Web Services Security
Kerberos Token Profile 1.1

Committee Draft – Tuesday 14 June 2005

OASIS identifier:

{*WSS: SOAP Message Security* }-{Kerboer Token Profile }-{*1.1*} (Word) (PDF)

Location:

http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-wss-kerberos-token-profile-1.1

Technical Commitee:

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 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).

Notices

63 Table of Contents

81

1 Introduction

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

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

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

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

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

2 Notations and Terminology

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

2.1 Notational Conventions

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 [2119].

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

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

2.2 Namespaces

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

```
http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd
http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd
http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-wss-wssecurity-
secext-1.1.xsd
```

Note that this specification does not introduce new schema elements.

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-wssecurity-secext-1.0.xsd

wsse11	http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-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#

124

125 The URLs provided for the *wsse* and *wsu* namespaces can be used to obtain the schema files.
126 URI fragments defined in this specification are relative to the following base URI unless otherwise
127 specified:

128 http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-wss-kerberos-token-profile-1.1

129 2.3 Terminology

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

131

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

134

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

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

137

3 Usage

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

Identification: http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-wss-kerberos-token-profile-1.1

3.1 Processing Model

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

3.2 Attaching Security Tokens

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

URI	Description
http://docs.oasis-open.org/wss/2005/xx/oasis-2005xx-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/2005/xx/oasis-2005xx-wss-kerberos-token-profile-1.1#GSS_Kerberosv5_AP_REQ	A GSS wrapped Kerberos v5 AP-REQ as defined in the GSSAPI specification. This ValueType is used when the ticket is an AP Request (ST + Authenticator).

It should be noted that the URIs in the table above also serves as the official URIs identifying the Kerberos token defined in this specification.

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

The octet sequence of the either the GSS wrapped Kerberos ticket or the Kerberos ticket (e.g. AP-REQ) is encoded using the indicated algorithm (e.g. base 64) and the result is placed inside of the <wsse:BinarySecurityToken> element.

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

```
<S11:Envelope xmlns:S11="...">
```



```
    <S11:Header>
        <wsse:Security xmlns:wsse="...">
            <wsse:BinarySecurityToken
            xmlns:wsse="... "
                wsu:Id="myToken"
                ValueType="...#Kerberosv5_AP_REQ"
                EncodingType="...#Base64Binary">
                MIIEZzCCA9CgAwIBAgIQEmtJZc0...
            </wsse:BinarySecurityToken>
            ...
        </wsse:Security>
    </S11:Header>
    <S11:Body>
        ...
    </S11:Body>
</S11:Envelope>
```

3.3 Identifying and Referencing Kerberos Tokens

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

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

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

Key Name references MUST NOT be used.

When a Kerberos Token is referenced using <wsse:SecurityTokenReference> the *@ValueType* attribute is not required. If specified, the URI listed above as Kerberos token type MUST be specified.

The <wsse:SecurityTokenReference> element from which the reference is made contains the <wsse:KeyIdentifier> element. The <wsse:KeyIdentifier> element MUST have a ValueType attribute with the value #Kerberosv5APREQSHA1 and its contents MUST be the SHA1 of GSS wrapped or unwrapped AP-REQ, encoded as per the <wsse:KeyIdentifier> element's EncodingType attribute.

Reference Identifier	**ValueType URI**	Description
Kerberos v5 AP-REQ	#Kerberosv5APREQSHA1	SHA1 of the v5 AP-REQ octets, either GSS wrapped Kerberos AP-REQ or just the Kerberos AP-REQ.

