

[MS-WSPOL-Diff]:

Web Services: Policy Assertions and WSDL Extensions

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (~~“this documentation”~~) for protocols, file formats, data portability, computer languages, and standards ~~as well as overviews of the interaction among each of these technologies~~ support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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~~ can make copies of it in order to develop implementations of the technologies ~~that are~~ described in ~~the Open Specifications~~ this documentation and ~~may~~ can distribute portions of it in your implementations ~~using~~ that use these technologies or ~~in~~ your documentation as necessary to properly document the implementation. You ~~may~~ can also distribute in your implementation, with or without modification, any ~~schema, IDL's~~ schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications- documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that ~~may~~ might cover your implementations of the technologies described in the Open Specifications- documentation. Neither this notice nor Microsoft's delivery of ~~the~~ this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open ~~Specification may~~ Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in ~~the Open Specifications~~ this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation ~~may~~ might 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.
- **Fictitious Names.** The example companies, organizations, products, domain names, ~~e-mail~~ email addresses, logos, people, places, and events ~~that are~~ 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 as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications ~~documentation~~ does 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 documents are intended for use in conjunction with publicly available ~~standard~~ standards specifications and network programming art; and ~~assumes, as such,~~ assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
9/25/2009	0.1	Major	First Release.
11/6/2009	0.1.1	Editorial	Changed language and formatting in the technical content.
12/18/2009	0.1.2	Editorial	Changed language and formatting in the technical content.
1/29/2010	1.0	Major	Updated and revised the technical content.
3/12/2010	2.0	Major	Updated and revised the technical content.
4/23/2010	2.0.1	Editorial	Changed language and formatting in the technical content.
6/4/2010	2.0.2	Editorial	Changed language and formatting in the technical content.
7/16/2010	3.0	Major	Updated and revised the technical content.
8/27/2010	3.0	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2010	3.0	None	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	3.0	None	No changes to the meaning, language, or formatting of the technical content.
1/7/2011	3.0	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2011	3.0	None	No changes to the meaning, language, or formatting of the technical content.
3/25/2011	3.0	None	No changes to the meaning, language, or formatting of the technical content.
5/6/2011	3.0	None	No changes to the meaning, language, or formatting of the technical content.
6/17/2011	3.1	Minor	Clarified the meaning of the technical content.
9/23/2011	3.2	Minor	Clarified the meaning of the technical content.
12/16/2011	4.0	Major	Updated and revised the technical content.
3/30/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/12/2012	5.0	Major	Updated and revised the technical content.
10/25/2012	5.0	None	No changes to the meaning, language, or formatting of the technical content.
1/31/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
8/8/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
11/14/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
2/13/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
5/15/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
6/30/2015	6.0	Major	Significantly changed the technical content.
10/16/2015	6.0	No ChangeNone	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1	Introduction	6
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	10
1.6	Applicability Statement	10
1.7	Versioning and Capability Negotiation	10
1.8	Vendor-Extensible Fields	11
1.9	Standards Assignments	11
2	Messages	12
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
2.2.3.1	Basic HTTP Authentication Policy Assertion	13
2.2.3.2	Digest HTTP Authentication Policy Assertion	14
2.2.3.3	NTLM HTTP Authentication Policy Assertion	14
2.2.3.4	Negotiate HTTP Authentication Policy Assertion	14
2.2.3.5	Streamed Message Framing Policy Assertion	14
2.2.3.6	Binary Encoding Policy Assertion	15
2.2.3.7	Message Framing Transport Security Policy Assertion	15
2.2.3.8	Message Framing Security Provider Negotiation Policy Assertion	15
2.2.3.9	One-way Policy Assertion	16
2.2.3.10	Composite Duplex Policy Assertion	17
2.2.3.11	UDP Retransmission-Enabled Policy Assertion	17
2.2.3.12	WebSocket Streamed Policy Assertion	17
2.2.3.13	WebSocket Streamed Request Policy Assertion	18
2.2.3.14	WebSocket Streamed Response Policy Assertion	18
2.2.3.15	SOAP-over-UDP SOAP Binding Transport URI	18
2.2.4	Complex Types	18
2.2.5	Simple Types	18
2.2.6	Attributes	18
2.2.6.1	Using Session WSDL Extension	19
2.2.6.2	Is Initiating WSDL Extension	19
2.2.6.3	Is Terminating WSDL Extension	19
2.2.7	Groups	20
2.2.8	Attribute Groups	20
3	Protocol Details	21
3.1	Server Details	21
3.2	Client Details	21
4	Protocol Examples	22
5	Security	23
5.1	Security Considerations for Implementers	23
5.2	Index of Security Parameters	23
6	Appendix A: Full WSDL	24
6.1	Basic HTTP Authentication Policy Assertion	24
6.2	Digest HTTP Authentication Policy Assertion	25

6.3	NTLM HTTP Authentication Policy Assertion.....	25
6.4	Negotiate HTTP Authentication Policy Assertion.....	26
6.5	Streamed Message Framing Policy Assertion.....	26
6.6	Binary Encoding Policy Assertion.....	26
6.7	Message Framing Transport Security Policy Assertion.....	27
6.8	Message Framing Security Provider Negotiation Policy Assertion.....	28
6.9	One-way and Composite Duplex Policy Assertions.....	28
6.10	UDP Retransmission-Enabled Policy Assertion.....	29
6.11	WebSocket Streamed Policy Assertion.....	29
6.12	WebSocket Streamed Request Policy Assertion.....	30
6.13	WebSocket Streamed Response Policy Assertion.....	30
6.14	SOAP-over-UDP Transport URI.....	30
6.15	Using Session, Is Initiating, and Is Terminating WSDL Extensions.....	31
7	Appendix B: Product Behavior.....	32
8	Change Tracking.....	33
9	Index.....	34

1 Introduction

This document specifies a collection of **Web service** policy assertions and Web Services Description Language (WSDL) extensions, which define domain-specific behavior for the interaction between two Web service entities. This document does not define any specific Web service endpoints or message exchanges.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative ~~and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [RFC2119]. Sections 1.5 and 1.9 are also normative but do not contain those terms.~~ All other sections and examples in this specification are informative.

