

# [MS-WSTEP]: WS-Trust X.509v3 Token Enrollment Extensions

---

## Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](http://www.microsoft.com/trademarks).
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

## Revision Summary

Date	Revision History	Revision Class	Comments
12/05/2008	0.1	Major	Initial Availability
01/16/2009	0.1.1	Editorial	Revised and edited the technical content.
02/27/2009	0.1.2	Editorial	Revised and edited the technical content.
04/10/2009	0.1.3	Editorial	Revised and edited the technical content.
05/22/2009	0.2	Minor	Updated the technical content.
07/02/2009	1.0	Major	Updated and revised the technical content.
08/14/2009	1.1	Minor	Updated the technical content.
09/25/2009	2.0	Major	Updated and revised the technical content.
11/06/2009	3.0	Major	Updated and revised the technical content.
12/18/2009	4.0	Major	Updated and revised the technical content.
01/29/2010	5.0	Major	Updated and revised the technical content.
03/12/2010	5.1	Minor	Updated the technical content.
04/23/2010	5.1.1	Editorial	Revised and edited the technical content.
06/04/2010	5.1.2	Editorial	Revised and edited the technical content.
07/16/2010	6.0	Major	Significantly changed the technical content.
08/27/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
10/08/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/07/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/11/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/25/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
05/06/2011	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
06/17/2011	6.1	Minor	Clarified the meaning of the technical content.

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
09/23/2011	6.1	No change	No changes to the meaning, language, or formatting of the technical content.
12/16/2011	7.0	Major	Significantly changed the technical content.
03/30/2012	7.0	No change	No changes to the meaning, language, or formatting of the technical content.
07/12/2012	7.1	Minor	Clarified the meaning of the technical content.
10/25/2012	7.2	Minor	Clarified the meaning of the technical content.
01/31/2013	7.2	No change	No changes to the meaning, language, or formatting of the technical content.
08/08/2013	8.0	Major	Significantly changed the technical content.
11/14/2013	8.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/13/2014	8.0	No change	No changes to the meaning, language, or formatting of the technical content.
05/15/2014	8.1	Minor	Clarified the meaning of the technical content.

# Contents

<b>1 Introduction</b>	<b>6</b>
1.1 Glossary	6
1.2 References	7
1.2.1 Normative References	7
1.2.2 Informative References	8
1.3 Overview	8
1.4 Relationship to Other Protocols	10
1.5 Prerequisites/Preconditions	11
1.6 Applicability Statement	11
1.7 Versioning and Capability Negotiation	11
1.8 Vendor-Extensible Fields	11
1.9 Standards Assignments	11
<b>2 Messages</b>	<b>12</b>
2.1 Transport	12
2.2 Common Message Syntax	12
2.2.1 Namespaces	12
2.2.2 Messages	12
2.2.3 Elements	12
2.2.4 Complex Types	12
2.2.5 Simple Types	12
2.2.6 Attributes	13
2.2.7 Groups	13
2.2.8 Attribute Groups	13
<b>3 Protocol Details</b>	<b>14</b>
3.1 SecurityTokenService Server Details	14
3.1.1 Abstract Data Model	15
3.1.1.1 Authentication	15
3.1.1.1.1 Kerberos Authentication	15
3.1.1.1.2 X.509v3 Certificate Authentication	15
3.1.1.1.3 Username and Password Authentication	15
3.1.1.1.4 No (Anonymous) Authentication	15
3.1.2 Timers	15
3.1.3 Initialization	15
3.1.4 Message Processing Events and Sequencing Rules	16
3.1.4.1 wst:RequestSecurityToken2	16
3.1.4.1.1 Messages	16
3.1.4.1.1.1 wst:RequestSecurityTokenMsg	16
3.1.4.1.1.2 wst:RequestSecurityTokenResponseCollectionMsg	16
3.1.4.1.2 Elements	17
3.1.4.1.2.1 wstep:CertificateEnrollmentWSDetail	17
3.1.4.1.2.2 DispositionMessage	17
3.1.4.1.2.3 wst:KeyExchangeToken	17
3.1.4.1.2.4 RequestID	17
3.1.4.1.2.5 wst:RequestSecurityToken	17
3.1.4.1.2.6 RequestSecurityTokenResponseCollection	17
3.1.4.1.2.7 wst:RequestType	18
3.1.4.1.2.8 wst:TokenType	18
3.1.4.1.3 Complex Types	18

3.1.4.1.3.1	DispositionMessageType .....	18
3.1.4.1.3.2	wst:RequestedSecurityTokenType .....	18
3.1.4.1.3.3	wst:RequestSecurityTokenType .....	19
3.1.4.1.3.4	wst:RequestSecurityTokenResponseType.....	21
3.1.4.1.3.5	wst:RequestSecurityTokenResponseCollectionType .....	22
3.1.4.1.3.6	wst:RequestTypeEnum .....	23
3.1.4.1.3.7	wstep:CertificateEnrollmentWSDetailType.....	23
3.1.4.1.4	Attributes.....	23
3.1.4.2	Processing Rules.....	24
3.1.4.2.1	WSTEP Action: Request Security Token Processing Rules .....	24
3.1.4.2.1.1	New and Renewal Request Processing .....	24
3.1.4.2.1.2	QueryTokenStatus Request Processing .....	25
3.1.4.2.2	KET Action: Request Security Token Processing Rules.....	25
3.1.4.2.2.1	Key Exchange Token Request Processing.....	25
3.1.5	Timer Events .....	25
3.1.6	Other Local Events .....	26
<b>4</b>	<b>Protocol Examples.....</b>	<b>27</b>
4.1	RequestSecurityToken Request/Response Message Sequence.....	27
4.1.1	Standard Certificate Request .....	27
4.1.1.1	RequestSecurityToken Message (Issue Request).....	27
4.1.1.2	Server RequestSecurityToken Response .....	28
4.1.2	Key Exchange Token Request .....	30
4.1.2.1	Client Exchange Token Request.....	30
4.1.2.2	Server Key Exchange Token Response .....	31
4.1.3	Retrieval of a previously pended certificate request with Query Token Status.....	32
4.1.3.1	Client Request.....	32
4.1.4	Message exchange with a server fault.....	33
4.1.4.1	Client Request.....	33
4.1.4.2	Server Fault Response.....	33
4.1.5	Certificate Renewal.....	33
4.1.5.1	Client Renewal Request .....	33
4.1.5.2	Server Request Security Token Response .....	36
<b>5</b>	<b>Security.....</b>	<b>39</b>
5.1	Security Considerations for Implementers.....	39
5.2	Index of Security Parameters .....	39
<b>6</b>	<b>Appendix A: Full WSDL.....</b>	<b>40</b>
<b>7</b>	<b>Appendix B: Product Behavior.....</b>	<b>41</b>
<b>8</b>	<b>Change Tracking.....</b>	<b>42</b>
<b>9</b>	<b>Index .....</b>	<b>44</b>

# 1 Introduction

The WS-Trust X.509v3 Token Enrollment Protocol Extensions are extensions of WS-Trust that are used by a system to request that a **certificate** be issued.

The communication is initiated by a requesting client who requests a new certificate, retrieval of an issued certificate, or retrieval of a server certificate. The server processes the request and generates a response based on the request type.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

## 1.1 Glossary

The following terms are defined in [\[MS-GLOS\]](#):

- Abstract Syntax Notation One (ASN.1)**
- certificate**
- certification authority (CA)**
- globally unique identifier (GUID)**
- Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**
- Public Key Cryptography Standards (PKCS)**
- SOAP action**
- SOAP body**
- SOAP fault**
- SOAP fault code**
- SOAP fault detail**
- SOAP header**
- SOAP header block**
- SOAP message**
- SOAP mustUnderstand attribute**
- Unicode**
- Uniform Resource Locator (URL)**
- Web Services Description Language (WSDL)**
- WSDL message**
- WSDL port type**
- WSDL operation**
- X.509**
- XML**
- XML namespace**
- XML schema (XSD)**

The following terms are specific to this document:

**Certificate Management Messages over CMS (CMC):** An internet standard for transport mechanisms for CMS [\[RFC2797\]](#).

**Cryptographic Message Syntax (CMS):** An internet standard for cryptographically protected messages [\[RFC3852\]](#).

**Security Token Service (STS):** A special type of server defined in WS-Trust [\[WSTrust1.3\]](#).

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## 1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[MS-WCCE] Microsoft Corporation, "[Windows Client Certificate Enrollment Protocol](#)".

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>

[RFC2797] Myers, M., Liu, X., Schaad, J., and Weinstein, J., "Certificate Management Messages Over CMS", RFC 2797, April 2000, <http://www.ietf.org/rfc/rfc2797.txt>

[RFC2986] Nystrom, M., and Kaliski, B., "PKCS#10: Certificate Request Syntax Specification", RFC 2986, November 2000, <http://www.ietf.org/rfc/rfc2986.txt>

[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", RFC 3066, January 2001, <http://www.ietf.org/rfc/rfc3066.txt>

[RFC3852] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004, <http://www.ietf.org/rfc/rfc3852.txt>

[RFC5246] Dierks, T., and Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008, <http://www.ietf.org/rfc/rfc5246.txt>

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[WSS] OASIS, "Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)", February 2006, <http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf>

[WSSUTP] OASIS Standard, "Web Services Security UsernameToken Profile 1.0", March 2004, <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0.pdf>

[WSTrust1.3] Lawrence, K., Kaler, C., Nadalin, A., et al., "WS-Trust 1.3", March 2007, <http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html>

[WSTrust1.3Schema] OASIS Standard, "WS-Trust 1.3", <http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3.xsd>

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1] Thompson, H.S., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

[XMLSCHEMA2] Biron, P.V., and Malhotra, A., Eds., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

## 1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)".

[SCEP] Nourse, A., and Vilhuber, J. Ed., "Cisco Systems' Simple Certificate Enrollment Protocol", April 2009, <http://tools.ietf.org/html/draft-nourse-scep-19>

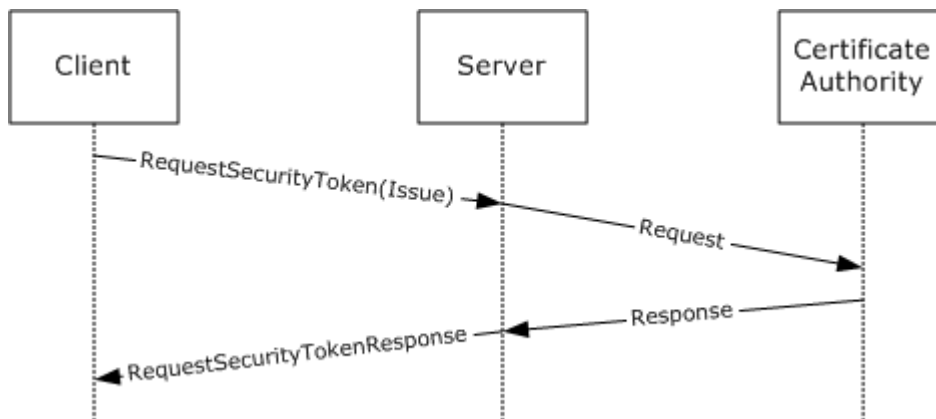
## 1.3 Overview

The WS-Trust X.509v3 Token Enrollment Extensions (WSTEP) defines the token enrollment profile for WS-Trust [\[WSTrust1.3\]](#) to allow a client to request **X.509v3** certificates.

Existing **certificate authorities** support **ASN.1** formats such as PKCS#10 ([\[RFC2986\]](#)), PKCS#7 ([\[RFC3852\]](#)), or CMC ([\[RFC2797\]](#)) to encode a certificate request, and those requests are carried in an existing protocol, such as Windows Client Certificate Enrollment Protocol [\[MS-WCCE\]](#) or Cisco's SCEP ([\[SCEP\]](#)). WSTEP also carries those requests from the client to the issuer.

WSTEP provides for issuance, renewal, and delayed-issuance scenarios for X.509v3 digital certificates. The server is known in WS-Trust [\[WSTrust1.3\]](#) terminology as a **Security Token Service (STS)**.

