Lesson 23 – Secure Business Process

Service Oriented Architectures Security

Module 1 -Basic technologies

Unit 1 – Introduction

Ernesto Damiani

Università di Milano

Web Evolution

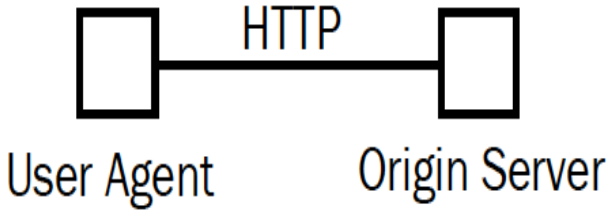
Current: Human and some automated usage

- **Interactive Web pages**
- **XML technology (data exchange, data representation)**
- **RESTful Web Services (URI, CRUD)**
- **SOAP Web Services (WSDL, SOAP)**
- **Semantic Web (RDF, OWL, RuleML, Web databases)**

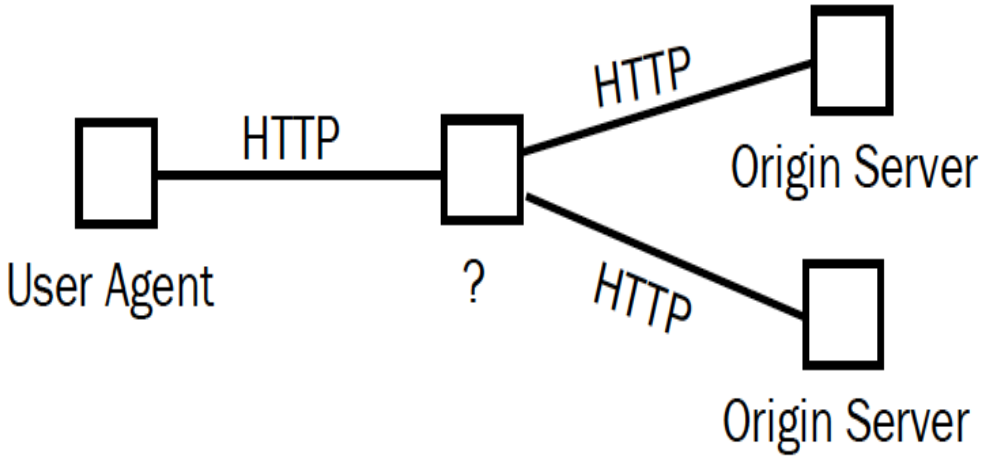
Future: Semantic Web Services

REST vs SOAP

- The basic REST design elements do not take composition into account



- WS-BPEL is the standard Web service composition language. Business process models are used to specify how a collection of services is orchestrated into a composite service
- Can we apply WS-BPEL to RESTful services?



WS-Security reminder

WS-Security (Web Services Security): a communications protocol providing a means for applying security to Web Services

From: originally by IBM, Microsoft, and VeriSign, the protocol is now officially called WSS and developed via committee in Oasis-Open

Defines how integrity and confidentiality can be enforced on Web Services messaging

Use of SAML and Kerberos, and certificate formats

Incorporates security features in the header of a SOAP message, working in the application layer (different from TLS-based security)

WS Policy reminder

WS-Policy: a specification that allows web services to use XML to advertise their policies (on security, Quality of Service, etc.)

- **Used by web service consumers to specify their policy requirements**

Inherent Security of Web Services

Security of the services

Security Software ≠ Software Security

Problems

- **Late binding adds flexibility at the expense of reduced safety**

- **We are moving from the safety of *pre* run-time structuring&verification to the complete freedom of dynamic composition, while we are providing service**
  - **we need to go beyond traditional *pre* run-time testing and validation!**