1.1 Glossary

~~The~~This document uses the following terms ~~are specific to this document:~~

certificate: A certificate is a collection of attributes (1) and extensions that can be stored persistently. The set of attributes in a certificate can vary depending on the intended usage of the certificate. A certificate securely binds a public key to the entity that holds the corresponding private key. A certificate is commonly used for authentication (2) and secure exchange of information on open networks, such as the Internet, extranets, and intranets. Certificates are digitally signed by the issuing certification authority (CA) and can be issued for a user, a computer, or a service. The most widely accepted format for certificates is defined by the ITU-T X.509 version 3 international standards. For more information about attributes and extensions, see [RFC3280] and [X509] sections 7 and 8.

client: A computer on which the remote procedure call (RPC) client is executing.

initiating operation: A WSDL operation that is the first operation sent by the client.

Initiating Stream: The protocol stream that flows from the initiator (2).

input message: The **WSDL** message referred to by the input element in a **WSDL operation**.

notification operation: An operation in which the endpoint sends a message, as specified in [WSDL].

one-way operation: An operation in which the endpoint receives a message, as specified in [WSDL].

output message: The **WSDL** message referred to by the output element in a **WSDL operation**.

processing operation: A **WSDL operation** that is not a **terminating operation**.

sessionful transport: A transport that associates messages into message groups defined by the transport.

SOAP message: An **XML** document consisting of a mandatory SOAP envelope, an optional SOAP header, and a mandatory SOAP body. See [SOAP1.2-1/2007] section 5 for more information.

SSL/TLS handshake: The process of negotiating and establishing a connection protected by Secure Sockets Layer (SSL) or Transport Layer Security (TLS). For more information, see [SSL3] and [RFC2246].

terminating operation: A **WSDL operation** that is the last operation sent by a **client**.

web service: A unit of application logic that provides data and services to other applications and can be called by using standard Internet transport protocols such as HTTP, Simple Mail Transfer

Protocol (SMTP), or File Transfer Protocol (FTP). Web services can perform functions that range from simple requests to complicated business processes.

Web Services Description Language (WSDL): An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

WSDL extension: Represents a requirement or a capability of a **Web service**, which is defined by using the **WSDL** extensibility model.

WSDL operation: A single action or function of a web service. The execution of a WSDL operation typically requires the exchange of messages between the service requestor and the service provider.

WSDL port type: A named set of logically-related, abstract **Web Services Description Language (WSDL)** operations and messages.

XML: The Extensible Markup Language, as described in [XML1.0].

XML namespace: A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [RFC3986]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [XMLNS-2ED].

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

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

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

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. We will assist you in finding the relevant information.

[MC-NBFSE] Microsoft Corporation, ".NET Binary Format: SOAP Extension".

[MC-NBFS] Microsoft Corporation, ".NET Binary Format: SOAP Data Structure".

[MC-NMF] Microsoft Corporation, ".NET Message Framing Protocol".

[MC-NPR] Microsoft Corporation, ".NET Packet Routing Protocol".

[MS-NNS] Microsoft Corporation, ".NET NegotiateStream Protocol".

[MS-NTHT] Microsoft Corporation, "NTLM Over HTTP 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>

[RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., et al., "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999, <http://www.rfc-editor.org/rfc/rfc2617.txt>

[RFC4346] Dierks, T., and Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006, <http://www.ietf.org/rfc/rfc4346.txt>

[RFC4559] Jaganathan, K., Zhu, L., and Brezak, J., "SPNEGO-based Kerberos and NTLM HTTP Authentication in Microsoft Windows", RFC 4559, June 2006, <http://www.rfc-editor.org/rfc/rfc4559.txt>

[RFC6455] Fette, I., and Melnikov, A., "The WebSocket Protocol", RFC 6455, December 2011, <http://www.ietf.org/rfc/rfc6455.txt>

[SOAP-UDP] Combs, H., Justice, J., Kakivaya, G., et al., "SOAP-over-UDP", September 2004, <http://specs.xmlsoap.org/ws/2004/09/soap-over-udp/soap-over-udp.pdf>

[WS-Policy] Siddharth, B., Box, D., Chappell, D., et al., "Web Services Policy 1.2 - Framework (WS-Policy)", April 2006, <http://www.w3.org/Submission/2006/SUBM-WS-Policy-20060425/>

[WSAddressing] Box, D., et al., "Web Services Addressing (WS-Addressing)", August 2004, <http://www.w3.org/Submission/ws-addressing/>

[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>

[WSPolicyAtt] BEA Systems, IBM, Microsoft Corporation, SAP, Sonic Software, VeriSign, "Web Services Policy 1.2 - Attachment (WS-PolicyAttachment)", April 2006, <http://www.w3.org/Submission/WS-PolicyAttachment/>

[WSSP1.2] OASIS Standard, "WS-SecurityPolicy 1.2", July 2007, <http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/ws-securitypolicy-1.2-spec-os.pdf>

[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>

[XMLNS-2ED] World Wide Web Consortium, "Namespaces in XML 1.0 (Second Edition)", August 2006, <http://www.w3.org/TR/2006/REC-xml-names-20060816/>

[XMLSCHEMA1] Thompson, H., 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/>

1.2.2 Informative References

None.

1.3 Overview

WS-Policy (Web Services Policy Framework) [WS-Policy] and WS-PolicyAttachment (Web Services Policy Attachment) [WSPolicyAtt] collectively define a framework, model, and grammar for expressing the requirements and general characteristics of entities in an **XML** web services-based system. This document specifies the following policy assertions as defined in [WS-Policy]:

- **Basic HTTP Authentication**

The Basic HTTP Authentication policy assertion indicates that a Web service endpoint requires authentication using the Basic Authentication scheme, as specified in [RFC2617] section 2.
- **Digest HTTP Authentication**

The Digest HTTP Authentication policy assertion indicates that a Web service endpoint requires authentication using the Digest Access Authentication scheme, as specified in [RFC2617] section 3.
- **NTLM HTTP Authentication**

The NTLM HTTP Authentication policy assertion indicates that a Web service endpoint requires authentication using the NTLM over HTTP Protocol, as specified in [MS-NTHT].
- **Negotiate HTTP Authentication**

The Negotiate HTTP Authentication policy assertion indicates that a Web service endpoint requires authentication using the HTTP Negotiate Authentication scheme, as specified in [RFC4559] section 4.
- **Streamed Message Framing**

