WS-Policy Interop Scenarios

27 January 2008 14:14

Version 1.5

Editors

Asir S Vedamuthu, Microsoft Corporation
David Orchard, BEA Systems, Inc.
Frederick Hirsch, Nokia
Maryann Hondo, IBM Corporation
Prasad Yendluri, webMethods, Inc.
Toufic Boubez, Layer 7 Technologies

Abstract

This document describes the scenarios and test cases for WS-Policy Framework and WS-PolicyAttachment.

Status

This document is an editors' copy that has no official standing.

Table of Contents

1. Introduction

This specification describes two scenarios and four test cases for WS-Policy and WS-PolicyAttachment. To ensure a shared understanding of the basic interoperability, this specification bundles Round 1 and 2 test cases.

This specification is intended to help implementers to write WS-Policy processors that comply with WS-Policy and WS-PolicyAttachment specifications, and interoperate with other WS-Policy processors that comply with WS-Policy and WS-PolicyAttachment specifications.

Interop scenarios have two participants: a requestor (R) and a service end point (S). S exposes the conditions on interaction between R and S using Policy Expressions attached to Policy Subjects that have WSDL descriptions. R uses one of the Policy Alternatives to interact with S.

There are two scenarios and two test cases for each of them:

- Transport Security Policy – related test cases are numbered Tx
- X509 Security Policy – related test cases are numbered Axx

The chosen interop scenarios exercise substantial parts of the Policy Framework and Policy Attachment for WSDL in the context of a security policy domain [WS-SecurityPolicy, WS-SecurityPolicyInterop Scenarios].
To ensure a shared understanding of the basic interoperability of the domain independent parts of the framework,

- Appendix C – describes Round 1 test cases for testing policy normalize, merge and intersect operations and,
- Appendix E – describes Round 2 test cases for testing effective policy computations for policies attached to WSDL.

2. Notation

2.1 XML Namespaces

Table 1 lists XML namespaces that are used in this specification. The choice of any namespace prefix is arbitrary and not semantically significant.

Table 1: Prefixes and XML Namespaces used in this specification.

Prefix	XML Namespace	Specification(s)
s11	http://schemas.xmlsoap.org/soap/envelope/	[SOAP 1.1]
s12	http://www.w3.org/2003/05/soap-envelope	[SOAP 1.2]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL 1.1]
wsdl20	http://www.w3.org/ns/wsdl	[WSDL 2.0 Core]
wsoap	http://www.w3.org/ns/wsdl/soap	[WSDL 2.0 Adjuncts]
wsoap12	http://schemas.xmlsoap.org/wsdl/soap12/	[WSDL 1.1, SOAP 1.2]
wsp	http://www.w3.org/ns/ws-policy	[WS-Policy, WS-PolicyAttachment]
sp	http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702	[WS-SecurityPolicy]

wsse	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd	[WSS10]
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd	[WSS10]
x	http://example.com/ws/2004/09/policy	This specification

2.2 Notational Conventions

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

3. Round 3

3.1 Assumptions

- S exposes the conditions on interaction between R and S using Policy Expressions attached to Policy Subjects that have WSDL descriptions (WSDL 1.1 or 2.0).
- R has acquired these WSDL descriptions. How R got these WSDL descriptions is out of scope for the interop scenarios.
- S implements the two scenarios and six test cases in this specification.
- SOAP 1.2 is the messaging protocol for interactions between R and S.
- HTTP is the underlying transport protocol for interactions between R and S.
- S may be implemented as one or more endpoints. Each endpoint may implement one or more test cases.
- R has an X509 certificate for S. How R got this certificate is out of scope for the interop scenarios.
- R transmits a simple echo request and S responds with a simple echo response.
- For convenience, to get started with interop testing, S exposes a simple endpoint that does not use any of these security features.

3.2 Round 3 Scenario - Transport Security Policy

3.2.1 Description

This scenario illustrates the use of a transport binding security policy assertion [WS-SecurityPolicy]. S has defined two alternatives [Appendix B- Round3 WSDL]. R chooses one of the policy alternatives that contains transport security policy assertion, constructs a message and transmits to S. S receives the message, validates and transmits a response message. R receives the response message and validates that the response payload is the same as the original payload sent.

3.2.1.1 Test Case T1

Test case T1 uses no client certificate, includes a timestamp and uses no supporting user name token. The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 1; and, the Policy Expression is attached to WSDL Binding.

Policy Expression 1

```
<wsp:Policy wsu:Id="T1Endpoint" >
  <!-- Policy alternative T1 - Anonymous client -->
  <sp:TransportBinding>
    <wsp:Policy>
      <sp:TransportToken>
        <wsp:Policy>
          <sp:HttpsToken >
            <wsp:Policy />
          </sp:HttpsToken>
        </wsp:Policy>
      </sp:TransportToken>
      <sp:AlgorithmSuite>
        <wsp:Policy>
          <sp:Basic256Rsa15 />
        </wsp:Policy>
      </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
      <sp:IncludeTimestamp />
    </wsp:Policy>
  </sp:TransportBinding>
</wsp:Policy>
```

3.2.1.2 Test Case T3

Test case T3 uses no client certificate, includes a timestamp and uses a supporting user name token. Please refer to Appendix A – below for user name tokens. The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 2; and, the Policy Expression is attached to WSDL Binding.

Policy Expression 2

```
<wsp:Policy wsu:Id="T3Endpoint" >
  <!-- Policy alternative T3 - Authenticated client with
username/password token -->
  <sp:TransportBinding>
    <wsp:Policy>
      <sp:TransportToken>
        <wsp:Policy>
          <sp:HttpsToken >
            <wsp:Policy />
          </sp:HttpsToken>
        </wsp:Policy>
      </sp:TransportToken>
      <sp:AlgorithmSuite>
```

```
        <wsp:Policy>
          <sp:Basic256Rsa15 />
        </wsp:Policy>
      </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
      <sp:IncludeTimestamp />
    </wsp:Policy>
  </sp:TransportBinding>
  <sp:SignedSupportingTokens>
    <wsp:Policy>
      <sp:UsernameToken sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient" >
        <wsp:Policy>
          <sp:WssUsernameToken10 />
        </wsp:Policy>
      </sp:UsernameToken>
    </wsp:Policy>
  </sp:SignedSupportingTokens>
  <sp:Wss10>
    <wsp:Policy>
      <sp:MustSupportRefKeyIdentifier />
      <sp:MustSupportRefIssuerSerial />
    </wsp:Policy>
  </sp:Wss10>
</wsp:Policy>
```

3.2.1.3 Sample Messages

The following is a sample request message for test case T1:

```
<s:Envelope xmlns:s=" http://www.w3.org/2003/05/soap-envelope "
   xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <s:Header>
    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2006-01-19T02:49:53.914Z</u:Created>
        <u:Expires>2006-01-19T02:54:53.914Z</u:Expires>
      </u:Timestamp>
    </o:Security>
  </s:Header>
  <s:Body>
```

```
    <EchoRequest xmlns="http://example.com/ws/2004/09/policy">Hello
World! (T1)</EchoRequest>
  </s:Body>
</s:Envelope>
```

The following is a sample request message for test case T3:

```
<s:Envelope xmlns:s=" http://www.w3.org/2003/05/soap-envelope "
   xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <s:Header>
    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="_0">
        <u:Created>2006-01-19T02:49:56.368Z</u:Created>
        <u:Expires>2006-01-19T02:54:56.368Z</u:Expires>
      </u:Timestamp>
      <o:UsernameToken>
        <o:Username>Alice</o:Username>
        <o:Password>ecilA</o:Password>
      </o:UsernameToken>
    </o:Security>
  </s:Header>
  <s:Body>
    <EchoResponse xmlns="http://example.com/ws/2004/09/policy">Hello
World! (T3)</EchoResponse>
  </s:Body>
</s:Envelope>
```

3.3 Round 3 Scenario - X509 Security Policy

3.3.1 Description

This scenario illustrates the use of an asymmetric binding security policy assertion. The AsymmetricBinding assertion as described in the Security Policy document, is intended to be used in scenarios in which message protection is provided by means defined in WSS: SOAP Message Security.

