Web Services Security
X.509 Certificate Token Profile

Working Draft 08, 6th August 2003

Document identifier:

urn:oasis:names:tc:WSS:1.0:profiles:X509-08

Location:

http://www.oasis-open.org/committees/download.php/2427/WSS-X509-08.pdf

Editors:

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

Contributors:

Current voting members of the WSS TC (as of July 1st 2003)

Note: It is assumed that we will update this on the day of Committee Spec to be the current list

Gene	Thurston	AmberPoint
Frank	Siebenlist	Argonne National Lab
Merlin	Hughes	Baltimore Technologies
Irving	Reid	Baltimore Technologies
Peter	Dapkus	BEA
Hal	Lockhart	BEA
Symon	Chang	CommerceOne
Thomas	DeMartini	ContentGuard
Guillermo	Lao	ContentGuard
TJ	Pannu	ContentGuard
Shawn	Sharp	Cyclone Commerce
Ganesh	Vaideeswaran	Documentum
Sam	Wei	Documentum
John	Hughes	Entegrity
Tim	Moses	Entrust
Toshihiro	Nishimura	Fujitsu
Tom	Rutt	Fujitsu
Jason	Rouault	HP
Yutaka	Kudo	Hitachi
Maryann	Hondo	IBM
Kelvin	Lawrence	IBM (co-Chair)
Anthony	Nadalin	IBM
Nataraj	Nagaratnam	IBM
Don	Flinn	Individual
Bob	Morgan	Individual
Paul	Cotton	Microsoft
Vijay	Gajjala	Microsoft

Chris	Kaler	Microsoft (co-Chair)
Chris	Kurt	Microsoft
John	Shewchuk	Microsoft
Prateek	Mishra	Netegrity
Frederick	Hirsch	Nokia
Senthil	Sengodan	Nokia
Lloyd	Burch	Novell
Ed	Reed	Novell
Charles	Knouse	Oblix
Steve	Anderson	OpenNetwork (Sec)
Vipin	Samar	Oracle
Jerry	Schwarz	Oracle
Eric	Gravengaard	Reactivity
Stuart	King	Reed Elsevier
Andrew	Nash	RSA Security
Rob	Philpott	RSA Security
Peter	Rostin	RSA Security
Martijn	de Boer	SAP
Pete	Wenzel	SeeBeyond
Jonathan	Tourzan	Sony
Yassir	Elley	Sun Microsystems
Jeff	Hodges	Sun Microsystems
Ronald	Monzillo	Sun Microsystems
Jan	Alexander	Systinet
Michael	Nguyen	The IDA of Singapore
Don	Adams	TIBCO
John	Weiland	US Navy
Phillip	Hallam-Baker	VeriSign
Morten	Jorgensen	Vordel

Contributors of input documents (if not already listed above) :

Bob	Blakley	IBM
Joel	Farrell	IBM
Satoshi	Hada	IBM
Hiroshi	Maruyama	IBM
David	Melgar	IBM
Bob	Atkinson	Microsoft
Allen	Brown	Microsoft
Giovanni	Della-Libera	Microsoft
Johannes	Klein	Microsoft
Scott	Konersmann	Microsoft
Brian	LaMacchia	Microsoft
Paul	Leach	Microsoft
John	Manferdell	Microsoft
Dan	Simon	Microsoft
Hervey	Wilson	Microsoft
Hemma	Prafullchandra	VeriSign

Abstract:

This document describes how to use X.509 Certificates with the **WS-Security** specification.

Status:

This is an interim draft.

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 wsscomment@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 WS-Security TC web page (http://www.oasis-open.org/committees/wss/ipr.php).

Table of Contents

1 Introduction (Non-Normative)

This specification describes the use of the X.509 authentication framework with the Web Services Security: SOAP Message Security specification [WS-Security].

An X.509 certificate specifies a binding between a public key and a set of attributes that includes (at least) a subject name, issuer name, serial number and validity interval. This binding may be subject to subsequent revocation advertised by mechanisms that include issuance of CRLs, OCSP tokens or mechanisms that are outside the X.509 framework, such as XKMS.