The Streamed Message Framing policy assertion indicates that a Web service endpoint requires messages to be transferred to it using the framing protocol specified in [MC-NMF] with "Singleton Un sized" mode, as specified in [MC-NMF] section 2.2.3.2.
- **Binary Encoding**

The Binary Encoding policy assertion indicates that **SOAP messages** are required to be formatted as specified in [MC-NBFS] or [MC-NBFSE].
- **Message Framing Transport Security**

The Message Framing Transport Security policy assertion indicates that a Web service endpoint requires messages to be transferred to it using the framing protocol specified in [MC-NMF] with an "application/ssl-tls" protocol upgrade, as specified in [MC-NMF] section 2.2.3.5.
- **Message Framing Security Provider Negotiation**

The Message Framing Security Provider Negotiation policy assertion indicates that a Web service endpoint requires messages to be transferred to it using the framing protocol specified in [MC-NMF] with an "application/negotiate" protocol upgrade, as specified in [MC-NMF] section 2.2.3.5.
- **One-way**

The One-way policy assertion indicates that a Web service endpoint treats all **input messages** as **one-way operations** and all **output messages** as **notification operations**. This policy assertion also indicates whether to send messages as .NET packets, as specified in [MC-NPR] section 2.2.2.
- **Composite Duplex**

The Composite Duplex policy assertion indicates that a Web service endpoint requires two separate transport connections for messages to and from it.
- **UDP Retransmission Enabled**

The UDP Retransmission Enabled policy assertion indicates that a Web service endpoint has enabled retransmission, as specified in [SOAP-UDP].

- **WebSocket Streamed**

The WebSocket Streamed policy assertion indicates that a Web service endpoint intends to send and receive messages as a stream of bytes, as specified in [RFC6455].

- **WebSocket Streamed Request**

The WebSocket Streamed Request policy assertion indicates that a Web service endpoint intends to receive messages as a stream of bytes.

- **WebSocket Streamed Response**

The WebSocket Streamed Response policy assertion indicates that a Web service endpoint intends to send messages as a stream of bytes.

This document specifies the following **WSDL extensions** using the extensibility model described in [WSDL]:

- **Using Session**

The Using Session WSDL extension, applicable over a **WSDL port type**, indicates whether a port type defines any **initiating operations**.

- **Is Initiating**

The Is Initiating WSDL extension, applicable over a **WSDL operation**, indicates whether this operation is an initiating operation.

- **Is Terminating**

The Is Terminating WSDL extension, applicable over a WSDL operation, indicates whether this operation is a **terminating operation**.

This document specifies the following **WSDL URIs** using the extensibility model described in [WSDL]:

- **SOAP-over-UDP** -- <http://schemas.microsoft.com/soap/udp>

The SOAP-over-UDP transport defines the following URI: <http://schemas.microsoft.com/soap/udp>, which indicates that a Web service endpoint requires messages to be transferred using the [SOAP-UDP] protocol.

1.4 Relationship to Other Protocols

This document only defines policy assertions and WSDL extensions for existing protocols and does not define any new protocols.

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

None.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

This document only defines policy assertions and WSDL extensions for existing protocols and does not define any new messages.

2.1 Transport

None.

2.2 Common Message Syntax

2.2.1 Namespaces

This specification defines and references the following **XML namespaces** using the mechanisms specified in [XMLNS-2ED], which **MUST** be used by the implementations of this specification.

Prefix	Namespace URI	Reference
http	http://schemas.microsoft.com/ws/06/2004/policy/http	
msf	http://schemas.microsoft.com/ws/2006/05/framing/policy	
msb	http://schemas.microsoft.com/ws/06/2004/mspolicy/netbinary1	
ow	http://schemas.microsoft.com/ws/2005/05/routing/policy	
cdp	http://schemas.microsoft.com/net/2006/06/duplex	
msc	http://schemas.microsoft.com/ws/2005/12/wsdli/contract	
wSDL	http://schemas.xmlsoap.org/wsdli/	[WSDL]
wsp	http://schemas.xmlsoap.org/ws/2004/09/policy	[WS-Policy]
sp	http://schemas.xmlsoap.org/ws/2005/07/securitypolicy	[WSSP1.2]
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd	[WSS]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
sud	http://schemas.microsoft.com/ws/06/2010/policy/soap/udp	
mswsp	http://schemas.microsoft.com/soap/websocket/policy	

2.2.2 Messages

This specification does not define any messages.

2.2.3 Elements

The following table summarizes the set of common XML Schema element definitions defined by this specification.

Element	Description
BasicAuthentication	Indicates that clients are authenticated using the Basic Authentication scheme.

Element	Description
DigestAuthentication	Indicates that clients are authenticated using the Digest Access Authentication scheme.
NtlmAuthentication	Indicates that clients are authenticated using the NTLM over HTTP Protocol.
NegotiateAuthentication	Indicates that clients are authenticated using the HTTP Negotiate Authentication scheme.
Streamed	Indicates that messages are exchanged using the .NET Message Framing Protocol with a particular framing mode.
BinaryEncoding	Indicates that messages are exchanged using the binary format with in-band dictionary specified.
SslTransportSecurity	Indicates that messages are exchanged using the .NET Message Framing Protocol with a particular preamble.
WindowsTransportSecurity	Indicates that messages are exchanged using the .NET Message Framing Protocol with a particular preamble.
OneWay	Indicates that all input messages are treated as input messages in one-way operations and all output messages as notification operations.
CompositeDuplex	Indicates that messages sent back to the client are sent using the endpoint reference provided by the client in the ReplyTo header.
RetransmissionEnabled	Indicates that the Web service endpoint has enabled retransmission of SOAP-over-UDP messages.
Streamed	Indicates that the Web service endpoint intends to send and receive messages as a stream of bytes over the WebSockets protocol.
StreamedRequest	Indicates that the Web service endpoint intends to receive messages as a stream of bytes over the WebSockets protocol.
StreamedResponse	Indicates that the Web service endpoint intends to send messages as a stream of bytes over the WebSockets protocol.

The following sections contain the **XML schema** description for the policy assertions and WSDL extensions specified in this document.

