



Web Services Security X509 Certificate Token Profile

Working Draft 05, 6th June 2003

Document identifier:

WSS-X509-05

Location:

<http://www.oasis-open.org/committees/documents.php>

Editors:

Phillip Hallam-Baker, VeriSign
Chris Kaler, Microsoft
Ronald Monzillo, Sun
Anthony Nadalin, IBM

Contributors:

TBD – Revise this list to include WSS TC contributors

Bob Atkinson, Microsoft	John Manferdelli, Microsoft
Giovanni Della-Libera, Microsoft	Hiroshi Maruyama, IBM
Satoshi Hada, IBM	Anthony Nadalin, IBM
Phillip Hallam-Baker, VeriSign	Nataraj Nagaratnam, IBM
Maryann Hondo, IBM	Hemma Prafullchandra, VeriSign
Chris Kaler, Microsoft	John Shewchuk, Microsoft
Johannes Klein, Microsoft	Dan Simon, Microsoft
Brian LaMacchia, Microsoft	Kent Tamura, IBM
Paul Leach, Microsoft	Hervey Wilson, Microsoft

Abstract:

This document describes how to use X509 Certificates with the [WS-Security](#) specification.

Status:

This is an interim draft. Please send comments to the editors.

Committee members should send comments on this specification to the wss@lists.oasis-open.org list. Others should subscribe to and send comments to the wss-comment@lists.oasis-open.org list. To subscribe, visit <http://lists.oasis-open.org/ob/adm.pl>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Security Services TC web page (<http://www.oasis-open.org/who/intellectualproperty.shtml>).

30 **Table of Contents**

31 1 Introduction 3
32 2 Notations and Terminology..... 4
33 2.1 Notational Conventions 4
34 2.2 Namespaces 4
35 2.3 Terminology 4
36 3 Usage 5
37 3.1 Processing Model 5
38 3.2 Attaching Security Tokens 5
39 3.3 Identifying Certificates 6
40 3.3.1 Identifying End Entity Certificates by Value 6
41 3.3.2 Identifying and Referencing Certificate Chains 6
42 3.3.3 Identifying End Entity Certificates by Reference 7
43 3.4 Authentication..... 8
44 3.5 Encryption 8
45 3.6 Error Codes 8
46 3.7 Threat Model and Countermeasures 8
47 4 Acknowledgements10
48 5 References11
49 Appendix A: Revision History.....12
50 Appendix B: Notices13
51

52 **1 Introduction**

53 This specification describes the use of X509 certificates with respect to the [WS-Security](#)
54 specification.

55 An X.509 Certificate specifies a binding between a public key and a set of attributes that include a
56 subject name, issuer name, serial number and validity interval. This binding may be subject to
57 subsequent revocation advertised by mechanisms that include issue of CRLs, OCSP tokens or
58 mechanisms that are outside the X.509 framework such as XKMS.

59 An X.509 Certificate may be used to establish the authenticity of a public key used to authenticate
60 a WS-Security enhanced message or to identify the public key under which a WS-Security
61 enhanced message is encrypted.

62 Note that Section 1 is non-normative.

63 2 Notations and Terminology

64 This section specifies the notations, namespaces, and terminology used in this specification.

65 2.1 Notational Conventions

66 This document uses the notational conventions defined in the [WS-Security SOAP Message Security](#) document.

68 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",
69 "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be
70 interpreted as described in RFC2119.

71 Readers are presumed to be familiar with the terms in the [Internet Security Glossary](#).

72 2.2 Namespaces

73 The [XML namespace](#) URIs that MUST be used by implementations of this specification are as
74 follows (note that different elements in this specification are from different namespaces):

```
75 http://schemas.xmlsoap.org/ws/2002/xx/secext  
76 http://schemas.xmlsoap.org/ws/2002/xx/utility
```

77 The following namespaces are used in this document:

Prefix	Namespace
S	http://www.w3.org/2001/12/soap-envelope
ds	http://www.w3.org/2000/09/xmlsig#
xenc	http://www.w3.org/2001/04/xmlenc#
wsse	http://schemas.xmlsoap.org/ws/2002/xx/secext
wsu	http://schemas.xmlsoap.org/ws/2002/xx/utility

78 2.3 Terminology

79 This specification employs the terminology defined in the WS-Security Core Specification.

80 3 Usage

81 This section describes the profile (specific mechanisms and procedures) for the X509
82 binding of [WS-Security](#).

83 **Identification:** urn:oasis:names:tc:WSS:1.0:profiles:WSS-X509-token

84 **Contact information:** TBD

85 **Description:** Given below.