WS AND BP SECURITY CERTIFICATION

Goals

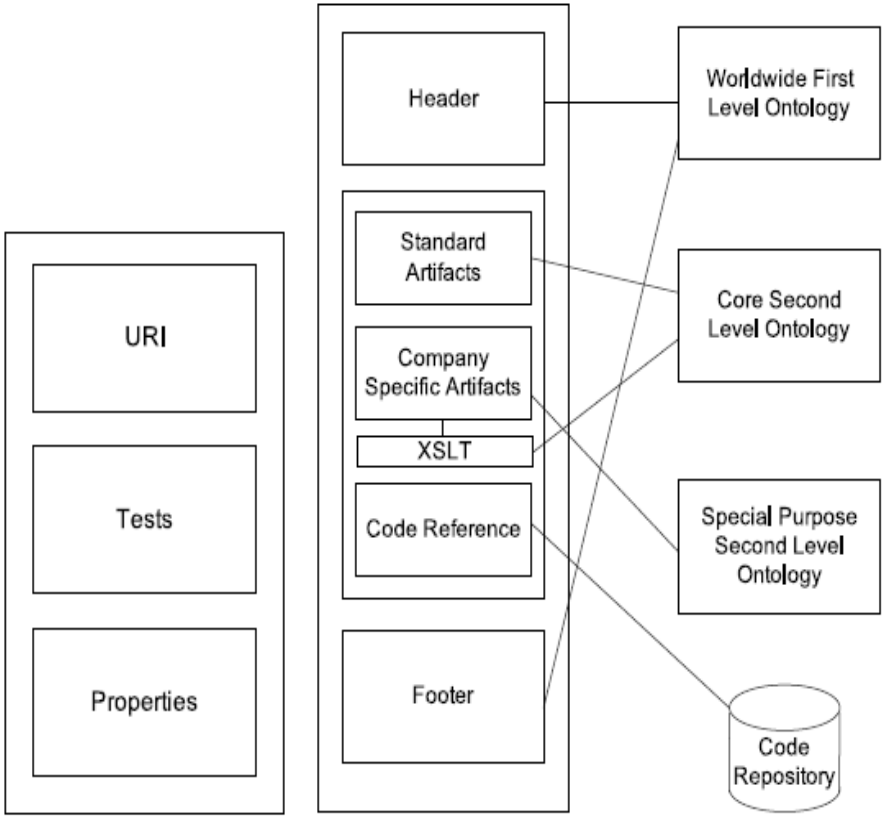
Certification scheme for services

Service composition process driven by the analysis of certified properties of individual services at selection time

- **A (certifiably correct) inference process that starting from certified properties of individual services computes the properties of the composed process**

