Web Services Security Kerberos Token Profile 1.1

OASIS Standard Specification, 1 February 2006

OASIS identifier:
> wss-v1.1-spec-os-KerberosTokenProfile

Location:
> http://docs.oasis-open.org/wss/v1.1

Technical Committee:
> Web Service Security (WSS)

Chairs:
> Kelvin Lawrence, IBM
> Chris Kaler, Microsoft

Editors:
> Anthony Nadalin, IBM
> Chris Kaler, Microsoft
> Ronald Monzillo, Sun
> Phillip Hallam-Baker, Verisign

Abstract:
> This document describes how to use Kerberos [Kerb] tickets (specifically the AP-REQ packet) with the WSS: SOAP Message Security [WSS] specification.

Status:

> This is an OASIS Standard document produced by the Web Services Security Technical Committee. It was approved by the OASIS membership on 1 February 2006. Check the current location noted above for possible errata to this document.
>
> Technical Committee members should send comments on this specification to the technical Committee's email list. Others should send comments to the

30	Technical Committee by using the "Send A Comment" button on the Technical
31	Committee's web page at www.oasisopen.org/committees/wss.
32	
33	For information on whether any patents have been disclosed that may be
34	essential to implementing this specification, and any offers of patent licensing
35	terms, please refer to the Intellectual Property Rights section of the Security
36	Services TC web page (http://www.oasis-pen.org/who/intellectualproperty.shtml).

37 Notices

38 OASIS takes no position regarding the validity or scope of any intellectual property or other rights
39 that might be claimed to pertain to the implementation or use of the technology described in this
40 document or the extent to which any license under such rights might or might not be vailable;
41 neither does it represent that it has made any effort to identify any such rights. Information on

42 OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS
43 website. Copies of claims of rights made available for publication and any assurances of licenses
44 to be made available, or the result of an attempt made to obtain a general license or permission
45 for the use of such proprietary rights by implementors or users of this specification, can be
46 obtained from the OASIS Executive Director. OASIS invites any interested party to bring to its
47 attention any copyrights, patents or patent applications, or other proprietary rights which may
48 cover technology that may be required to implement this specification. Please address the
49 information to the OASIS Executive Director.

50

74

75 This section is non-normative.

76   Table of Contents

94

95 1 Introduction

96 This specification describes the use of Kerberos [Kerb] tokens with respect to the WSS: SOAP
97 Message Security specification [WSS].

98 Specifically, this document defines how to encode Kerberos tickets and attach them to SOAP
99 messages. As well, it specifies how to add signatures and encryption to the SOAP message, in
100 accordance with WSS: SOAP Message Security, which uses and references the Kerberos
101 tokens.

102 For interoperability concerns, and for some security concerns, the specification is limited to using
103 the AP-REQ packet (service ticket and authenticator) defined by Kerberos as the Kerberos token.
104 This allows a service to authenticate the ticket and interoperate with existing Kerberos
105 implementations.

106 It should be noted that how the AP-REQ is obtained is out of scope of this specification as are
107 scenarios involving other ticket types and user-to-user interactions.

108 Note that Sections 2.1, 2.2, all of 3, and indicated parts of 6 are normative. All other sections are
109 non-normative.

110 2 Notations and Terminology

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

112 2.1 Notational Conventions

113 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",
114 "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be
115 interpreted as described in RFC2119 [2119].

116

117 Namespace URIs (of the general form "some-URI") represent some application-dependent or
118 context-dependent URI as defined in RFC2396 [URI].

119

120 This specification is designed to work with the general SOAP [S11, S12] message structure and
121 message processing model, and should be applicable to any version of SOAP. The current SOAP
122 1.2 namespace URI is used herein to provide detailed examples, but there is no intention to limit
123 the applicability of this specification to a single version of SOAP.

124 2.2 Namespaces

125 The XML namespace [XML-ns] URIs that MUST be used by implementations of this specification
126 are as follows (note that different elements in this specification are from different namespaces):

127

```
128   http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
129   secext-1.0.xsd
130   http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
131   utility-1.0.xsd
132   http://docs.oasis-open.org/wss/oasis-wss-wssecurity-secext-1.1.xsd
```

133 Note that this specification does not introduce new schema elements.

134 The following namespaces are used in this document:

Prefix	Namespace
S11	`http://schemas.xmlsoap.org/soap/envelope/`
S12	`http://www.w3.org/2003/05/soap-envelope`
wsse	`http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-`

