempt made to obtain a general license or permission for the
72 use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS
73 Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any
74 information or list of intellectual property rights will at any time be complete, or that any claims in such list
75 are, in fact, Essential Claims.

76 The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be
77 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 Normative References.....5
84 1.3 Non-normative References.....5
85 1.4 Conformance.....5
86 1.4.1 SAML V2.0 Holder-of-Key Assertion Profile.....5
87 2 SAML V2.0 Holder-of-Key Assertion Profile.....6
88 2.1 Required Information.....6
89 2.2 Profile Description.....6
90 2.3 X.509 Certificate Usage.....6
91 2.4 Issuing Holder-of-Key Assertions.....7
92 2.4.1 KeyInfo Usage.....7
93 2.4.2 Example.....8
94 2.5 Processing Holder-of-Key Assertions.....9
95 2.6 Security and Privacy Considerations.....10
96 2.6.1 ASN.1 Encoding.....10
97 2.6.2 X.509 Serial Number.....11
98 Appendix A. Acknowledgments.....12
99 Appendix B. Revision History.....13

100

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 [standard X.509-v3 certificate \[RFC5280\]](#).

This profile involves a SAML issuer and a SAML relying party, each with an X.509-v3 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 subject 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 Normative References

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

153 1.3 Non-normative References

- 154 [RFC3820] S. Tuecke, V. Welch, D. Engert, L. Pearlman, M. Thompson. *Internet X.509*
155 *Public Key Infrastructure (PKI) Proxy Certificate Profile*. IETF RFC 3820, June
156 2004. <http://www.ietf.org/rfc/rfc3820.txt>
- 157 [RFC4346] T. Dierks, E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.1*.
158 IETF RFC 4346, April 2006. <http://www.ietf.org/rfc/rfc4346.txt>

159 1.4 Conformance

160 1.4.1 SAML V2.0 Holder-of-Key Assertion Profile

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

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

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

168 Likewise a conforming relying party implementation MUST support the <ds:X509Certificate>
169 element specified in section 2.5. Support for the remaining child elements specified in section 2.5 is
170 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). 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-v3 certificate to the <saml:SubjectConfirmation> element of a holder-of-key assertion.

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-v3 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-v3 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.

2.3 X.509 Certificate Usage

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

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

~~In what follows, we use the term X.509 certificate to refer to an X.509-v3 certificate conforming to [RFC5280].~~

211 2.4 Issuing Holder-of-Key Assertions

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

215 Suppose a SAML issuer wishes to issue a response containing one or more holder-of-key assertions. As
216 a prerequisite, the SAML issuer MUST possess an X.509 certificate known to be associated with the
217 subject. The SAML issuer binds some or all of the X.509 data in the certificate to the
218 `<saml:SubjectConfirmation>` element of a SAML assertion. The expected content of a holder-of-
219 key `<saml:SubjectConfirmation>` element is specified in the next section.

220 The SAML issuer binds a `<ds:KeyInfo>` element to a SAML assertion. The `<ds:KeyInfo>` element
221 contains one or more of the following elements: `<ds:X509Certificate>`, `<ds:X509SKI>`,
222 `<ds:X509SubjectName>`, or `<ds:X509IssuerSerial>`. A `<ds:X509Certificate>` element
223 contains a base64 encoding of the certificate possessed by the SAML issuer. A `<ds:X509SKI>` element
224 contains the base64 encoding of the Subject Key Identifier (SKI) extension (if there is one) bound to the
225 certificate. A `<ds:X509SubjectName>` element contains the subject distinguished name (DN) bound to
226 the certificate. A `<ds:X509IssuerSerial>` element contains the issuer DN and the issuer serial
227 number bound to the certificate. In each case, the content of the `<ds:KeyInfo>` element conforms to
228 the XML Signature specification [XMLSig]. These requirements are spelled out more clearly in the next
229 section.

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

236 Since the `<ds:KeyInfo>` element is extensible [XMLSig], other fields or extensions from the X.509
237 certificate may be bound to the holder-of-key `<saml:SubjectConfirmation>` element. These are
238 provided as a convenience to the relying party, so that the relying party need not have to decode and
239 parse the certificate. All such extensions are [out](#) of scope with respect to this profile, however.