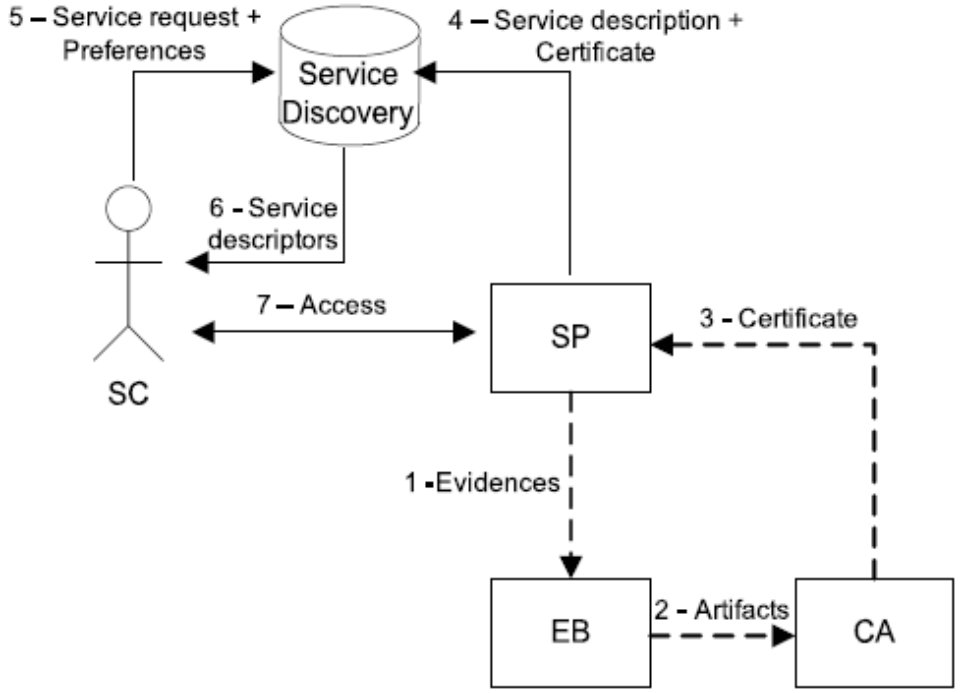
Test-based Certification

- **Evidence-based proofs that a test carried out on the service has given a result and some properties holds**
- **Require machine-readable (XML-based) certificates**
- **Support dynamic selection of single services**



Test-based Certification (2)

- A service certification infrastructure provides
  - A certification process
  - A certificate-aware service discovery
  - (Semi-)automatic matching of consumer preferences and certificates
  - A mechanism to associate certificates with WSDL of the service



Model-based certifications: certifying compositions

Predicting non-functional properties of services obtained by composition

Model-checking techniques look promising for computing properties at run-time but

- **run-time model checking not always possible**

Contract-based approach check simple properties based on services' pre-conditions, post-conditions, and invariants

Certifying Compositions (2)

"Properties come first" strategy

- Security properties to achieve dictate a "safe" service composition scheme
- Run-time verification or validation

Example: cardinality-based privacy properties

- No more than *k* component services simultaneously hold a given information
- Easily imposed for series-only compositions
- Need for proving such a property run-time can drive the composition topology

Example

Property to be certified is clique avoidance, i.e., the impossibility of certain information-sharing cliques to arise

Properties of individual services in an orchestration

- **“Invocation parameters are retained by the invoked service for less than 5 msec”**
- **“Invocation parameters cannot be inferred from other internal variables or results”**

Example

Orchestrator will invoke component services in a linear sequence

Orchestration timing (certifiably) shows that invocations are being clocked at 10 msec from each other