An X.509 certificate may be used to validate a public key that may be used to authenticate a WS-Security-enhanced message or to identify the public key with which a WS-Security-enhanced message has been encrypted.

2 Notations and Terminology

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

2.1 Notational Conventions

This document uses the notational conventions defined in [WS-Security].

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

2.2 Namespaces

The XML namespace URIs that MUST be used by implementations of this specification are as follows (note that elements used in this specification are defined in one or other of these namespaces):

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

The following namespace prefixes are used in this document:

Prefix	Namespace
S	http://www.w3.org/2001/12/soap-envelope
ds	http://www.w3.org/2000/09/xmldsig#
xenc	http://www.w3.org/2001/04/xmlenc#
wsse	urn:oasis:names:tc:WSS:1.0
wsu	http://schemas.xmlsoap.org/ws/2002/xx/utility

Table 1- Namespace prefixes

2.3 Terminology

This specification adopts the terminology defined in [WS-Security].

Readers are presumed to be familiar with the definitions of terms in the Internet Security Glossary [Glossary].

3 Usage

This section describes the syntax and processing rules for the X.509 binding of WS-Security.

3.1 Token types

This profile defines the syntax of, and processing rules for, three types of token:

Token	QName	Description
Single certificate	wsse:X509v3	An X.509 v3 signature-verification certificate
Set of certificates and CRLs	wsse:X509PKIPathv1	A list of X.509 certificates packaged in a PKIPath
	wsse:PKCS7	A list of X.509 certificates and (optionally) CRLs packaged in a PKCS#7 wrapper

Table 2 – Token types

In order to ensure a consistent processing model across all the token types supported by WS-Security, the wsse:SecurityTokenReference element SHOULD be used to specify all references to X.509 token types in signature or encryption elements.

3.1.1 wsse:X509v3 Token Type

The type of the end-entity that is authenticated by a certificate used in this manner is a matter of policy that is outside the scope of this specification.

3.1.2 wsse:X509PKIPathv1 Token Type

The wsse:BinarySecurityToken element MAY contain a binary object that represents a certificate path. It is RECOMMENDED that applications use the PKIPath object for this purpose.

3.1.3 wsse:PKCS7 Token Type

The wsse:BinarySecurityToken element MAY contain a binary object that represents a certificate path. It is RECOMMENDED that applications use the PKIPath object for this purpose. The PKCS#7 SignedData object MAY be used instead.

The order of the certificates in a PKCS#7 data structure is not significant. If an ordered certificate path is converted to PKCS#7 encoded bytes and then converted back, the order of the certificates may not be preserved. Processors SHALL NOT assume any significance to the order of the certificates in the data structure. See [PKCS7] for more information.

3.2 Token References

A wsse:SecurityTokenReference MAY reference an X.509 token type by one of the following means:

Key Identifier

The wsse:SecurityTokenReference element contains a wsse:KeyIdentifier element that specifies the token data by means of a URI reference.

Reference to a Binary Security Token

The wsse:SecurityTokenReference element contains a wsse:Reference element that references a wsse:BinarySecurityToken element that contains the token data itself.

Reference to an Issuer and Serial Number

The wsse:SecurityTokenReference element contains a wsse:Reference element that references a wsse:Embedded element which contains a ds:X509IssuerSerial element that uniquely identifies an end entity certificate.

3.2.1 Key Identifier Reference

The wsse:KeyIdentifier is used to specify a reference to an X.509 security token by means of a URI.

The wsse:SecurityTokenReference from which the indirect reference is made contains the wsse:KeyIdentifier element. The attributes of the wsse:KeyIdentifier element include a ValueType whose value specifies a an X.509 token type and a URI Identifier that identifies the token.

