

***A flexible
access control model
for web services***

**Elisa Bertino
CERIAS and CS&ECE Departments
Purdue University**

Outline

- Motivations
- Overview of *Ws*-Attribute Based Access control (*Ws*-ABA)
- Underlying technologies
 - Digital identity management
 - Trust negotiation system
- Access control model
- System architecture
- Conclusions and future work

Web Services

- A Web service is a Web-Based application that can be
 - Published
 - Located
 - Invoked
- Compared to centralized systems and client-server environments, a Web service is much more *dynamic* and *security* for such an environment poses unique challenges

Promises of Web Services

- Interoperability across lines of business and enterprises
 - Regardless of platform, programming language and operating system
- End-to-end exchange of data
 - Without custom integration
- Loosely-coupled integration across applications
 - Using Simple Object Access Protocol (SOAP) and XML

Why HTTPS Is not Enough for Web Services

- HTTPS is protocol-level security
 - Point-to-point: lasts only for the duration of the connection
 - Does not secure solutions that use other protocols
 - “All or nothing” encryption only
 - Does not support other security mechanisms

Building Blocks for Web Service Security

- XML Encryption
 - Encrypt all or parts of an XML message
 - Separation of encryption information from encrypted data
- XML Signature
 - Apply to all or parts of a document
 - Facilitates production of composite documents while preserving the signature
 - Multiple signature with different characteristics over the same content
- SAML
 - XML format for exchanging authentication, authorization, and attribute assertions
- WS-Security
 - Originally defined by Microsoft, IBM, and Verisign
 - It defines how to attach signature, encryption, and security tokens to SOAP messages

Web Services: Access Control

An important issue is represented by the development of suitable access control models, able to restrict access to Web services to authorized users.



Web services are quite different with respect to objects typically protected in conventional systems, since they consist of software modules, to be executed, upon service requests, according to a set of associated input parameters.

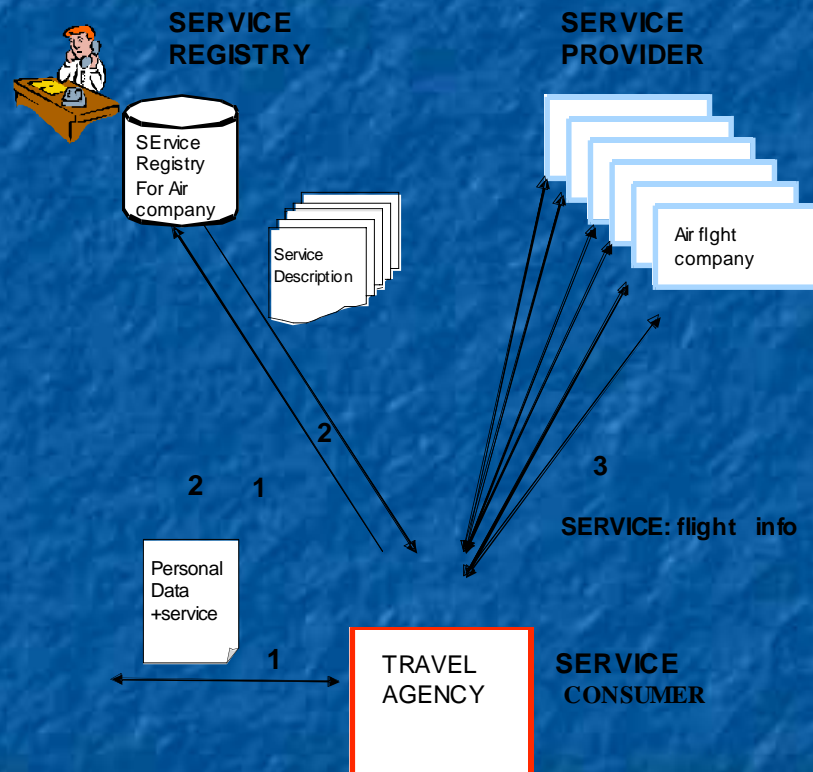


security technologies commonly adopted for Web sites and traditional access control models are not enough!