86 **Updates:** None.

87 3.1 Processing Model

88 The processing model for [WS-Security](#) with X509 certificates is no different from that
89 of [WS-Security](#) with other token formats as described in [WS-Security](#) .

90 3.2 Attaching Security Tokens

91 X.509 Certificates that are attached as security tokens within a [WS-Security](#)
92 enhanced message SHOULD be attached by means of the
93 `<wsse:BinarySecurityToken>` element.

94 The [WS-Security](#) specification indicates that X.509 certificates MAY be described
95 inside of a `<ds:KeyInfo>` element, however, it is RECOMMENDED that they be
96 specified using a `<wsse:BinarySecurityToken>`. If, however, an implementation
97 needs to use `<ds:KeyInfo>`, it SHOULD place the `<ds:KeyInfo>` element as a child
98 of the `<wsse:Security>` header rather than embedded within the signature. This
99 allows receivers to have a single processing model.

100 The following values are defined for the ValueType attribute of the
101 `<wsse:BinarySecurityToken>` element.

QName	Description
wsse:X509v3	X.509 v3 end entity certificate
wsse:PKIPath	An X.509 certificate chain packaged in a PKIPATH wrapper
wsse:PKCS7	An X.509 certificate chain packaged in a PKCS#7 wrapper

102 The following example illustrates a SOAP message with an X509 Certificate.

```
103 <S:Envelope xmlns:S="...">  
104   <S:Header>  
105     <wsse:Security xmlns:wsse="...">  
106  
107       <wsse:BinarySecurityToken  
108         xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext "  
109         Id="myToken"  
110         ValueType="wsse:X509v3"  
111         EncodingType="wsse:Base64Binary">  
112           MIIIEZzCCA9CgAwIBAgIQEmtJZc0...  
113         </wsse:BinarySecurityToken>  
114
```

115
116
117
118
119
120
121

```
    ...  
    </wsse:Security>  
</S:Header>  
<S:Body>  
    ...  
</S:Body>  
</S:Envelope>
```

122 3.3 Identifying Certificates

123 3.3.1 Identifying End Entity Certificates by Value

124 An attached X.509 certificate that identifies an end entity is attached by means of
125 the wsse:BinarySecurityToken element and referenced by means of a
126 wsse:SecurityTokenReference element that contains a wsse:KeyIdentifier element.
127 The wsu:Id attribute of the wsse:KeyIdentifier element references the value of the
128 wsu:Id attribute specified in the wsse:BinarySecurityToken.

129 The types of end entities that are authenticated by means of the Certificates is
130 outside the scope of this specification. WS-Security applications MAY require the use
131 of certificates that identify a particular type of end-entity such as a client or a
132 service.

133 The following example shows a SOAP message that contains an X509v3 Certificate as
134 a binary token:

135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154

```
Example TBS  
<S:Envelope xmlns:S="...">  
  <S:Header>  
    <wsse:Security xmlns:wsse="...">  
  
      <wsse:BinarySecurityToken  
        xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext "  
        Id="myToken "  
        ValueType="wsse:X509v3 "  
        EncodingType="wsse:Base64Binary">  
          MIEZzCCA9CgAwIBAgIQEmtJZc0...  
        </wsse:BinarySecurityToken>  
  
      ...  
    </wsse:Security>  
  </S:Header>  
<S:Body>  
  ...  
</S:Body>  
</S:Envelope>
```

155 3.3.2 Identifying and Referencing Certificate Chains

156 An attached X.509 certificate that identifies an end entity is attached by means of
157 the wsse:BinarySecurityToken element and referenced by means of a
158 wsse:SecurityTokenReference element that contains a wsse:KeyIdentifier element.
159 The wsu:Id attribute of the wsse:KeyIdentifier element references the value of the
160 wsu:Id attribute specified in the wsse:BinarySecurityToken.