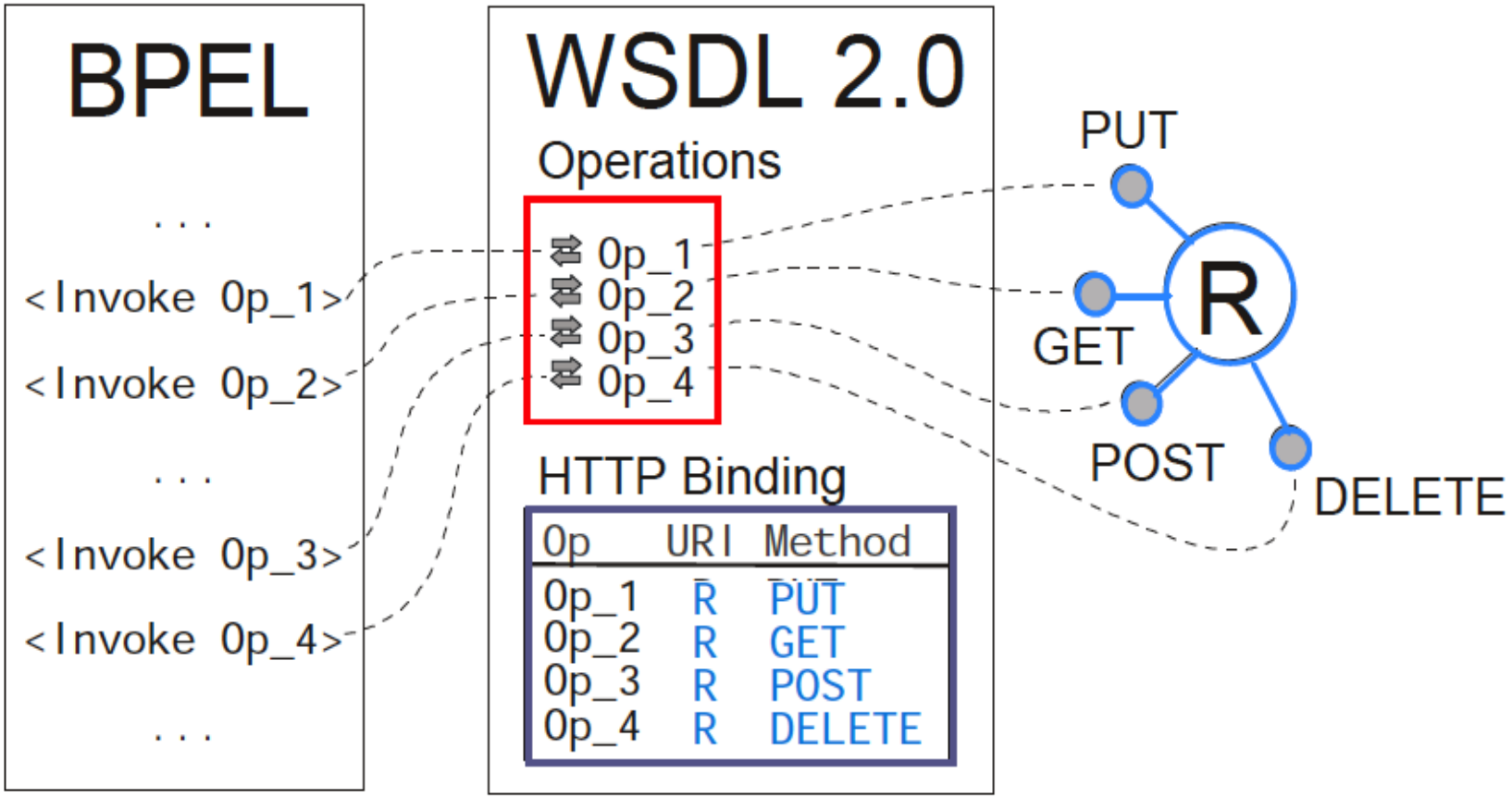
- **No information-sharing clique of services can be formed**

A process-wide security property of clique avoidance is inferred dynamically from individual services' certifications and process execution context