240 2.4.1 KeyInfo Usage

241 According to the SAML V2.0 specification, a holder-of-key `<saml:SubjectConfirmation>` element
242 MUST contain at least one `<ds:KeyInfo>` element and that the `<ds:KeyInfo>` element [MUST](#)
243 conform to the XML Signature specification. The current specification requires that the `<ds:KeyInfo>`
244 element MUST conform to the *Second Edition* of the XML Signature specification [XMLSig] and further
245 constrains the content of each `<ds:KeyInfo>` element to contain exactly one `<ds:X509Data>` element.
246 The `<ds:X509Data>` element MUST NOT contain a `<ds:X509CRL>` element. Instead, the following
247 content options are specified, at least one of which MUST be satisfied:

- 248 • The `<ds:X509Data>` element MAY contain a `<ds:X509Certificate>` element. If it does, the
249 `<ds:X509Certificate>` element MUST contain a base64 encoding of the X.509 certificate
250 possessed by the SAML issuer.
- 251 • The `<ds:X509Data>` element MAY contain a `<ds:X509SKI>` element. If it does, the
252 `<ds:X509SKI>` element MUST contain the base64 encoding of the plain (i.e., *not* DER-encoded)
253 value of the Subject Key Identifier (SKI) extension (as specified in [XMLSig]) of the X.509
254 certificate possessed by the SAML issuer. If the certificate does not contain an SKI extension, the
255 `<ds:X509Data>` element MUST NOT contain a `<ds:X509SKI>` element.

256 • The <ds:X509Data> element MAY contain a <ds:X509SubjectName> element. If it does, the
257 <ds:X509SubjectName> element MUST contain the subject distinguished name (DN) bound to
258 the X.509 certificate possessed by the SAML issuer.

259 • The <ds:X509Data> element MAY contain a <ds:X509IssuerSerial> element. If it does,
260 the <ds:X509IssuerSerial> element MUST contain the issuer DN and the issuer serial
261 number (as specified in [XMLSig]) bound to the X.509 certificate possessed by the SAML issuer.

262 Use of the <ds:X509Certificate> element or the <ds:X509IssuerSerial> element is most
263 restrictive since each implies that the exact same certificate is used by both the SAML issuer and the
264 relying party. Use of the <ds:X509SKI> element or the <ds:X509SubjectName> element is less
265 restrictive since each permits a different certificate to be used by the relying party provided the certificate
266 contains the same key or DN (resp.) [in the certificate](#) used by the SAML issuer.