2.2.3.1 Basic HTTP Authentication Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2004/policy/http"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="BasicAuthentication" />
</xs:schema>
```

The following describes the content model of the **BasicAuthentication** element.

/http:BasicAuthentication: A Web service endpoint with Basic HTTP Authentication policy assertion MUST authenticate clients using the Basic Authentication scheme, as specified in [RFC2617] section 2.

2.2.3.2 Digest HTTP Authentication Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2004/policy/http"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="DigestAuthentication" />
</xs:schema>
```

The following describes the content model of the **DigestAuthentication** element.

/http:DigestAuthentication: A Web service endpoint with Digest HTTP Authentication policy assertion MUST authenticate clients using the Digest Access Authentication scheme, as specified in [RFC2617] section 3.

2.2.3.3 NTLM HTTP Authentication Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2004/policy/http"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="NtlmAuthentication" />
</xs:schema>
```

The following describes the content model of the **NtlmAuthentication** element.

/http:NtlmAuthentication: A Web service endpoint with NTLM HTTP Authentication policy assertion MUST authenticate clients using the NTLM over HTTP Protocol, as specified in [MS-NTHT].

2.2.3.4 Negotiate HTTP Authentication Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2004/policy/http"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="NegotiateAuthentication" />
</xs:schema>
```

The following describes the content model of the **NegotiateAuthentication** element.

/http:NegotiateAuthentication: A Web service endpoint with Negotiate HTTP Authentication policy assertion MUST authenticate clients using the HTTP Negotiate Authentication scheme, as specified in [RFC4559] section 4.

2.2.3.5 Streamed Message Framing Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/2006/05/framing/policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Streamed" />
</xs:schema>
```

The following describes the content model of the **Streamed** element.

/msf:Streamed: A Web service endpoint with Streamed Message Framing policy assertion MUST exchange messages using the .NET Message Framing Protocol [MC-NMF]. The framing mode MUST be Singleton Unsized (as described in [MC-NMF] section 2.2.3.2).

2.2.3.6 Binary Encoding Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2004/mspolicy/netbinary1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="BinaryEncoding" />
</xs:schema>
```

The following describes the content model of the **BinaryEncoding** element.

/msb:BinaryEncoding: A Web service endpoint with a Binary Encoding policy assertion and configured with a **sessionful transport** MUST exchange messages using the binary format with in-band dictionary specified in [MC-NBFSE]. A Web service endpoint with a Binary Encoding policy assertion and configured with a transport that is not a sessionful transport MUST exchange messages using the binary format specified in [MC-NBFS].<1>

2.2.3.7 Message Framing Transport Security Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/2006/05/framing/policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="SslTransportSecurity">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="RequireClientCertificate" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

The following describes the content model of the **SslTransportSecurity** element.

/msf:SslTransportSecurity: A Web service endpoint with the Message Framing Transport Security policy assertion MUST exchange messages using the .NET Message Framing Protocol [MC-NMF]. The preamble MUST include an upgrade request for "application/ssl-tls", as specified in [MC-NMF] section 2.2.3.5. The Web service endpoint MUST accept an upgrade request for "application/ssl-tls".

/msf:SslTransportSecurity/msf:RequireClientCertificate: A parameter that specifies that a client MUST provide a server-recognizable **certificate**, as specified in [RFC4346] section 7.4.6, during the initial **SSL/TLS handshake** described in [RFC4346] section 7.3.

The SslTransportSecurity element is nested inside the sp:TransportBinding/wsp:Policy/sp:TransportToken/wsp:Policy element of the TransportBinding Assertion, as specified in [WSSP1.2], to indicate that the SOAP message protection is provided by the Transport Layer Security Protocol [RFC4346].

2.2.3.8 Message Framing Security Provider Negotiation Policy Assertion

```
<xs:schema
```

```

        attributeFormDefault="unqualified"
        elementFormDefault="qualified"
        targetNamespace="http://schemas.microsoft.com/ws/2006/05/framing/policy"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
    <xs:element name="WindowsTransportSecurity">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="ProtectionLevel">
                    <xs:simpleType>
                        <xs:restriction base="xs:string">
                            <xs:enumeration value="None"/>
                            <xs:enumeration value="Sign"/>
                            <xs:enumeration value="EncryptAndSign"/>
                        </xs:restriction>
                    </xs:simpleType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

The following describes the content model of the **WindowsTransportSecurity** element.

/msf:WindowsTransportSecurity: A Web service endpoint with the Message Framing Security Provider Negotiation policy assertion MUST exchange messages using the .NET Message Framing Protocol [MC-NMF]. The preamble MUST include an upgrade request for "application/negotiate", as specified in [MC-NMF] section 2.2.3.5. The Web service endpoint MUST accept an upgrade request for "application/negotiate".

/msf:WindowsTransportSecurity/msf:ProtectionLevel: A parameter that specifies the minimal level of protection that MUST be applied to protect the **Initiating Stream**.

The protection level MUST be set to one of the following values:

Value	Meaning
None	Specifies that the Initiating Stream SHOULD be unsigned and SHOULD be unencrypted. The Initiating Stream MAY be signed and MAY be encrypted.
Sign	Specifies that the Initiating Stream MUST be signed. The signed Initiating Stream SHOULD be unencrypted. The signed Initiating Stream MAY be encrypted.
EncryptAndSign	Specifies that the Initiating Stream MUST be encrypted and then signed.

The WindowsTransportSecurity element is nested inside the sp:TransportBinding/wsp:Policy/sp:TransportToken/wsp:Policy element of the TransportBinding Assertion, as specified in [WSSP1.2], to indicate that the SOAP message protection is provided by the .NET NegotiateStream Protocol [MS-NNS].

2.2.3.9 One-way Policy Assertion

```

<xs:schema
    attributeFormDefault="unqualified"
    elementFormDefault="qualified"
    targetNamespace="http://schemas.microsoft.com/ws/2005/05/routing/policy"
    xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="OneWay">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="PacketRoutable" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

```



```
</xs:schema>
```

The following describes the content model of the **OneWay** element.

/ow:OneWay: A Web service endpoint with a One-way policy assertion MUST treat all input messages as input messages in one-way operations. The Web service endpoint MUST NOT send replies to a received message. The Web service endpoint MUST treat all output messages as output messages in notification operations. The Web service endpoint MUST NOT accept replies from sent messages.

/ow:OneWay/ow:PacketRoutable: When present, indicates that messages sent to the Web service endpoint MUST be sent as .NET packets, as specified in [MC-NPR] section 2.2.2.

2.2.3.10 Composite Duplex Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/net/2006/06/duplex"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="CompositeDuplex" />
</xs:schema>
```

The following describes the content model of the **CompositeDuplex** element.