Other families of certifiable properties

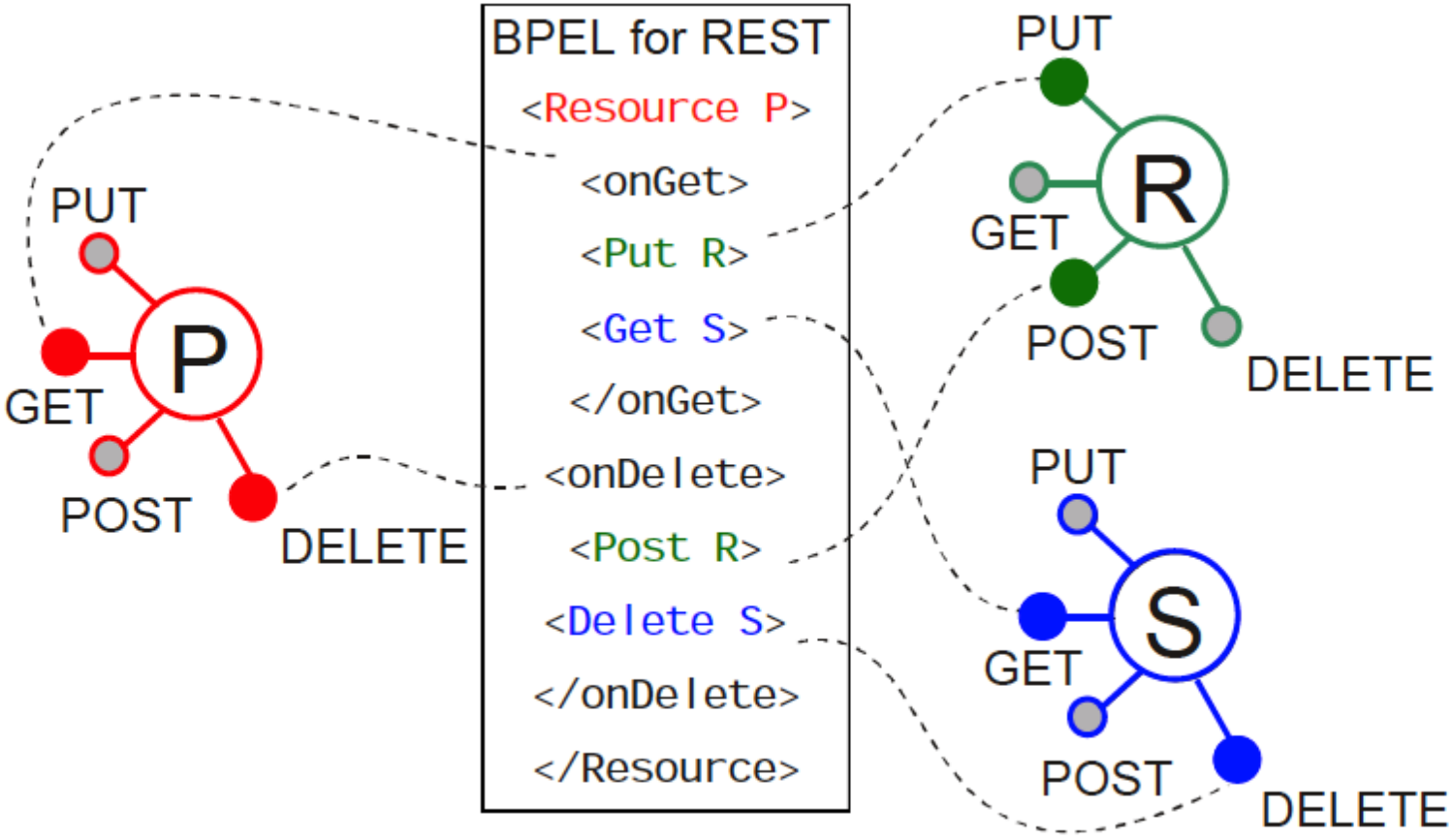
- Knowledge sharing/privacy
- Separation of duty
- Deadline completion
- ….

In principle, it can hold for REST as well..



But unfortunately no BPEL support (yet?)

Integrated REST and BPEL



BPEL Fault-Handler

BPEL faults are of two types:

- **business faults: application specific faults returned by a service invoked**
- **runtime faults: triggered or detected by the environment**

Handling runtime faults requires finding a replacement for the failed service

Reactions

Retry

- **transient faults**

Rebind

- **find a suitable replacement for previous service**

Restructure (local reconfiguration)

- **find a collection of services that satisfies request, or merge given collection into one**

S&R-by-Design Considerations

WHAT I HAVE

- **Separation of code from the actual business logic**
- **Reliability, based on the recovery from a failed component service.**
  - **This involves: rerouting of processes that have already begun execution**