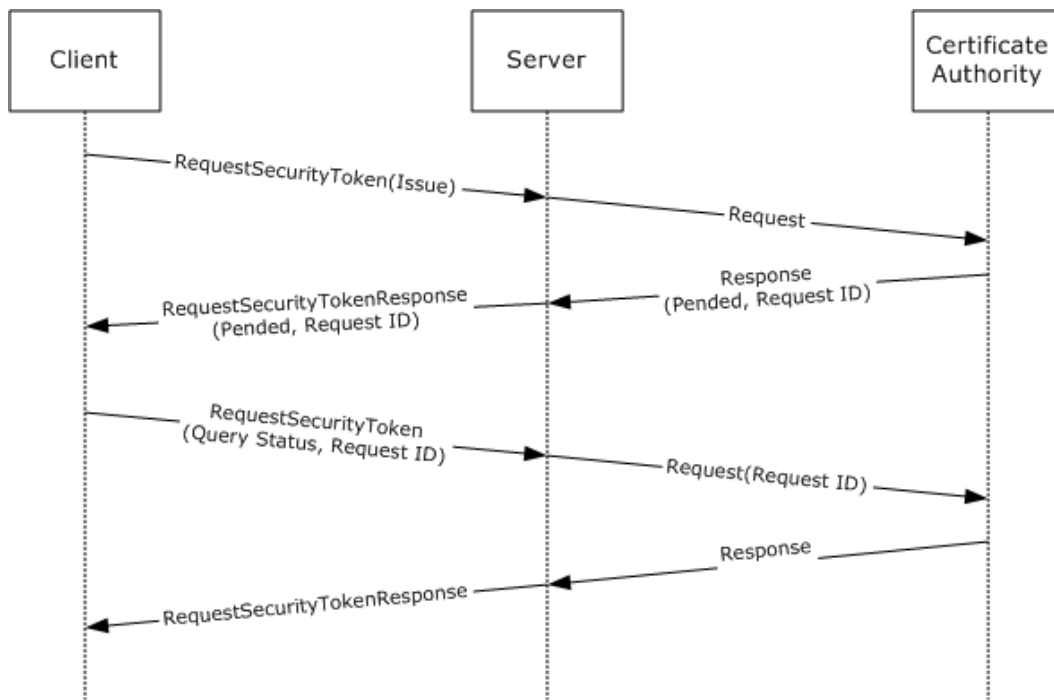
The WS-Trust protocol [\[WSTrust1.3\]](#) definition provides the framework for the STS and for enrollment profile extensions. A typical client interacts with a STS with a request security token (RST) message. The STS responds to a client request security token message with a request security token response (RSTR) or a **SOAP fault**.



**Figure 1: Typical sequence for certificate enrollment**

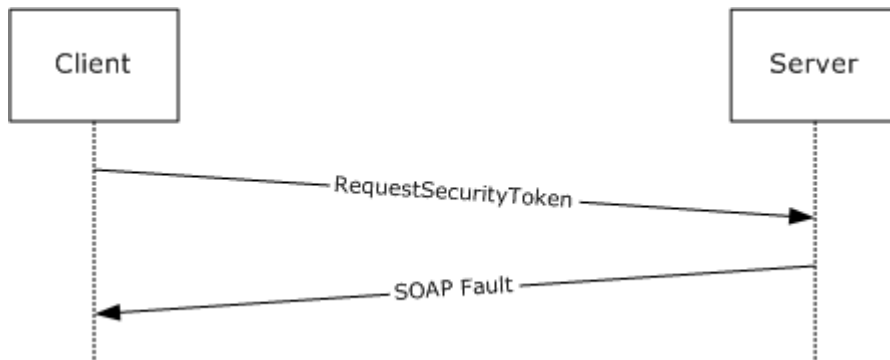
The following figure shows a scenario in which a request cannot be satisfied immediately. In this scenario, the client makes a request, and the server reply indicates that the request is pending some other action. The client then queries the request at a later time, presumably after any conditions for its satisfaction have been met, and receives a reply that the request was issued, rejected, or is still pending.





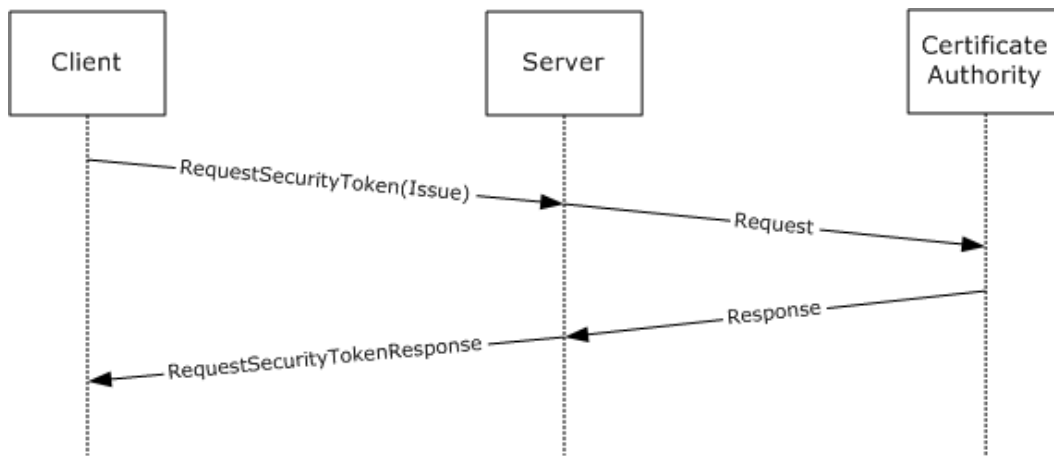
**Figure 2: Typical sequence for a pended certificate enrollment request**

In some circumstances, the client request may be rejected. In these instances, the STS responds with a SOAP fault. The following figure shows the typical sequence.



**Figure 3: Typical sequence for a rejected certificate renewal request**

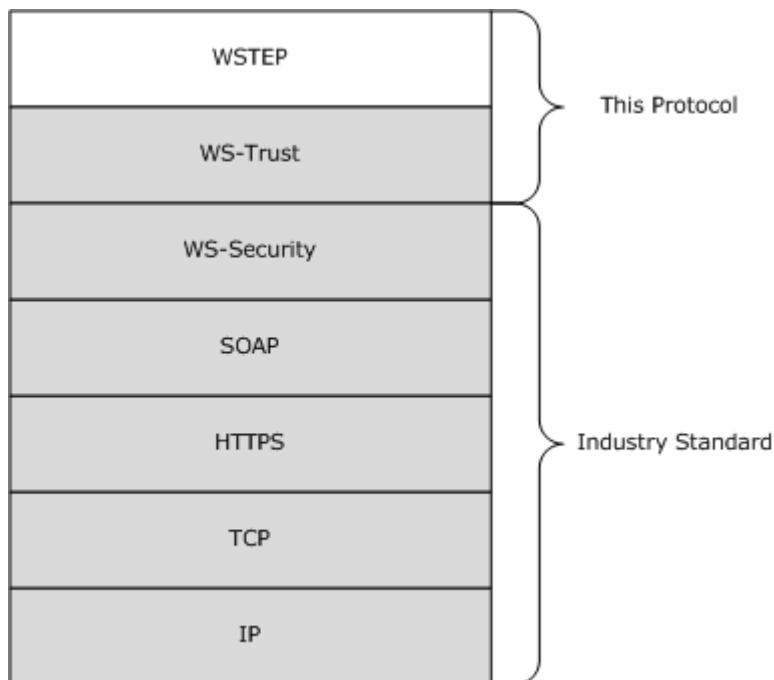
The following figure is an example of a message exchange for a renewal request. A renewal request uses an existing certificate and requests a new lifespan. From the point of view of the WSTEP protocol, this is the same as an issue request, as the message format is unchanged.



**Figure 4: Typical sequence for a certificate renewal request**

### 1.4 Relationship to Other Protocols

The following figure shows the WSTEP Protocol stack diagram.



**Figure 5: WSTEP Protocol stack diagram**

The WSTEP protocol specification is a profile of the WS-Trust Protocol [\[WSTrust1.3\]](#) and makes use of the SOAP and **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)** protocols for messaging and security.

## 1.5 Prerequisites/Preconditions

The WSTEP protocol specification facilitates the issuance of X.509v3 certificates. A server implementation of the protocol requires the functionality of a certificate authority, capable of interpreting requests in at least one of PKCS#7, PKCS#10, or **Certificate Management Messages over CMS (CMC)**.

## 1.6 Applicability Statement

The WSTEP protocol specification is applicable only for requests for X.509v3 certificates.

## 1.7 Versioning and Capability Negotiation

The WSTEP protocol specification does not include versioning and capability negotiation.

## 1.8 Vendor-Extensible Fields

The WSTEP protocol specification does not include any vendor-extensible fields. WSTEP adheres to the WS-Trust 1.3 [\[WSTrust1.3\]](#) provided extension points.

## 1.9 Standards Assignments

None.

## 2 Messages

### 2.1 Transport

SOAP version 1.2 MUST be used for messaging for the WSTEP protocol. HTTPS protocol MUST be used as the transport.

### 2.2 Common Message Syntax

This section contains common definitions used by this protocol. The syntax of the definitions uses the **XML schema** as defined in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and the **Web Services Description Language** as defined in [\[WSDL\]](#).

#### 2.2.1 Namespaces

This specification defines and references various **XML namespaces**, using the mechanisms specified in [\[XMLNS\]](#). Although this specification associates a specific **XML namespace** prefix for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefixes and XML namespaces used in this specification are as follows.

Prefix	Namespace URI	Reference
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>	<a href="#">[XMLSCHEMA1]</a>
wst	<a href="http://docs.oasis-open.org/ws-sx/ws-trust/200512">http://docs.oasis-open.org/ws-sx/ws-trust/200512</a>	<a href="#">[WSTrust1.3]</a>
auth	<a href="http://schemas.xmlsoap.org/ws/2006/12/authorization">http://schemas.xmlsoap.org/ws/2006/12/authorization</a>	<a href="#">[XMLSCHEMA1]</a>
wsu	<a href="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd</a>	
wsse	<a href="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-secext-1.0.xsd</a>	
wstep	<a href="http://schemas.microsoft.com/windows/pki/2009/01/enrollment">http://schemas.microsoft.com/windows/pki/2009/01/enrollment</a>	MS-WSTEP

#### 2.2.2 Messages

None.

#### 2.2.3 Elements

This specification does not define any common XML schema element definitions.

#### 2.2.4 Complex Types

This specification does not define any common XML schema complex type definitions.

#### 2.2.5 Simple Types

The WSTEP protocol specification does not define any common XML schema simple type definitions.

### **2.2.6 Attributes**

The WSTEP protocol specification does not define any common XML schema attribute definitions.

### **2.2.7 Groups**

The WSTEP protocol specification does not define any common XML schema group definitions.

### **2.2.8 Attribute Groups**

The WSTEP protocol specification does not define any common XML schema attribute group definitions.

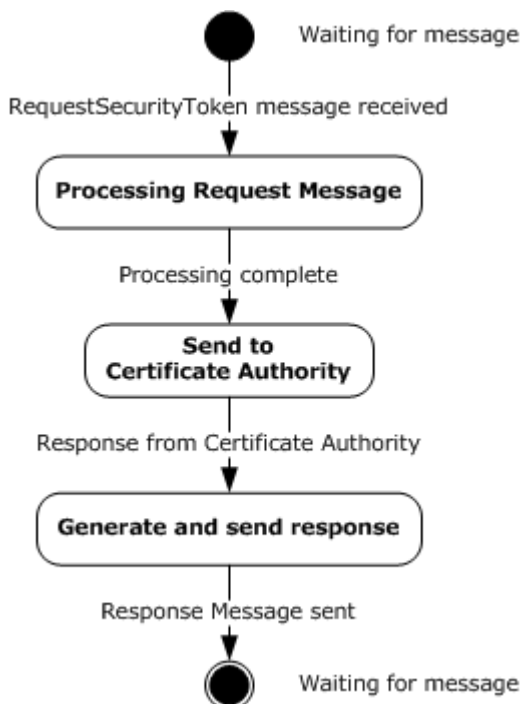
### 3 Protocol Details

The client side of this protocol is a simple pass-through. No additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport layer, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

This section addresses the message processing model for the protocol. It includes related information required by an implementation to successfully send and consume protocol messages.