This binding has two binding specific token properties; [Initiator Token] and [Recipient Token]. If the message pattern requires multiple messages, this binding defines that the [Initiator Token] is used for the message signature from initiator to the recipient, and for encryption from recipient to initiator. The [Recipient Token] is used for encryption from initiator to recipient, and for the message signature from recipient to initiator. In our test case here, we use X509 V3 tokens for both initiator and recipient. [1]

R chooses one of the policy alternatives that contains an asymmetric security policy assertion, constructs a message and transmits to S. S receives the message,

[1] WS-SecurityPolicy

validates and transmits a response message. R receives the response message and validates that the response payload is the same as the original payload sent.

3.3.1.1 Test Case A11

Test case A11 illustrates an asymmetric binding with X509 Version 3 tokens as specified in [WSS: X509 Certificate Token Profile 1.0] for mutual authentication and AES 256 XML encryption method, signing of headers and body has been specified, and it includes a timestamp.

There are two Policy Expressions:

(1) The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 3; and, the Policy Expression is attached to WSDL Binding.

Policy Expression 3

```
<wsp:Policy wsu:Id="A11Endpoint" >
  <!-- Asymmetric Policy A11 - X509 with mutual authentication and AES 
256 -->
  <sp:AsymmetricBinding>
    <wsp:Policy>
      <sp:RecipientToken>
        <wsp:Policy>
          <sp:X509Token sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/Never" >
            <wsp:Policy>
              <sp:WssX509V3Token10 />
            </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:RecipientToken>
      <sp:InitiatorToken>
        <wsp:Policy>
          <sp:X509Token sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient" >
            <wsp:Policy>
              <sp:WssX509V3Token10 />
            </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:InitiatorToken>
      <sp:AlgorithmSuite>
        <wsp:Policy>
          <sp:Basic256Rsa15 />
        </wsp:Policy>
      </sp:AlgorithmSuite>
      <sp:Layout>
        <wsp:Policy>
          <sp:Lax />
        </wsp:Policy>
      </sp:Layout>
      <sp:IncludeTimestamp />
      <sp:OnlySignEntireHeadersAndBody />
    </wsp:Policy>
  </sp:AsymmetricBinding>
  <sp:Wss10>
    <wsp:Policy>
      <sp:MustSupportRefKeyIdentifier />
      <sp:MustSupportRefIssuerSerial />
    </wsp:Policy>
  </sp:Wss10>
</wsp:Policy>
```

(2) The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 4; and, the Policy Expression is attached to WSDL Message Policy Subjects – input and output.

Policy Expression 4

```
<wsp:Policy wsu:Id="AMessage" >
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts>
        <sp:Body />
      </sp:SignedParts>
      <sp:EncryptedParts>
        <sp:Body />
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```

3.3.1.2 Test Case A12

Test case A11 uses X509 Version 3 token as specified in [WSS: X509 Certificate Token Profile 1.0] is the same as A11 but with TRIPLEDES specified as the algorithm. .

There are two Policy Expressions:

(1) The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 5; and the Policy Expression is attached to WSDL Binding,

Policy Expression 5

```
<wsp:Policy wsu:Id="A12Endpoint" >
  <!-- Asymmetric Policy A12 - X509 with mutual authentication and 3DES
-->
  <sp:AsymmetricBinding>
    <wsp:Policy>
      <sp:RecipientToken>
```

```
        <wsp:Policy>
          <sp:X509Token sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/Never" >
            <wsp:Policy>
              <sp:WssX509V3Token10 />
            </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:RecipientToken>
      <sp:InitiatorToken>
        <wsp:Policy>
          <sp:X509Token sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient" >
            <wsp:Policy>
              <sp:WssX509V3Token10 />
            </wsp:Policy>
          </sp:X509Token>
        </wsp:Policy>
      </sp:InitiatorToken>
      <sp:AlgorithmSuite>
        <wsp:Policy>
          <sp:TripleDesRsa15 />
        </wsp:Policy>
      </sp:AlgorithmSuite>
      <sp:Layout>
        <wsp:Policy>
          <sp:Lax />
        </wsp:Policy>
      </sp:Layout>
      <sp:IncludeTimestamp />
      <sp:OnlySignEntireHeadersAndBody />
    </wsp:Policy>
  </sp:AsymmetricBinding>
  <sp:Wss10>
    <wsp:Policy>
      <sp:MustSupportRefKeyIdentifier />
      <sp:MustSupportRefIssuerSerial />
    </wsp:Policy>
  </sp:Wss10>
</wsp:Policy>
```

(2) The normal form of the Policy Expression MUST be equivalent to the normal form of Policy Expression 4; and, the Policy Expression is attached to WSDL Message Policy Subjects – input and output.

3.3.1.3 Sample Message

The following is a sample request message for test case A11:

```
<s:Envelope xmlns:s=" http://www.w3.org/2003/05/soap-envelope "
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <s:Header>
    <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <u:Timestamp u:Id="uuid-328a1cc5-304d-4c6d-a3e0-d5379e6747f8-1">
        <u:Created>2006-01-18T17:51:33.412Z</u:Created>
        <u:Expires>2006-01-18T17:56:33.412Z</u:Expires>
      </u:Timestamp>
      <o:BinarySecurityToken u:Id="uuid-a2a99bec-5688-468d-85df-
37f1350039a6-1" 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-
soap-message-security-
1.0#Base64Binary">MIIDDDCCAfSgAwIBAgIQM6YEf7FVYx/tZyEXgVComTANBgkqhkiG9
w0BAQUFADAwMQ4wDAYDVQQKDAVPQVNJUzEeMBwGA1UEAwwVT0FTSVMgSW50ZXJvcCBUZXN0
IENBMB4XDTA1MDMxOTAwMDAwMFoXDTE4MDMxOTIzNTk1OVowQjEOMAwGA1UECgwFT0FTSVM
xIDAeBgNVBAsMF09BU0lTIEludGVyb3AgVGVzdCBDZXJ0MQ4wDAYDVQQDDAVBbGljZTCBnz
ANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAoqi99By1VYo0aHrkKCNT4DkIgPL/SgahbeKdG
hrbu3K2XG7arfD9tqIBIKMfrX4Gp90NJa85AV1yiNsEyvq+mUnMpNcKnLXLOjkTmMCqDYbb
kehJlXPnaWLzve+mW0pJdPxtf3rbD4PS/cBQIvtpjmrDAU8VsZKT8DN5Kyz+EZsCAwEAAaO
BkzCBkDAJBgNVHRMEAjAAMDMGA1UdHwQsMCowKKImhiRodHRwOi8vaW50ZXJvcC5iYnRlc3
QubmV0L2NybC9jYS5jcmwwDgYDVR0PAQH/BAQDAgSwMB0GA1UdDgQWBBQK4l0TUHZ1QV3V2
QtlLNDm+PoxiDAfBgNVHSMEGDAWgBTAnSj8wes1oR3WqqqgHBpNwkkPDzANBgkqhkiG9w0B
AQUFAAOCAQEABTqpOpvW+6yrLXyUlP2xJbEkohXHI5OWwKWleOb9hlkhWntUalfcFOJAgUy
H30TTpHldzx1+vK2LPzhoUFKYHE1IyQvokBN2JjFO64BQukCKnZhldLRPxGhfkTdxQgdf5r
CK/wh3xVsZCNTfuMNmlAM6lOAg8QduDah3WFZpEA0s2nwQaCNQTNMjJC8tav1CBr6+E5FAm
wPXP7pJxn9Fw9OXRyqbRA4v2y7YpbGkG2GI9UvOHw6SGvf4FRSthMMO35YbpikGsLix3vAs
XWWi4rwfVOYzQK0OFPNi9RMCUdSH06m9uLWckiCxjos0FQODZE9l4ATGy9s9hNVwryOJTw=
=</o:BinarySecurityToken>
      <e:EncryptedKey Id="_0"
xmlns:e="http://www.w3.org/2001/04/xmlenc#">
        <e:EncryptionMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <o:SecurityTokenReference>
            <o:KeyIdentifier ValueType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-
1.0#X509SubjectKeyIdentifier" EncodingType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-soap-message-security-
1.0#Base64Binary">Xeg55vRyK3ZhAEhEf+YT0z986L0=</o:KeyIdentifier>
          </o:SecurityTokenReference>
        </KeyInfo>
        <e:CipherData>

<e:CipherValue>aS6obCAnGkz+u/AXUCsiFjpTxcb10HvAHsPoLaJJgDKeCLs4/g4B1Ssu
kLPTnfR1A5TVUWyqf78Bv9cR4qfL5Wj5lOfkLZ9rr+KIADclpocy1AktWHboaJ1B+m4yi6C
aoJkE8yZAFIy+e9hJuSzeXQ3/53uekMrmLSRl8zYpBuo=</e:CipherValue>
        </e:CipherData>
        <e:ReferenceList>
          <e:DataReference URI="#_2"/>
        </e:ReferenceList>
      </e:EncryptedKey>
      <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
        <SignedInfo>
          <CanonicalizationMethod
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
          <SignatureMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
          <Reference URI="#_1">
            <Transforms>
              <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#"/>
            </Transforms>
            <DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <DigestValue>azfCbpY2wRg78KLYSnD/mjzCMCM=</DigestValue>
          </Reference>
          <Reference URI="#uuid-328a1cc5-304d-4c6d-a3e0-d5379e6747f8-
1">
            <Transforms>
              <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#"/>
            </Transforms>
            <DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <DigestValue>j4hXJ8mcQaWZVEkP4gzOHPlPQMs=</DigestValue>
          </Reference>
        </SignedInfo>

<SignatureValue>ladw8G5ToSccwF2Na/3fN77sy7INO4KJqoO5EmhX769iCDiqt4hp/fL
kFSY31POLu+NqSDxpgtr4bxvLyuo9k77RtppH5IIlxa+NOLZcBV8+9bmKw76ntpquTDPRnp
BFu7KIueEvzU7YgE9cXQeb27+T4C4eYDH6bUTnmX02Bck=</SignatureValue>
        <KeyInfo>
          <o:SecurityTokenReference>
            <o:Reference ValueType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"
URI="#uuid-a2a99bec-5688-468d-85df-37f1350039a6-1"/>
          </o:SecurityTokenReference>
        </KeyInfo>
      </Signature>
    </o:Security>
  </s:Header>
  <s:Body u:Id="_1" xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
    <e:EncryptedData Id="_2"
Type="http://www.w3.org/2001/04/xmlenc#Content"
xmlns:e="http://www.w3.org/2001/04/xmlenc#">
```

```
      <e:EncryptionMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#aes256-cbc"/>
      <e:CipherData>

