Key Points

1. Keep it simple, but not simplistic

2. Need a standard language to support intersection of Assertions/Attributes, not just intersection of choices

3. Let WSDL specify mechanisms for binding policies to services
Key Points

➔ Keep it simple, but not simplistic
• Need a standard language to support intersection of Assertions/Attributes, not just intersection of choices
• Let WSDL specify mechanisms for binding policies to services
“Web services policy”

- Working Definition:

  The requirements and abilities of a web service in its interactions with other web services or consumers.

  Endpoints in a web services interaction must agree on one set of parameters from the intersection of their policies in order to interact successfully.
A Standard* Policy Usage Model

Policy Enforcement Point (PEP)

Attributes

Policy Decision Request

Policy Decision (Permit/Deny)

Policy Decision Point (PDP)

Policies

*IETF/DMTF
Some aspects of “policy”

- Policy statement
  - Vocabulary elements
  - Vocabulary sets ("domains")
  - Vocabulary mappings between domains
  - Constraints on vocabulary elements
  - Choices and combinations of constraints (WS-Policy)
- Binding to target (WS-PolicyAttachments); discovery
- Authoring: permission, storage, update, authentication
- Distribution of policies
- Contracts about policies: enforcement, monitor compliance
- Reasoning about policies; policy analysis
- Policy intersection or composition
Key Points

• Keep it simple, but not simplistic

➔ Need a standard language to support intersection of Assertions/Attributes, not just intersection of choices

• Let WSDL specify mechanisms for binding policies to services
Interaction example

User/Consumer

Service/Provider

On-line Movie Download Service
Another interaction example

User/Consumer  Service/Provider(1)  Service/Provider(2)

On-line Movie Download Service

Movie Distributor Service

May require Provider 1 to represent User to Provider 2
Policy Intersection

User/Consumer

Service/Provider

On-line Movie Download Service

User's Policy

Service's Policy
Policy intersection components

1) Intersection of choices or combinations of constraints (WS-Policy)

2) Intersection of constraints on individual vocabulary elements
Intersection of choices: Convert to DNF (OR of AND's)
Intersection of choices:
Select matching rows

<table>
<thead>
<tr>
<th></th>
<th>Policy 1</th>
<th></th>
<th>Policy 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td>A</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Intersection:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>
Intersection of constraints: WS-Policy style

Client Policy:

\[<\text{xx:BuyPrice}>1,000,000</\text{xx:BuyPrice}>\]

Service Policy:

\[<\text{xx:SellPrice}>999,999</\text{xx:SellPrice}>\]

Must understand semantics of \(<\text{BuyPrice}>\) and \(<\text{SellPrice}>\)
Intersection of constraints: alternative

Client Policy:

<integer-less-than-or-equal>
  <Attribute Id="yy:Price"/>
  <Value>1,000,000</Value>
</integer-less-than-or-equal>

Service Policy:

<integer-greater-than-or-equal>
  <Attribute Id="yy:Price"/>
  <Value>999,999</Value>
</integer-greater-than-or-equal>

Must understand semantics of integers
Translating schemas: XSLT

<xx:BuyPrice>....</xx:BuyPrice>

<integer-less-than-or-equal>
  <Attribute Id="yy:price"/>
  <value>....</value>
</integer-less-than-or-equal>

Specifies match-critical semantics of “BuyPrice”
Translating schemas: XPath

<xx:BuyPrice>....</xx:BuyPrice>

<integer-less-than-or-equal>
  <Selector xpath="...xpath to BuyPrice value..."/>
  <value>....</value>
</integer-less-than-or-equal>

Specifies match-critical semantics of “BuyPrice”
Policy matching with constraint language

Knows datatype semantics

Clients

Service Broker

Services

Client Policy

Know attribute identifier semantics

Know attribute identifier semantics

Service Policy
Policy composition with constraint language

Services

A

A's Policy

B

B's Policy

C

C's Policy

Policy composer

A/B/C Policy

Know attribute id semantics

Knows data type semantics
Why does it matter?