#### 3.1 SecurityTokenService Server Details

The **SecurityTokenService** hosts a message endpoint that receives **RequestSecurityToken** messages. When received, the server processes the client request and sends it to the certificate authority. Upon receiving a response from the certificate authority, a response is generated, and the server sends either a **RequestSecurityTokenResponse** message or a SOAP fault. When the message has been sent to the client, the server returns to the waiting state.



**Figure 6: Security token service state model**

The items of information that are communicated between the server and the certificate authority are specified in this section, but the method of communication used, including timeout and error handling (local API, local remote procedure call (RPC), or some other protocol) is not specified.

The certificate authority MAY have additional requirements that MUST be met in order to issue an X509v3 Certificate, such as manager approval, payment processing, or validation of request information. In these instances, a certificate authority response indicating the issuance is pending.

### 3.1.1 Abstract Data Model

A server supporting the WSTEP protocol maintains a relationship to an issuer which processes messages submitted by the server. When communicating with requestors, a server can support a variety of languages.

**Issuer:** An address of a certificate authority (CA). The format of the stored address is specific to the implementation and to the form of communication used between the Issuer and the Server.

**SupportedLanguages:** A list of language identifiers supported by the server. The set of languages are of type `xml:lang` and defined in [\[RFC3066\]](#).

**DefaultLanguage:** The default language for the server. `DefaultLanguage` is of type `xml:lang`, and the set of supported languages is defined in [\[RFC3066\]](#).

#### 3.1.1.1 Authentication

The WS-Trust X509v3 Enrollment Extensions use the authentication provisions in WS-Security ([\[WSS\]](#)) for the X509v3 Security Token issuer to authenticate the X509v3 Security Token requestor. This section defines the schema used to express the credential descriptor for each supported credential type.

##### 3.1.1.1.1 Kerberos Authentication

Authentication using Kerberos is done at the transport layer.

##### 3.1.1.1.2 X.509v3 Certificate Authentication

Authentication using X509v3 certificates is done at the transport level using Transport Level Security (TLS) 1.2 as defined in [\[RFC5246\]](#).

##### 3.1.1.1.3 Username and Password Authentication

The username and password credential is provided in a request message using the WS-Security Username Token Profile 1.0. The username is provided as defined in section 3.1 of the Ws-Security document [\[WSSUTP\]](#).

##### 3.1.1.1.4 No (Anonymous) Authentication

If no authentication is provided at either the transport layer or the message layer, the request is considered to be anonymous. Anonymous authentication is supported only for renewal requests, where the signature from the existing certificate on the request object serves as authentication for the X509v3 Security Token requestor.

### 3.1.2 Timers

None.

### 3.1.3 Initialization

The *SupportedLanguages* object MUST be initialized with the set of languages that the server supports.

The *DefaultLanguage* parameter MUST be initialized with the language that is to be used by the server when a request does not define a language preference, or the preference is not in *SupportedLanguages*.<1>

### 3.1.4 Message Processing Events and Sequencing Rules

Operation	Description
<a href="#">wst:RequestSecurityToken2</a>	The wst:RequestSecurityToken2 operation is the sole operation in the WSTEP protocol. It provides the mechanism for certificate enrollment requests, retrieval of pending certificate status, and the request of the server key exchange certificate. The wst:RequestSecurityToken2 operation is defined in WS-Trust 1.3 <a href="#">[WSTrust1.3]</a> .

#### 3.1.4.1 wst:RequestSecurityToken2

The wst:RequestSecurityToken2 operation provides the mechanism for certificate enrollment requests, retrieval of pending certificate status, and the request of the server key exchange certificate. The wst:SecurityTokenService port and wst:RequestSecurityToken2 operation are defined in the [\[WSTrust1.3\]](#) WSDL wsdl:portType definition.

```
<wsdl:operation name="RequestSecurityToken2">
  <wsdl:input message="wst:RequestSecurityTokenMsg" />
  <wsdl:output message="wst:RequestSecurityTokenResponseCollectionMsg" />
</wsdl:operation>
```

WSTEP makes use of the wst:RequestSecurityToken2 operation. The wst:RequestSecurityToken operation defined in the SecurityTokenService operation is not used. The wst:RequestSecurityTokenMsg message consists of a single object definition: the client request. The client request is made using the acceptable SOAP actions as defined in section [3.1.4.2](#) and RequestType values, as defined in section [3.1.4.1.2.7](#).

##### 3.1.4.1.1 Messages

The following WSDL message definitions are specific to this operation.

###### 3.1.4.1.1.1 wst:RequestSecurityTokenMsg

The wst:RequestSecurityTokenMsg is an incoming message, and is defined in WS-Trust 1.3 [\[WSTrust1.3\]](#) WSDL.

**wst:RequestSecurityToken:** An instance of a **wst:RequestSecurityTokenType** complex type as defined in section [3.1.4.1.3.3](#). The **wst:RequestSecurityToken** element defines the client request and the required information for it to be processed.

###### 3.1.4.1.1.2 wst:RequestSecurityTokenResponseCollectionMsg

The wst:RequestSecurityTokenResponseCollectionMsg is an outgoing message, and is defined in WS-Trust 1.3 [\[WSTrust1.3\]](#) WSDL.

**wst:RequestSecurityTokenResponseCollectionMsg:** An instance of a **wst:RequestSecurityTokenResponseCollection** element as defined in section [3.1.4.1.2.6](#). This element contains the results of the client request.



### 3.1.4.1.2 Elements

#### 3.1.4.1.2.1 wstep:CertificateEnrollmentWSDetail

The **wstep:CertificateEnrollmentWSDetail** element is used to convey additional information to a client as part of the SOAP fault structure when a server returns a SOAP fault.

```
<xs:element name="CertificateEnrollmentWSDetail" nillable="true"
type="wstep:CertificateEnrollmentWSDetailType" />
```

**wstep:CertificateEnrollmentWSDetail:** An instance of a `<wstep:CertificateEnrollmentWSDetailType>` as defined in section [3.1.4.1.3.7](#). If there is no additional information, the **wstep:CertificateEnrollmentWSDetail** SHOULD be omitted in the SOAP fault.

#### 3.1.4.1.2.2 DispositionMessage

```
<xs:element name="DispositionMessage"
type="wstep:DispositionMessageType" nillable="true" />
```

**DispositionMessage:** An instance of a `DispositionMessageType` object as defined in section [3.1.4.1.3.1](#).

#### 3.1.4.1.2.3 wst:KeyExchangeToken

The `<wst:KeyExchangeToken>` element is defined in WS-Trust 1.3 [\[WSTrust1.3\]](#) section 8.4.

**wst:KeyExchangeToken:** The `wst:KeyExchangeToken` element provides a key exchange token that can be used in certificate enrollment requests that include the private key.

#### 3.1.4.1.2.4 RequestID

```
<xs:element name="RequestID"
type="xs:string" nillable="true"/>
```

**RequestID:** A string identifier used to identify a request.

#### 3.1.4.1.2.5 wst:RequestSecurityToken

The `<wst:RequestSecurityToken>` element is defined in WS-Trust 1.3 [\[WSTrust1.3\]](#), section 3.1.

**wst:RequestSecurityToken:** An instance of a **wst:RequestSecurityTokenType** object as specified in section [3.1.4.1.3.3](#).

#### 3.1.4.1.2.6 RequestSecurityTokenResponseCollection

The `RequestSecurityTokenResponseCollection` is defined in WS-Trust 1.3 [\[WSTrust1.3\]](#), section 3.2.

**RequestSecurityTokenResponseCollection:** An instance of a **wst:RequestSecurityTokenResponseCollectionType** object as specified in section [3.1.4.1.3.5](#).

### 3.1.4.1.2.7 wst:RequestType

The <wst:RequestType> element is defined in [\[WSTrust1.3\]](#) section 3.1.

**wst:RequestType:** An instance of a <wst:RequestTypeOpenEnum> object as defined in [\[WSTrust1.3\]](#) XML schema definition(XSD).

The <wst:RequestType> MUST have one of the following values:

```
"http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue"  
"http://schemas.microsoft.com/windows/pki/2009/01/enrollment/QueryTokenStatus"  
"http://docs.oasis-open.org/ws-sx/ws-trust/200512/KET"
```

If the <wst:RequestType> has any other value, the server MUST respond with a SOAP fault.

### 3.1.4.1.2.8 wst:TokenType

The <TokenType> element is defined in [\[WSTrust1.3\]](#), section 3.1.

**wst:TokenType:** For the X.509 enrollment extension to WS-Trust, the <wst:tokentype> element MUST be <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3>.

### 3.1.4.1.3 Complex Types

The following XML schema complex type definitions are specific to this operation.

#### 3.1.4.1.3.1 DispositionMessageType

The DispositionMessageType is an extension to the string type that allows an attribute definition of the language for the string. The DispositionMessageType is used to provide additional information about the server processing.

```
<xs:complexType name="DispositionMessageType">  
  <xs:simpleContent>  
    <xs:extension base="xs:string">  
      <xs:attribute ref="xml:lang" use="optional" />  
    </xs:extension>  
  </xs:simpleContent>  
</xs:complexType>
```

**xs:string:** The string element contains the literal string disposition message returned from the server. The string element contains an xml:lang attribute that defines the language for the string. The language SHOULD be provided for each string element instance.

**xml:lang:** The language reference xml:lang, indicating the natural or formal language the string element content is written in.

#### 3.1.4.1.3.2 wst:RequestedSecurityTokenType

The wst:RequestedSecurityTokenType is defined in WS-Trust XML schema definition (XSD) [\[WSTrust1.3Schema\]](#).

```

<xs:complexType name="RequestedSecurityTokenType">
  <xs:sequence>
    <xs:any namespace="##any" processContents="lax" />
  </xs:sequence>
</xs:complexType>

```

The WSTEP extends wst: RequestedSecurityTokenType with two additional elements.

```

<xs:element ref="wsse:BinarySecurityToken" />
<xs:element ref="wsse:SecurityTokenReference" />

```

**wsse:BinarySecurityToken:** The wsse:BinarySecurityToken element contains the issued certificate. The issued certificate follows the encoding and data structure defined in [\[MS-WCCE\]](#) section 2.2.2.8.

**wsse:SecurityTokenReference:** A URI reference used to indicate where a pending Certificate Request can be retrieved. The server MUST provide its own URI as the value of the <wsse:BinarySecurityTokenReference:Reference> element as specified in [\[WS-Trust1.3\]](#) section 4.2.