The following example shows the use of a Key Identifier reference in a wsse:Security encryption header:

```
<S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope">
   <S:Header>
      <wsse:Security xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
          wsu:Id="A1UdAQQ8MDqAEEVv">
          <ds:KeyInfo ds=”http://www.w3.org/2000/09/xmldsig#”>
                <wsse:SecurityTokenReference>
                   <wsse:KeyIdentifier
                       ValueType=”wsse:X509v3”
                       URI="http://example.com/certs/…" />
                </wsse:SecurityTokenReference>
          </ds:KeyInfo>
          <xenc:EncryptedKey>…</xenc:EncryptedKey>
      </wsse:Security>
   </S:Header>
   <S:Body>
       ...
   </S:Body>
</S:Envelope>
```

3.2.2 Reference to a Security Token data

The wsse:BinarySecurityToken element is used to reference X.509 security token data by value.

The wsse:BinarySecurityToken element is a child of a wsse:Security header and MUST contain a wsu:Id attribute. The wsse:BinarySecurityToken element is referenced by means of a wsse:SecurityTokenReference element that contains a wsse:Reference whose value is the same as that of the wsu:Id attribute of the wsse:BinarySecurityToken element.

The following example shows an example of a certificate referenced by value to establish the trustworthiness of a public key used for signature.

```
<S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope">
   <S:Header>
      <wsse:Security xmlns:wsse="urn:oasis:names:tc:WSS:1.0">
         <wsse:BinarySecurityToken
             wsu:Id="A1UdAQQ8MDqAEEVs"
             wsu:ValueType="wsse:X509v3"
             wsu:EncodingType="wsse:Base64Binary">
             MIIEZzCCA9CgAwIBAgIQEmtJZc0...
          </wsse:BinarySecurityToken>
          <ds:Signature
                xmlns:ds=”http://www.w3.org/2000/09/xmldsig#”>
             <ds:SignedInfo>
                 ...
             </ds:SignedInfo>
             <ds:SignatureValue>…</ds:SignatureValue>
                <wsse:SecurityTokenReference>
                   <wsse:Reference URI="#A1UdAQQ8MDqAEEVs" />
                </wsse:SecurityTokenReference>
         </ds:Signature>
      </wsse:Security>
   </S:Header>
   <S:Body>
        ...
   </S:Body>
</S:Envelope>
```

3.2.3 Reference to an Issuer and Serial Number

The ds:KeyInfo element is used to specify a reference to an X.509 security token by means of the certificate issuer name and serial number.

The ds:KeyInfo element is a child of a wsse:Security header and MUST contain a wsu:Id attribute. The ds:KeyInfo element is referenced by means of a wsse:SecurityTokenReference element that contains a wsse:Reference whose value is the same as that of the wsu:Id attribute of the ds:KeyInfo element.

The following example shows the use of a certificate reference by means of the certificate issuer name and serial number to a private key used for encryption.

```
<S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope">
   <S:Header>
      <wsse:Security xmlns:wsse="urn:oasis:names:tc:WSS:1.0">

         <ds:KeyInfo ds=”http://www.w3.org/2000/09/xmldsig#”
               wsu:Id="A1UdAQQ8MDqAEEVr">
            <ds:X509Data>
               <ds:X509IssuerSerial>
                  <ds:X509IssuerName>DC=ACMECorp, DC=com
                  </ds:X509IssuerName>
                  <ds:X509SerialNumber>12345678</X509SerialNumber>
               </ds:X509IssuerSerial>
            </ds:X509Data>
         </ds:KeyInfo>
          <ds:Signature
                xmlns:ds=”http://www.w3.org/2000/09/xmldsig#”>
             <ds:SignedInfo>
                 ...
             </ds:SignedInfo>
             <ds:SignatureValue>…</ds:SignatureValue>
                <wsse:SecurityTokenReference>
                   <wsse:Reference URI="#A1UdAQQ8MDqAEEVr" />
                </wsse:SecurityTokenReference>
         </ds:Signature>
      </wsse:Security>
   </S:Header>
   <S:Body>
   </S:Body>
</S:Envelope>
```

3.3 Signature

Signed data MAY specify the certificate used for signing using any of the X.509 security token types.