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

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

274 2.4.2 Example

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

```
277 <saml:SubjectConfirmation
278   xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
279   Method="urn:oasis:names:tc:SAML:2.0:cm:holder-of-key">
280   <saml:SubjectConfirmationData
281     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
282     xsi:type="saml:KeyInfoConfirmationDataType">
283     <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
284       <ds:X509Data>
285
286         <!-- a base64 encoding of an X.509 certificate -->
287         <ds:X509Certificate>
288 MIIDuDCCAqACQCJZK8wF0xVXjANBgkqhkiG9w0BAQQFADCBnTELMaKGA1UEBhMCQlIxZARBgNV
289 BAgTC1NvbWUuU3RhdGUxEjAQBGNVBAcTCVNvbWUuU210eTESMBAGA1UEChMJR1NvQyAyMDA4MRIw
290 EAYDVQQLEw1HU29DIDIwMDgxFzAVBGNVBAMTDkpvYW5hIFRyaW5kYW5kYWRlMSQwIgyJKoZiIhvcNAQkBB
291 FhVzb211LWFkZHZHJlc3NAAG9zdC5vcmcwHhcNMMDgWnJE2MTcyMTQzWhcNMMDkwnJE2MTcyMTQzWjCB
292 nTELMaKGA1UEBhMCQlIxZARBgNVBAgTC1NvbWUuU3RhdGUxEjAQBGNVBAcTCVNvbWUuU210eTES
293 MBAGA1UEChMJR1NvQyAyMDA4MRIwEAYDVQQLEw1HU29DIDIwMDgxFzAVBGNVBAMTDkpvYW5hIFRya
294 aW5kYWRlMSQwIgyJKoZiIhvcNAQkBBFhVzb211LWFkZHZHJlc3NAAG9zdC5vcmcwggEiMA0GCSqGSIb3
295 DQEBAQUAA4IBDwAwggEKAoIBAQDIDVkdO2CCVYA0TspOPmcSNnivjQq7jCacrgRPawKi3/pTuvnW
296 3c2XCpyT2s6Sks3Eg5T4HIXta5E+lOpN8VbTunVdSrac54r2uK8x+8AqX7M0wQw+98iGw9E2an5q
297 xRZfqqE1T5jWL/a/G1/e2TG1mp521W3k1nNtF8rYH39JpwBSZMeW7uHOSZokT/pVvqPTgG7vUQT6
298 BiRh7PfwsLrLOmubbeQ6Z2m3Vnsv20E1FbPzswz4X1gXj9bnyI2UsoisW9Y4p4byjL3GJ/hxp
299 mjRjXs+aIpzi0V3MH+jVJ98eomhlUFLaE83xycC8lns+FcCSQZ8RsbnaLZrtC8r7AgMBAAEwDQYJ
300 KoZiIhvcNAQEEBQADggEBACwnWSEpwq5aE7QBdDNNXyok34RIonYi9690yw7i+JU7R/QdE42GERJS
301 DVKBN959ELLJf5d0vybGv08QWbZVQ7eBgn9xaZ7MhSnblyNDXs9vuv1V2Dy32q1J5nCSzqpJDyln
302 lVFWe9UQMCJ006ibUtWLhIDQ49kmMabgyYfX28qB6oRdVL+mDI/XTt+mkCgk4Rs78n4kbX6qnRlj
303 dE/YnibP1A7iMh8pQkv49J6sP9SeUmQ2zxKct3tSrZzypWc8JjOZGuBYGQH19Xm7Wes4CXS7iZJW
304 E32frMATavMcTM/gnDtCc8tZAx12PSLOF1954vapfMjBhg3VTI6QRW//wPE=
305   </ds:X509Certificate>
306
307   <!-- the above X.509 certificate does not contain a
308   Subject Key Identifier extension so the SAML issuer
309   must not include a <ds:X509SKI> element -->
310
```



```

311         <!-- the subject DN (in RFC 5414 format) bound to the
312             above X.509 certificate -->
313         <ds:X509SubjectName>emailAddress=some-address@host.org,CN=Joana
314 Trindade,OU=GSoC 2008,O=GSoC 2008,L=Some-City,ST=Some-
315 State,C=BR</ds:X509SubjectName>
316
317         <!-- the issuer DN (in RFC 5414 format) and the issuer serial
318             number (in decimal) bound to the above X.509 certificate -->
319         <ds:X509IssuerSerial>
320         <ds:X509IssuerName>emailAddress=some-address@host.org,CN=Joana
321 Trindade,OU=GSoC 2008,O=GSoC 2008,L=Some-City,ST=Some-
322 State,C=BR</ds:X509IssuerName>
323         <ds:X509SerialNumber>9900230501951362398</ds:X509SerialNumber>
324         </ds:X509IssuerSerial>
325
326     </ds:X509Data>
327 </ds:KeyInfo>
328 </saml:SubjectConfirmationData>
329 </saml:SubjectConfirmation>

```

330 A relying party can confirm the subject by the matching the available X.509 data to any of the above child
331 elements of the `<ds:X509Data>` element.

332 2.5 Processing Holder-of-Key Assertions

333 A relying party wishing to confirm the subject of a holder-of-key assertion MUST possess an X.509
334 certificate known to be associated with the [presenter's subject](#) of the assertion. The relying party confirms
335 the subject of the assertion by comparing the X.509 data in the certificate to the X.509 data bound to the
336 assertion. If the X.509 data in the certificate matches the X.509 data bound to the assertion, the subject is
337 said to be *confirmed*.