### 3.1.4.1.3.3 wst:RequestSecurityTokenType

The **wst:RequestSecurityTokenType** complex type contains the elements for the security token request in the RequestSecurityTokenMsg message. It is the client-provided object for a certificate enrollment request. **wst:RequestSecurityTokenType** is defined in the WS-Trust [\[WS-Trust1.3\]](#) XML schema definition (XSD).

```

<xs:complexType name="RequestSecurityTokenType">
  <xs:annotation>
    <xs:documentation>
      Actual content model is non-deterministic, hence wildcard. The following shows intended
      content model:
      <xs:element ref='wst:TokenType' minOccurs='0' />
      <xs:element ref='wst:RequestType' />
      <xs:element ref='wsp:AppliesTo' minOccurs='0' />
      <xs:element ref='wst:Claims' minOccurs='0' />
      <xs:element ref='wst:Entropy' minOccurs='0' />
      <xs:element ref='wst:Lifetime' minOccurs='0' />
      <xs:element ref='wst:AllowPostdating' minOccurs='0' />
      <xs:element ref='wst:Renewing' minOccurs='0' />
      <xs:element ref='wst:OnBehalfOf' minOccurs='0' />
      <xs:element ref='wst:Issuer' minOccurs='0' />
      <xs:element ref='wst:AuthenticationType' minOccurs='0' />
      <xs:element ref='wst:KeyType' minOccurs='0' />
      <xs:element ref='wst:KeySize' minOccurs='0' />
      <xs:element ref='wst:SignatureAlgorithm' minOccurs='0' />
      <xs:element ref='wst:Encryption' minOccurs='0' />
      <xs:element ref='wst:EncryptionAlgorithm' minOccurs='0' />
      <xs:element ref='wst:CanonicalizationAlgorithm' minOccurs='0' />
      <xs:element ref='wst:ProofEncryption' minOccurs='0' />
      <xs:element ref='wst:UseKey' minOccurs='0' />
      <xs:element ref='wst:SignWith' minOccurs='0' />
      <xs:element ref='wst:EncryptWith' minOccurs='0' />
      <xs:element ref='wst:DelegateTo' minOccurs='0' />
      <xs:element ref='wst:Forwardable' minOccurs='0' />
    </xs:documentation>
  </xs:annotation>
</xs:complexType>

```

```

    <xs:element ref='wst:Delegatable' minOccurs='0' />
    <xs:element ref='wsp:Policy' minOccurs='0' />
    <xs:element ref='wsp:PolicyReference' minOccurs='0' />
    <xs:any namespace='##other' processContents='lax' minOccurs='0' maxOccurs='unbounded'
  />
</xs:documentation>
</xs:annotation>
<xs:sequence>
  <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
</xs:sequence>
<xs:attribute name="Context" type="xs:anyURI" use="optional" />
<xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>

```

WSTEP extends <wst:RequestSecurityTokenType> with the following elements:

```

<xs:element ref="wsse:BinarySecurityToken" minOccurs="0"
maxOccurs="1" />
<xs:element ref="auth:AdditionalContext" minOccurs="0"
maxOccurs="1" />
<xs:element ref="wst:RequestKET" minOccurs="0" maxOccurs="1" />
<xs:element ref="wstep:RequestID" minOccurs="0" maxOccurs="1" />

```

Only the elements specified below are used in WSTEP. Any element received that is not specified below SHOULD be ignored.

**wst:TokenType:** Refers to the wst:TokenType definition in section [3.1.4.1.2.8](#).

**wst:RequestType:** Refers to the wst:RequestType definition in section [3.1.4.1.2.7](#). The wst:RequestType is used to identify the type of the security token request.

**wst:RequestKET:** Used when requesting a key exchange token as defined in [\[WSTrust1.3\]](#) section 8.4.

**wsse:BinarySecurityToken:** Provides the DER ASN.1 representation of the certificate request. The type of token is defined by the wst:TokenType element. For the X.509 enrollment extension the wst:TokenType MUST be specified as in section [3.1.4.1.2.8](#). The certificate request follows the formatting from [\[MS-WCCE\]](#) section 2.2.2.6. The EncodingType attribute of the wsse:BinarySecurityToken element MUST be set to base64Binary.

**auth:AdditionalContext:** The auth:AdditionalContext element is used to provide extra information in a wst:RequestSecurityToken message. It is an optional element, and SHOULD be omitted if there is no extra information to be passed.

**wstep:RequestID:** An instance of **wstep:RequestID** as specified in section [3.1.4.1.2.4](#).

WSTEP extends <wst:RequestSecurityTokenType> with an additional attribute:

```

<xs:attribute name="PreferredLanguage" type="xs:language"
use="optional" />

```

Only the attribute specified below is used in WSTEP. Any attribute received that is not specified below SHOULD be ignored.

**wstep:PreferredLanguage:** The **wstep:PreferredLanguage** attribute defines the preferred language to be used in a server response.

### 3.1.4.1.3.4 wst:RequestSecurityTokenResponseType

The wst:RequestSecurityTokenResponseType contains the elements that are part of a server response to a wst:RequestSecurityToken message. wst:RequestSecurityTokenResponseType is defined in the WS-Trust [\[WSTrust1.3\]](#) XML schema definition (XSD).

```
<xs:complexType name="RequestSecurityTokenResponseType">
  <xs:annotation>
    <xs:documentation>
      Actual content model is non-deterministic, hence wildcard. The following shows intended
      content model:
      <xs:element ref='wst:TokenType' minOccurs='0' />
      <xs:element ref='wst:RequestType' />
      <xs:element ref='wst:RequestedSecurityToken' minOccurs='0' />
      <xs:element ref='wsp:AppliesTo' minOccurs='0' />
      <xs:element ref='wst:RequestedAttachedReference' minOccurs='0' />
      <xs:element ref='wst:RequestedUnattachedReference' minOccurs='0' />
      <xs:element ref='wst:RequestedProofToken' minOccurs='0' />
      <xs:element ref='wst:Entropy' minOccurs='0' />
      <xs:element ref='wst:Lifetime' minOccurs='0' />
      <xs:element ref='wst:Status' minOccurs='0' />
      <xs:element ref='wst:AllowPostdating' minOccurs='0' />
      <xs:element ref='wst:Renewing' minOccurs='0' />
      <xs:element ref='wst:OnBehalfOf' minOccurs='0' />
      <xs:element ref='wst:Issuer' minOccurs='0' />
      <xs:element ref='wst:AuthenticationType' minOccurs='0' />
      <xs:element ref='wst:Authenticator' minOccurs='0' />
      <xs:element ref='wst:KeyType' minOccurs='0' />
      <xs:element ref='wst:KeySize' minOccurs='0' />
      <xs:element ref='wst:SignatureAlgorithm' minOccurs='0' />
      <xs:element ref='wst:Encryption' minOccurs='0' />
      <xs:element ref='wst:EncryptionAlgorithm' minOccurs='0' />
      <xs:element ref='wst:CanonicalizationAlgorithm' minOccurs='0' />
      <xs:element ref='wst:ProofEncryption' minOccurs='0' />
      <xs:element ref='wst:UseKey' minOccurs='0' />
      <xs:element ref='wst:SignWith' minOccurs='0' />
      <xs:element ref='wst:EncryptWith' minOccurs='0' />
      <xs:element ref='wst:DelegateTo' minOccurs='0' />
      <xs:element ref='wst:Forwardable' minOccurs='0' />
      <xs:element ref='wst:Delegatable' minOccurs='0' />
      <xs:element ref='wsp:Policy' minOccurs='0' />
      <xs:element ref='wsp:PolicyReference' minOccurs='0' />
      <xs:any namespace='##other' processContents='lax' minOccurs='0' maxOccurs='unbounded'
    />
  </xs:documentation>
</xs:annotation>
  <xs:sequence>
    <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="Context" type="xs:anyURI" use="optional" />
  <xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
```

WSTEP extends the wst:RequestSecurityTokenResponseType with the following elements:

```

    <xs:element ref="wstep:DispositionMessage" />
    <xs:element ref="wsse:BinarySecurityToken" minOccurs="0" maxOccurs="1" />
    <xs:element ref="wstep:RequestID" minOccurs="0" maxOccurs="1" />
    <xs:element ref="wst:KeyExchangeToken" minOccurs="0" maxOccurs="1" />
  />

```

Only the elements documented as follows are used by WSTEP. Any element received that is not documented as follows SHOULD be ignored.

**wst:TokenType:** Refers to the TokenType definition in section [3.1.4.1.2.8](#).

**wstep:DispositionMessage:** Refers to the definition in section [3.1.4.1.2.2](#). The wstep:DispositionMessage element is used to convey any additional server disposition information as part of the response message.

**wsse:BinarySecurityToken:** Refers to the wsse:BinarySecurityToken definition in section [3.1.4.1.3.2](#).

**wst:KeyExchangeToken:** Refers to the wst:KeyExchangeToken definition in section [3.1.4.1.2.3](#).

**wst:RequestedSecurityToken:** An instance of a wst:RequestedSecurityTokenType object as defined in section [3.1.4.1.3.2](#).

**wstep:RequestID:** An instance of a **wstep:RequestID** as defined in section [3.1.4.1.2.4](#) that conveys the request identifier of the originating request.

### 3.1.4.1.3.5 wst:RequestSecurityTokenResponseCollectionType

The <wst:RequestSecurityTokenResponseCollectionType> is defined in the [\[WSTrust1.3\]](#) XML schema definition (XSD) as a collection of one or more <wst:RequestSecurityTokenResponse> elements. The WS-Trust X.509v3 Token Enrollment Extensions further constrain the [\[WSTrust1.3\]](#) definition and the <wst:RequestSecurityTokenResponseCollection> collection MUST contain at most one <wst:RequestSecurityTokenResponse> element.

```

<xs:complexType name="RequestSecurityTokenResponseCollectionType">
  <xs:annotation>
    <xs:documentation>
      The <wst:RequestSecurityTokenResponseCollection> element (RSTRC) MUST be used to return a
      security token or response to a security token request on the final
      response.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element ref="wst:RequestSecurityTokenResponse" minOccurs="1" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:anyAttribute namespace="##other" processContents="lax" />
  </xs:complexType>

```

**wst:RequestSecurityTokenResponse:** An instance of a [wst:RequestSecurityTokenResponseType](#) object. The <wst:RequestSecurityTokenResponseCollectionType> MUST contain only one <RequestSecurityTokenResponse> element.

### 3.1.4.1.3.6 wst:RequestTypeEnum

The <wst:RequestTypeEnum> is defined in WS-Trust [\[WSTrust1.3\]](#) XML schema definition (XSD). WSTEP defines the following values for <wst:RequestTypeEnum>.

```
"http://schemas.microsoft.com/windows/pki/2009/01/enrollment/QueryTokenStatus"
```

WSTEP makes use of the Key Exchange Token request type defined in [\[WSTrust1.3\]](#) section 10:

```
"http://docs.oasis-open.org/ws-sx/ws-trust/200512/KET"
```

and the issue request type defined in [\[WSTrust1.3\]](#) XML Schema Definition (XSD)] :

```
"http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue"
```

### 3.1.4.1.3.7 wstep:CertificateEnrollmentWSDetailType

The <wstep:CertificateEnrollmentWSDetailType> contains additional information pertaining to error conditions.

```
<xs:complexType name="CertificateEnrollmentWSDetailType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="BinaryResponse" nillable="true"
type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="ErrorCode" nillable="true" type="xs:int"
/>
    <xs:element minOccurs="0" maxOccurs="1" name="InvalidRequest" nillable="true"
type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="RequestID" type="xs:string"
nillable="true" />
  </xs:sequence>
</xs:complexType>
```

**wstep:BinaryResponse:** The wstep:BinaryResponse element is used to provide a response if the Issuer generates one. If there is no response to provide, the wstep:BinaryResponse element MUST be nil.

**wstep:ErrorCode:** An integer value representing a server error. If there is no error to provide, wstep:ErrorCode MUST be specified as nil.

**wstep:InvalidRequest:** If the request is denied by the Issuer the server MUST return true. For other errors the wstep:InvalidRequest SHOULD be false.

**wstep:RequestID:** If the Issuer provides a wstep:RequestID to the server, it MUST be provided to a client. If no wstep:RequestID is provided by the Issuer, the wstep:RequestID element should be specified as nil.

### 3.1.4.1.4 Attributes

There are no attributes that are specific to this operation.

### 3.1.4.2 Processing Rules

An incoming **SOAP message** MUST be processed to evaluate the **SOAP actions** and authentication information.

If the user is authenticated successfully using the provided authentication information, message processing MUST continue, and the authentication information SHOULD be provided to the Issuer. If the authentication fails, the server MUST respond with a SOAP fault.

If the SOAP action is "http://schemas.microsoft.com/windows/pki/2009/01/enrollment/RST/wstep" the server must follow the Request Security Token Processing Rules per section [3.1.4.2.1](#).