/cdp:CompositeDuplex: A Web service endpoint with a Composite Duplex policy assertion MUST send any messages intended for the client to the endpoint reference provided by the client in the ReplyTo header. Messages sent to the Web service endpoint MUST specify an endpoint reference in the ReplyTo header [WSAddressing] of each request message. Messages sent by the Web service endpoint to the client MUST be sent using the WSDL binding for the Web service endpoint.

2.2.3.11 UDP Retransmission-Enabled Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/06/2010/policy/soap/udp"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="RetransmissionEnabled" />
</xs:schema>
```

The following describes the content model of the **RetransmissionEnabled** element.

/sud:RetransmissionEnabled: A Web service endpoint with retransmission enabled MUST retransmit messages. A client SHOULD enable a mechanism to detect duplicates and take appropriate action as messages are received from this Web service endpoint.<2>

2.2.3.12 WebSocket Streamed Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/soap/websocket/policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Streamed" />
</xs:schema>
```

The following describes the content model of the **Streamed** element.

/mswsp:Streamed: A Web service endpoint with WebSocket Streamed policy assertion MUST send and receive messages as a stream of bytes.<3>

2.2.3.13 WebSocket Streamed Request Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/soap/websocket/policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="StreamedRequest" />
</xs:schema>
```

The following describes the content model of the **StreamedRequest** element.

/mswsp:StreamedRequest: A client SHOULD send a message to a Web service endpoint with WebSocket Streamed Request policy assertion as a stream of bytes.<4>

2.2.3.14 WebSocket Streamed Response Policy Assertion

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/soap/websocket/policy"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="StreamedResponse" />
</xs:schema>
```

The following describes the content model of the **StreamedResponse** element.

/mswsp:StreamedResponse: A Web service endpoint with WebSocket Streamed Response policy assertion MUST send messages as a stream of bytes.<5>

2.2.3.15 SOAP-over-UDP SOAP Binding Transport URI

This protocol does not define any new element. However, this protocol defines a new transport URI, <http://schemas.microsoft.com/soap/udp>, which specifies that a Web service endpoint requires messages to be transferred using the [SOAP-UDP] protocol.<6>

2.2.4 Complex Types

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

2.2.5 Simple Types

This specification does not define any common XML Schema simple type definitions.

2.2.6 Attributes

The following table summarizes the set of common XML Schema attribute definitions defined by this specification.

Attribute	Description
usingSession	Specifies that session semantics are required.
isInitiating	Indicates that an operation is an initiating operation.
isTerminating	Indicates that an operation is a terminating operation.

The following sections contain the XML schema description for the WSDL extensions specified in this document.

2.2.6.1 Using Session WSDL Extension

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/2005/12/wsd/contract/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="usingSession" type="xs:boolean" />
</xs:schema>
```

The following describes the content model of the **usingSession** attribute.

/msc:usingSession: A WSDL port type having a Using Session WSDL extension with a true value specifies that:

- At least one initiating operation **MUST** be present.
- At least one terminating operation **MAY** be present.
- A client **MUST** request one or more initiating operations, followed by zero or more **processing operations**, followed by zero or one terminating operations.
- The Web service endpoint **MUST** process all operations in the order they were sent by the client.

2.2.6.2 Is Initiating WSDL Extension

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/2005/12/wsd/contract/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="isInitiating" type="xs:boolean" />
</xs:schema>
```

The following describes the content model of the **isInitiating** attribute.

/msc:isInitiating: A WSDL operation having an Is Initiating WSDL extension with a true value indicates that this operation is an initiating operation.

2.2.6.3 Is Terminating WSDL Extension

```
<xs:schema
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/ws/2005/12/wsd/contract/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="isTerminating" type="xs:boolean"/>
</xs:schema>
```

</xs:schema>

The following describes the content model of the **isTerminating** attribute.

/msc:isTerminating: A WSDL operation having an Is Terminating WSDL extension that has a true value indicates that the operation is a terminating operation.

2.2.7 Groups

This specification does not define any common XML Schema group definitions.

2.2.8 Attribute Groups

This specification does not define any common XML Schema attribute group definitions.

3 Protocol Details

The policy assertions defined in this document specify behavior over all messages sent to and from a Web service endpoint and so they MUST have the following policy subjects, as defined in [WS-Policy]:

- Endpoint policy subject

[WSPolicyAtt] defines a set of WSDL/1.1 [WSDL] policy attachment points for the policy subject noted previously.

The following is the list of WSDL/1.1 [WSDL] elements whose scope contains the policy subject for the policy assertions defined in this document, but which MUST NOT have the policy assertions attached:

- wsdl:portType
- wsdl:port

The following is the list of WSDL/1.1 [WSDL] elements whose scope contains the policy subject for the policy assertions defined in this document, and which MAY have the policy assertions attached:

- wsdl:binding

The assertions defined in this document MUST NOT contain a nested policy expression.

The assertions defined in this document MUST NOT be specified multiple times in the same policy alternative, as defined in [WS-Policy].

The Using Session WSDL extension defined in this document MAY be used on the following list of WSDL/1.1 [WSDL] elements:

- wsdl:portType

The Is Initiating and Is Terminating WSDL extensions defined in this document MAY be used on the following list of WSDL/1.1 [WSDL] elements:

- wsdl:operation

3.1 Server Details

None.

3.2 Client Details

None.

4 Protocol Examples

Section 6, Appendix A: Full WSDL, provides examples of all of the policy assertions specified in this document.

5 Security

5.1 Security Considerations for Implementers

Security considerations are discussed in detail under the security considerations section in [WS-Policy].

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation the full WSDLs with schemas are provided in the following sections.