<e:CipherValue>QzaE+MUuGKDE/bZJ3zXh+kO4vLbG5oNBeWL+xOJ34R+ldoWhsf3NypOu
bCsBYDPTeOb8cKJjQcE/vYDq8E0xaQXz7cWp1B14j3e7Mhf8hsqBItYrxb4PSFrHU1my5Ww
UUABS/LOaWnjps50N92tsYx9as8Cmvdx1fR9VIAvqpNo=</e:CipherValue>
      </e:CipherData>
    </e:EncryptedData>
  </s:Body>
</s:Envelope>
```

4. Acknowledgements

This specification has been developed as a result of joint work with many individuals and teams, including: Martin Gudgin (Microsoft), Kirill Gavrylyuk (Microsoft), Sara Wong (Microsoft), Francisco Curbera (IBM), Chris Sharp (IBM), Arthur Ryman (IBM), Bruce Rich, IBM

5. References

[RFC 2119]

S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, Harvard University, March 1997. (See http://www.ietf.org/rfc/rfc2119.txt.)

[SOAP 1.1]

D. Box, et al, "Simple Object Access Protocol (SOAP) 1.1," May 2000. (See http://www.w3.org/TR/2000/NOTE-SOAP-20000508.)

[SOAP 1.2]

M. Gudgin, et al, “SOAP Version 1.2 Part 1: Messaging Framework,” June 2003 (See http://www.w3.org/TR/2003/REC-soap12-part1-20030624/.)

[WSDL 1.1]

E. Christensen, et al, "Web Services Description Language (WSDL) 1.1," March 2001. (See http://www.w3.org/TR/2001/NOTE-wsdl-20010315.)

[WSDL 1.1, SOAP 1.2]

D. Angelov, et al, "WSDL 1.1 Binding Extension for SOAP 1.2," April 2006. (See http://www.w3.org/Submission/2006/SUBM-wsdl11soap12-20060405/.)

[WSDL 2.0 Core]

R. Chinnici, et al, "Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language" (See http://www.w3.org/TR/2006/CR-wsdl20-20060327/.)

[WSDL 2.0 Adjuncts]

R. Chinnici, et al, "Web Services Description Language (WSDL) Version 2.0 Part 2: Adjuncts" (See http://www.w3.org/TR/2006/CR-wsdl20-adjuncts-20060327/.)

[WS-Policy]

A. Vedamuthu, et al, "Web Services Policy 1.5 - Framework," Nov 2006. (See http://www.w3.org/TR/2006/WD-ws-policy-20061117.)

[WS-PolicyAttachment]

A. Vedamuthu, et al, "Web Services Policy 1.5 - Attachment," Nov 2006. (See http://www.w3.org/TR/2006/WD-ws-policy-attach-20061117.)

[WS-SecurityPolicy]
A. Nadalin, et al, “WS-SecurityPolicy 1.2,” July 2007. (See http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702/ws-securitypolicy-1.2-spec-os.pdf.)

[XML Schema, Part 1]
H. Thompson, et al, "XML Schema Part 1: Structures," October 2004. (See http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/.)

[XML Schema, Part 2]
P. Biron, et al, "XML Schema Part 2: Datatypes," October 2004. (See http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/.)

[WSS10]
A. Nadalin, et.al., "Web Services Security: SOAP Message Security 1.0,” March 2004 (See http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf.)

[WSS:UsernameToken Profile 1.0]
Nadalin, et.al., "Web Services Security: UsernameToken Profile 1.0," March 2004. (See http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0.pdf.)

Appendix A – Username and Password List

Username	*Password*	*E-mail Address*
Alice	ecilA	alice@fabrikam.com
Bob	boB	bob@fabrikam.com
Charlie	eilrahC	charlie@fabrikam.com
Dawn	nwaD	dawn@fabrikam.com
Evan	navE	evan@fabrikam.com

Appendix B – WSDL for Round 3 Scenarios

A non-normative copy of the WSDL description is listed below for convenience.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions
      targetNamespace="http://example.com/ws/2004/09/policy"
      xmlns:tns="http://example.com/ws/2004/09/policy"
      xmlns:sp="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702"
      xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:wsoap12="http://schemas.xmlsoap.org/wsdl/soap12/"
      xmlns:wsp="http://www.w3.org/ns/ws-policy"
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-utility-1.0.xsd" >
  <wsp:Policy wsu:Id="T1Endpoint" >
    <!-- Policy alternative T1 - Anonymous client -->
    <sp:TransportBinding>
```

```
      <wsp:Policy>
        <sp:TransportToken>
          <wsp:Policy>
            <sp:HttpsToken >
              <wsp:Policy />
            </sp:HttpsToken>
          </wsp:Policy>
        </sp:TransportToken>
        <sp:AlgorithmSuite>
          <wsp:Policy>
            <sp:Basic256Rsa15 />
          </wsp:Policy>
        </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
        <sp:IncludeTimestamp />
      </wsp:Policy>
    </sp:TransportBinding>
  </wsp:Policy>
  <wsp:Policy wsu:Id="T3Endpoint" >
    <!-- Policy alternative T3 - Authenticated client with
username/password token -->
    <sp:TransportBinding>
      <wsp:Policy>
        <sp:TransportToken>
          <wsp:Policy>
            <sp:HttpsToken >
              <wsp:Policy />
            </sp:HttpsToken>
          </wsp:Policy>
        </sp:TransportToken>
        <sp:AlgorithmSuite>
          <wsp:Policy>
            <sp:Basic256Rsa15 />
          </wsp:Policy>
        </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
        <sp:IncludeTimestamp />
      </wsp:Policy>
    </sp:TransportBinding>
    <sp:SignedSupportingTokens>
      <wsp:Policy>
        <sp:UsernameToken sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient"
>
        <wsp:Policy>
           <sp:WssUsernameToken10 />
          </wsp:Policy>
        </sp:UsernameToken>
      </wsp:Policy>
    </sp:SignedSupportingTokens>
    <sp:Wss10>
      <wsp:Policy>
        <sp:MustSupportRefKeyIdentifier />
        <sp:MustSupportRefIssuerSerial />
      </wsp:Policy>
    </sp:Wss10>
  </wsp:Policy>



  <wsp:Policy wsu:Id="A11Endpoint" >
    <!-- Asymmetric Policy A11 - X509 with mutual authentication and
AES 256 -->
    <sp:AsymmetricBinding>
      <wsp:Policy>
        <sp:RecipientToken>
          <wsp:Policy>
            <sp:X509Token sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/Never" >
              <wsp:Policy>
                <sp:WssX509V3Token10 />
              </wsp:Policy>
            </sp:X509Token>
          </wsp:Policy>
        </sp:RecipientToken>
        <sp:InitiatorToken>
          <wsp:Policy>
            <sp:X509Token sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient"
>
              <wsp:Policy>
                <sp:WssX509V3Token10 />
              </wsp:Policy>
            </sp:X509Token>
          </wsp:Policy>
        </sp:InitiatorToken>
        <sp:AlgorithmSuite>
          <wsp:Policy>
            <sp:Basic256Rsa15 />
          </wsp:Policy>
        </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
        <sp:IncludeTimestamp />
        <sp:OnlySignEntireHeadersAndBody />
      </wsp:Policy>
    </sp:AsymmetricBinding>
    <sp:Wss10>
      <wsp:Policy>
        <sp:MustSupportRefKeyIdentifier />
        <sp:MustSupportRefIssuerSerial />
      </wsp:Policy>
    </sp:Wss10>
  </wsp:Policy>
  <wsp:Policy wsu:Id="A12Endpoint" >
    <!-- Asymmetric Policy A12 - X509 with mutual authentication and
3DES -->
    <sp:AsymmetricBinding>
      <wsp:Policy>
        <sp:RecipientToken>
          <wsp:Policy>
            <sp:X509Token sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/Never" >
              <wsp:Policy>
                <sp:WssX509V3Token10 />
              </wsp:Policy>
            </sp:X509Token>
          </wsp:Policy>
        </sp:RecipientToken>
        <sp:InitiatorToken>
          <wsp:Policy>
            <sp:X509Token sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient"
>
              <wsp:Policy>
                <sp:WssX509V3Token10 />
              </wsp:Policy>
            </sp:X509Token>
          </wsp:Policy>
        </sp:InitiatorToken>
        <sp:AlgorithmSuite>
          <wsp:Policy>
            <sp:TripleDesRsa15 />
          </wsp:Policy>
        </sp:AlgorithmSuite>
        <sp:Layout>
          <wsp:Policy>
            <sp:Lax />
          </wsp:Policy>
        </sp:Layout>
        <sp:IncludeTimestamp />
        <sp:OnlySignEntireHeadersAndBody />
      </wsp:Policy>
    </sp:AsymmetricBinding>
    <sp:Wss10>
      <wsp:Policy>
        <sp:MustSupportRefKeyIdentifier />
        <sp:MustSupportRefIssuerSerial />
      </wsp:Policy>
    </sp:Wss10>
  </wsp:Policy>

  <wsp:Policy wsu:Id="AMessage" >
    <sp:SignedParts>
      <sp:Body />
    </sp:SignedParts>
    <sp:EncryptedParts>
      <sp:Body />
    </sp:EncryptedParts>
  </wsp:Policy>
  <wsdl:types>
    <xs:schema
      targetNamespace="http://example.com/ws/2004/09/policy"
            blockDefault="#all"
            elementFormDefault="qualified" >
      <xs:element name="EchoRequest" type="xs:string" />
      <xs:element name="EchoResponse" type="xs:string" />
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="EchoInMessage" >
    <wsdl:part name="Body" element="tns:EchoRequest" />
  </wsdl:message>
```

```
  <wsdl:message name="EchoOutMessage" >
    <wsdl:part name="Body" element="tns:EchoResponse" />
  </wsdl:message>
  <wsdl:portType name="Test" >
    <wsdl:operation name="Echo" >
      <wsdl:input message="tns:EchoInMessage" />
      <wsdl:output message="tns:EchoOutMessage" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="NoSecurityBinding" type="tns:Test" >
    <wsoap12:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="Echo" >
      <wsoap12:operation
soapAction="http://example.com/ws/2004/09/policy/Test/EchoRequest" />
      <wsdl:input>
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="T1Binding" type="tns:Test" >
    <wsoap12:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsp:PolicyReference URI="#T1Endpoint" wsdl:required="true" />
    <wsdl:operation name="Echo" >
      <wsoap12:operation
soapAction="http://example.com/ws/2004/09/policy/Test/EchoRequest" />
      <wsdl:input>
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="T3Binding" type="tns:Test" >
    <wsoap12:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsp:PolicyReference URI="#T3Endpoint" wsdl:required="true" />
    <wsdl:operation name="Echo" >
      <wsoap12:operation
soapAction="http://example.com/ws/2004/09/policy/Test/EchoRequest" />
      <wsdl:input>
```

```
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
    <wsdl:binding name="A11Binding" type="tns:Test" >
    <wsoap12:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsp:PolicyReference URI="#A11Endpoint" wsdl:required="true" />
    <wsdl:operation name="Echo" >
      <wsoap12:operation
soapAction="http://example.com/ws/2004/09/policy/Test/EchoRequest" />
      <wsdl:input>
        <wsoap12:body use="literal" />
        <wsp:PolicyReference URI="#AMessage" wsdl:required="true" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
        <wsp:PolicyReference URI="#AMessage" wsdl:required="true" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="A12Binding" type="tns:Test" >
    <wsoap12:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <wsp:PolicyReference URI="#A12Endpoint" wsdl:required="true" />
    <wsdl:operation name="Echo" >
      <wsoap12:operation
soapAction="http://example.com/ws/2004/09/policy/Test/EchoRequest" />
      <wsdl:input>
        <wsoap12:body use="literal" />
        <wsp:PolicyReference URI="#AMessage" wsdl:required="true" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
        <wsp:PolicyReference URI="#AMessage" wsdl:required="true" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>

  <wsdl:service name="Test" >
    <wsdl:port name="NoSecurityPort" binding="tns:NoSecurityBinding" >
      <wsoap12:address location="TBD" />
    </wsdl:port>
    <wsdl:port name="T1Port" binding="tns:T1Binding" >
      <wsoap12:address location="TBD" />
    </wsdl:port>
    <wsdl:port name="T3Port" binding="tns:T3Binding" >
      <wsoap12:address location="TBD" />
    </wsdl:port>
    <wsdl:port name="A11Port" binding="tns:A11Binding" >
      <wsoap12:address location="TBD" />
    </wsdl:port>
    <wsdl:port name="A12Port" binding="tns:A12Binding" >
      <wsoap12:address location="TBD" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

A non-normative copy of WSDL 2.0 description is listed below for your convenience.

