Trustworthy Semantic Webs

Dr. Thuraisingham
The University of Texas at Dallas

December 2008
Outline

• Origins
  – Tim Berners Lee’s Vision for Semantic Web (late 1990s)
  – NSF/EU Workshop October 2001

• Where am we now?: Trustworthy Semantic Webs
  – Beyond XML Security
  – Secure Geospatial semantic web
  – Application: Assured Information Sharing

• References
  – A semantic web primer: Antoniou and van Harmelen, MIT Press 2003
  – Building trustworthy semantic webs, Thuraisingham, CRC Press (Taylor and Francis), 2007
Security Constraints (Policies)
Late 1980s/Early 1990s

- Simple Constraint: Mission attribute of SHIP is Secret
- Content-based constraint: If relation MISSION contains information about missions in Europe, then MISSION is Secret
- Association-based Constraint: Ship’s location and mission taken together is Secret; individually each attribute is Unclassified
- Release constraint: After X is released Y is Secret
- Aggregate Constraint: Ten or more tuples taken together is Secret
- Dynamic Constraint: After the Mission is completed, information about the mission is Unclassified
- Logical Constraint: A Implies B; therefore if B is Secret then A must be at least Secret
Enforcement of Security Constraints (1990-1995)

User Interface Manager

Constraint Manager

Query Processor: Constraints during query and release operations

Update Processor: Constraints during update operation

Database Design Tool
Constraints during database design operation

Security Constraints

MLS/DBMS

MLS Database
Challenges
1996

- How do you specify the policies
  - Very difficult task
  - DoD Policies are very complex
- Reasoning about policies
  - Datalog based systems have performance problems
  - Need more complex reasoning strategies
- Inference problem is unsolvable
  - 1990 Computer Security Foundations Workshop (Thuraisingham)
- One can always make unauthorized inferences by using real world knowledge
  - How long do you keep the history; how do you represent the real world data
From Today’s Web to the Semantic Web
Berners Lee’s Vision in the late 1990s

• Today’s web
  – High recall, low precision: Too many web pages resulting in searches, many not relevant
  – Results sensitive to vocabulary: Different words even if they mean the same thing do not result in same web pages
  – Results are single web pages not linked web pages

• Tomorrow’s web
  – Machine understandable web pages
  – Activities on the web such as searching with little or no human intervention
  – Technologies for knowledge management, e-commerce, interoperability
  – Retrieving appropriate web pages, sensitive to vocabulary
Potential Applications

• Knowledge Management
  – Organizing knowledge, automated tools for maintaining knowledge, question answering, querying multiple documents

• E-Business
  – Develop software agents that can interpret privacy requirements, pricing and product information and display timely and correct information; provide information about the reputation of shops; negotiation on behalf of the user

• Other: web services, interoperability, data management, digital libraries
Layered Approach: Tim Berners Lee’s Technology Stack
XML Security (jointly with U of Milan)  
Credentials in XML

<Professor credID="9" subID = "16: C1ssuer = “2”>
  <name> Alice Brown </name>
  <university> University of X </university>
  <department> CS </department>
  <research-group> Security </research-group>
</Professor>

<Secretary credID="12" subID = "4: C1ssuer = “2”>
  <name> John James </name>
  <university> University of X </university>
  <department> CS </department>
  <level> Senior </level>
</Secretary>
Policies in XML

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

... 
<policy_spec ID='P1' cred_expr="/Professor[department='CS']" target="annual_report.xml" path="/Patent[@Dept='CS']/node()" priv="VIEW"/>
<policy_spec ID='P2' cred_expr="/Professor[department='CS']" target="annual_report.xml" path="/Patent[@Dept='IST']/Short-descr/node() and /Patent[@Dept='IST']/authors" priv="VIEW"/>
<policy_spec ID='P3' cred_expr="/Professor[department='IST']" target="annual_report.xml" path="/Patent[@Dept='IST']/node()" priv="VIEW"/>
<policy_spec ID='P4' cred_expr="/Professor[department='IST']" target="annual_report.xml" path="/Patent[@Dept='CS']/Short-descr/node() and /Patent[@Dept='CS']/authors" priv="VIEW"/>
<policy_spec ID='P5' cred_expr="/secretary[department='CS' and level='junior']" target="annual_report.xml" path="/Asset[@Dept='CS']/node()" priv="VIEW"/>
<policy_spec ID='P6' cred_expr="/secretary[department='CS' and level='senior']" target="annual_report.xml" path="/Asset[@Dept='IST']/Funds/@Type and /Asset[@Dept='IST']/Funds/@Funding-Date" priv="VIEW"/>
<policy_spec ID='P7' cred_expr="/secretary[department='IST' and level='junior']" target="annual_report.xml" path="/Asset[@Dept='IST']/node()" priv="VIEW"/>