If the SOAP action is "http://docs.oasis-open.org/ws-sx/ws-trust/200512/RST/KET" the server must follow the Key Exchange Token Processing Rules per section [3.1.4.2.2](#).

If any other SOAP action is defined, the server SHOULD respond with a SOAP fault.

#### 3.1.4.2.1 WSTEP Action: Request Security Token Processing Rules

A <wst:RequestSecurityTokenMsg> MUST contain a <wst:RequestType> element as defined in section [3.1.4.1.2.7](#). If the <wst:RequestType> element is absent, nil, or undefined, the server MUST respond with a SOAP fault.

If a **wstep:PreferredLanguage** attribute is not present in a <RequestSecurityTokenType> object, or the value is not in SupportedLanguages, the server SHOULD use DefaultLanguage.

If the <wst:RequestType> is "http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue", the server MUST process the request per section [3.1.4.2.1.1](#).

If the <wst:RequestType> is "http://schemas.microsoft.com/windows/pki/2009/01/enrollment/QueryTokenStatus" the server MUST process the request per section [3.1.4.2.1.2](#).

If the <wst:RequestType> is any other value, the server MUST respond with a SOAP fault.

##### 3.1.4.2.1.1 New and Renewal Request Processing

A wst:RequestSecurityToken message with a wst:RequestType value of "http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue" is used for the purposes of issuing an X.509v3 certificate or for renewal of an existing X.509v3 certificate.

For this type of message, a server has additional syntax constraints on the request message.

**wsse:BinarySecurityToken:** If the wsse:BinarySecurityToken element is absent or undefined, the server MUST respond with a SOAP fault.

**wstep:RequestID:** If the **wstep:RequestID** element is present and defined, the server SHOULD ignore it.

The server MUST provide the **wsse:BinarySecurityToken** to the Issuer and SHOULD provide the **auth:AdditionalContext** (see section [3.1.4.1.3.3](#)) to the Issuer.

If the Issuer responds with an error, the server MUST respond with a SOAP fault. If the Issuer indicates the issuance is pending, the server MUST use the Issuer response to generate a pending **wst:RequestSecurityTokenResponseCollectionMsg** message. If the Issuer responds with an issued certificate, the server MUST respond with a **wst:RequestSecurityTokenResponseCollectionMsg** message providing the issued certificate.



### 3.1.4.2.1.2 QueryTokenStatus Request Processing

A **wst:RequestSecurityToken** message with a <wst:RequestType> of "http://schemas.microsoft.com/windows/pki/2009/01/enrollment/QueryTokenStatus" is used to retrieve an issued certificate or check the status of a certificate request that was pending.

For this type of message, the server has additional syntax constraints on the request message.

The **wstep:RequestID** element is a null-terminated **Unicode** string that contains a certificate request identifier (as defined in section [3.1.4.1.2.4](#)). If the <wstep:RequestID> element is absent, defined as nil, or contains no value the server MUST return a SOAP fault.

The server MUST provide the **wstep:RequestID** to the Issuer.

If the Issuer responds with an error, the server MUST respond with a SOAP fault. If the Issuer indicates the issuance is pending, the server MUST use the Issuer response to generate a pending **wst:RequestSecurityTokenResponseCollectionMsg** message. If the Issuer responds with an issued certificate, the server MUST respond with a **wst:RequestSecurityTokenResponseCollectionMsg** message providing the issued certificate.

### 3.1.4.2.2 KET Action: Request Security Token Processing Rules

A **wst:RequestSecurityTokenMsg** MUST contain a <wst:RequestType> element as defined in section [3.1.4.1.2.7](#). If the <wst:RequestType> element is absent, nil, or undefined, the server MUST respond with a SOAP fault.

If the <wst:RequestType> is "http://docs.oasis-open.org/ws-sx/ws-trust/200512/KET" the server MUST process the request per section [3.1.4.2.2.1](#).

If the <wst:RequestType> is any other value, the server MUST respond with a SOAP fault.

#### 3.1.4.2.2.1 Key Exchange Token Request Processing

A RequestSecurityToken message of wst:RequestType of "http://docs.oasis-open.org/ws-sx/ws-trust/200512/KET" is used to retrieve the Key Exchange Token.

For this type of message, a server has additional syntax constraints on the **wst:RequestSecurityTokenMsg** message.

If the <wst:RequestKET> element is absent, the server MUST return a SOAP fault.

The server requests the Key Exchange Token from the issuer. If the issuer responds with an error, the server MUST respond with a SOAP fault. Otherwise, the server uses the Issuer response to generate a **wst:RequestSecurityTokenResponseCollectionMsg** message.

The <wst:RequestSecurityTokenResponse> element in the server response follows the [\[WSTrust1.3\]](#) definition in section 8, but for key exchange in the WSTEP protocol, the <wst:KeyExchangeToken> element MUST be present, and provides the key exchange token provided from the Issuer.

### 3.1.5 Timer Events

None.

### 3.1.6 Other Local Events

None.

## 4 Protocol Examples

### 4.1 RequestSecurityToken Request/Response Message Sequence

In the following message sequence, the username/password authentication headers have been included in the message sequences for clarity.

#### 4.1.1 Standard Certificate Request

##### 4.1.1.1 RequestSecurityToken Message (Issue Request)

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1">
http://schemas.microsoft.com/windows/pki/2009/01/enrollment/RST/wstep</a:Action>
    <a:MessageID>urn:uuid:b5d1a601-5091-4a7d-b34b-5204c18b5919</a:MessageID>
    <a:ReplyTo>
    <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <RequestSecurityToken xmlns="http://docs.oasis-open.org/ws-sx/ws-trust/200512">
    <TokenType>http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-
1.0#X509v3</TokenType>
    <RequestType>http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue</RequestType>
    <BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd#base64binary"
ValueType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd#PKCS7"
xmlns="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd">MIIEDCCAvQCAQAwADCCASiDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANpk
/1AOEvYikbMjvabzApKyJkqLnaXWm2FvnO6UNctXWf9WchbbumLqkIas9BUcMiSE
Eh4tVZNfugi3bahnjUjTG9MIvAZd3/C0YfuLX8y19mcIVWZhyYZVwUeMh4GYS5ht
90NFZP0vXb7c0brSRyvhwZq+kG7om24qMTZBgSIRsajcDVY+uGLdhixy4AtXNw5
pzzRdS/1QBF1wsDT3C0bceWy2uej2hsLYoLyGdd0fHkly/tOusoyjc3itw2o3P9j
k+bP4eDG2ukrjMMcjxQ500Bze7hXQf2hrNEJRTd6pPIOdAub8Hz/DiPYaEY75XN
EQepc11nLmq2GQ9YghcCAWEAAcCAcUwGgYKKwYBBAGCNw0CAzEMFgo2LjEuNzA1
My4yMGQCSsGAQQBgjcVDFXMFUCAUMLzktMTM1MUMwNDA1QS5kOS0xMzUxQzA0
MDZBLm50dGVzdC5taWVyb3NvZnQuY29tDBJEOS0xMzUxQzA0MDZBXXGFiYnkMC0N1
c1Rlc3QuZlMhMHQGCisGAQQBgjcNAgIxZjBkAgEBHlwATQBpAGMAcGvBvAHMAbWBM
AHQAIAABFAG4AaABhAG4AYwBlAGQAIABDAlAeQBWAHQABWBNhAHIAyQBWAAGAAQBJ
ACAAUABYAG8AdgBpAGQAZQByACAAdgAxAc4AMAMBADCBYgYJKoZIhvcNAQkOMYG8
MIG5MBCGCSsGAQQBgjcUAQKHHggAVQBzAGUAcjApBgNVHSUEIjAgBgorBgEEAYI3
CgMEBggrBgEFBQcDBAYIKwYBBQUHAWIwDgYDVR0PAQH/BAQDAgWgMEQGCSqGSIB3
DQEJJDwQ3MDUwDgYIKoZIhvcNAwICAgCAMA4GCCqGSIB3DQMEAgIAgDAHBGUrDgMC
BzAKBggqhkiG9w0DBzAdBgNVHQ4EFgQUavblZB2QWQ6vt+ag4T4jZMPFe3owDQYJ
KoZIhvcNAQEFBQADggEBAGId8Dv9gVNVNgnSHkNuTiErtwIacv609MnMt2WxhnAj
zGQZS4bZ9JNH+CR49yswieFCS3zFiP5PxGL5CCogn2XHGS7LCCzHtrltAZBACTC
tzLF5Qcj0Ki/H5GRa4Q+ZelUrcM1cSnD52zY+V1vFXX1Xc2P5hTB0bq8GbZME/MW
84XE1sz75NqZeQ2vh066ozAMyWmtC26Q+7DofBaPMxXrWgMQBm6qO/Yjj3vDY/U8
T9rpJqGHHTG7E7E+/3EcGPeKNExxf0n+VXRwL09C5wOS6Xy/JNGfuiPW+SzaRbPs
H5/6UiS+uqtSVzaJmA0a9vzxJQfgARCucr49wM3YUek=</BinarySecurityToken>
    <RequestID xsi:nil="true"
xmlns="http://schemas.microsoft.com/windows/pki/2009/01/enrollment" />
  </RequestSecurityToken>
</s:Body>
</s:Envelope>
```

#### 4.1.1.2 Server RequestSecurityToken Response

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:Action s:mustUnderstand="1">
http://schemas.microsoft.com/windows/pki/2009/01/enrollment/RSTRC/wstep</a:Action>
    <ActivityId CorrelationId="a0f231a3-ccf2-4b9c-99a6-bc353a59b5d0"
xmlns="http://schemas.microsoft.com/2004/09/ServiceModel/Diagnostics">
95427c83-902c-48db-9529-f61cc1d8c035</ActivityId>
    <a:RelatesTo>urn:uuid:b5d1a601-5091-4a7d-b34b-5204c18b5919</a:RelatesTo>
  </s:Header>
  <s:Body>
    <RequestSecurityTokenResponseCollection
xmlns="http://docs.oasis-open.org/ws-sx/ws-trust/200512">
    <RequestSecurityTokenResponse>
    <TokenType>http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-x509-token-profile-1.0#X509v3</TokenType>
    <DispositionMessage xml:lang="en-US"
xmlns="http://schemas.microsoft.com/windows/pki/2009/01/enrollment">
Issued</DispositionMessage>
    <BinarySecurityToken
ValueType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd#PKCS7"
EncodingType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd#base64binary"
xmlns="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd">MIIRLAYJKoZIHvcNAQcCoIIRhTCCEYECAQMxCzAJBgUrDgMCGGUAMH0GCCsGAQUF
BwwDoHEEbzbTMCgwIQIBAQYIKwYBBQUHBBwExEjAQAQEAMAMCAQEMBK1zc3VlZDZBC
AgECBgorBgEeAYI3CgoBMTEwLwIBADADAgEBMSUwIwYJKwYBBAGCNxURMRYYEFFis
145+YbEal zssa0G63KkQD6+OMAAwAKCCD0EwggNbMI ICQ6ADAgECAhAeqF9153Dz
n0o0G27H8w6RMAOGCSqGSIB3DQEBBQUAMQxGzAZBgNVBASteKlpY3Jvc29mdCBQ
S0kgVGVhbTEVMBMGA1UEAwMRkJfFRW50Um9vdENBMB4XDTA5MDMwMzAzMjQxMl oX
DTE0MDMwMzAzMzQxMjFoNDEBMBkGA1UECXMSTWljcm9zb2Z0IFBLSSBUZWFtMRUw
EwYDVQDDAxGQ19FbnRSb290Q0EwggEiMAOGCSqGSIB3DQEBBAQUAA4IBDwAwggEK
AoIBAQCInE154od1KuJpZ8BoaqVIsuE4BX9dXTsk0BmBVblPlYzI1RwMONE1Zr40
TdgZ/Nv69kwCozi0D0Eo58fHYz3FAh6rww4o+ABpx9nFJlJj69D9H7JIQWswdTOe
nQxwW59vzotQfMz00T//lNCilX3aBMj6VjArX51fYLCqBr2Qgw9BmEkaivntw9Vd
1gvJTPNoyG79c2V2Mux+4M9dzIR17xw8Mx4LhJrXXKQPZ1YgwVeWdAXelS5aaoXG
LI2GIx15LtsUqzYxcelsVotVcfr4NM31Xkis5x679DtxMoB2gyqjUhkB1hTLIQwK
8V5v5j jjsuy7tXP5qIEpOq7B6NCzAgMBAAGjaTBnMBMGCsGAQQBggj cUAgQGHgQA
QwBBMA4GA1UdDwEB/wQEAWIBhjAPBgNVHRMBAf8EBTADAQH/MB0GA1UdDgQWBBS9
oNbJuWz92VLUQS1J1Dzamt3dVTAQBgkrBgEeAYI3FQEAAwIBADANBgkqhkiG9w0B
AQUFAAOCAQEAvr8MMHHzHcnUUyKGFnBE8qNPKIHI9oDEee3jnChqO9wmKbEZV4701
+ejdiDj iC9FQlHHbuWxhKpJOnAtqXN48E9XLPzS/ezx/LwsEv5LlroiORBym8NbA
1dLJNFqskrC0FAhefg9Jc4c91Q3uyGUjMb4Hoa9b2cqNIeMerzV+L1oH0wVZpg9o
i8OoC1FX/woETKBryiLnXPLybdQu0E7brTkyYmXJsGuFPgLzj6DVFOdb1ZMmEJNy
6Opr98dFJYwcnhjdVx0FtRTsXnU8epeAYOEHWJCU01bWpcRPF6C6sJY0wmRaP7
iOCGXhoFO61cbL08fztvGpUkyZfDoHg3DCCBUwggQ0oAMCAQICcmEMfNwAAAA
AAIwDQYJKoZIhvcNAQEFBQAwNDEbMBkGA1UECXMSTWljcm9zb2Z0IFBLSSBUZWFt
MRUwEwYDVQDDAxGQ19FbnRSb290Q0EwHhcNMDkwMzAzMDMwMzAzMDMwMzAzMDMwMzAz
MDMzNjE2WjAzMRswGQYDVQQLExJNaWNYb3NvZnZnQGUETJFRlYw0xZDASBgNVBAMM
C0ZCX0VudFN1YkNBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKAAQEAzqt1
8VMe3t1durXs81ORjWBWoxDQtTPJAlYNQdqvs+H2HutrUNjvw/+vK0Am0ib8GR3u
D8IT+Kk8TJvzSGQuQAKxYtzaqjDt7Alc7UtsnelSiKDT5ZSf1pmfUvASKd28jJ4Y
B1SDJSiJmOyWqUZCnwwAwW0VXCrMk1QnyGjr3Akq+p6Mgo/ZqaeFuj4o7jJjI/em
```