WSS: Kerberos Token Profile   1 February 2006

	`wssecurity-secext-1.0.xsd`
wsse11	`http://docs.oasis-open.org/wss/oasis-wss-wssecurity-secext-1.1.xsd`
wsu	`http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd`
ds	`http://www.w3.org/2000/09/xmldsig#`
xenc	`http://www.w3.org/2001/04/xmlenc#`

135

136 The URLs provided for the `wsse` and `wsu` namespaces can be used to obtain the schema files.
137 URI fragments defined in this specification are relative to the following base URI unless otherwise
138 specified:

139 `http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1`

140 **2.3 Terminology**

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

142

143 This specification employs the terminology defined in the WSS: SOAP Message Security Core
144 Specification [WSS].

145

146 The following (non-normative) table defines additional acronyms and abbreviations for this
147 document.

Term	Definition
SHA	Secure Hash Algorithm
SOAP	Simple Object Access Protocol
URI	Uniform Resource Identifier
XML	Extensible Markup Language

148

149 3 Usage

150 This section describes the profile (specific mechanisms and procedures) for the Kerberos binding
151 of WSS: SOAP Message Security.

152 **Identification:** `http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-`
153 `profile-1.1`

154 3.1 Processing Model

155 The processing model for WSS: SOAP Message Security with Kerberos tokens is no different
156 from that of WSS: SOAP Message Security with other token formats as described in WSS: SOAP
157 Message Security.

158 3.2 Attaching Security Tokens

159 Kerberos tokens are attached to SOAP messages using WSS: SOAP Message Security by using
160 the `<wsse:BinarySecurityToken>` described in WSS: SOAP Message Security. When using
161 this element, the `@ValueType` attribute MUST be specified. This specification defines six
162 values for this attribute as defined in the table below:

URI	Description
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerberosv5_AP_REQ`	Kerberos v5 AP-REQ as defined in the Kerberos specification. This `ValueType` is used when the ticket is an AP Request.
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#GSS_Kerberosv5_AP_REQ`	A GSS-API Kerberos V5 mechanism token containing an KRB_AP_REQ message as defined in RFC-1964 [1964], Sec. 1.1 and its successor RFC-4121 [4121], Sec. 4.1. This `ValueType` is used when the ticket is an AP Request (ST + Authenticator).
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerberosv5_AP_REQ1510`	Kerberos v5 AP-REQ as defined in RFC1510. This `ValueType` is used when the ticket is an AP Request per RFC1510.
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#GSS_Kerberosv5_AP_REQ1510`	A GSS-API Kerberos V5 mechanism token containing an KRB_AP_REQ message as defined in RFC-1964, Sec. 1.1 and its

	successor RFC-4121, Sec. 4.1. This `ValueType` is used when the ticket is an AP Request (ST + Authenticator) per RFC1510.
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerberosv5_AP_REQ4120`	Kerberos v5 AP-REQ as defined in RFC4120. This `ValueType` is used when the ticket is an AP Request per RFC4120
`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#GSS_Kerberosv5_AP_REQ4120`	A GSS-API Kerberos V5 mechanism token containing an KRB_AP_REQ message as defined in RFC-1964, Sec. 1.1 and its successor RFC-4121, Sec. 4.1. This `ValueType` is used when the ticket is an AP Request (ST + Authenticator) per RFC4120.

163 It should be noted that the URIs in the table above also serve as the official URIs identifying the
164 Kerberos tokens defined in this specification.

165

166 All token types defined in this section use the type 0x8003 defined in RFC1964 for the checksum
167 field of the authenticator inside the AP_REQ.

168

169 The octet sequence of either the GSS-API framed KRB_AP_REQ token or an unwrapped
170 AP_REQ is encoded using the indicated encoding (e.g. base 64) and the result is placed inside of
171 the `<wsse:BinarySecurityToken>` element.

172 The following example illustrates a SOAP message with a Kerberos token.

```
173   <S11:Envelope xmlns:S11="..." xmlns:wsu="...">
174       <S11:Header>
175           <wsse:Security xmlns:wsse="...">
176               <wsse:BinarySecurityToken EncodingType="http://docs.
177                   oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-
178   security-1.0#Base64Binary" ValueType=" http://docs.oasis-
179   open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerb
180   erosv5_AP_REQ" wsu:Id="MyToken">boIBxDCCAcCgAwIBBaEDAgEOogcD...
181               </wsse:BinarySecurityToken>
182               ...
183           </wsse:Security>
184       </S11:Header>
185       <S11:Body>
186           ...
187       </S11:Body>
188   </S11:Envelope>
```

189

190 3.3 Identifying and Referencing Kerberos Tokens

191 A Kerberos Token is referenced by means of the `<wsse:SecurityTokenReference>`
192 element. This mechanism, defined in WSS: SOAP Message Security, provides different
193 referencing mechanisms. The following list identifies the supported and unsupported
194 mechanisms:

195 The `wsu:Id` MAY be specified on the `<wsse:BinarySecurityToken>` element allowing the
196 token to be directly referenced.