161 The wsse:BinarySecurityToken element contains a binary object that specifies a
162 certificate chain. It is RECOMMENDED that applications use the PKIPath object for
163 this purpose. The PKCS#7 SignedData object MAY be used instead, although as

164 noted, this usage depends on the ordering of the certificates being preserved by the
165 processing of the PKCS#7 object.

166 If the ValueType wsse:PKIPath is specified the wsse:BinarySecurityToken element
167 contains an ASN.1 DER encoded sequence of certificates, as specified in [TBS]. The
168 PKIPath element is defined as follows:

169 PkiPath ::= SEQUENCE OF Certificate

170 Within the sequence, the order of certificates is such that the subject of the first
171 certificate is the issuer of the second certificate, etc. Each certificate in PkiPath shall
172 be unique. No certificate may appear more than once in a value of Certificate in
173 PkiPath.

174 If the ValueType wsse:PKCS7 is specified the wsse:BinarySecurityToken element
175 contains a PKCS#7 SignedData object, with the only significant field being
176 certificates. In particular, the signature and the contents are ignored. If no
177 certificates are present, a zero-length CertPath is assumed.

178 Warning: PKCS#7 does not maintain the order of certificates in a certification path.
179 This means that if a CertPath is converted to PKCS#7 encoded bytes and then
180 converted back, the order of the certificates may change, potentially rendering the
181 CertPath invalid. Users should be aware of this behavior. See [PKCS7] for more
182 information.

183 The following example shows a SOAP message that contains an X509v3 Certificate
184 chain encoded inside a PKCS#7 package:

```
185 Example TBS
186 <S:Envelope xmlns:S="...">
187   <S:Header>
188     <wsse:Security xmlns:wsse="...">
189
190       <wsse:BinarySecurityToken
191         xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext"
192         Id="myToken"
193         ValueType="wsse:PKCS7"
194         EncodingType="wsse:Base64Binary">
195         MIIIEZzCCA9CgAwIBAgIQEmtJZc0...
196       </wsse:BinarySecurityToken>
197
198       ...
199     </wsse:Security>
200   </S:Header>
201   <S:Body>
202     ...
203   </S:Body>
204 </S:Envelope>
```

205 3.3.3 Identifying End Entity Certificates by Reference

206 An X.509 certificate that identifies an end entity that is not attached to the message
207 payload is referenced by means of the Issuer Subject Name and Serial Number using
208 the XML Signature <X509IssuerSerial> element.

209 The following example shows a SOAP message that identifies an X.509v3 end entity
210 certificate by reference to the Issuer and serial number:

```
211 Example TBS
212 <S:Envelope xmlns:S="...">
213   <S:Header>
```

214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233

```
<wsse:Security xmlns:wsse="...">
  <ds:KeyInfo>
    <X509Data>
      <X509IssuerSerial>
        <X509IssuerName>CN=TAMURA Kent, OU=TRL, O=IBM,
          L=Yamato-shi, ST=Kanagawa, C=JP</X509IssuerName>
        <X509SerialNumber>12345678</X509SerialNumber>
      </X509IssuerSerial>
      <X509SKI>31d97bd7</X509SKI>
    </X509Data>
  </ds:KeyInfo>
  ...
</wsse:Security>
</S:Header>
<S:Body>
  ...
</S:Body>
</S:Envelope>
```

234 **3.4 Authentication**

235 When an X.509 certificate is used to specify a signature key, the [signature](#) algorithm
236 MUST be a digital signature algorithm.

237 The value of the signature key is the value of the public key specified in the
238 certificate.

239 **3.5 Encryption**

240 When an X.509 certificate is used to specify an encryption key, the encryption
241 algorithm MUST be a public key encryption algorithm.

242 The certificate that specifies the encryption key SHOULD be identified by reference
243 since the receiver only requires the use of the certificate to identify the
244 corresponding encryption key and does not require the certificate or the certificate
245 chain to authenticate the public key it holds.