An Important Requirement: to be Policy-based

- A *policy* is a set of capabilities, requirements, preferences and general characteristics about entities in a system
- The elements of a policy (*policy assertions*) can express:
 - Security requirements or capabilities
 - Various Quality of Service (QoS) characteristics

An Example



- Suppose to have a travel agency selling flight tickets to generic customers offering a service, whose goal is to offer competitive flight tickets fare to requesting customers.
- As sketched (arrow 1), a customer request is sent by including also a set of attributes describing relevant properties of the customer and his/her preference or needs, to customize service release.
- The agency, in turn, forwards customer requests to flight companies.

Ws - Attribute Based Access Control

- Implementation independent access control model for Web services, for use within the SOAP standard, characterized by capabilities for negotiating service parameters
- The goal of *Ws-Aba*, is to express, validate and enforce application-based access control policies without assuming pre-established trust in the users of web services

Underlying Technologies

Digital Identity Management

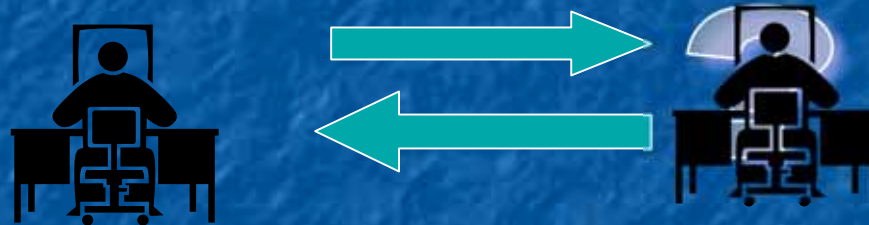
- What is digital identity?
 - *Digital identity can be defined as the digital representation of the information known about a specific individual or organization*
- The term *DI* usually refers to two different concepts:
 - *Nym* – a nym gives a user an identity under which to operate when interacting with other parties. Nyms can be strongly bound to a physical identity
 - *Partial identity* – partially identities refer to the set of properties that can be associated with an individual, such as name, birth-date, credit cards. Any subset of such properties represents a partial identity of the user

Underlying Technologies

Trust Negotiation

■ Interactions between strangers

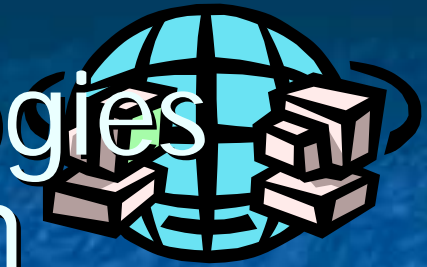
- In conventional systems user identity is known in advance and can be used for performing access control
- In open systems participants may have no pre-existing relationship and may not share a common security domain



■ Mutual authentication

- Assumption on the counterpart honesty no longer holds
- Both participants need to authenticate each other

Underlying Technologies Trust Negotiation



- A promising approach for open systems where most of the interactions occur between strangers
- The **goal**: establish trust between parties in order to exchange sensitive information and services
- The **approach**: establish trust by verifying **properties** of the other party

Ws-Aba access control model

- Access conditions
 - expressed in terms of *partial identities*
 - take into account also the parameters characterizing web services
- Concept of *access negotiation*
 - Web service negotiation in Ws-Aba deals with the possibility for trusted users to dynamically change their access requests in order to obtain authorizations

Ws-Aba access control policies

- An access control policy is defined by three elements:
 - A service identifier
 - A set of parameter specifications
 - A parameter specification is a pair
 - Parameter-name, parameter-value-range
 - A set of conditions against partial identities
- A WS-policy specification of our policy language has been developed