...
</policy_base>
System Architecture for Access Control

User

Pull/Query

X-Access

X-Admin

Policy base

Credential base

Admin Tools

Push/result

XML Documents
Third-Party Architecture

• The Owner is the producer of information. It specifies access control policies.
• The Publisher is responsible for managing (a portion of) the Owner information and answering subject queries.
• Goal: Untrusted Publisher with respect to Authenticity and Completeness checking.

The diagram illustrates the flow of information and credentials between the XML Source base, policy base, and the SE-XML system. The process involves queries, credentials, reply documents, and SE-XML interactions.
Secure Web Service Architecture
Confidentiality, Authenticity, Integrity

Query

UDDI

Service requestor

BusinessService

Service provider

BusinessEntity

<dsig:Signature>

PublisherAssertion

BindingTemplate

BusinessService

tModel
NSF/EU Workshop October 2001

• An invitational workshop was held in October 2001 to determine directions for semantic web
• My presentation was on Security for the semantic web; this was the beginning of security research for the semantic web
• Since then two major accomplishments (2001-2004)
  – Finin, Kagal et al at UMBC; REI policy language, framework and reasoned (Kagal PhD Thesis)
  – Policy Aware Web Project initiated at MIT
• More recently Pietro Bonatti et al’s (U of Naples) research on policy and semantic web
Beyond XML Security

Why do we need RDF, OWL Security?

• Why do we need RDF and OWL?
  – More expressive as well as reasoning power than XML
  – Inferencing capabilities
• Policies can be expressed in RDF and OWL
• Need to secure RDF and OWL documents
• Inference and Privacy problems can be better handled with RDF and OWL
• Some early research on RDF security with Elena Ferrari and Barbara Carminati (2003-4)
RDF Specification

- RDF specifications have been given for Attributes, Types Nesting, Containers, etc.
- How can security policies be included in the specification?
- Example: consider the statement “Berners Lee is the Author of the book Semantic Web”
- Do we allow access to the connection between author and book? Do we allow access to the connection but not to the author name and book name?
RDF Policy Specification

<rdf: RDF
    xmlns: xsd = “http:// - - -
    xmlns: uni = “http:// - - - -

<rdf: Description: rdf: about = “949352”
    <uni: name = Berners Lee</uni:name>
    <uni: title> Professor < uni:title>
Level = L1
    </rdf: Description>

<rdf: Description rdf: about: “ZZZ”
    < uni: bookname> semantic web <uni:bookname>
    < uni: authoredby: Berners Lee <uni:authoredby>
Level = L2
    </rdf: Description>

</rdf: RDF>
Security and Ontologies

• Access control for Ontologies
  – Who can access which parts of the Ontologies
  – E.g, Professor can access all patents of the department while the Secretary can access only the descriptions of the patents in the patent ontology

• Ontologies for Security Applications
  – Use ontologies for specifying security/privacy policies
  – Ontology reasoning techniques for reasoning about policies
  – Integrating heterogeneous policies may involve developing ontologies and resolving inconsistencies
Confidentiality, Privacy and Trust

CPT

- **Trust**
  - Trust is established between say a web site and a user based on credentials or reputations.

- **Privacy**
  - When a user logs into a website to make say a purchase, the web site will specify that its privacy policies are. The user will then determine whether he/she wants to enter personal information.
  - That is, if the web site will give out say the user’s address to a third party, then the user can decide whether to enter this information.
  - However before the user enters the information, the user has to decide whether he trusts the web site.
  - This can be based on the credential and reputation.
  - if the user trusts the web site, then the user can enter his private information if he is satisfied with the policies. If not, he can choose not to enter the information.