WSDL or schema name	Assertion/WSDL extension/Transport URI	Section
Basic HTTP Authentication Policy Assertion	http:BasicAuthentication	6.1
Digest HTTP Authentication Policy Assertion	http:DigestAuthentication	6.2
NTLM HTTP Authentication Policy Assertion	http:NtlmAuthentication	6.3
Negotiate HTTP Authentication Policy Assertion	http:NegotiateAuthentication	6.4
Streamed Message Framing Policy Assertion	msf:Streamed	6.5
Binary Encoding Policy Assertion	msb:BinaryEncoding	6.6
Message Framing Transport Security Policy Assertion	msf:SslTransportSecurity	6.7
Message Framing Security Provider Negotiation Policy Assertion	msf:WindowsTransportSecurity	6.8
One-way Policy Assertion	ow:OneWay	6.9
Composite Duplex Policy Assertion	cdp:CompositeDuplex	6.9
UDP Retransmission Enabled Policy Assertion	sud:RetransmissionEnabled	6.10
WebSocket Streamed Policy Assertion	mwsdp:Streamed	6.11
WebSocket Streamed Request Policy Assertion	mwsdp:StreamedRequest	6.12
WebSocket Streamed Response Policy Assertion	mwsdp:StreamedResponse	6.13
SOAP-over-UDP transport URI	http://schemas.microsoft.com/soap/udp	6.14
Using Session WSDL Extension	msc:UsingSession	6.15
Is Initiating WSDL Extension	msc:IsInitiating	6.15
Is Terminating WSDL Extension	msc:IsTerminating	6.15

6.1 Basic HTTP Authentication Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
  targetNamespace="http://tempuri.org/"
  xmlns:tns="http://tempuri.org/"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsm="http://schemas.xmlsoap.org/ws/2004/09/mex"
  xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
  xmlns:wsdl=http://schemas.xmlsoap.org/wsdl/
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <http:BasicAuthentication />
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
```



```

<!-- omitted elements -->
<wsdl:binding name="MyBinding" type="tns:MyPortType">
  <wsp:PolicyReference URI="#MyPolicy" />
  <!-- omitted elements -->
</wsdl:binding>
<!-- omitted elements -->
</wsdl:definitions>

```

6.2 Digest HTTP Authentication Policy Assertion

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
  targetNamespace="http://tempuri.org/"
  xmlns:tns="http://tempuri.org/"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
  xmlns:wsdl=http://schemas.xmlsoap.org/wsdl/
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp:All>
        <!-- omitted elements -->
        <http:DigestAuthentication />
        <!-- omitted elements -->
      </wsp:All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy" />
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>

```

6.3 NTLM HTTP Authentication Policy Assertion

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
  targetNamespace="http://tempuri.org/"
  xmlns:tns="http://tempuri.org/"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
  xmlns:wsdl=http://schemas.xmlsoap.org/wsdl/
  xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp:All>
        <!-- omitted elements -->
        <http:NtlmAuthentication />
        <!-- omitted elements -->
      </wsp:All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy" />
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>

```

6.4 Negotiate HTTP Authentication Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:http="http://schemas.microsoft.com/ws/06/2004/policy/http">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <http:NegotiateAuthentication />
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy" />
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

6.5 Streamed Message Framing Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:msf="http://schemas.microsoft.com/ws/2006/05/framing/policy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <msf:Streamed/>
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

6.6 Binary Encoding Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy">
```

```

        xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
        xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
        xmlns:msb="http://schemas.microsoft.com/ws/06/2004/mspolicy/netbinary1"
    <wsp:Policy wsu:Id="MyPolicy">
        <wsp:ExactlyOne>
            <wsp>All>
                <!-- omitted elements -->
                <msb:BinaryEncoding />
                <!-- omitted elements -->
            </wsp>All>
        </wsp:ExactlyOne>
    </wsp:Policy>
    <!-- omitted elements -->
    <wSDL:binding name="MyBinding" type="tns:MyPortType">
        <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
        <!-- omitted elements -->
    </wSDL:binding>
    <!-- omitted elements -->
</wSDL:definitions>

```

6.7 Message Framing Transport Security Policy Assertion

```

<?xml version="1.0" encoding="utf-8"?>
<wSDL:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:msf="http://schemas.microsoft.com/ws/2006/05/framing/policy"
    xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
    <wsp:Policy wsu:Id="MyPolicy">
        <wsp:ExactlyOne>
            <wsp>All>
                <!-- omitted elements -->
                <sp:TransportBinding >
                    <wsp:Policy>
                        <sp:TransportToken>
                            <wsp:Policy>
                                <msf:SslTransportSecurity >
                                    <msf:RequireClientCertificate>
                                    </msf:RequireClientCertificate>
                                </msf:SslTransportSecurity>
                            </wsp:Policy>
                        </sp:TransportToken>
                        <!-- omitted elements -->
                    </wsp:Policy>
                </sp:TransportBinding>
                <!-- omitted elements -->
            </wsp>All>
        </wsp:ExactlyOne>
    </wsp:Policy>
    <!-- omitted elements -->
    <wSDL:binding name="MyBinding" type="tns:MyPortType">
        <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
        <!-- omitted elements -->
    </wSDL:binding>
    <!-- omitted elements -->
</wSDL:definitions>

```

6.8 Message Framing Security Provider Negotiation Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:msf="http://schemas.microsoft.com/ws/2006/05/framing/policy"
    xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <sp:TransportBinding >
          <wsp:Policy>
            <sp:TransportToken>
              <wsp:Policy>
                <msf:WindowsTransportSecurity >
                  <msf:ProtectionLevel>EncryptAndSign</msf:ProtectionLevel>
                </msf:WindowsTransportSecurity>
              </wsp:Policy>
            </sp:TransportToken>
            <!-- omitted elements -->
          </wsp:Policy>
        </sp:TransportBinding>
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

6.9 One-way and Composite Duplex Policy Assertions

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:cdp="http://schemas.microsoft.com/net/2006/06/duplex"
    xmlns:ow="http://schemas.microsoft.com/ws/2005/05/routing/policy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <cdp:CompositeDuplex />
        <ow:OneWay />
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy" />
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

```

    </wsdl:binding>
    <!-- omitted elements -->
</wsdl:definitions>

```

6.10 UDP Retransmission-Enabled Policy Assertion

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:sud="http://schemas.microsoft.com/ws/06/2010/policy/soap/udp">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <sud:RetransmissionEnabled />
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy" />
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>

```

6.11 WebSocket Streamed Policy Assertion

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:mwsp="http://schemas.microsoft.com/soap/websocket/policy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <mwsp:Streamed/>
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>