Ws-Aba access control policies examples

■ Policy Pol1

- (FlightRes; Discount[0,30]; Age > 65)
- It authorizes subjects older than 65 to reserve a flight with a discount up to 30%;


■ Policy Pol2

- (FlightRes; {Fare [Standard, Gold], Discount[0,50]}; {Partnership=TravelCorporation, Seniority >3, Age>65})
- It authorizes subjects that are older than 65 and have a 3 year seniority and have a partnership with TravelCorporation to get a fare between standard and gold and a discount up to 50%


Ws-Aba: how it works

 Access requests are received

- ✓ specified by constraining service parameters, and subject partial identities
- ✓ Note: a subject before releasing partial identity information may require to establish trust by using trust negotiation


 The system extracts the corresponding access control policies, in order to establish whether the subject request can be:

- ✓ accepted as it is
- ✓ must be rejected
- ✓ has to be negotiated

 A request negotiation results in eliminating and/or modifying some of the service parameters specified within an access request that made it not immediately acceptable

Access responses in Ws-Aba

- Upon an access request three replies are possible:

 The submitted attributes match with a policy for the specified service request and the specified service parameters are acceptable by the policy




Request is granted

 The submitted attributes do not match with any policy for the specified service request



Request is rejected

 The submitted attributes match with a policy for the specified service request but the specified service parameters are not acceptable by the policy



Access responses in Ws-Aba - example

- Policy Pol1 - (FlightRes; Discount[0,30]; Age > 65)
- Policy Pol2 - (FlightRes; {Fare [Standard, Gold]; Discount[0,50]});
{Partnership=TravelCorporation, Seniority >3, Age>65})
- Requests:
 - <[Partnership:TravelCorporation, Seniority:5, Age:70]; FlightRes; [Fare:Gold, Discount:30]>
 - It complies with Pol2 and can be fully accepted
 - <[Age:70; FlightRes; [Discount:50]>
 - It complies with Pol1; however it must be negotiated since the parameter value is outside the range specified in Pol1
 - <[University:Milano; FlightRes; [Discount:30]>
 - It is rejected since it does not match the subject specification of any policy

Certificates supported

- WS-Aba accepts SOAP messages for service invocation
- To promote interoperability and flexibility we do not restrict our system to a specific implementation, we adopt a specific proposal to connect our system to the PKC infrastructure: X.509 AC

