WS-Policy

Unit-V

Introduction

- **Need for WS-Policy specification**
- allows web services to advertise their policies (on security, quality of service, etc.)
- for web service consumers to specify their policy requirements in XML format

WS-Policy Framework

- The WS-Policy framework
- governs assembly and structure of policy description documents
- association of policies to Web resources

- This framework is comprised of the following three specifications:
- WS-Policy
- WS-PolicyAttachments
- WS-PolicyAssertions

Introduction to WS-Policy

- WS-Policy can express requirements, capabilities and assertions

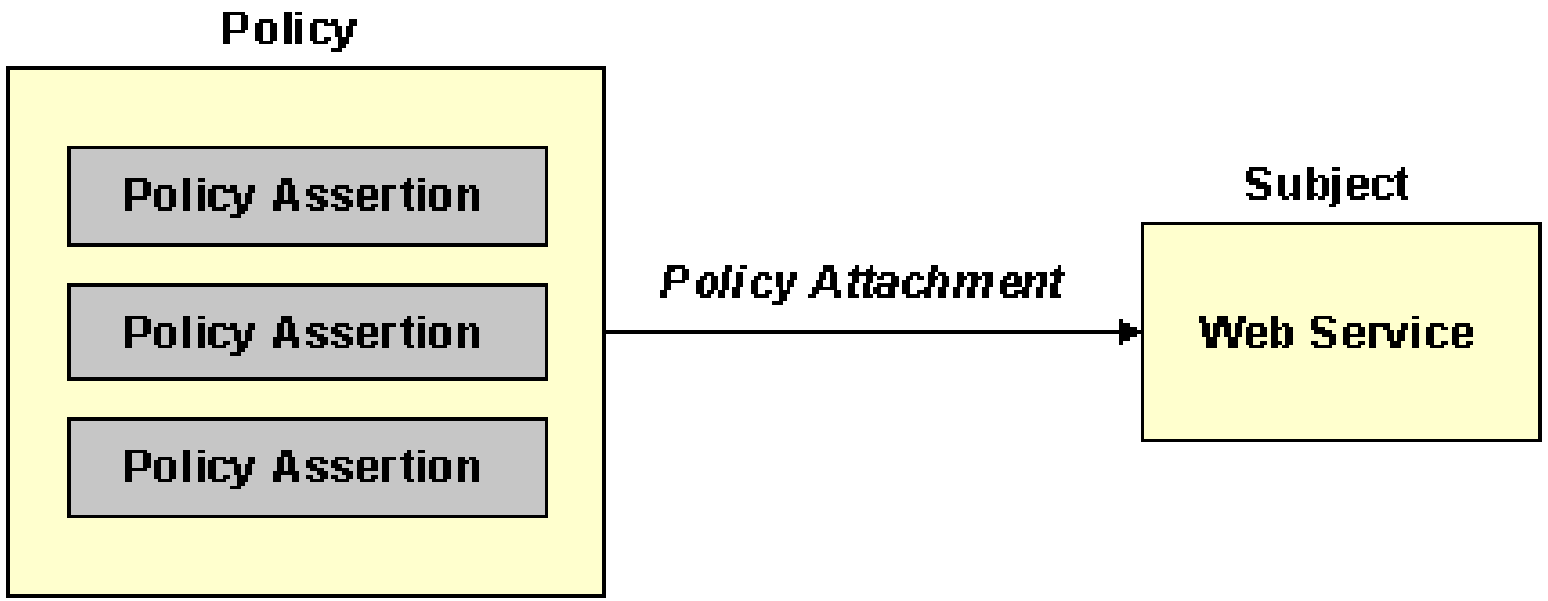
- Example:
  - a policy can indicate that a Web Service only accepts requests containing a valid signature
  - a certain message size should not be exceeded.

Terminology

- **Policy**: refers to the set of information being expressed as *policy assertions*
- **Policy Assertion**: represents an individual preference, requirement, capability, etc.
- **Policy Expression**: is an XML Infoset representation of a policy, interoperable form
- **Policy Subject**: an entity to which a *policy expression* can be bound

Terminology

- **Policy Attachment**: the mechanism for associating policy expressions with one or more subjects



Policy Namespaces

- WS-Policy schema defines all constructs that can used in a *policy expression*

Prefix	Description	Namespace
wsp	WS-Policy, WS-PolicyAssertions, and WS-PolicyAttachment	http://schemas.xmlsoap.org/ws/2002/12/policy
wsse	WS-SecurityPolicy	http://schemas.xmlsoap.org/ws/2002/12/secext
wsu	WS utilty schema	http://schemas.xmlsoap.org/ws/2002/07/utility
msp	WSE 2.0 policy schema	http://schemas.microsoft.com/wse/2003/06/Policy