197 A `<wsse:KeyIdentifier>` element MAY be used which specifies the identifier for the
198 Kerberos ticket. This value is computed as the SHA1 of the pre-encoded octets that were used to
199 form the contents of the `<wsse:BinarySecurityToken>` element. The
200 `<wsse:KeyIdentifier>` element contains the encoded form the of the `KeyIdentifier`
201 which is defined as the base64 encoding of the SHA1 result.

202 Key Name references MUST NOT be used.

203 When a Kerberos Token is referenced using `<wsse:SecurityTokenReference>` the
204 `@wsse11:TokenType` attribute SHOULD be specified. If the `@wsse11:TokenType` is specified
205 its value MUST be the URI that identifies the Kerberos token type as defined for a corresponding
206 `BinarySecurityToken/@ValueType` attribute. The `Reference/@ValueType` attribute is
207 not required. If specified, its value MUST be equivalent to that of the `@wsse11:TokenType`
208 attribute..

209 The `<wsse:SecurityTokenReference>` element from which the reference is made contains
210 the `<wsse:KeyIdentifier>` element. The `<wsse:KeyIdentifier>` element MUST have a
211 `ValueType` attribute on the `<wsse:KeyIdentifier>` element with the value
212 `#Kerberosv5APREQSHA1` and its contents MUST be the SHA1 of GSS-API framed
213 KRB_AP_REQ token or unwrapped AP-REQ, as appropriate, encoded as per the
214 `<wsse:KeyIdentifier>` element's `EncodingType` attribute.

215

Reference Identifier	ValueType URI	Description
Kerberos v5 AP-REQ	`http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerberosv5APREQSHA1`	SHA1 of the v5 AP-REQ octets, either GSS-API framed KRB_AP_REQ token or just the Kerberos AP-REQ.

216

217 The following example illustrates using ID references to a Kerberos token:

218

```
219    <S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="...">
220        <S11:Header>
```

```
221            <wsse:Security>
222                <wsse:BinarySecurityToken EncodingType="http://docs.
223    oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-
224    1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/oasis-wss-
225    kerberos-token-profile-1.1#Kerberosv5_AP_REQ" wsu:Id="MyToken">
226                        boIBxDCCAcCgAwIBBaEDAgEOogcD...
227                </wsse:BinarySecurityToken>
228            ...
229                <wsse:SecurityTokenReference>
230                    <wsse:Reference URI="#MyToken"
231    ValueType="http://docs.oasis-open.org/wss/oasis-wss-kerberos-token-
232    profile-1.1#Kerberosv5_AP_REQ">
233                    </wsse:Reference>
234                </wsse:SecurityTokenReference>
235            ...
236        </wsse:Security>
237    </S11:Header>
238    <S11:Body>
239        ...
240    </S11:Body>
241 </S11:Envelope>
242
```

The AP-REQ packet is included in the initial message to the service, but need not be attached to
subsequent messages exchanged between the involved parties. Consequently, the
`KeyIdentifier` reference mechanism SHOULD be used on subsequent exchanges as
illustrated in the example below:

```
249 <S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="...">
250     <S11:Header>
251         <wsse:Security>
252             ...
253                 <wsse:SecurityTokenReference>
254    wsse11:TokenType="http://docs.oasis-open.org/wss/oasis-wss-kerberos-
255    token-profile-1.1#Kerberosv5_AP_REQ"
256                     <wsse:KeyIdentifier ValueType="http://docs.oasis-
257    open.org/wss/oasis-wss-kerberos-token-profile-1.1#Kerb
258    erosv5APREQSHA1">GbsDt+WmD9XlnUUWbY/nhBveW8I=
259                     </wsse:KeyIdentifier>
260                 </wsse:SecurityTokenReference>
261             ...
262         </wsse:Security>
263     </S11:Header>
264     <S11:Body>
265         ...
266     </S11:Body>
267 </S11:Envelope>
268
```

3.4 Authentication

When a Kerberos ticket is referenced as a signature key, the signature algorithm [DSIG] MUST be a hashed message authentication code.