BN5mM01x0y0jQF79bQbJagd0bidQkfSASDW3HAycTcWuKfMfjF5vPMZqLWcaRjAS  
v2tEn9urPDJv4mhXrTz53ETup1kaOmA95BQULnLXXHreXImUkGacdPhd1hCWm9KE  
v1Q1vsks9UUBfHUcKwIDAQAB04ICXzCCA1swEAYJKwYBBAGCNxUBBAMCAQAwHQYD  
VR00BBYEFJ+3jZGC0QUd0DHiPfaXeoF15VzIMBkGCSsGAQQBgjcUAgQMhGoAUwB1  
AGIAQwBBMA4GA1UdDwEB/wQEAWIBhjAPBgNVHRMBAf8EBTADAQH/MB8GA1UdIwQY  
MBAaFL2glsm5bP3a8u5B1gmUPNqaDd1VMIHsBgNVHR8EgeQwgeEwgD6ggduggdiG  
gdVsZGFwOi8vL0N0PUZCX0VudFJvb3RDQsxDtj05LTEzNTFDMDQwNkEsQ049Q0RQ  
LENOPVB1YmXpYyUyMETleSUyMfN1cnZpY2VzLENOPVN1cnZpY2VzLENOPUNvbmZp  
Z3VyYXRpb24sREM9ZDktMTM1MUMwNDA2QSxEQz1udHRlc3QsREM9bWljcm9zb2Z0  
LERDPWNvbT9jZXJ0aWZpY2F0ZVJldm9jYXRpb25MaXN0P2Jhc2U/b2JqZWN0Q2xh  
c3M9Y1JMRGlzdHJpYnV0aW9uUG9pbmQwgdGCSsGAQUFBwEBBIHOMIHLMIHIBggr  
BgEFBQcwAoBu2xkYXA6Ly8vQ049RkRFRW50Um9vdenBLENOPUFJQSxDTj1QdWJs  
aWMLMjBLZXk1MjBTZXJ2aWN1cyxDTj1TZXJ2aWN1cyxDTj1Db25maWd1cmF0aW9u  
LERDPWQ5LTEzNTFDMDQwNkEsREM9bnR0ZXN0LERDPW1pY3Jvc29mdCxEQz1jb20/  
Y0FDZXJ0aWZpY2F0ZT9iYXN1P29iamVjdENsYXNzPWN1cnRpZmljYXRpb25BdXRo  
b3JpdHkwDQYJKoZIhvcNAQEFBQAAdggEBACUFJf5b34y2bob4+rmjcJ2F4MRRg8C3  
v91tkpai68nerYc2tNU0C+hav5QZ2DjW8KwNs+rRblUW/5iRa/7uKhNHYs14Yv0  
I/LddTlv5F5uv5CK0YbmQIGx37aLLHjDngerh346N+z75kzJ18eEVnUPZUQ/ZB  
btqb7GAKcIRViJJ+9rspGRJoIFjFIAUI7jThssG1zobaLjyW4IvMX0+VpGpr8zxM  
ZfK0P4kr3Ou+TsvK1H0cBSFy4SgkcsdxVFkoyCEm7Pr4osxVhCfKc9JgihYldi1  
xgt4U54nFagUrTbHjB+JYjHLqQYafPGCYfB9bR4M1/1jhV05F1V61zIwggAOMIIF  
dqADAgECAgoY2dTAQAAAAA9MA0GCSqGSIb3DQEBAQUAAQIBAQwYBBAQCCgMGA1Ud  
Y3Jvc29mdCBQSOkgVGVhbTEUMBIGA1UEAwLRkRFRW50U3ViQ0EwHhcNMDkMzA1  
MTgyNTQ1WhcNMTAwMzA1MTgyNTQ1WjCBvjETMBEGCgmsJomT8ixkARKWA2NvbTEZ  
MbcGcmSjomT8ixkARKWCW1pY3Jvc29mdDEWMBQGCgmsJomT8ixkARKWBm50dGVz  
dDEdMBSGcmSjomT8ixkARKWdWQ5LTEzNTFDMDQwNkEsREM9bnR0ZXN0LERDPW1pY3Jv  
c29mdCwYDQDEwRhmYmJ5MTYwNAYJKoZIhvcNAQkBFidHmJ5QE05LTEzNTFDMDQw  
NkEuTlRURVNULk1JQ1JPU09GVC5DT00wgGEMa0GCSqGSIb3DQEBAQUAA4IBDwAw  
gGKAoIBAQDT5P5QDhL2IpgZCb2m82qSsiZKi521lpthb5zulDQrVln/VnIW27pi  
6pCGPQVHD1khhBeLVWTRboIt22oZ41I0xvTCLwGXd/wtGh7i1/MpfZnCFVYcmG  
VcFHjIeBmEuYbFdDRWT9L12+3NG60kcr4b1s6vpBu6JtuKjE2QYeiEbGo3A1WPrh  
i3YyscuALVzcOac8X0Uv9UARdcLA09wtG3H1strho9obC2KJchnXdh5XNcv7TrrK  
Mo3N4rcNqNz/Y5Pmz+HgxtrpEYzDHI6sUOdDgc3u4V0H9oazRCU3eqTyDnQLm/B  
8/w4j2GhDwEB/wQEAWIFoDbrBgNVHREZDBioDcGcisGAQQBgjcUAgOgKQwnYwJi  
eUBkOS0xmZuXqZA0MDZBLm50dGVzdC5taWNyb3NvZnQuY29tZG9mYmJ5QE05LTEz  
NTFDMDQwNkEuTlRURVNULk1JQ1JPU09GVC5DT00wgesGA1UdHwSB4zCB4DCB3aCB  
2qCB14aB1GxkYXA6Ly8vQ049RkRFRW50U3ViQ0EwS049OS0xmZuXqZA0MDdBLENO  
PUNEUCxDTj1QdWJsawMLMjBLZXk1MjBTZXJ2aWN1cyxDTj1TZXJ2aWN1cyxDTj1D  
b25maWd1cmF0aW9uLERDPWQ5LTEzNTFDMDQwNkEsREM9bnR0ZXN0LERDPW1pY3Jv  
c29mdCxEQz1jb20/Y0FDZXJ0aWZpY2F0ZT9iYXN1P29iamVjdENsYXNzPWN1cnRp  
ZmljYXRpb25BdXRob3JpdHkwHQYDVR0OBByEFgr25WQdkFhur7fmoOE+I2TDxXt6  
MA4GA1UdDwEB/wQEAWIFoDbrBgNVHREZDBioDcGcisGAQQBgjcUAgOgKQwnYwJi  
eUBkOS0xmZuXqZA0MDZBLm50dGVzdC5taWNyb3NvZnQuY29tZG9mYmJ5QE05LTEz  
NTFDMDQwNkEuTlRURVNULk1JQ1JPU09GVC5DT00wgesGA1UdHwSB4zCB4DCB3aCB  
2qCB14aB1GxkYXA6Ly8vQ049RkRFRW50U3ViQ0EwS049OS0xmZuXqZA0MDdBLENO  
PUNEUCxDTj1QdWJsawMLMjBLZXk1MjBTZXJ2aWN1cyxDTj1TZXJ2aWN1cyxDTj1D  
b25maWd1cmF0aW9uLERDPWQ5LTEzNTFDMDQwNkEsREM9bnR0ZXN0LERDPW1pY3Jv  
c29mdCxEQz1jb20/Y2VydG1maWNhdGVsZXZvY2F0aW9uTG1zdD9iYXN1P29iamVj  
dENsYXNzPWNSTERpc3RyaWJldG1vb1BvaW50MB8GA1UdIwQYMBaAFJ+3jZGC0QUd  
0DHiPfaXeoF15VzIMCkGA1UdJQIiMCAGCisGAQQBgjcKAwGCCsGAQUFBwEBBggr  
BgEFBQcwDAjANBgkqhkiG9w0BAQUFAAOCAQEAAwLndycUwoaIU8itsOhThdSiImen  
Izf4Gz4fOicwKuc10RaHbeKDBp5e91x1Z6My+4fbGn2o58vB05wRMfYa31HIngO/  
+C2W+9p0xsPGXSFR8qwfYcPou6aZC40G21TSkclpZvuCqcOED2CPaHocW9f2OF  
Z8OQyi+zFTNFY04B5219isyCoOwivF4EsbamTLlBqTRdPLBJ2Vz+BB/nwTemZz1  
vd7ZBw89C9qebI6QVb8udRi21x50LqQZUtJyiWpv/3zhpR6eL9DobsxIrhc4Np7r  
ft2hfopnHulZLhwFRk2GXGRZqBabDN06kCe1qV6X2gw1v6NbDRXc/OFFOTGCAakw  
ggG1AgEBMEIwNDEbMBkGA1UECXMSTW1jcm9zb2Z0IFBLSzBUZWFtMRUwEwYDVQQD  
DAxGQ19FbnRsb290Q0ECCmEmfNwAAAAAAAIwCQYFKw4DAhoFAKA+MBcGCSqGSIb3  
DQEJAzEKBggrBgEFBQcMMAzAjBgkqhkiG9w0BCCQXxFgQUY/W4vpjAGswflv6yOirp  
M+kawEwDQYJKoZIhvcNAQEBBQAEGgEAkQr0469R/EGRQaf4jdV1z/NGG+7WPnv8  
SKS0h3U5ZwKUPHjCtLkfa7oJDgCMAA026/VVN3nAwCiFtA3ZCc+FA5d+5rmUtUq  
k25eeCTaBH/6NyKvOddsgyKrt1Gzvj+uavqQ0uNhQeZyWBjNgw10Q09ENXcw4qe  
I05oofT8Td5Z2wACKunoU/xrvrWoPoS58TtkSDGkd/BWq8R6ZzaFdfjAF26bGB7Hi f5  
OMzCxn5Sd/AILOzG3F99/wnqijCxnsFPP6EYrVGs8gjDKfduaIb+cyw4X+wFZch2  
DMryOkIN+6YDN60SNPgnHCMq+1P95IjX2aCk3EAYx3NWtf/BOu4KcW==</BinarySecurityToken>  
<RequestedSecurityToken>

```

    <BinarySecurityToken ValueType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-x509-token-profile-1.0#X509v3"
EncodingType="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd#base64binary"
xmlns="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd">MIIGjjCCBxagAwIBAgIKGNnV0wAAAAAPTANBgkqhkiG9w0BAQUFADAZMRswGQYD
VQQLExJNaWYyb3NvZnQvUETJIFRlYW0xZDASBgNVBAMMC0ZCX0VudFNlYkNBMB4X
DTA5MDMwNTE4MjU0NVVoXDTEwMDMwNTE4MjU0NVVwgb4xEzARBgoJkiaJk/IsZAEZ
FgNjb20xGTAXBgJkiaJk/IsZAEZFGltawNyb3NvZnQvFjAUBgoJkiaJk/IsZAEZ
FgZudHRlc3QxHTAbBgoJkiaJk/IsZAEZFGlkOS0xMzUxQzA0MDZBMQ4wDAYDVQQD
EwVvc2Vyc2ENMAsGA1UEAxMEYwJieTE2MDQGCSqGSIB3DQEJARyYnYwJieUBEOS0x
MzUxQzA0MDZBLk5UVEVTVc5NSUNST1NPRlQuQ09NMIIBIjANBgkqhkiG9w0BAQEF
AAOCAQ8AMIIBCgKCAQEAO+T+UA4S9iKRswm9pvnqkrImSoudpdabYW+c7pQOKldZ
/1ZyFtu6YUqQhgz0FRwyJIQSHi1Vkw06CLdtqGeNSNmB0wi8B13f8LRh+4tfzKX2
ZwhVzmHJh1XBR4yHgZhLmG33Q0Vk/S9dvtzRutJHK+G9bOr6QubibbioxNkGBIhG
xqNwNVj64Yt2GLHLGClc3DmnPNF1L/VAEXXCwNpcLRtx5bLa56PaGwtiixIz13R8
eTXL+066yJKNzeK3Dajc/2OT5s/h4Mba6RGMwxyOrFDnQ4HN7uFdB/aGs0QLFN3q
k8g50C5vvpfP8OI9hoRjv1c0RB61zXWcuarYZD1iCFwIDAQABo4IDFjCCAxIwRAYJ
KoZIHvcNAQkPBDcwNTAObggqhkig9w0DAGICAIwDgYIKoZIhvcNAwQCAGCAMAcG
BSs0AwIHMMAoGCCqGSIb3DQMHEMCCGCCSsGAQQBggjUAgQKHHgAVQBzAGUAcjCB2gYI
KwYBQUHAQEgEgc0wgcowgccGCCSsGAQUFBzAchoG6bGRhcDovLy9DTj1GQl9FbnRT
dWJdQsxDtj1DRFAsQ049UHvibG1jJTIwS2V5JT1wU2Vydm1jZXMsQ049U2Vydm1j
ZXMsQ049Q29uZmlndXJhdGlvbixEQz1kOS0xMzUxQzA0MDZBLERDPW50dGVzdCxE
Qz1taWYyb3NvZnQsREM9Y29tP2NBQ2VydGlmawNhdGU/YmFzZT9vYmplY3RDdbGFz
cz1jZXJ0aWZpY2F0aW9uQXV0aG9yaXR5X29uZm9Ga1UdDgQWBBRq9uVkhZBYbq+35qDh
PiNkw8V7eJA0BgNVHQ8BAf8EBAMCBAAwawYDVR0RBGQwYqA3BgorBgEEAYI3FAID
oCkMJ2FiYn1AZDktMTM1MUMwNDA2QS5udHRlc3QubW1jcm9zb2Z0LmNvbYEnYWJie
UBEOS0xMzUxQzA0MDZBLk5UVEVTVc5NSUNST1NPRlQuQ09NMIHRBgNVHR8EgeMw
geAwgd2ggdgggdeGgdRsZGFwOi8vLONOPUZCX0VudFNlYkNBLENOPtktMTM1MUMw
NDA3QSxDtj1DRFAsQ049UHvibG1jJTIwS2V5JT1wU2Vydm1jZXMsQ049U2Vydm1j
ZXMsQ049Q29uZmlndXJhdGlvbixEQz1kOS0xMzUxQzA0MDZBLERDPW50dGVzdCxE
Qz1taWYyb3NvZnQsREM9Y29tP2N1cnRpZmljYXRlUmV2b2NhdGlvbkxpc3Q/YmFz
ZT9vYmplY3RDdbGFzcz1jUkxEaXN0cmliX25Qb21udDAfBgNVHSMEGDAWgBSf
t42RgtEFHdAx4j32l3qBdeVcyDAPBgNVHSUEIjAgBggrBgEEAYI3CgMEBggrBgEF
BQCDBAIIKwYBBQUHAWIwDQYJKoZIhvcNAQEFBQADggEBAMC53cnFMKGmiFPiRbDo
U4XUoiJnpyM3+Bs+HziHMCrnJdEWh23igwaeXvZcdWejMvuH2xp9qOfLwdOceTH2
Gt9RyJ4Dv/gt1vvaTsbDx10n0fKsH4mAqS6LummQuNBttU0pHJaWb7gqnDhA9gj2
h6HFvX9jhWfDkMovsxbTRWNOAeUx9ovbMgqDlorxeBLG2pkyy26k0XTywSdlc/gQ
f58E3pmc9b3e2QcPPQvanmyOkFW/LnUYtpceTi6kGVLScolqb/984aUeni/Q6G7M
SK4XODae637doX6KZx7pWS4cBUZNh1xkWagWmwzDOPAntalel9oMJB+jWw0V3Pzn
xTk=</BinarySecurityToken>
  </RequestedSecurityToken>
  <RequestID
xmlns="http://schemas.microsoft.com/windows/pki/2009/01/enrollment">61</RequestID>
</RequestSecurityTokenResponse>
</RequestSecurityTokenResponseCollection>
</s:Body>
</s:Envelope>