Identity and attributes: X.509 AC

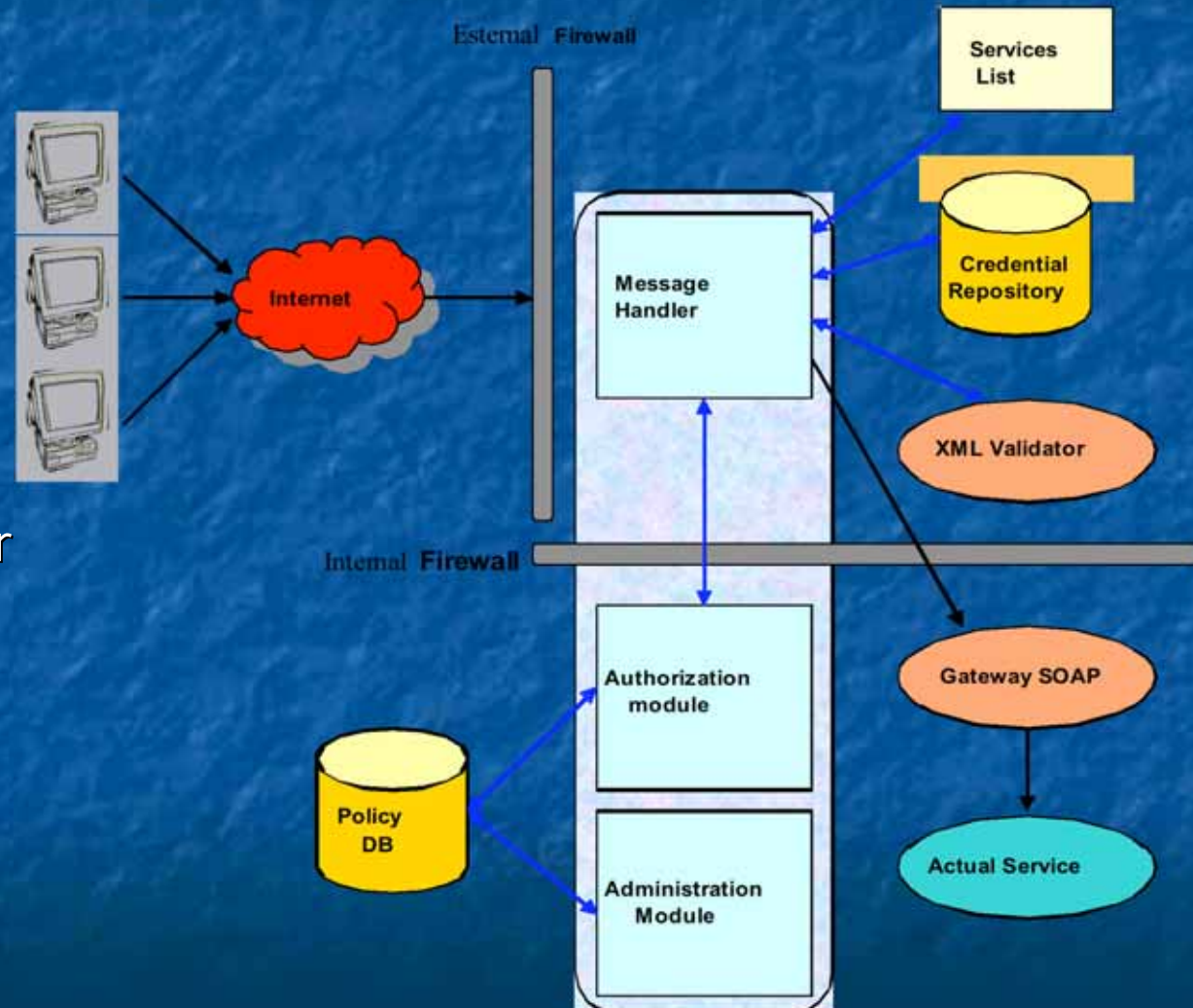
X.509 AC provides a binding between attributes and an identity. It is composed of two nested elements: the former describing the conveyed information, that is, the AttributeCertificateInfo element and the Signature element, carrying the signature

```
<element name="Attributes" type="ac:AttributesType"/>
<complexType name="AttributesType">
  <sequence>
    <element ref="ac:ServiceAuthenticationInformation" minOccurs="0"/>
    <element ref="ac:AccessIdentity" minOccurs="0"/>
    <element ref="ac:ChargingIdentity" minOccurs="0"/>
    <element ref="ac:Group" minOccurs="0"/>
    <element ref="ac:Role" minOccurs="0"/>
    <element ref="ac:Clearance" minOccurs="0"/>
    <element ref="ac:GenericAttribute" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="Id" type="ID" use="optional"/>
</complexType>
```

WS- Aba System Architecture

- Three main modules:

- Message Handler
- Authorization module
- Authorization management



Open issues

- Policy selection:
 - If a request complies with several policies, how do we choose a policy to apply?
- Negotiation of parameters:
 - How can subjects negotiate service parameters?
- Delegation:
 - How to manage delegated access requests?
- Cached policies:
 - How and where keep track of previous access requests?
- Policy protection:
 - How to protect UDDI registries where AC policies are stored?

Future work

- Delegation mechanisms for credentials
- Automated mechanisms supporting negotiations of parameters
- Automated mechanisms for policy configurations – for making policies active or passive depending on specific events and context conditions
- Granularity levels of policies: policies that apply to group of services
- Authorization derivation rules, allowing authorizations on a service to be automatically other services