```
<?xml version="1.0" encoding="UTF-8"?>

<wsdl20:description
targetNamespace="http://example.com/ws/2004/09/policy"
    xmlns:tns="http://example.com/ws/2004/09/policy"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:wsdl20="http://www.w3.org/ns/wsdl"
xmlns:wsoap="http://www.w3.org/ns/wsdl/soap"
    xmlns:wsp="http://www.w3.org/ns/ws-policy"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    xmlns:sp="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

    <wsp:Policy wsu:Id="T1Endpoint">
        <!-- Policy alternative T1 - Anonymous client -->
        <sp:TransportBinding>
            <wsp:Policy>
                <sp:TransportToken>
                    <wsp:Policy>
                        <sp:HttpsToken>
                            <wsp:Policy />
                        </sp:HttpsToken>
                    </wsp:Policy>
                </sp:TransportToken>
                <sp:AlgorithmSuite>
                    <wsp:Policy>
                        <sp:Basic256Rsa15/>
                    </wsp:Policy>
                </sp:AlgorithmSuite>
                <sp:Layout>
                    <wsp:Policy>
                        <sp:Lax/>
                    </wsp:Policy>
                </sp:Layout>
                <sp:IncludeTimestamp/>
            </wsp:Policy>
        </sp:TransportBinding>
    </wsp:Policy>
    <wsp:Policy wsu:Id="T3Endpoint">
        <!-- Policy alternative T3 - Authenticated client with
username/password token -->
        <sp:TransportBinding>
            <wsp:Policy>
                <sp:TransportToken>
                    <wsp:Policy>
                        <sp:HttpsToken>
                            <wsp:Policy />
                        </sp:HttpsToken>
                    </wsp:Policy>
                </sp:TransportToken>
                <sp:AlgorithmSuite>
                    <wsp:Policy>
                        <sp:Basic256Rsa15/>
                    </wsp:Policy>
                </sp:AlgorithmSuite>
                <sp:Layout>
                    <wsp:Policy>
                        <sp:Lax/>
                    </wsp:Policy>
                </sp:Layout>
                <sp:IncludeTimestamp/>
            </wsp:Policy>
        </sp:TransportBinding>
        <sp:SignedSupportingTokens>
            <wsp:Policy>
                <sp:UsernameToken
                    sp:IncludeToken="http://docs.oasis-open.org/ws-
sx/ws-securitypolicy/200702/IncludeToken/AlwaysToRecipient">
                    <wsp:Policy>
                        <sp:WssUsernameToken10/>
                    </wsp:Policy>
                </sp:UsernameToken>
            </wsp:Policy>
        </sp:SignedSupportingTokens>
        <sp:Wss10>
            <wsp:Policy>
                <sp:MustSupportRefKeyIdentifier/>
                <sp:MustSupportRefIssuerSerial/>
            </wsp:Policy>
        </sp:Wss10>
    </wsp:Policy>
    <wsp:Policy wsu:Id="A11Endpoint">
        <!-- Asymmetric Policy A11 - X509 with mutual authentication
and AES 256 -->
        <sp:AsymmetricBinding>
            <wsp:Policy>
                <sp:RecipientToken>
                    <wsp:Policy>
                        <sp:X509Token
                            sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/Never">
                            <wsp:Policy>
                                <sp:WssX509V3Token10/>
                            </wsp:Policy>
                        </sp:X509Token>
                    </wsp:Policy>
                </sp:RecipientToken>
                <sp:InitiatorToken>
                    <wsp:Policy>
                        <sp:X509Token
                            sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-
securitypolicy/200702/IncludeToken/AlwaysToRecipient">
                            <wsp:Policy>
                                <sp:WssX509V3Token10/>
                            </wsp:Policy>
                        </sp:X509Token>
                    </wsp:Policy>
                </sp:InitiatorToken>
                <sp:AlgorithmSuite>
                    <wsp:Policy>
                        <sp:Basic256Rsa15/>
                    </wsp:Policy>
                </sp:AlgorithmSuite>
                <sp:Layout>
                    <wsp:Policy>
                        <sp:Lax/>
                    </wsp:Policy>
                </sp:Layout>
                <sp:IncludeTimestamp/>
                <sp:OnlySignEntireHeadersAndBody/>
            </wsp:Policy>
        </sp:AsymmetricBinding>
        <sp:Wss10>
            <wsp:Policy>
                <sp:MustSupportRefKeyIdentifier/>
                <sp:MustSupportRefIssuerSerial/>
            </wsp:Policy>
        </sp:Wss10>
    </wsp:Policy>
    <wsp:Policy wsu:Id="A12Endpoint">
        <!-- Asymmetric Policy A12 - X509 with mutual authentication
and 3DES -->
        <sp:AsymmetricBinding>
            <wsp:Policy>
                <sp:RecipientToken>
                    <wsp:Policy>
                        <sp:X509Token
                            sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/IncludeToken/Never">
                            <wsp:Policy>
                                <sp:WssX509V3Token10/>
                            </wsp:Policy>
                        </sp:X509Token>
                    </wsp:Policy>
                </sp:RecipientToken>
                <sp:InitiatorToken>
                    <wsp:Policy>
                        <sp:X509Token
                            sp:IncludeToken="http://docs.oasis-
open.org/ws-sx/ws-securitypolicy/200702/AlwaysToRecipient">
                            <wsp:Policy>
                                <sp:WssX509V3Token10/>
                            </wsp:Policy>
                        </sp:X509Token>
                    </wsp:Policy>
                </sp:InitiatorToken>
                <sp:AlgorithmSuite>
                    <wsp:Policy>
                        <sp:TripleDesRsa15/>
                    </wsp:Policy>
                </sp:AlgorithmSuite>
                <sp:Layout>
                    <wsp:Policy>
                        <sp:Lax/>
                    </wsp:Policy>
                </sp:Layout>
                <sp:IncludeTimestamp/>
                <sp:OnlySignEntireHeadersAndBody/>
            </wsp:Policy>
        </sp:AsymmetricBinding>
        <sp:Wss10>
            <wsp:Policy>
                <sp:MustSupportRefKeyIdentifier/>
                <sp:MustSupportRefIssuerSerial/>
            </wsp:Policy>
        </sp:Wss10>
    </wsp:Policy>
    <wsp:Policy wsu:Id="AMessage">
        <sp:SignedParts>
            <sp:Body/>
        </sp:SignedParts>
        <sp:EncryptedParts>
            <sp:Body/>
        </sp:EncryptedParts>
    </wsp:Policy>