```

6.12 WebSocket Streamed Request Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:mwsp="http://schemas.microsoft.com/soap/websocket/policy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <mwsp:StreamedRequest/>
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

6.13 WebSocket Streamed Response Policy Assertion

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
    targetNamespace="http://tempuri.org/"
    xmlns:tns="http://tempuri.org/"
    xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:mwsp="http://schemas.microsoft.com/soap/websocket/policy">
  <wsp:Policy wsu:Id="MyPolicy">
    <wsp:ExactlyOne>
      <wsp>All>
        <!-- omitted elements -->
        <mwsp:StreamedResponse/>
        <!-- omitted elements -->
      </wsp>All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="tns:MyPortType">
    <wsp:PolicyReference URI="#MyPolicy"></wsp:PolicyReference>
    <!-- omitted elements -->
  </wsdl:binding>
  <!-- omitted elements -->
</wsdl:definitions>
```

6.14 SOAP-over-UDP Transport URI

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
    xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/">
  <!-- omitted elements -->
  <wsdl:binding name="MyBinding" type="MyPortType">
```

```

    <soap12:binding transport="http://schemas.microsoft.com/soap/udp"/>
    <wsdl:operation name="MyOperation">
      <!-- omitted elements -->
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="MyService">
    <wsdl:port name="MyPort" binding="MyBinding">
      <soap12:address location="soap.udp://myhost/" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>

```

6.15 Using Session, Is Initiating, and Is Terminating WSDL Extensions

```

<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
  targetNamespace="http://tempuri.org/"
  xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <!-- omitted elements -->
  <wsdl:portType msc:usingSession="true" name="IContractName">
    <wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="A">
      <!-- omitted elements -->
    </wsdl:operation>
    <wsdl:operation msc:isInitiating="false" msc:isTerminating="false" name="B">
      <!-- omitted elements -->
    </wsdl:operation>
    <wsdl:operation msc:isInitiating="false" msc:isTerminating="true" name="C">
      <!-- omitted elements -->
    </wsdl:operation>
  <!-- omitted elements -->
</wsdl:portType>
<!-- omitted elements -->
</wsdl:definitions>

```

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.

This document specifies version-specific details in the Microsoft .NET Framework. For information about which versions of .NET Framework are available in each released windows product or as supplemental software, see [MS-NETOD] section 4.

- Microsoft .NET Framework 3.0
- Microsoft .NET Framework 3.5
- Microsoft .NET Framework 4.0
- Microsoft .NET Framework 4.5
- Microsoft .NET Framework 4.6

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 2.2.3.6: The Windows implementation of this protocol uses a transport that is not a sessionful transport for the "http" and "https" schemes.

<2> Section 2.2.3.11: UDP retransmission-enabled policy assertion is not available in .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4.0.

<3> Section 2.2.3.12: WebSocket Streamed policy assertions are not available in .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4.0.

<4> Section 2.2.3.13: WebSocket Streamed Request policy assertions are not available in .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4.0.

<5> Section 2.2.3.14: WebSocket Streamed Response policy assertions are not available in .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4.0.

<6> Section 2.2.3.15: SOAP-over-UDP transport URI is not available in .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4.0.

8 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

9 Index

A

- Applicability 10
- Attribute groups 20
- Attributes 18
 - Is Initiating WSDL Extension 19
 - Is Terminating WSDL Extension 19
 - isInitiating 19
 - isTerminating 19
 - overview 18
 - Using Session WSDL Extension 19
 - usingSession 19

B

- Basic HTTP Authentication policy assertion 13
- Basic HTTP Authentication Policy Assertion element 13
- BasicAuthentication element 13
- Binary Encoding policy assertion 15
- Binary Encoding Policy Assertion element 15
- BinaryEncoding element 15

C

- Capability negotiation 10
- Change tracking 33
- Client - details 21
- Complex types 18
- Composite Duplex policy assertion 17
- Composite Duplex Policy Assertion element 17
- CompositeDuplex element 17

D

- Details
 - client 21
 - overview 21
 - server 21
- Digest HTTP Authentication policy assertion 14
- Digest HTTP Authentication Policy Assertion element 14
- DigestAuthentication element 14

E

- Elements
 - Basic HTTP Authentication Policy Assertion 13
 - BasicAuthentication 13
 - Binary Encoding Policy Assertion 15
 - BinaryEncoding 15
 - Composite Duplex Policy Assertion 17
 - CompositeDuplex 17
 - Digest HTTP Authentication Policy Assertion 14
 - DigestAuthentication 14
 - Message Framing Security Provider Negotiation Policy Assertion 15
 - Message Framing Transport Security Policy Assertion 15
 - Negotiate HTTP Authentication Policy Assertion 14
 - NegotiateAuthentication 14
 - NTLM HTTP Authentication Policy Assertion 14
 - NtlmAuthentication 14
 - OneWay 16
 - One-way Policy Assertion 16
 - SOAP-over-UDP SOAP Binding Transport URI 18

- SslTransportSecurity 15
- Streamed 14
- Streamed Message Framing Policy Assertion 14
- UDP Retransmission-Enabled Policy Assertion 17
- WebSocket Streamed Policy Assertion 17
- WebSocket Streamed Request Policy Assertion 18
- WebSocket Streamed Response Policy Assertion 18
- WindowsTransportSecurity 15
- Examples - overview 22

F

- Fields - vendor-extensible 11
- Full WSDL 24
 - Basic HTTP Authentication Policy Assertion 24
 - Binary Encoding Policy Assertion 26
 - Composite Duplex Policy Assertion 28
 - Digest HTTP Authentication Policy Assertion 25
 - Is Initiating Policy Assertion 31
 - Is Terminating Policy Assertion 31
 - Message Framing Security Provider Negotiation Policy Assertion 28
 - Message Framing Transport Security Policy Assertion 27
 - Negotiate HTTP Authentication Policy Assertion 26
 - NTLM HTTP Authentication Policy Assertion 25
 - One-way and Composite Duplex Policy Assertions 28
 - One-way Policy Assertion 28
 - overview 24
 - SOAP-over-UDP Transport URI 30
 - Streamed Message Framing Policy Assertion 26
 - UDP Retransmission-Enabled Policy Assertion 29
 - Using Session - Is Initiating - and Is Terminating WSDL Extensions 31
 - Using Session Policy Assertion 31
 - WebSocket Streamed Policy Assertion 29
 - WebSocket Streamed Request Policy Assertion 30
 - WebSocket Streamed Response Policy Assertion 30