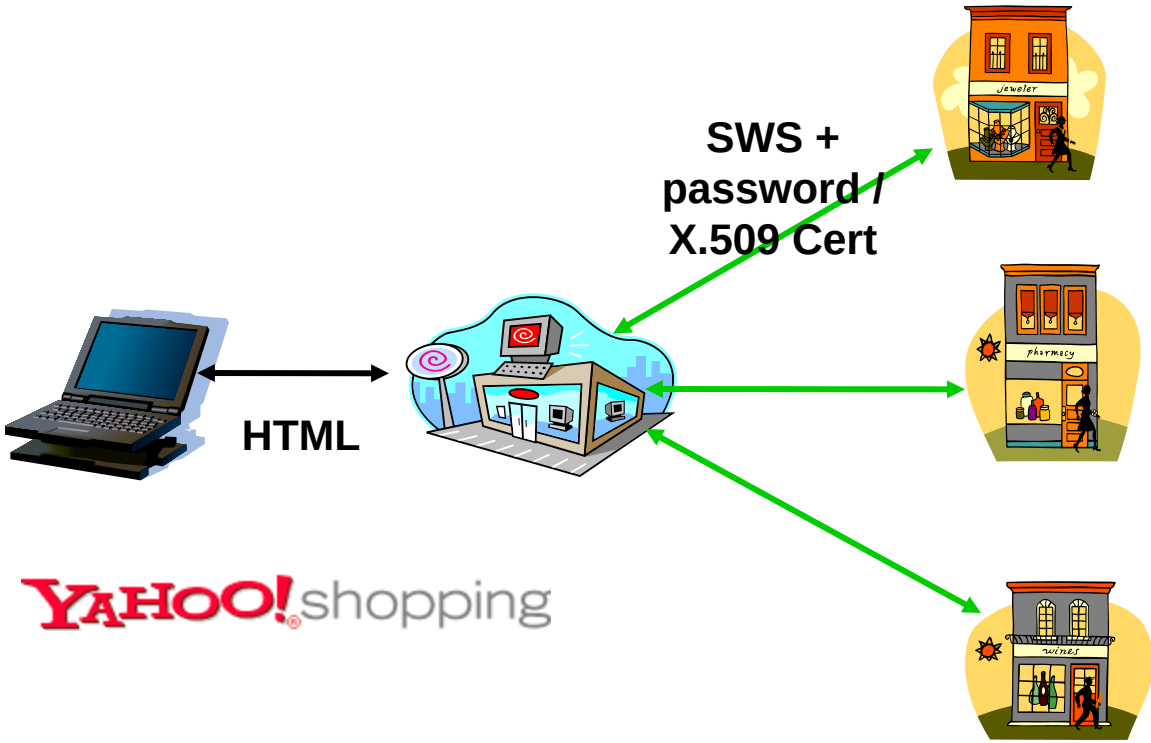
WHAT I NEED

- **Efficient recovery from a failure of component services**
- **Service selection based on the composition constraints (business rules)**

Security and Privacy - Today

Today transactions are secured using WSS toolkits to implement the Web Service security standards

Usually support for X.509 Certificates or password credentials



Basic Business Process Security Requirements

Identity Management: Each entity must be able to identity itself to the party it wants to communicate with

Policy Management: Each entity enforces policies with other entities. E.g. message format, who has access to what, what one needs to process

Secure Messaging: authentication, confidentiality, integrity, non-repudiation

Service Selection Algorithm

- **Finds a replacement service if documented: the algorithm will find a replacement service if it is explicitly defined or selection is based on the service categories**

- **Is correct: selection of the service must be such that it satisfies the business and security requirements given as rules**

- **Efficiency: The time complexity of the algorithm primarily arises from the inference engine – small number of rules**