    <wsdl20:types>
        <xs:schema
targetNamespace="http://example.com/ws/2004/09/policy"
blockDefault="#all"
            elementFormDefault="qualified">
            <xs:element name="EchoRequest" type="xs:string"/>
            <xs:element name="EchoResponse" type="xs:string"/>
        </xs:schema>
    </wsdl20:types>

    <wsdl20:interface name="Test">
        <wsdl20:operation name="Echo"
pattern="http://www.w3.org/ns/wsdl/in-out">
            <wsdl20:input element="tns:EchoRequest" />
            <wsdl20:output element="tns:EchoResponse" />
        </wsdl20:operation>
    </wsdl20:interface>

    <wsdl20:binding name="NoSecurityBinding" interface="tns:Test"
        type="http://www.w3.org/ns/wsdl/soap"
        wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP/">
        <wsdl20:operation ref="tns:Echo"
wsoap:action="http://example.com/ws/2004/09/policy/Test/EchoRequest">
            <wsdl20:input> </wsdl20:input>
            <wsdl20:output> </wsdl20:output>
        </wsdl20:operation>
    </wsdl20:binding>

    <wsdl20:binding name="T1Binding" interface="tns:Test"
type="http://www.w3.org/ns/wsdl/soap"
        wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP/">
        <wsp:PolicyReference URI="#T1Endpoint" wsdl20:required="true"/>
        <wsdl20:operation ref="tns:Echo"
wsoap:action="http://example.com/ws/2004/09/policy/Test/EchoRequest">
            <wsdl20:input> </wsdl20:input>
            <wsdl20:output> </wsdl20:output>
        </wsdl20:operation>
    </wsdl20:binding>

    <wsdl20:binding name="T3Binding" interface="tns:Test"
type="http://www.w3.org/ns/wsdl/soap"
        wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP/">
        <wsp:PolicyReference URI="#T3Endpoint" wsdl20:required="true"/>
        <wsdl20:operation ref="tns:Echo"
wsoap:action="http://example.com/ws/2004/09/policy/Test/EchoRequest">
            <wsdl20:input> </wsdl20:input>
            <wsdl20:output> </wsdl20:output>
        </wsdl20:operation>
    </wsdl20:binding>

    <wsdl20:binding name="A11Binding" interface="tns:Test"
        type="http://www.w3.org/ns/wsdl/soap"
        wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP/">
        <wsp:PolicyReference URI="#A11Endpoint"
wsdl20:required="true"/>
        <wsdl20:operation ref="tns:Echo"
wsoap:action="http://example.com/ws/2004/09/policy/Test/EchoRequest">
            <wsdl20:input>
                <wsp:PolicyReference URI="#AMessage"
wsdl20:required="true"/>
            </wsdl20:input>
            <wsdl20:output>
                <wsp:PolicyReference URI="#AMessage"
wsdl20:required="true"/>
            </wsdl20:output>
        </wsdl20:operation>
    </wsdl20:binding>

    <wsdl20:binding name="A12Binding" interface="tns:Test"
        type="http://www.w3.org/ns/wsdl/soap"
        wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/HTTP/">
        <wsp:PolicyReference URI="#A12Endpoint"
wsdl20:required="true"/>
        <wsdl20:operation ref="tns:Echo"
wsoap:action="http://example.com/ws/2004/09/policy/Test/EchoRequest">
            <wsdl20:input>
                <wsp:PolicyReference URI="#AMessage"
wsdl20:required="true"/>
            </wsdl20:input>
            <wsdl20:output>
                <wsp:PolicyReference URI="#AMessage"
wsdl20:required="true"/>
            </wsdl20:output>
        </wsdl20:operation>
    </wsdl20:binding>

    <wsdl20:service name="Test" interface="tns:Test">
        <wsdl20:endpoint name="NoSecurityPort"
binding="tns:NoSecurityBinding" address="TBD"> </wsdl20:endpoint>
        <wsdl20:endpoint name="T1Port" binding="tns:T1Binding"
address="TBD"> </wsdl20:endpoint>
        <wsdl20:endpoint name="T3Port" binding="tns:T3Binding"
address="TBD"> </wsdl20:endpoint>
        <wsdl20:endpoint name="A11Port" binding="tns:A11Binding"
address="TBD"> </wsdl20:endpoint>
        <wsdl20:endpoint name="A12Port" binding="tns:A12Binding"
address="TBD"> </wsdl20:endpoint>
    </wsdl20:service>

</wsdl20:description>
```

Appendix C – Round 1 Unit Test Cases for Normalize, Merge and Intersect

Although this specification focuses on two WS-Security Policy Interop Scenarios, to ensure a shared understanding of the basic interoperability, this section describes Round 1 test cases for testing policy normalize, merge and intersect operations.

Input and expected output data for Round 1 test cases are bundled with this specification. These are WS-Policy unit test cases. These are not interop test cases. Implementers may wrap these tests as JUnit / NUnit / VSTS implementations, Web service endpoints, command line executables, user interface code, etc. This section describes Round 1 test cases using a simple Web service endpoint wrapper. Appendix D provides a WSDL 1.1 description for the Web service endpoint wrapper.

This section describes the sequence of messages for testing policy operations: normalize, merge and intersect. Base test cases and expected results are bundled with this specification.

Assumptions

- S exposes normalize, merge and intersect as web service operations. Appendix D – contains the WSDL for these operations.
- SOAP 1.2 is the messaging protocol for interactions between R and S.
- HTTP is the underlying transport protocol for interactions between R and S.
- R transmits one or more policy expressions and S responds with a policy expression in normal form.

Normalize

The normalize web service operation accepts a single policy expression and returns a single policy expression in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/Normalize/Request
```