```

## 4.1.2 Key Exchange Token Request

### 4.1.2.1 Client Exchange Token Request

```

<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1">
http://docs.oasis-open.org/ws-sx/ws-trust/200512/RST/KET</a:Action>
    <a:MessageID>urn:uuid:c2884a79-b943-45c6-ac02-7256071de309</a:MessageID>

```







## 4.1.4 Message exchange with a server fault

### 4.1.4.1 Client Request

See section [4.1.1.1](#) for an example of a client request.

### 4.1.4.2 Server Fault Response

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:Action s:mustUnderstand="1">http://schemas.microsoft.com/net/2005/12/
windowscommunicationfoundation/dispatcher/fault</a:Action>
    <a:RelatesTo>urn:uuid:ce330bb2-0ca2-473b-a29a-19e9264666ff</a:RelatesTo>
    <ActivityId CorrelationId="4f0e4425-4883-41c1-b704-771135d18f84"
xmlns="http://schemas.microsoft.com/2004/09/ServiceModel/Diagnostics">
eda7e63d-0c42-455d-9c4f-47ab85803a50</ActivityId>
  </s:Header>
  <s:Body>
    <s:Fault>
      <s:Code>
        <s:Value>s:Receiver</s:Value>
        <s:Subcode>
          <s:Value xmlns:a="http://schemas.microsoft.com/net/2005/12/windowscommunicationfoundation/
dispatcher">a:InternalServiceFault</s:Value>
        </s:Subcode>
      </s:Code>
      <s:Reason>
        <s:Text xml:lang="en-US">The server was unable to process the request
due to an internal error. For more information about the error, either turn
on IncludeExceptionDetailInFaults (either from ServiceBehaviorAttribute or
from the <serviceDebug> configuration behavior) on the server in order to
send the exception information back to the client, or turn on tracing as per
the Microsoft .NET Framework 3.0 SDK documentation and inspect the server
trace logs.</s:Text>
      </s:Reason>
    </s:Fault>
  </s:Body>
</s:Envelope>
```

## 4.1.5 Certificate Renewal

### 4.1.5.1 Client Renewal Request

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1">http://schemas.microsoft.com/windows/pki/2009/
01/enrollment/RST/wstep</a:Action>
    <a:MessageID>urn:uuid:b0a9b388-2581-451d-8c03-270d4ffe2928</a:MessageID>
    <a:ReplyTo>
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
  </s:Header>
```