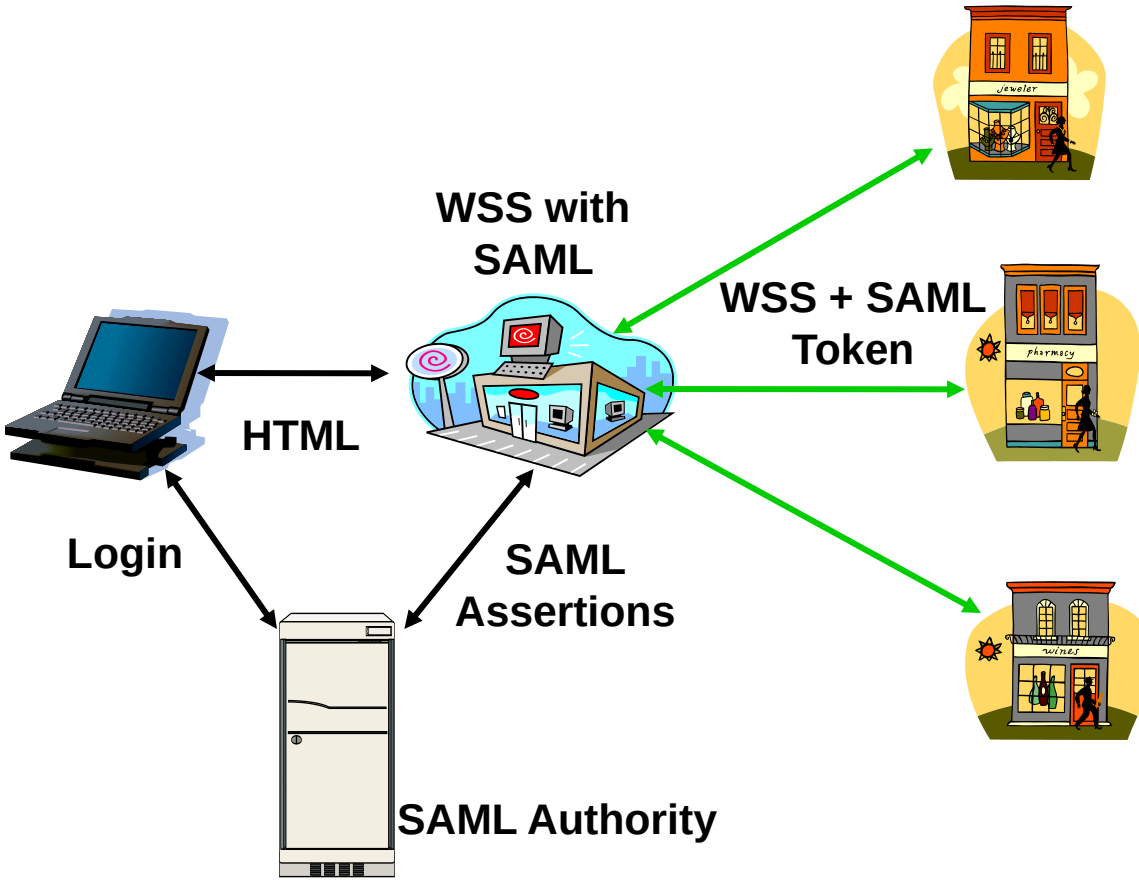
Security and Privacy – "Tomorrow"