246 The value of the encryption key is the value of the public key specified in the
247 certificate.

248 **3.6 Error Codes**

249 When using X509 Certificates the error codes defined in the [WS-Security](#)
250 specification MUST be used.

251 If an implementation requires the use of a custom error it is recommended that a
252 sub-code be defined as an extension of one of the codes defined in the [WS-Security](#)
253 specification.

254 **3.7 Threat Model and Countermeasures**

255 The use of X509 certificates with [WS-Security](#) introduces no new threats beyond
256 those identified for WS-Security with other types of security tokens.

257 Message alteration and eavesdropping can be addressed by using the integrity and
258 confidentiality mechanisms described in WS-Security. Replay attacks can be

259 addressed by using message timestamps and caching, as well as other application-
260 specific tracking mechanisms. For X.509 certificates ownership is verified by use of
261 keys, man-in-the-middle attacks are generally mitigated.
262 It is strongly RECOMMENDED that all relevant and immutable message data be
263 signed.
264 It should be noted that transport-level security MAY be used to protect the message
265 and the security token.

266 **4 Acknowledgements**

267 This specification was developed as a result of joint work of many individuals from
268 the WSS TC including: TBD

269 The input specifications for this document were developed as a result of joint work
270 with many individuals and teams, including: Keith Ballinger, Microsoft, Bob Blakley,
271 IBM, Allen Brown, Microsoft, Joel Farrell, IBM, Mark Hayes, VeriSign, Kelvin
272 Lawrence, IBM, Scott Konersmann, Microsoft, David Melgar, IBM, Dan Simon,
273 Microsoft, Wayne Vicknair, IBM.

274 5 References

- 275 **[DIGSIG]** Informational RFC 2828, "[Internet Security Glossary](#)," May 2000.
- 276 **[KEYWORDS]** S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels,"
277 [RFC 2119](#), Harvard University, March 1997
- 278 **[SOAP]** W3C Note, "[SOAP: Simple Object Access Protocol 1.1](#)," 08 May 2000.
- 279 **[URI]** T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers
280 (URI): Generic Syntax," [RFC 2396](#), MIT/LCS, U.C. Irvine, Xerox
281 Corporation, August 1998.
- 282 **[WS-Security]** TBS – point to the OASIS draft
- 283 **[XML-ns]** W3C Recommendation, "[Namespaces in XML](#)," 14 January 1999.
- 284 **[XML Signature]** W3C Recommendation, "[XML Signature Syntax and Processing](#)," 12
285 February 2002.
- 286 **[PKCS7]** TBS <http://www.rsasecurity.com/rsalabs/pkcs/pkcs-7/index.html>
- 287 **[X509]** TBS
- 288 **[PKIPATH]** TBS
289 [ftp://ftp.bull.com/pub/OSIdirectory/DefectResolution/TechnicalCorrigenda](ftp://ftp.bull.com/pub/OSIdirectory/DefectResolution/TechnicalCorrigenda/ApprovedTechnicalCorrigendaToX.509/8%7CX.509-TC1(4th).pdf)
290 [/ApprovedTechnicalCorrigendaToX.509/8%7CX.509-TC1\(4th\).pdf](ftp://ftp.bull.com/pub/OSIdirectory/DefectResolution/TechnicalCorrigenda/ApprovedTechnicalCorrigendaToX.509/8%7CX.509-TC1(4th).pdf).
291

Appendix A: Revision History

Rev	Date	What
01	18-Sep-02	Initial draft based on input documents and editorial review
03	30-Jan-03	Changes in title
04	19-May-03	Added by reference and pkipath modes of cert identification. Added section 1 introduction, changes to formatting etc.

Appendix B: Notices

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

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

307 Copyright © OASIS Open 2002. *All Rights Reserved.*

308 This document and translations of it may be copied and furnished to others, and derivative works
309 that comment on or otherwise explain it or assist in its implementation may be prepared, copied,
310 published and distributed, in whole or in part, without restriction of any kind, provided that the
311 above copyright notice and this paragraph are included on all such copies and derivative works.
312 However, this document itself does not be modified in any way, such as by removing the
313 copyright notice or references to OASIS, except as needed for the purpose of developing OASIS
314 specifications, in which case the procedures for copyrights defined in the OASIS Intellectual
315 Property Rights document must be followed, or as required to translate it into languages other
316 than English.

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

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

324