When a Kerberos ticket is referenced as an encryption key, the encryption algorithm MUST be a symmetric encryption algorithm.

The value of the signature or encryption key is constructed from the value of the Kerberos sub-key when it is present in the authenticator or a session key from the ticket if the sub-key is absent, either by using the Kerberos sub-key or session key directly or using a key derived from that key using a mechanism agreed to by the communicating parties.

3.5 Encryption

When a Kerberos ticket is referenced as an encryption key, the encryption algorithm MUST be a symmetric encryption algorithm.

The value of the signature or encryption key is constructed from the value of the Kerberos sub-key when it is present in the authenticator or a session key from the ticket if the sub-key is absent, either by using the Kerberos sub-key or session key directly or using a key derived from that key using a mechanism agreed to by the communicating parties..

3.6 Principal Name

Kerberos principal name definition and mapping of non-Kerberos names to Kerberos V principal names are out of scope of this document.

3.7 Error Codes

When using Kerberos tokens, it is RECOMMENDED to use the error codes defined in the WSS: SOAP Message Security specification. However, implementations MAY use custom errors, defined in private namespaces if they desire. Care should be taken not to introduce security vulnerabilities in the errors returned.

296 4 Threat Model and Countermeasures

297 The use of Kerberos assertion tokens with WSS: SOAP Message Security introduces no new
298 message-level threats beyond those identified for Kerberos itself or by WSS: SOAP Message
299 Security with other types of security tokens.

300

301 One potential threat is that of key re-use. The mechanisms described in WSS: SOAP Message
302 Security can be used to prevent replay of the message; however, it is possible that for some
303 service scopes, there are host security concerns of key hijacking within a Kerberos infrastructure.
304 The use of the AP-REQ and its associated authenticator and sequencer mitigate this threat.

305

306 Message alteration and eavesdropping can be addressed by using the integrity and confidentiality
307 mechanisms described in WSS: SOAP Message Security. Replay attacks can be addressed by
308 using message timestamps and caching, as well as other application-specific tracking
309 mechanisms. For Kerberos tokens ownership is verified by use of keys, so man-in-the-middle
310 attacks are generally mitigated.

311

312 It is strongly recommended that GSS wrapped AP-REQ be used or that unwrapped AP-REQ be
313 combined with timestamp be used to prevent replay attack.

314

315 It is strongly recommended that all relevant and immutable message data be signed to prevent
316 replay attacks.

317

318 It should be noted that transport-level security MAY be used to protect the message and the
319 security token in cases where neither a GSS-API framed KRB_AP_REQ token or an unwrapped
320 AP-REQ combined with timestamp and signature are being used.

321 5 References

322 The following are normative references

323 324	**[2119]**	S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, Harvard University, March 1997
325 326	**[Kerb]**	J. Kohl and C. Neuman, "The Kerberos Network Authentication Service (V5)," RFC 1510, September 1993, http://www.ietf.org/rfc/rfc1510.txt .
327 328	**[KEYWORDS]**	S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, Harvard University, March 1997
329	**[S11]**	W3C Note, "SOAP: Simple Object Access Protocol 1.1," 08 May 2000.
330 331	**[S12]**	W3C Recommendation, "SOAP Version 1.2 Part 1: Messaging Framework", 23 June 2003.
332 333 334	**[URI]**	T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," RFC 3986, MIT/LCS, Day Software, Adobe Systems, January 2005.
335 336 337 338	**[WSS]**	A. Nadalin et al., Web Services Security: SOAP Message Security 1.1 (WS-Security 2004), OASIS Standard, http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.1.pdf.
339 340	**[1964]**	J. Linn , The Kerberos Version 5 GSS-API Mechanism, RFC 1964, June 1996.
341 342 343	**[4121]**	L, Zhu, K. Jaganathan, S. Hartman, The Kerberos Version 5 Generic Security Service Application Program Interface (GSS-API) Mechanism: Version 2, RFC 4121, July 2005.

344 The following are non-normative references

345	**[ISG]**	Informational RFC 2828, "Internet Security Glossary," May 2000.
346	**[XML-ns]**	W3C Recommendation, "Namespaces in XML," 14 January 1999.
347 348 349	**[DSIG]**	D. Eastlake, J. R., D. Solo, M. Bartel, J. Boyer , B. Fox , E. Simon. *XML-Signature Syntax and Processing*, W3C Recommendation, 12 February 2002. http://www.w3.org/TR/xmldsig-core/.