SAML Tokens for use in WSS security headers to support Federated Identities

User Authentication supplied by CT/FIM

Requests SAML assertions from SAML authority to build SAML tokens

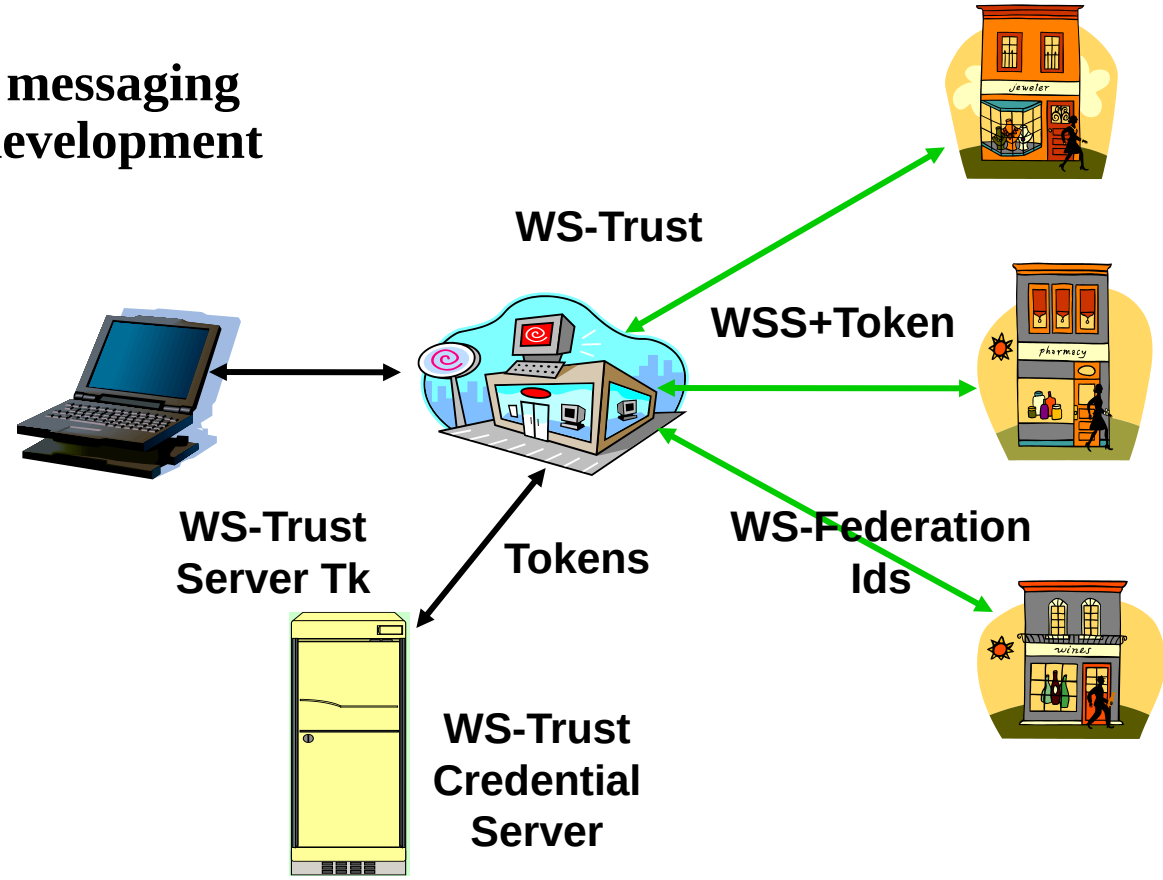
Crossover from Browser/User security world to Web Services



Security and Privacy – "Tomorrow"

Web services infrastructure moves toward WS-Trust credential servers for token issuance and support of WS-Federation

WS-Trust toolkits provide messaging and protocol support for development of clients and servers



Reading List

- **Web Services Choreography Working Group , http://www.w3.org/2002/ws/chor/**
- **Web Services Federation Language (WS-Federation), http://msdn.microsoft.com/webservices/webservices/understanding/advancedwebservices/default.aspx?pull=/library/en-us/dnglobspec/html/ws-federation.asp**
- **A Case Study of the WS-Security Framework, http://www.cs.ucsb.edu/~gayatri/Presentations/WS%20Case%20Study.ppt**
- **Web Services Choreography and Process Algebra, http://www.daml.org/services/swsl/materials/WS-CDL.ppt**
- **WS Choreography Overview, http://xml.coverpages.org/BurdettWSChoreographyOverview200306.ppt**
- **BPEL Overview, http://www.oracle.com/technology/tech/webservices/ppt/BPELOverview.ppt#1**
- **Ernesto Damiani, Antonio Maña: Toward WS-certificate. SWS 2009: 1-2 http://portal.acm.org/citation.cfm?doid=1655121.1655123**
- **M. Anisetti, C. Ardagna, F. Guida, S. Gürgens, V. Lotz, A. Maña, C. Pandolfo, J.-C. R. Pazzaglia, G. Pujol, G. Spanoudakis: ASSERT4SOA: Toward Security Certification of Service-Oriented Applications. OTM Workshops 2010: 38-40 http://www.springerlink.com/content/131162771vw62450/**
- **C. Pautasso, BPEL for REST, Proc. of the 6th International Conference on Business Process Management (BPM 2008), Milan,Italy, September 2008.**

Secure orchestrations via Transformations

BPEL process seen as a graph
Graph transformation rules express possible local changes

- **Unroll loops**
- **split parallel node composition into a sequence parallel node composition**

Restructure

- **BPEL process seen as a graph**
  - Graph transformation rules express possible local changes
- **At this stage we consider**
  - split a node into a sequence
  - parallel node composition
  - branch composition