G

- Glossary 6
- Groups 20

I

- Implementer - security considerations 23
- Index of security parameters 23
- Informative references 8
- Introduction 6
- Is Initiating WSDL Extension attribute 19
- Is Terminating WSDL Extension attribute 19
- isInitiating attribute 19
- isTerminating attribute 19

M

- Message Framing Security Provider Negotiation policy assertion 15
- Message Framing Security Provider Negotiation Policy Assertion element 15
- Message Framing Transport Security policy assertion 15
- Message Framing Transport Security Policy Assertion element 15
- Messages
 - attribute groups 20
 - attributes 18
 - Basic HTTP Authentication policy assertion 13
 - Basic HTTP Authentication Policy Assertion element 13
 - BasicAuthentication element 13
 - Binary Encoding policy assertion 15

- Binary Encoding Policy Assertion element 15
- BinaryEncoding element 15
- complex types 18
- Composite Duplex policy assertion 17
- Composite Duplex Policy Assertion element 17
- CompositeDuplex element 17
- Digest HTTP Authentication policy assertion 14
- Digest HTTP Authentication Policy Assertion element 14
- DigestAuthentication element 14
- elements 12
- enumerated 12
- groups 20
- Is Initiating WSDL Extension attribute 19
- Is Terminating WSDL Extension attribute 19
- isInitiating attribute 19
- isTerminating attribute 19
- Message Framing Security Provider Negotiation policy assertion 15
- Message Framing Security Provider Negotiation Policy Assertion element 15
- Message Framing Transport Security policy assertion 15
- Message Framing Transport Security Policy Assertion element 15
- namespaces 12
- Negotiate HTTP Authentication policy assertion 14
- Negotiate HTTP Authentication Policy Assertion element 14
- NegotiateAuthentication element 14
- NTLM HTTP Authentication policy assertion 14
- NTLM HTTP Authentication Policy Assertion element 14
- NtlmAuthentication element 14
- OneWay element 16
- One-way policy assertion 16
- One-way Policy Assertion element 16
- overview 12
- simple types 18
- SOAP-over-UDP SOAP Binding Transport URI element 18
- SslTransportSecurity element 15
- Streamed element 14
- Streamed Message Framing policy assertion 14
- Streamed Message Framing Policy Assertion element 14
- syntax 12
- transport 12
- UDP Retransmission-Enabled Policy Assertion element 17
- Using Session WSDL Extension attribute 19
- usingSession attribute 19
- WebSocket Streamed Policy Assertion element 17
- WebSocket Streamed Request Policy Assertion element 18
- WebSocket Streamed Response Policy Assertion element 18
- WindowsTransportSecurity element 15

N

- Namespaces 12
- Negotiate HTTP Authentication policy assertion 14
- Negotiate HTTP Authentication Policy Assertion element 14
- NegotiateAuthentication element 14
- Normative references 7
- NTLM HTTP Authentication policy assertion 14
- NTLM HTTP Authentication Policy Assertion element 14
- NtlmAuthentication element 14

O

- OneWay element 16
- One-way policy assertion 16
- One-way Policy Assertion element 16
- Overview 8
- Overview (synopsis) 8

P

- Parameters - security index 23
- Policy assertion
 - Basic HTTP Authentication 13
 - Binary Encoding 15
 - Composite Duplex 17
 - Digest HTTP Authentication 14
 - Message Framing Security Provider Negotiation 15
 - Message Framing Transport Security 15
 - Negotiate HTTP Authentication 14
 - NTLM HTTP Authentication 14
 - One-way 16
 - Streamed Message Framing 14
- Preconditions 10
- Prerequisites 10
- Product behavior 32
- Protocol Details
 - overview 21

R

- References 7
 - informative 8
 - normative 7
- Relationship to other protocols 10

S

- Security
 - implementer considerations 23
 - parameter index 23
- Security - implementer considerations 23
- Security - implementer considerations 23
- Server - details 21
- Simple types 18
- SOAP-over-UDP SOAP Binding Transport URI element 18
- SslTransportSecurity element 15
- Standards assignments 11
- Streamed element 14
- Streamed Message Framing policy assertion 14
- Streamed Message Framing Policy Assertion element 14
- Syntax 12

T

- Tracking changes 33
- Transport 12
- Types
 - complex 18
 - simple 18

U

- UDP Retransmission-Enabled Policy Assertion element 17
- Using Session WSDL Extension attribute 19
- usingSession attribute 19

V

- Vendor-extensible fields 11
- Versioning 10

W

- WebSocket Streamed Policy Assertion element 17
- WebSocket Streamed Request Policy Assertion element 18
- WebSocket Streamed Response Policy Assertion element 18
- WindowsTransportSecurity element 15
- WSDL 24
 - Basic HTTP Authentication Policy Assertion 24
 - Binary Encoding Policy Assertion 26
 - Composite Duplex Policy Assertion 28
 - Digest HTTP Authentication Policy Assertion 25
 - Is Initiating Policy Assertion 31
 - Is Terminating Policy Assertion 31
 - Message Framing Security Provider Negotiation Policy Assertion 28
 - Message Framing Transport Security Policy Assertion 27
 - Negotiate HTTP Authentication Policy Assertion 26
 - NTLM HTTP Authentication Policy Assertion 25
 - One-way and Composite Duplex Policy Assertions 28
 - One-way Policy Assertion 28
 - overview 24
 - SOAP-over-UDP Transport URI 30
 - Streamed Message Framing Policy Assertion 26
 - UDP Retransmission-Enabled Policy Assertion 29
 - Using Session - Is Initiating - and Is Terminating WSDL Extensions 31
 - Using Session Policy Assertion 31
 - WebSocket Streamed Policy Assertion 29
 - WebSocket Streamed Request Policy Assertion 30
 - WebSocket Streamed Response Policy Assertion 30