Appendix A. Acknowledgments

350

351 Current Contributors:

Michael	Hu	Actional
Maneesh	Sahu	Actional
Duane	Nickull	Adobe Systems
Gene	Thurston	AmberPoint
Frank	Siebenlist	Argonne National Laboratory
Hal	Lockhart	BEA Systems
Denis	Pilipchuk	BEA Systems
Corinna	Witt	BEA Systems
Steve	Anderson	BMC Software
Rich	Levinson	Computer Associates
Thomas	DeMartini	ContentGuard
Merlin	Hughes	Cybertrust
Dale	Moberg	Cyclone Commerce
Rich	Salz	Datapower
Sam	Wei	EMC
Dana S.	Kaufman	Forum Systems
Toshihiro	Nishimura	Fujitsu
Kefeng	Chen	GeoTrust
Irving	Reid	Hewlett-Packard
Kojiro	Nakayama	Hitachi
Paula	Austel	IBM
Derek	Fu	IBM
Maryann	Hondo	IBM
Kelvin	Lawrence	IBM
Michael	McIntosh	IBM
Anthony	Nadalin	IBM
Nataraj	Nagaratnam	IBM
Bruce	Rich	IBM
Ron	Williams	IBM
Don	Flinn	Individual
Kate	Cherry	Lockheed Martin
Paul	Cotton	Microsoft
Vijay	Gajjala	Microsoft
Martin	Gudgin	Microsoft
Chris	Kaler	Microsoft
Frederick	Hirsch	Nokia
Abbie	Barbir	Nortel
Prateek	Mishra	Oracle
Vamsi	Motukuru	Oracle
Ramana	Turlapi	Oracle
Ben	Hammond	RSA Security
Rob	Philpott	RSA Security

Blake	Dournaee	Sarvega
Sundeep	Peechu	Sarvega
Coumara	Radja	Sarvega
Pete	Wenzel	SeeBeyond
Manveen	Kaur	Sun Microsystems
Ronald	Monzillo	Sun Microsystems
Jan	Alexander	Systinet
Symon	Chang	TIBCO Software
John	Weiland	US Navy
Hans	Granqvist	VeriSign
Phillip	Hallam-Baker	VeriSign
Hemma	Prafullchandra	VeriSign

352 Previous Contributors:

Peter	Dapkus	BEA
Guillermo	Lao	ContentGuard
TJ	Pannu	ContentGuard
Xin	Wang	ContentGuard
Shawn	Sharp	Cyclone Commerce
Ganesh	Vaideeswaran	Documentum
Tim	Moses	Entrust
Carolina	Canales-Valenzuela	Ericsson
Tom	Rutt	Fujitsu
Yutaka	Kudo	Hitachi
Jason	Rouault	HP
Bob	Blakley	IBM
Joel	Farrell	IBM
Satoshi	Hada	IBM
Hiroshi	Maruyama	IBM
David	Melgar	IBM
Kent	Tamura	IBM
Wayne	Vicknair	IBM
Phil	Griffin	Individual
Mark	Hayes	Individual
John	Hughes	Individual
Peter	Rostin	Individual
Davanum	Srinivas	Individual
Bob	Morgan	Individual/Internet2
Bob	Atkinson	Microsoft
Keith	Ballinger	Microsoft
Allen	Brown	Microsoft
Giovanni	Della-Libera	Microsoft
Alan	Geller	Microsoft
Johannes	Klein	Microsoft
Scott	Konersmann	Microsoft
Chris	Kurt	Microsoft

Brian	LaMacchia	Microsoft
Paul	Leach	Microsoft
John	Manferdelli	Microsoft
John	Shewchuk	Microsoft
Dan	Simon	Microsoft
Hervey	Wilson	Microsoft
Jeff	Hodges	Neustar
Senthil	Sengodan	Nokia
Lloyd	Burch	Novell
Ed	Reed	Novell
Charles	Knouse	Oblix
Vipin	Samar	Oracle
Jerry	Schwarz	Oracle
Eric	Gravengaard	Reactivity
Andrew	Nash	Reactivity
Stuart	King	Reed Elsevier
Martijn	de Boer	SAP
Jonathan	Tourzan	Sony
Yassir	Elley	Sun
Michael	Nguyen	The IDA of Singapore
Don	Adams	TIBCO
Morten	Jorgensen	Vordel

353 # Appendix B. Revision History

Rev	Date	By Whom	What

354