The following describes a request message to normalize operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp=”http://www.w3.org/ns/ws-policy”
    xmlns:x=>
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from normalize operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

Test cases for normalize and expected policy expressions in normal form are organized and bundled with the specification as follows,

```
Round1/Policy1.xml ← test cases
....
Round1/Policy20.xml
Round1/Normalized/Policy1.xml <- expected policy expression in normal
form
....
Round1/Normalized/Policy20.xml
```

Merge

The merge web service operation accepts one or more policy expressions and returns a single policy expression that is the combination of input policy expressions in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/Merge/Request
```

The following describes a request message to merge operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy" >
  <s12:Body>
    (<wsp:Policy> ... </wsp:Policy>)+
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from merge operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

Test cases for merge and expected merged policy expressions are organized and bundled with the specification as follows,

```
Round1/Policy21.xml ← test cases
....
Round1/Policy25.xml
Round1/Merged/Policy21-21.xml <- expected policy expression in normal form
Round1/Merged/Policy21-22.xml
....
Round1/Merged/Policy25-24.xml
Round1/Merged/Policy25-25.xml
```

Intersect

The intersect web service operation accepts one or more policy expressions and returns a single policy expression that is the intersection of input policy expressions in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/Intersect/Request
```

The following describes a request message to intersect operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy"     xmlns:>
  <s12:Body>
    (<wsp:Policy> ... </wsp:Policy>)+
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from intersect operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

Test cases for intersect and expected intersect policy expressions are organized and bundled with the specification as follows,

```
Round1/Policy21.xml ← test cases
....
Round1/Policy25.xml
Round1/Intersected/Policy21-21.xml <- expected policy expression in normal form
Round1/Intersected/Policy21-22.xml
....
Round1/Intersected/Policy25-24.xml
Round1/Intersected/Policy25-25.xml
```

Appendix D – WSDL for Round 1 - Normalize, Merge and Intersect

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:wsoap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://example.com/ws/2004/09/policy"
  targetNamespace="http://example.com/ws/2004/09/policy"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema elementFormDefault="qualified"
      targetNamespace="http://example.com/ws/2004/09/policy">
      <xs:complexType name="any" mixed="true">
        <xs:sequence>
```

```
          <xs:any minOccurs="0" maxOccurs="unbounded" />
        </xs:sequence>
      </xs:complexType>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="NormalizeSoapIn">
    <wsdl:part name="policy" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="NormalizeSoapOut">
    <wsdl:part name="NormalizeResult" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="IntersectSoapIn">
    <wsdl:part name="policies" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="IntersectSoapOut">
    <wsdl:part name="IntersectResult" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="MergeSoapIn">
    <wsdl:part name="policies" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="MergeSoapOut">
    <wsdl:part name="MergeResult" type="tns:any" />
  </wsdl:message>
  <wsdl:portType name="PolicySoap">
    <wsdl:operation name="Normalize">
      <wsdl:input message="tns:NormalizeSoapIn" />
      <wsdl:output message="tns:NormalizeSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="Intersect">
      <wsdl:input message="tns:IntersectSoapIn" />
      <wsdl:output message="tns:IntersectSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="Merge">
      <wsdl:input message="tns:MergeSoapIn" />
      <wsdl:output message="tns:MergeSoapOut" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="PolicySoap" type="tns:PolicySoap">
    <wsoap12:binding
      transport="http://schemas.xmlsoap.org/soap/http"
      style="document" />
    <wsdl:operation name="Normalize">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/Normalize/Request"
```

```
        style="document" />
      <wsdl:input>
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Intersect">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/Intersect/Request"
        style="document" />
      <wsdl:input>
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Merge">
      <wsoap12:operation
        soapAction="http://example.com/ws/2004/09/policy/Merge/Request"
        style="document" />
      <wsdl:input>
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="Policy">
    <wsdl:port name="PolicySoap" binding="tns:PolicySoap">
      <wsoap12:address location="TBD" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Appendix E – Round 2 Unit Test Cases for Computing Effective Policy

Although this specification focuses on two WS-Security Policy Interop Scenarios, to ensure a shared understanding of the basic interoperability, this section describes Round 2 test cases for testing effective policy computations for policies attached to WSDL.

Input and expected output data for Round 2 test cases are bundled with this specification. These are WS-Policy unit test cases. These are not interop test cases. Implementers may wrap these tests as JUnit / NUnit / VSTS implementations, Web service endpoints, command line executables, user interface code, etc. This section describes Round 2 test cases using a simple Web service endpoint wrapper. Appendix F provides a WSDL 1.1 description for the Web service endpoint wrapper.

This section describes the sequence of messages for testing effective policy computation for message, operation, endpoint, and service policy subjects. Base test case and expected results are bundled with this specification.

Assumptions

- S exposes effective policy computation as web service operations. Appendix F – contains the WSDL for these operations.
- SOAP 1.2 is the messaging protocol for interactions between R and S.
- HTTP is the underlying transport protocol for interactions between R and S.
- R transmits a WSDL and S responds with an effective policy expression in normal form.
- Policy expressions are either embedded within the WSDL or assumed to be made available out of band of the interaction between R and S..

Effective-Policy-4-Input

The Effective-Policy-4-Input operation accepts a single WSDL, that has policies attached, and returns a single policy expression that is the effective policy for the input policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveInputPolicy/Request
```

R MUST use the following SOAP Header to indicate the service, port and target QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
  <x:port>PortName</x:port>
  <x:operation>OperationName</x:operation>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Input operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
      <x:port>PortName</x:port>
      <x:operation>OperationName</x:operation>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
```

```
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Input operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for input message policy subject are organized and bundled with the specification as follows,

```
Round2/WSDL11/PolicyAttachments.wsdl ← test cases for WSDL 1.1
Round2/EffectivePolicy/Policy-for-Input.xml <- expected effective
policy expression in normal form for WSDL 1.1

Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Effective-Policy-4-Output

The Effective-Policy-4-Output operation accepts a single WSDL, that has policies attached, and returns a single policy expression that is the effective policy for the output policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveOutputPolicy/Request
```

R MUST use the following SOAP Header to indicate the target service, port and operation QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
  <x:port>PortName</x:port>
  <x:operation>OperationName</x:operation>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Output operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
      <x:port>PortName</x:port>
      <x:operation>OperationName</x:operation>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Output operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for output message policy subject are organized and bundled with the specification as follows,

```
Round2/PolicyAttachments.wsdl ← test cases for WSDL 1.1
Round2/EffectivePolicy/Policy-for-Output.xml <- expected effective
policy expression in normal form for WSDL 1.1

Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Effective-Policy-4-Fault

The Effective-Policy-4-Fault operation accepts a single WSDL, that has policies attached, and returns a single policy expression that is the effective policy for the fault message policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveFaultPolicy/Request
```

R MUST use the following SOAP Header to indicate the target service, port and operation QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
  <x:port>PortName</x:port>
  <x:operation>OperationName</x:operation>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Fault operation,

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
      <x:port>PortName</x:port>
      <x:operation>OperationName</x:operation>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Fault operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for fault message policy subject are organized and bundled with the specification as follows,

```
Round2/PolicyAttachments.wsdl ← test cases for WSDL 1.1 policy subjects
Round2/EffectivePolicy/Policy-for-Fault.xml <- expected effective
policy expression in normal form for WSDL 1.1

Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Effective-Policy-4-Operation

The Effective-Policy-4-Operation operation accepts a single WSDL, that has policies attached, and returns a single policy expression that is the effective policy for the operation policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveOperationPolicy/Request
```

R MUST use the following SOAP Header to indicate the target service, port and operation QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
  <x:port>PortName</x:port>
  <x:operation>OperationName</x:operation>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Operation operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
      <x:port>PortName</x:port>
      <x:operation>OperationName</x:operation>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Operation operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for operation message policy subject are organized and bundled with the specification as follows,

```
Round2/PolicyAttachments.wsdl ← test cases for WSDL 1.1
Round2/EffectivePolicy/Policy-for-Operation.xml <- expected effective
policy expression in normal form for WSDl 1.1

Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Effective-Policy-4-Endpoint

The Effective-Policy-4-Endpoint operation accepts a single WSDL, that has policies attached, and returns a single policy expression that is the effective policy for the endpoint policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveEndpointPolicy/Request
```

R MUST use the following SOAP Header to indicate the target service and port QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
  <x:port>PortName</x:port>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Endpoint operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
      <x:port>PortName</x:port>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Endpoint operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for endpoint message policy subject are organized and bundled with the specification as follows,

```
Round2/PolicyAttachments.wsdl ← test cases for WSDL 1.1
Round2/EffectivePolicy/Policy-for-Endpoint.xml <- expected effective
policy expression in normal form for WSDL 1.1

Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Effective-Policy-4-Service

The Effective-Policy-4-Service operation accepts a single WSDL, that has policies attached as references to both locally embedded policy expressions and externally provided policy expressions, and returns a single policy expression that is the effective policy for the service policy subject in normal form.

SOAP Action Feature URI for this operation is,

```
http://example.com/ws/2004/09/policy/EffectiveServicePolicy/Request
```

R MUST use the following SOAP Header to indicate the target service QNames,

```
<x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
  <x:service>Service</x:service>
</x:PolicySubject>
```

The following describes a request message to Effective-Policy-4-Service operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" >
  <s12:Header>
    <x:PolicySubject xmlns:x="http://example.com/ws/2004/09/policy">
      <x:service>Service</x:service>
    </x:PolicySubject>
  </s12:Header>
  <s12:Body>
    <wsdl:definitions> ... </wsdl:definitions>
  </s12:Body>
</s12:Envelope>
```

The following describes a response message from Effective-Policy-4-Service operation,

```
<?xml version="1.0" encoding="utf-8"?>
<s12:Envelope
    xmlns:s12="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsp="http://www.w3.org/ns/ws-policy">
  <s12:Body>
    <wsp:Policy> ... </wsp:Policy>
  </s12:Body>
</s12:Envelope>
```

The test case for computing effective policy and expected effective policy expression for service message policy subject are organized and bundled with the specification as follows,

```
Round2/PolicyAttachments.wsdl ← locally embedded test cases for WSDL 1.1
Round2/ExternalPolicy.xml ← externally provided policy expression
Round2/EffectivePolicy/Policy-for-Service.xml <- expected effective policy expression in normal form for WSDL 1.1
```

```
Round2/WSDL20/PolicyAttachments.wsdl <-test cases for WSDL 2.0
Round2/EffectivePolicy/*.xml <- expected effective policy expression in
normal form for WSDl 2.0
```

Appendix F – WSDL for Round 2 - Computing Effective Policies

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:wsoap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://example.com/ws/2004/09/policy"
  targetNamespace="http://example.com/ws/2004/09/policy"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema elementFormDefault="qualified"
      targetNamespace="http://example.com/ws/2004/09/policy">
      <xs:complexType name="any" mixed="true">
        <xs:sequence>
          <xs:any minOccurs="0" maxOccurs="unbounded" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="PolicySubjectType">
        <xs:sequence>
          <xs:element name="service" type="xs:QName"/>
          <xs:element name="port" type="xs:QName" minOccurs="0"/>
          <xs:element name="operation" type="xs:QName" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
      <xs:element name="PolicySubject" type="tns:PolicySubjectType"/>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="WSDLIn">
    <wsdl:part name="policy" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="PolicyOut">
    <wsdl:part name="EffectivePolicyResult" type="tns:any" />
  </wsdl:message>
  <wsdl:message name="PolicySubjectHeader">
    <wsdl:part name="PolicySubject" element="tns:PolicySubject" />
  </wsdl:message>
  <wsdl:portType name="EffectivePolicyInterface">
    <wsdl:operation name="EffectivePolicy4Input">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Output">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Fault">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Operation">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Endpoint">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Service">
      <wsdl:input message="tns:WSDLIn" />
      <wsdl:output message="tns:PolicyOut" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="EffectivePolicyBinding"
type="tns:EffectivePolicyInterface">
    <wsoap12:binding
      transport="http://schemas.xmlsoap.org/soap/http"
      style="document" />
    <wsdl:operation name="EffectivePolicy4Input">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/EffectiveInputPolicy/R
equest"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Output">
      <wsoap12:operation
```

```
soapAction="http://example.com/ws/2004/09/policy/EffectiveOutputPolicy/
Request"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Fault">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/EffectiveFaultPolicy/R
equest"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Operation">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/EffectiveOperationPoli
cy/Request"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Endpoint">
      <wsoap12:operation
```

```
soapAction="http://example.com/ws/2004/09/policy/EffectiveEndpointPolic
y/Request"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EffectivePolicy4Service">
      <wsoap12:operation

soapAction="http://example.com/ws/2004/09/policy/EffectiveServicePolicy
/Request"
        style="document" />
      <wsdl:input>
        <wsoap12:header message="tns:PolicySubjectHeader"
part="PolicySubject" use="literal" />
        <wsoap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <wsoap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="EffectivePolicyService">
    <wsdl:port name="EffectivePolicyPort"
binding="tns:EffectivePolicyBinding">
      <wsoap12:address location="TBD" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Appendix G – Change Log

Date	Editor	Description
20070207	ASV	First draft. This draft is based on a contribution from IBM and Microsoft.
20070207	ASV	Updated list of editors, added W3C license text, updated status and added change log.
20070212	PY	Added W3C Document License Text up front. Refreshed TOC

20070306	ASV	s/example.org/example.com/g to make examples consistent with Round 3 WSDL document.
20070306	ASV	Removed the trailing / in the URI in the examples - http://example.com/ws/2004/09/policy - to make them consistent with Round 3 WSDL document.
20070315	ASV	Added timestamp (save date) – editorial action 185
20070315	ASV	Implemented the resolution for issue 4370 – editorial action 188
20070607	ASV	Implemented the resolution for issue 4368 – editorial action 271
20080127	ASV	Implemented the resolution for issue 5299.