- **Confidentiality**
  - Here the user is requesting information from the web site;
  - the web site checks its confidentiality policies and decides what information to release to the user.
  - The web set can also check the trust it has on the user and decide whether to give the information to the user.
Inference/Privacy Control

Interface to the Semantic Web

Inference Engine/Rules Processor

Semantic web engine

Policies
Ontologies
Rules

Technology
By UTDallas

XML, RDF, OWL Documents
Web Pages, Databases
Inference/Privacy Control
Prototype-1

Technology
By UTDallas

Policies
Ontologies
Rules
In RDF

Inference Engine/
Rules Processor
e.g., Pellet

JENA RDF Engine

Interface to the Semantic Web

RDF Documents
Inference/Privacy Control
Prototype - II

Interface to the Semantic Web

Inference Engine/Rules Processor e.g., RDF Reasoner

Policies
Ontologies
Rules
In RDF

Oracle RDF Data Manager

RDF Documents

Technology
By UTDallas
Query Modification and SPARQL

- Extensive research on SQL query modification based on access control rules/policies
- Extended for inference problem
- Extensive research on developing SPARQL (query language) for RDF documents
- SPARQL query modification implemented on top of RDF Data Manager
- Integrate SPARQL engine into the Inference/Privacy controller
DAGIS (Automatic Discovery of Geospatial Information Services) Vision

Bringing the web to its full potential for Geospatial Domain

Dynamic

Geo-Web Services
UDDI, WSDL, SOAP, OGC -WS

Static

WWW
URI, HTML, HTTP

Geospatial Semantic Web Services

Geospatial Semantic Web
GRDF

DAGIS
Motivating Scenario

**Query:** “Find movie theaters within 30 miles of 75080”
within, near, overlap – Geospatial Operators
Theaters, Restaurants – Businesses (Non-Geospatial data)
Miles – Distance Unit
75080, Richardson – Geo References
DAGIS System Architecture

- DAGIS Query Interface
- OWL-S MatchMaker
- OWL-DL Reasoner for Matchmaker
- Service Providers
DAGIS for Complex Queries

Find Movie Theaters within 30 Miles from Richardson, TX
## DAGIS Integration Scenarios

<table>
<thead>
<tr>
<th>Query</th>
<th>Availability</th>
<th>Service Type</th>
<th>Service Invoked</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find Movie Theaters within 30 Miles of 75080</td>
<td>YES</td>
<td>Atomic (Single) Service Provider</td>
<td>GetTheatersAndMoviesService</td>
</tr>
<tr>
<td>Find Movie Theaters within 30 Miles of Richardson,TX</td>
<td>NO</td>
<td>DAGIS Composes two Atomic Services: ZipCodeFinder, GetTheatersAndMovies</td>
<td></td>
</tr>
<tr>
<td>Find Movie Theaters within 30 Miles of 75080 QoS: Response Time 30 Sec</td>
<td>YES</td>
<td>Atomic Service</td>
<td>GetTheatersAndMoviesService QoS Response Time 40 Sec</td>
</tr>
<tr>
<td>Find Movie Theaters within 30 Miles of 75080</td>
<td>NO</td>
<td>Atomic Service</td>
<td>GetTheatersService</td>
</tr>
</tbody>
</table>
Geospatial Data Integration

Upper-level ontologies

Abstract Definitions of Main Geospatial Concepts

Mid-level ontology (GRDF)

Concrete Definitions of All Relevant Geospatial Concepts

Domain ontologies

Hydrology ontology
Cartography ontology
Image ontology
GRDF (Beyond OGC’s GML)

GRDF (Geospatial Resource Description Framework)

– Adds semantics to data
– Loosely-structured (easy to freely mix with other non-geospatial data)
– Semantically extensible

ComputerScience Building hasExtent (33.98111, -96.4011)
(33.989999, -96.4022)
Semantic Access Control (SAC)

Motivation
Shortcomings of Traditional Access Control
- Proprietary systems
- Lack of modularity
- Changes in access control schemas break the system
- Changes in data schemas break the system
- Path to resources (e.g., XPATH) is clumsy
  //school/department/professor/personal/ssn – LONG!
- Non-optimal for distributed/federation environment
SAC Ontology

- Written in OWL (Web Ontology Language)
- User-centric
- Modular
- Easily extensible
- Available at:

http://utd61105.campus.ad.utdallas.edu/geo/voc/newacceso
sonto
Geo-WS Security

• Data providers (e.g., geospatial clearinghouses, research centers) need access control on serviceable resources.