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

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

- 346 • If the `<ds:X509Data>` element contains a `<ds:X509Certificate>` element, and the relying
347 party chooses to confirm the subject based on this element, the relying party MUST ensure that
348 the certificate bound to the assertion matches the X.509 certificate in its possession. Matching is
349 done by comparing the base64-decoded certificates, or the hash values of the base64-decoded
350 certificates, byte-for-byte.
- 351 • If the `<ds:X509Data>` element contains a `<ds:X509SKI>` element, and the relying party
352 chooses to confirm the subject based on this element, the relying party MUST ensure that the
353 value bound to the assertion matches the Subject Key Identifier (SKI) extension bound to the
354 X.509 certificate. Matching is done by comparing the base64-decoded SKI values byte-for-byte.
355 If the X.509 certificate does not contain an SKI extension, the subject is not confirmed and the
356 relying party SHOULD disregard the assertion.
- 357 • If the `<ds:X509Data>` element contains a `<ds:X509SubjectName>` element, and the relying
358 party chooses to confirm the subject based on this element, the relying party MUST ensure that
359 the subject distinguished name (DN) bound to the assertion matches the DN bound to the X.509
360 certificate. If, however, the relying party does not trust the certificate issuer to issue such a DN,
361 the subject is not confirmed and the relying party SHOULD disregard the assertion.

362 • If the `<ds:X509Data>` element contains a `<ds:X509IssuerSerial>` element, and the relying
363 party chooses to confirm the subject based on this element, the relying party MUST ensure that
364 the issuer DN and issuer serial number bound to the assertion match the issuer DN and the
365 issuer serial number (resp.) bound to the X.509 certificate. If the relying party does not trust the
366 certificate issuer to issue X.509 certificates, however, the subject is not confirmed and the relying
367 party SHOULD disregard the assertion.

368 In the case of a `<ds:X509Certificate>` element or a `<ds:X509SKI>` element, the matching process
369 is relatively straightforward. If the `<ds:X509Data>` element contains a `<ds:X509SubjectName>`
370 element or a `<ds:X509IssuerSerial>` element, however, and the relying party chooses to confirm the
371 subject based on one of these elements, the relying party MUST trust the issuer of the X.509- certificate
372 before the subject can be considered confirmed. If such a trust relationship between the relying party and
373 the certificate issuer does not exist, the relying party SHOULD disregard the [enclosing](#) assertion.

374 If the `<saml:SubjectConfirmationData>` element includes `NotBefore` or `NotOnOrAfter`
375 attributes, and the relying party trusts the issuer of the X.509 certificate, the relying party MUST confirm
376 that the current time is greater than or equal to (resp., less than or equal to) the value of the `NotBefore`
377 (resp., the `NotOnOrAfter`) attribute. If this requirement is not met, the subject is not confirmed and the
378 relying party SHOULD disregard the assertion.

379 2.6 Security and Privacy Considerations

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

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

393 2.6.1 ASN.1 Encoding

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

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

407 The [basic](#) problem is that the ASN.1 encoding of an X.509 certificate is not specified. While it is true that
408 an X.509 certificate is often DER-encoded, a robust implementation must be prepared to handle other

409 ASN.1 encodings besides DER, mainly BER and CER. Consequently it is anticipated that deployments
410 will prefer the `<ds:X509Certificate>` element for maximum interoperability. In fact, this preference is
411 reflected in the conformance requirements of this profile (section 1.4).

412 **2.6.2 X.509 Serial Number**

413 Note that some CAs use large random numbers as serial numbers to prevent sequence guessing,
414 ~~but~~However, not all XML libraries are capable of dealing with large integers in the
415 `<ds:X509IssuerSerial>` element. The problem is that the `<ds:X509SerialNumber>` child element
416 of the `<ds:X509IssuerSerial>` element is typed as an arbitrary integer in [XMLSig] yet conforming
417 implementations are required to support only 18 decimal digits. Thus the `<ds:X509IssuerSerial>`
418 element should be used with care.

419 **Appendix A. Acknowledgments**

420 | The editor would like to acknowledge the contributions of the OASIS Security Services [\(SAML\)](#) Technical
421 | Committee, whose voting members at the time of publication were:

- 422 | • TBD

423 | The editor would also like to acknowledge the following contributors:

- 424 | • Joana M. F. da Trindade, Universidade Federal do Rio Grande do Sul (Brazil)
- 425 | • The members of the IETF PKIX Working Group
- 426 | • Peter Sylvester, EdelWeb [\(France\)](#)

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 samlp:
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