Policy Namespaces

- wsp:Policy
  - Representation of a *policy expression*
  - Container for *policy assertions*

```
<wsp:Policy xmlns:wsp="..."
xmlns:wsu="..." wsu:Id="..."
    Name="..." TargetNamespace="...">
    <!-- policy assertions go here -->
</wsp:Policy>
```

  - The wsu:Id attribute assigns the *policy expression* an ID value as a URI

Policy Expression Naming

- A full ID is formed by:
  <base URI>#<wsu:Id value>
- Policy Expression:

```
<wsp:Policy xmlns:wsp="..."
    xmlns:wsu="..." wsu:Id="MyPolicies" >
    ...</wsp:Policy>
```

- Policy Reference:

```
...
<wsp:PolicyReference xmlns:wsp="..."
  URI="http://virginia.edu/isis/policy.xml#MyPolicies"/>
...
```

Policy Expression Naming

- Alternatively, use namespace-qualified name
  - Add `Name` and `TargetNamespace`:

```
<wsp:Policy xmlns:wsp="..." Name="MyPolicies"
      TargetNamespace="http://virginia.edu/policies">
...</wsp:Policy>
```

- Reference:

```
...
<wsp:PolicyReference xmlns:wsp="..."
      xmlns:p="http://virginia.edu/policies"
      Ref="p:MyPolicies"/>
...
```

Policy Assertions

- A policy assertion:
  - represents an individual preference, requirement, capability, or other characteristic

```
<wsp:Policy xmlns:wsp="..." xmlns:wsu="..." wsu:Id="..."
      Name="..." TargetNamespace="..." >
      <Assertion wsp:Usage="..." wsp:Preference="..." />
      <Assertion wsp:Usage="..." wsp:Preference="..." />
... </wsp:Policy>
```

The Usage Qualifier

- wsp:Usage specifies how assertions are processed

Value	Meaning
wsp:Required	The assertion must be applied, otherwise an error results
wsp:Rejected	The assertion is not supported and, if present, will cause failure
wsp:Optional	The assertion may be made of the subject, but is not required
wsp:Observed	The assertion will be applied to all subjects and requestors are told
wsp:Ignored	The assertion will be ignored if present and requestors are told

Assertion Example

```
<wsp:Policy xmlns:wsp="..." xmlns:wsse="...">
  <wsse:SecurityToken wsp:Usage="wsp:Required">
<wsse:TokenType>wsse:Kerberosv5ST</wsse:TokenType>
  </wsse:SecurityToken>
  <wsse:Integrity wsp:Usage="wsp:Required">
    <wsse:Algorithm Type="wsse:AlgSignature"
      URI="http://www.w3.org/2000/09/xmlenc#aes" />
  </wsse:Integrity>
</wsp:Policy>
```

Two policy assertions:

1. Security Token is required
2. Use of AES is required

Assertion Preference

- wsp:Preference attribute:
  - Used to specify the service's preference as an integer value
  - Larger integer => higher preference
  - Omitted preference attribute is interpreted as a 0

Assertion Preference Example

```
<wsp:Policy xmlns:wsp="..." xmlns:wsse="...">
  <wsse:SecurityToken wsp:Usage="wsp:Optional">

<wsse:TokenType>wsse:UsernameToken</wsse:TokenType>
  </wsse:SecurityToken>
  <wsse:SecurityToken wsp:Usage="wsp:Optional"
    wsp:Preference="1">
    <wsse:TokenType>wsse:x509v3</wsse:TokenType>
  </wsse:SecurityToken>
</wsp:Policy>
```

- The subject prefers X.509 certificates over UsernameTokens

Standard Policy Assertions

- WS-PolicyAssertions defines four general policy assertions for any subject

Policy Assertion	Description
wsp:TextEncoding	Specifies a character encoding
wsp:Language	Specifies a natural language (xml:Lang)
wsp:SpecVersion	Specifies a version of a particular specification
wsp:MessagePredicate	Specifies a predicate that can be tested against the message (XPath expressions by default)

General Assertion Example

```
<wsp:Policy xmlns:wsse="...">
  <wsp:TextEncoding wsp:Usage="wsp:Required"
Encoding="utf-8"/>
  <wsp:Language wsp:Usage="wsp:Required" Language="en"/>
  <wsp:SpecVersion wsp:Usage="wsp:Required"
    URI="http://www.w3.org/TR/2000/NOTE-SOAP-20000508/" />
  ...
</wsp:Policy>
```

- The subject requires
  1. The UTF-8 character encoding
  2. Any form of the English language
  3. SOAP version 1.1

General Assertion Example

```
<wsp:Policy xmlns:wsp="..." xmlns:wsse="...">
  <wsp:MessagePredicate wsp:Usage="wsp:Required">
    count(wsp:GetHeader(.)/wsse:Security) = 1
  </wsp:MessagePredicate>
  <wsp:MessagePredicate wsp:Usage="wsp:Required">
    count(wsp:GetBody(.)/*) = 1
  </wsp:MessagePredicate>
  ...
</wsp:Policy>
```