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

```
<S11:Envelope xmlns:S11="...">
    <S11:Header>
        <wsse:Security xmlns:wsse="...">
            <wsse:BinarySecurityToken
            xmlns:wsse="... "
                wsu:Id="myToken"
                ValueType="...#Kerberosv5_AP_REQ"
                EncodingType="...#Base64Binary">
                MIIEZzCCA9CgAwIBAgIQEmtJZc0...
            </wsse:BinarySecurityToken>
            ...
               <wsse:SecurityTokenReference>
                   <wsse:Reference URI="#myToken"/>
               </wsse:SecurityTokenReference>
            ...
        </wsse:Security>
    </S11:Header>
    <S11:Body>
        ...
    </S11:Body>
</S11:Envelope>
```

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:

```
<S11:Envelope xmlns:S11="...">
    <S11:Header>
        <wsse:Security xmlns:wsse="...">
            ...
               <wsse:SecurityTokenReference
<wsse:KeyIdentifier    ValueType="...#Kerberosv5APREQSHA1">
                       EZzCCA9CgAwIB...
                   <wsse:KeyIdentifier>
               </wsse:SecurityTokenReference>
            ...
        </wsse:Security>
    </S11:Header>
    <S11:Body>
        ...
    </S11:Body>
</S11:Envelope>
```

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.

4 Threat Model and Countermeasures

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

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

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

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

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

It should be noted that transport-level security MAY be used to protect the message and the security token if either a wrapped AP-REQ or that unwrapped AP-REQ be combined with timestamp and signature are not being used.

5 References

The following are normative references

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

[Kerb] J. Kohl and C. Neuman, "The Kerberos Network Authentication Service (V5)," RFC 1510, September 1993, http://www.ietf.org/rfc/rfc1510.txt .

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

[S11] W3C Note, "SOAP: Simple Object Access Protocol 1.1," 08 May 2000.

[S12] W3C Recommendation, "SOAP Version 1.2 Part 1: Messaging Framework", 23 June 2003.

[URI] T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," RFC 3986, MIT/LCS, Day Software, Adobe Systems, January 2005.

The following are non-normative references

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

[WSS] A. Nadalin et al., Web Services Security: SOAP Message Security 1.0 (WS-Security 2004), OASIS Standard 200401, March 2004, http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf.

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

[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

This specification was developed as a result of joint work of many individuals from the WSS TC.

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

Gene	Thurston	AmberPoint
Frank	Siebenlist	Argonne National Lab
Merlin	Hughes	Baltimore Technologies
Irving	Reid	Baltimore Technologies
Peter	Dapkus	BEA
Hal	Lockhart	BEA
Steve	Anderson	BMC (Sec)
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
Yutaka	Kudo	Hitachi
Jason	Rouault	HP
Bob	Blakley	IBM
Joel	Farrell	IBM
Satoshi	Hada	IBM
Maryann	Hondo	IBM
Hiroshi	Maruyama	IBM

David	Melgar	IBM
Anthony	Nadalin	IBM
Nataraj	Nagaratnam	IBM
Wayne	Vicknair	IBM
Kelvin	Lawrence	IBM (co-Chair)
Don	Flinn	Individual
Bob	Morgan	Individual
Bob	Atkinson	Microsoft
Keith	Ballinger	Microsoft
Allen	Brown	Microsoft
Paul	Cotton	Microsoft
Giovanni	Della-Libera	Microsoft
Vijay	Gajjala	Microsoft
Johannes	Klein	Microsoft
Scott	Konermann	Microsoft
Chris	Kurt	Microsoft
Brian	LaMacchia	Microsoft
Paul	Leach	Microsoft
John	Manferdell	Microsoft
John	Shewchuk	Microsoft
Dan	Simon	Microsoft
Hervey	Wilson	Microsoft
Chris	Kaler	Microsoft (co-Chair)
Prateek	Mishra	Netegrity
Frederick	Hirsch	Nokia
Senthil	Sengodan	Nokia
Lloyd	Burch	Novell
Ed	Reed	Novell
Charles	Knouse	Oblix
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
Symon	Chang	TIBCO
John	Weiland	US Navy
Phillip	Hallam-Baker	VeriSign
Mark	Hays	Verisign
Hemma	Prafullchandra	VeriSign

333

Appendix B. 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	20-Jan-04	Revise based on comments, switch to new URLs and formats and recent decisions in TC
05	27-Jul-04	Revise based on comments and recent decisions in TC
06	16-May-05	Revise based on comments and recent decisions in TC. Issues 381, 382, 383, 384, 385, 386, 387
07	17-May-05	Formatting Issues
08	14-June-05	Issues 396