3.3.1 Referencing a Security Token

An X.509 certificate specifies a binding between a public key and a set of attributes that includes (at least) a subject name, issuer name, serial number and validity interval. Other attributes MAY specify constraints on the use of the certificate or affect the recourse that may be open to a relying party that depends on the certificate. A given public key may be specified in more than one X.509 certificate; consequently a given public key MAY be bound to two or more distinct sets of attributes.

It is therefore necessary to ensure that a signature created under an X.509 certificate token uniquely and irrefutably specify the certificate under which the signature is created.

Implementations SHOULD protect against this attack by including either the certificate itself or an immutable reference to the certificate within the scope of a signature according to the method used to reference the signature as follows:

3.3.1.1 Key Identifier

The wsse:KeyIdentifier element does not guarantee an immutable reference to the security token referenced. Consequently implementations that use this form of reference within a signature SHOULD include both the wsse:KeyIdentifier element that contains the reference and the referenced data in the scope of the signature.

Example

```
<S:TBS/>
```

3.3.1.2 Reference to a Binary Security Token

The signature SHOULD contain an XPath reference to the wsse:BinarySecurityToken element that contains the security token referenced.

Example

```
<S:TBS/>
```

3.3.1.3 Reference to an Issuer and Serial Number

The signature SHOULD contain an XPath reference to the ds:KeyInfo element that contains the security token referenced.

Example

```
<S:TBS/>
```

3.4 Encryption

Encrypted data MAY identify a key required for decryption by identifying the corresponding key used for encryption using any of the X.509 security token types specified.

Since the sole purpose is to identify the decryption key it is not necessary to specify either a trust path or the specific contents of the certificate itself.

It is recommended that implementations specify an encryption key by reference to the Issuer and Serial Number of an X509v3 certificate security token.

Implementations MAY specify an encryption key by means of a Key Identifier reference to an X509v3 certificate security token. This usage requires each recipient to dereference the Key Identifier in order to determine whether it refers to a key the recipient holds.

3.5 Error Codes

When using X.509 certificates, the error codes defined in the WS-Security specification MUST be used.

If an implementation requires the use of a custom error it is recommended that a sub-code be defined as an extension of one of the codes defined in the WS-Security specification.

3.6 Threat Model and Countermeasures

The use of X.509 certificates with WS-Security introduces no new threats beyond those identified for WS-Security with other types of security tokens.

Message alteration and eavesdropping can be addressed by using the integrity and confidentiality mechanisms described in WS-Security. Replay attacks can be addressed by using message timestamps and caching, as well as other application-specific tracking mechanisms. For X.509 certificates, identity is authenticated by use of keys, man-in-the-middle attacks are generally mitigated.

It is strongly RECOMMENDED that all relevant and immutable message data be signed.

It should be noted that transport-level security MAY be used to protect the message and the security token as an alternative.

4 References

[Gloassry] Informational RFC 2828, "Internet Security Glossary," May 2000.

[KEYWORDS] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, Harvard University, March 1997

[SOAP] W3C Note, "SOAP: Simple Object Access Protocol 1.1," 08 May 2000.

[URI] T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," RFC 2396, MIT/LCS, U.C. Irvine, Xerox Corporation, August 1998.

[WS-Security] http://www.oasis-open.org/committees/download.php/1686/WSS-SOAPMessageSecurity-12-04021.pdf

[XML-ns] W3C Recommendation, "Namespaces in XML," 14 January 1999.

[XML Signature] W3C Recommendation, "XML Signature Syntax and Processing," 12 February 2002.

[PKCS7] **TBS** http://www.rsasecurity.com/rsalabs/pkcs/pkcs-7/index.html

[X509] **TBS**

.

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.
05	6 June 2003	
06	20 June 2003	Included examples showing how tokens must be referenced from signatures and cipher values. Defined how key-agreement keys are to be conveyed in a Security header.
07	4 August 2003	Modifications to KeyIdentifier handling and use of SecurityTokenReference. Changes to the acknowledgements section.

Appendix B: Notices

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

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

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