If intersection engine must know semantics of vocabulary elements:

- Engine must be configured to support each specific element
- Engine may need new code modules to support specific elements
- Customers can't define their own policy schemas without doing above
- Standard engines can't support vocabulary schema updates
- Standard engines can't support proprietary vocabulary schemas
XACML profile for Web Services

• Example of such a constraint language
• Also known as Web Services Policy Language (WSPL)
• Based on OASIS XACML Standard
• Specifies how to match all constraints
• Rich set of data types
• Rich set of constraint functions
WSPL policy diagram

Movie Service PolicySet
- Authorization Policy
- Service Option Policy
- Privacy Policy

Service Option Policy
- Gold-level Rule
  - Monthly-fee = $20
  - Movies-per-month = 5
  - Bandwidth ≥ 320kbps

Gold-level Rule
- Tin-level Rule
WSPL policy intersection (1)

- Pair rules in all possible combinations

User's Policy

Policy 1
- Rule 1A
- Rule 1B

Policy 2
- Rule 2A
- Rule 2B

Service's Policy

Merged Policy

- Rule 1A + 2A
- Rule 1A + 2B
- Rule 1B + 2A
- Rule 1B + 2B
WSPL policy intersection (2)

- Merge rules

**Rule 1A (User)**
- Monthly-fee ≤ $25
- Movies-per-month ≥ 4
- Bandwidth ≥ 200kbps

**Rule 2A (Service)**
- Monthly-fee = $20
- Movies-per-month = 5
- Bandwidth = 320kbps

**Merged Rule**
- Monthly-fee = $20
- Movies-per-month = 5
- Bandwidth = 320kbps
WSPL policy intersection (3)

- Eliminate incompatible rules

**Merged Policy**
- Rule 1A + 2A
- Rule 1B + 2A
- Rule 1B + 2B

**Rule 1A (User)**
- Monthly-fee $\leq$ $25
- Movies-per-month $\geq$ 4
- Bandwidth $\geq$ 200kbps

**Rule 2B (Service)**
- Monthly-fee = $2
- Movies-per-month = 1
- Bandwidth = 32kbps

Copyright © 2004 Sun Microsystems, Inc. All rights reserved.
WSPL policy intersection (4)

• Eliminate unusable rules

Example:

Current time of day:

- timeOfDay == 6pm

Rule says:

- timeOfDay ≥ 9am
- timeOfDay ≤ 5pm
Other potential XACML/WSPL contributions

• Combining algorithms
  • Preferences
• Arithmetic functions
  • E.g. (Birthdate + 21) <= CurrentDate
• Policy and rule “targets”
  • Efficient identification of applicable policies
Key Points

• Keep it simple, but not simplistic
• Need a standard language to support intersection of Assertions/Attributes, not just intersection of choices

➔ Let WSDL specify mechanisms for binding policies to services
WSDL: bind policies to services

Using extension elements:

```xml
<interface name="example4">
  <operation .../>
  <operation .../>
  <xx:[Policy Language] wsdl:required="true">
    ...policy expression...
  </xx:[Policy Language]>
</interface>
```

Using extension attributes:

```xml
<interface name="example4"
  xx:[Policy Language Reference]="...URL to policy description...">  
  <operation .../>
  <operation .../>
  <operation .../>
</interface>
```
Binding granularity

• Message payload or fraction thereof
  • e.g. XPath to <Password> payload fragment
• Message
• Operation
• PortType / Interface
• Binding
• Port / Endpoint
• Service
Key Points

1. Keep it simple, but not simplistic
2. Need a standard language to support intersection of Assertions/Attributes, not just intersection of choices
3. Let WSDL specify mechanisms for binding policies to services
Further Information

• XACML profile for Web Services

• Web-services policy language use-cases and requirements
  OASIS XACML TC Working Draft 04, 16 April 2003,

• Danfeng Yao's WSPL prototype and demo
  http://www.cs.brown.edu/people/dyao/wspl.html

• Sun's open source XACML implementation
  http://sunxacml.sourceforge.net/

• Comparing WSPL and WS-Policy (IEEE Policy 2004 presentation)

• Anne Anderson <Anne.Anderson@sun.com>
• Roberto Chinnici <Roberto.Chinnici@sun.com>