0QUd0DHiPfaXeoF15VzIMCkGA1UdJQQiMCAGCisGAQQBggjckAwQGCCsGAQUFBwME  
BggrBgEFBQcDAjANBggkqhkiG9w0BAQUFAAOCAQEAge5W3XJ3766qGf8Y+r2YulGm  
Q0cskpa7tqeZxwZW51skiv6i0S1NPo67HNp08+UfLyRpdghF+ZwtphnCdtPieAVn  
0Hmd1LvO+LxTGA16oVCixa3qRGlx2NUQLH7VzH34ugf/YEaJDPJ/9818TJhQQ7iP  
Ti3jlu9e52rYmNLb450egdmVJhbKJIftpcQqbWZY7HP3tzizf50dXhhzg3CGkOJ  
WqSQPQQRknUkVxKjvFIIovrvNM3IluGDjIu9EjjlgE1ZSqA4Lixjko2M/hzhWpej  
KnVAEV9eb+ppWh25jbievk6WrYmgnwPgS9gRBB150q01ahPi6Vj5EOfXoe99KTAN  
BgkqhkiG9w0BAQUFAAOCAQEAL89MjFAUL4Bi2e/8KvIXY6r8cLEsXVVU5NNv16r4  
wPdkWeuGNteclvyufhxr3jNAEVwaeckKQcIGKNz5cyd6fBE4LDPP+fiI4Ywb1bPU  
gJVozQpSVJ5j2+dSux5TiJQYYeXlqrQfG3ioz7m7WSOvoFlmBomGpFUmHZMMdsVSj  
xFvJ7DShQeOJomRKc5G1Xp7o5/W5Xi6bn4TORTIGd6vtSbvAta1DJ/jais6urlv8  
yqedfAJIE62BbbEWovIOa6wqdhCLLUXilPv3eZy5SsyZdDBwkbJsVfEhr0yivq6q  
0ErE0fcDm/WBqOWhifUgJwIUF9DTE7vykt7n043VZYQLTaCCBPiwggaOMIIFdqAD  
AgECAgoY2d8GAAAAAAMA0GCSqGSIB3DQEBBQUAMDMxGzAZBgNVBAsTEk1pY3Jv  
c29mdCBQSO0kgVGVhbTEUMBIGA1UEAwWLRkJfRW50U3ViQ0EwHhcNMDkwMzA1MTgy  
NjE3WhcnMTAwMzA1MTgyNjE3WjCBvjETMBEGCgmSjOMT8ixkARkWA2NvbTEZMBcG  
CgmSjOMT8ixkARkWCWlpY3Jvc29mdDEWMBQGCgmSjOMT8ixkARkWBm50dGVzdDEd  
MBsGCGmSjOMT8ixkARkWDQ5LTEzNTFDMDQwNkExDjAMBGNVBAAMTBVzZXJzMQ0w  
CwYDVQQDEwRhYmJ5MTYwNAJJKoZiHvcNAQkBFidhYmJ5QE5LTEzNTFDMDQwNkEu  
TlRURVNUk1JQ1JPU09GVC5DT00wggEiMA0GCSqGSIB3DQEBQUAA4IBDwAwggEK  
AoIBAQCUIUF1eRkRjXgChj0u0lmiL+Gq1uG85wgfSz2th+w0jM+BA+1KLe57dbCc+  
FqzpZqJruPfgSdSAfMP46Kk8roM/4kPEVSYIbidn3hRx2txSR7HrcsLo8/xhnx  
WY7m8WjpcFro2mBV/JbOnTT5KfU0Z+YSSCGzEVahJqN2Wj11z3VBZ8YcJ3BEUWY1  
UDYp33zDnPAMULKDPUPJlMXUmlX+pUL4vycfnm1on4iGw0kHHCqfM77LNPYJfkdZ  
SgeTFRd2qSPdfUeOurwoS8whYFvPe2LFT5BRcAoF4dIaRK5DYSCP8yv1xQ+6z/yq  
D+tZ9WpR0TC7gFi1xeHrU3TpBq5AgMBAAGjggMWMIIDEjBEBGkqhkiG9w0BCQ8E  
NzAlMA4GCCqGSIB3DQMCAGIAgDAOBggqhkkiG9w0DBAICAIawBwYFKw4DAgcwCgYI  
KoZiHvcNAwcfWYJKwYBBAGCNxQCBAoeCAVAHMAZQBByMIHaBggrBgEFBQcBAQSB  
zTCBjyCBxwYIKwYBBQUHMAKggbpsZGFwOi8vL0NOPUZCX0VudFN1YknBLENOPUFJ  
QSxDtj1QdWJsaWMLMjBLZXXk1MjBTZXJ2aWNLcyxDTj1TZXJ2aWNLcyxDTj1Db25m  
aWdlcmF0aW9uLERDPWQ5LTEzNTFDMDQwNkEsREM9bnR0ZXN0LERDPWlpY3Jvc29m  
dCxEQz1jb20/Y0FDZxJ0aWZpY2F0ZT9iYXNlP29iamVjdENsYXNzPWNlcnRpZmlj  
YXRrb25BdXR0b3JpdHkwHwQYDVR0BBYEFaIBMzPGoRkGpPN1ZHy7InY1lG+MA4G  
A1UdDwE/wQEAWFI0DBrBgNVHREEZDBioDcGCisGAQQBggjckAwQGCCsGAQUFBwMEBggrBgEF  
BQcDAjANBggkqhkiG9w0BAQUFAAOCAQEAge5W3XJ3766qGf8Y+r2YulGmQ0cskpa7  
tqeZxwZW51skiv6i0S1NPo67HNp08+UfLyRpdghF+ZwtphnCdtPieAVn0Hmd1LvO  
+LxTGA16oVCixa3qRGlx2NUQLH7VzH34ugf/YEaJDPJ/9818TJhQQ7iPTi3jlu9e  
52rYmNLb450egdmVJhbKJIftpcQqbWZY7HP3tzizf50dXhhzg3CGkOJWqSQPQQR  
KnUkVxKjvFIIovrvNM3IluGDjIu9EjjlgE1ZSqA4Lixjko2M/hzhWpejKnVAEV9e  
b+ppWh25jbievk6WrYmgnwPgS9gRBB150q01ahPi6Vj5EOfXoe99KTGCAWggwGfK  
AgEBMEwMzEzMBkGA1UECmSTWljcm9zb2Z0IFBLSzBzZWZtMRQwEgYDVQDDAtG  
Q19FbnRtdWJdQkIKGNnfBgAAAAAPjAJBgUrDgMCGGUAMA0GCSqGSIB3DQEBQUAA  
BIIBAF06/80HTk6v4fx5rYiJOEpz43tvLOQk/0SfXeg4Nlm47SAzqDzNSZ3QljLJ  
vZoBnz4E2vc1ITZsLYpMN0o4rxflZwc+2X7MtoYbnbmV1lZnTnQINDfbmIiXyi+L  
zkjw+ZOTZUxqNYIXhevKru3P3ndhFENhSm/qc5Wovg7igCsDh9XJ/G6zkQ8SEbl  
vkBU21RjPoyKYaEUXz/Y0yViIxpYCPFrByDU50ngXhwOhBcbAc5RImhI807xEO4W  
YQ13sBxW1IsFuxMsmzWQ1TjrFauvjoPt96Hflog96p9w8D1zKxt1hqCI+XqI1qur  
30aWtKmxTQTxG8uBCrczYAgfWGk=</BinarySecurityToken>  
<RequestID xsi:nil="true"  
xmlns="http://schemas.microsoft.com/windows/pki/2009/01/enrollment" />  
</RequestSecurityToken>  
</s:Body>  
</s:Envelope>







## **5 Security**

### **5.1 Security Considerations for Implementers**

None.

### **5.2 Index of Security Parameters**

None.

## 6 Appendix A: Full WSDL

The WSTEP protocol is a profile extension of WS-Trust1.3. As such, it does not have a WSDL.

**WS-Trust 1.3 WSDL:** The full WSDL for WS-Trust can be found at: <http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3.wsdl>.

**WSTEP XML Schema:** For the convenience of implementation, the XML schema is provided here.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:wstep="http://schemas.microsoft.com/windows/pki/2009/01/enrollment"
  targetNamespace="http://schemas.microsoft.com/windows/pki/2009/01/enrollment"
  elementFormDefault="qualified">

  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd" />

  <xs:element name="DispositionMessage" type="wstep:DispositionMessageType" nillable="true"
  />
  <xs:complexType name="DispositionMessageType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang" use="optional" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:element name="CertificateEnrollmentWSDetail" nillable="true"
    type="wstep:CertificateEnrollmentWSDetailType" />
  <xs:complexType name="CertificateEnrollmentWSDetailType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="BinaryResponse" nillable="true"
        type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="ErrorCode" nillable="true" type="xs:int"
      />
      <xs:element minOccurs="0" maxOccurs="1" name="InvalidRequest" nillable="true"
        type="xs:boolean" />
      <xs:element minOccurs="0" maxOccurs="1" name="RequestID" type="xs:string"
        nillable="true" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="RequestID" type="xs:string" nillable="true" />
  <xs:attribute name="PreferredLanguage" type="xml:language" use="optional"/>
</xs:schema>
```



## 7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Windows 7 operating system
- Windows Server 2008 R2 operating system
- Windows 8 operating system
- Windows Server 2012 operating system
- Windows 8.1 operating system
- Windows Server 2012 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 3.1.3:](#) Windows servers set the *DefaultLanguage* parameter to the installed language.

## 8 Change Tracking

This section identifies changes that were made to the [MS-WSTEP] protocol document between the February 2014 and May 2014 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.

- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
<a href="#">3.1.4.1.3.3</a> <a href="#">wst:RequestSecurityTokenType</a>	70951 Changed RequestKET namespace from "wstep:RequestKET" to "wst:RequestKET".	N	Content updated.

## 9 Index

### A

[Abstract data model](#) 15  
[Applicability](#) 11  
[Attribute groups](#) 13  
Attributes  
  [overview](#) 13  
  [RequestSecurityToken2](#) 23  
[Authentication](#) 15

### C

[Capability negotiation](#) 11  
[Change tracking](#) 42  
Complex types  
  [overview](#) 12  
  RequestSecurityToken2  
    [DispositionMessageType](#) 18  
    [overview](#) 18  
    [RequestedSecurityTokenEnrollmentType](#) 18  
    [RequestSecurityTokenEnrollmentType](#) 19  
    [RequestSecurityTokenResponseCollectionEnrollmentType](#) 22  
    [RequestSecurityTokenResponseEnrollmentType](#) 21  
    [RequestTypeEnum](#) 23

### D

[Data model - abstract](#) 15

### E

Elements  
  [overview](#) 12  
  RequestSecurityToken2  
    [DispositionMessage](#) 17  
    [RequestID](#) 17  
    [RequestSecurityToken](#) 17  
    [RequestSecurityTokenResponseCollection](#) 17  
    [RequestType](#) 18  
    [TokenType](#) 18  
Events  
  [local](#) 26  
  [timer](#) 25  
[Examples - RequestSecurityToken request/response message sequence](#) 27

### F

[Fields - vendor-extensible](#) 11  
[Full WSDL](#) 40

### G

[Glossary](#) 6  
[Groups](#) 13

### I

[Implementer - security considerations](#) 39  
[Index of security parameters](#) 39  
[Informative references](#) 8  
[Initialization](#) 15  
[Introduction](#) 6  
[Issue request processing](#) 24

### K

[Key Exchange Token request processing](#) 25

### L

[Local events](#) 26

### M

Message processing  
  [overview](#) 16  
  RequestSecurityToken2  
    [attributes](#) 23  
    [complex types](#) 18  
    [elements](#) 17  
    [messages](#) 16  
    [overview](#) 16  
Messages  
  [attribute groups](#) 13  
  [attributes](#) 13  
  [complex types](#) 12  
  [elements](#) 12  
  [enumerated](#) 12  
  [groups](#) 13  
  [namespaces](#) 12  
  [simple types](#) 12  
  [syntax](#) 12  
  [transport](#) 12

### N

[Namespaces](#) 12  
[Normative references](#) 7

### O

[Overview \(synopsis\)](#) 8

### P

[Parameters - security index](#) 39  
[Preconditions](#) 11  
[Prerequisites](#) 11  
Processing rules  
  [Issue request](#) 24  
  [Key Exchange Token request](#) 25  
  [overview](#) 24

[QueryTokenStatus request](#) 25  
[Renew request](#) 24  
[Product behavior](#) 41

## Q

[QueryTokenStatus request processing](#) 25

## R

### References

[informative](#) 8  
[normative](#) 7  
[Relationship to other protocols](#) 10  
[Renew request processing](#) 24  
[RequestSecurityToken request/response message sequence example](#) 27  
RequestSecurityToken2  
[attributes](#) 23  
complex types  
[DispositionMessageType](#) 18  
[overview](#) 18  
[RequestedSecurityTokenEnrollmentType](#) 18  
[RequestSecurityTokenEnrollmentType](#) 19  
[RequestSecurityTokenResponseCollectionEnrollmentType](#) 22  
[RequestSecurityTokenResponseEnrollmentType](#) 21  
[RequestTypeEnum](#) 23  
elements  
[DispositionMessage](#) 17  
[RequestID](#) 17  
[RequestSecurityToken](#) 17  
[RequestSecurityTokenResponseCollection](#) 17  
[RequestType](#) 18  
[TokenType](#) 18  
messages  
[overview](#) 16  
[RequestSecurityTokenMsg](#) 16  
[RequestSecurityTokenResponseCollectionMsg](#) 16  
[overview](#) 16

## S

### Security

[implementer considerations](#) 39  
[parameter index](#) 39  
SecurityTokenService server  
[abstract data model](#) 15  
[initialization](#) 15  
[local events](#) 26  
message processing  
[overview](#) 16  
[RequestSecurityToken2](#) 16  
[overview](#) 14  
processing rules  
[Issue request](#) 24  
[Key Exchange Token request](#) 25  
[overview](#) 24  
[QueryTokenStatus request](#) 25  
[Renew request](#) 24

sequencing rules  
[overview](#) 16  
[RequestSecurityToken2](#) 16  
[timer events](#) 25  
[timers](#) 15  
Sequencing rules  
[overview](#) 16  
RequestSecurityToken2  
[attributes](#) 23  
[complex types](#) 18  
[elements](#) 17  
[messages](#) 16  
[overview](#) 16  
[Server - SecurityTokenService - overview](#) 14  
[Simple types](#) 12  
[Standards assignments](#) 11  
[Syntax - messages - overview](#) 12

## T

[Timer events](#) 25  
[Timers](#) 15  
[Tracking changes](#) 42  
[Transport](#) 12  
Types  
[complex](#) 12  
[simple](#) 12

## V

[Vendor-extensible fields](#) 11  
[Versioning](#) 11

## W

[WSDL](#) 40