- Must be:
  1. Exactly one wsse:Security header element
  2. Exactly one child within the soap:Body element

WS-SecurityPolicy

- Defines a set of security-related assertions

Policy Assertion	Description
wsse:SecurityToken	Specifies a type of security token (defined by WS-Security)
wsse:Integrity	Specifies a signature format (defined by WS-Security)
wsse:Confidentiality	Specifies an encryption format (defined by WS-Security)
wsse:Visibility	Specifies portions of a message that MUST be able to be processed by an intermediary or endpoint
wsse:SecurityHeader	Specifies how to use the <Security> header defined in WS-Security
wsse:MessageAge	Specifies the acceptable time period before messages are declared "stale" and discarded

Combining Multiple Assertions

- *Policy operators* are used to combine assertions
  - Can nest operators

Policy Operator	Description
wsp:All	Requires that all of its child elements be satisfied
wsp:ExactlyOne	Requires that exactly one child to be satisfied
wsp:OneOrMore	Requires that at least one child be satisfied
wsp:Policy	Same as wsp:All

Assertion Combination Example

```
<wsp:Policy xmlns:wsp="..." xmlns:wsse="...">
  <wsp:ExactlyOne wsp:Usage="Required">
    <wsse:SecurityToken>
      <wsse:TokenType>wsse:UsernameToken</wsse:TokenType>
    </wsse:SecurityToken>
    <wsse:SecurityToken wsp:Preference="10">
      <wsse:TokenType>wsse:x509v3</wsse:TokenType>
    </wsse:SecurityToken>
    <wsse:SecurityToken wsp:Preference="1">
      <wsse:TokenType>wsse:Kerberosv5ST</wsse:TokenType>
    </wsse:SecurityToken>
  </wsp:ExactlyOne>
</wsp:Policy>
```

- Exactly one child must be satisfied

Policy Reference

- Mechanism to share policy assertions across policy expressions
- Uses the naming conventions discussed above

```
<wsp:Policy xmlns:wsp="...">
  ...
  <wsp:PolicyReference URI="..."
                       Ref="..."
                       Digest="..."
                       DigestAlgorithm="..." />
  ...
</wsp:Policy>
```

Policy Reference Example

```
<wsp:Policy wsu:Id="tokens" xmlns:wsp="..." xmlns:wsse="...">
  <wsp:ExactlyOne wsp:Usage="Required">
    <wsse:SecurityToken>
      <wsse:TokenType>wsse:UsernameToken</wsse:TokenType>
    </wsse:SecurityToken>
    <wsse:SecurityToken wsp:Preference="10">
      <wsse:TokenType>wsse:x509v3</wsse:TokenType>
    </wsse:SecurityToken>
    <wsse:SecurityToken wsp:Preference="1">
      <wsse:TokenType>wsse:Kerberosv5ST</wsse:TokenType>
    </wsse:SecurityToken>
  </wsp:ExactlyOne>
</wsp:Policy>
```

Policy Reference Example

```
<wsp:Policy wsu:Id="tokensWithSignature"
  xmlns:wsp="..." xmlns:wsse="...">
  <wsp:PolicyReference URI="#tokens" />
  <wsse:Integrity wsp:Usage="wsp:Required">
    ...
  </wsse:Integrity>
</wsp:Policy>

<wsp:Policy wsu:Id="tokensWithEncryption"
  xmlns:wsp="..." xmlns:wsse="...">
  <wsp:PolicyReference URI="#tokens" />
  <wsse:Confidentiality wsp:Usage="Required">
  ...
  </wsse:Confidentiality>
</wsp:Policy>
```

Policy Attachments

- WS-PolicyAttachment defines mechanisms to associate expressions with subjects
- Specifically defines mechanisms for:
  - XML elements
  - WSDL definitions
  - UDDI entries
- Uses attributes
  1. wsp:PolicyURIs – list of URIs
  2. wsp:PolicyPrefs – list of QNames

Policy Attachments

- The attribute wsp:PolicyAttachment binds an endpoint to a policy expression
  - Requires no change to the web service

```
<wsp:PolicyAttachment>
  <wsp:AppliesTo>
    <wsa:EndpointReference xmlns:s="...">
      <wsa:Address>http://virginia.edu/someendpoint</wsa:Address>
      <wsa:PortType>s:SomePortType</wsa:PortType>
      <wsa:ServiceName>s:SomeService</wsa:ServiceName>
    </wsa:EndpointReference>
  </wsp:AppliesTo>
  <wsp:PolicyReference URI="http://virginia.edu/policy.xml" />
  <wsse:Security>
    <ds:Signature> ... </ds:Signature>
  </wsse:Security>
</wsp:PolicyAttachment>
```

Summary

- The policy specifications define a standard framework
- Developers can:
  - express requirements, capabilities, and preferences in an interoperable way
  - select web services more meaningfully
- Policies provide support for standard assertions