• Access policies have geospatial dimension
  – Bob has access on Building A
  – Bob does NOT have access on Building B
  – Building A and B have overlapping area

• Current access control mechanisms are static and non-modular.
Geo-WS Security: Architecture

Client

DAGIS

Geospatial Semantic WS Provider

Enforcement Module

Decision Module

Authorization Module

Semantic-enabled Policy DB

Web Service Client Side

Web Service Provider Side
Geo-WS Security: Inferencing

- **Semantic-enabled Policy DB**
  - Obvious facts
  - Deduced facts

- Inferencing Module

- Geospatial Data Store
SAC in Action

- Environment: University Campus
- Campus Ontology
  [http://utd61105.campus.ad.utdallas.edu/geo/voc/campusonto](http://utd61105.campus.ad.utdallas.edu/geo/voc/campusonto)
- Main Resources
  - Computer Science Building
  - Pharmacy Building
  - Electric Generator in each Building
- User Access:
  - Bob has ‘execute’ access to all Building Resources
  - Bob doesn’t have any access to CS Building
  - Bob has ‘modify’ access to Building resources within a certain geographic extent
- Policy File located at
  [http://utd61105.campus.ad.utdallas.edu/geo/voc/policyfile1](http://utd61105.campus.ad.utdallas.edu/geo/voc/policyfile1)
Directions for Secure Geospatial Semantic Web

- QoS based Selection for Complex Queries
- Automatic Trust Negotiation for DAGIS
  - Examine the work of Winslett et al (Trust Builder) and Bertino et al (Trust-X)
- Define a specification for access control semantics
- Geospatial dataset development
- Application: Crime Analysis
Assured Information Sharing: Need to Know to Need to Share

• Need to know policies during the cold war; even if the user has access, does the user have a need to know?

• Pose 9/11 the emphasis is on need to share
  – User may not have access, but needs the data
    • Do we give the data to the user and then analyze the consequences
    • Do we analyze the consequences and then determine the actions to take
    • Do we simply not give the data to the user
    • What are risks involved?
Assured Information Sharing: Architecture

- Data/Policy for Coalition
  - Export Data/Policy
    - Component Data/Policy for Agency A
    - Component Data/Policy for Agency B
  - Export Data/Policy
    - Component Data/Policy for Agency C

Trustworthy Partners
Semi-Trustworthy Partners
Untrustworthy Partners
Our Approach

- Integrate the Medicaid claims data and mine the data; next enforce policies and determine how much information has been lost (Trustworthy partners); Prototype system
- Examine RBAC and UCON for coalitions (George Mason University – Ravi Sandhu)
- Apply game theory and probing to extract information from semi-trustworthy partners
- Trust for Peer to Peer Networks
- Conduct information operations (defensive and offensive) and determine the actions of an untrustworthy partner.
Architectural Elements of the Prototype

• **Policy Enforcement Point (PEP):**
  • Enforces policies on requests sent by the Web Service.
  • Translates this request into an XACML request; sends it to the PDP.

• **Policy Decision Point (PDP):**
  • Makes decisions regarding the request made by the web service.
  • Conveys the XACML request to the PEP.

**Policy Files:**

- Policy Files are written in XACML policy language. Policy Files specify rules for “Targets”. Each target is composed of 3 components: Subject, Resource and Action; each target is identified uniquely by its components taken together. The XACML request generated by the PEP contains the target. The PDP’s decision making capability lies in matching the target in the request file with the target in the policy file. These policy files are supplied by the owner of the databases (Entities in the coalition).

**Databases:**

- The entities participating in the coalition provide access to their databases.

*Migrating to RDF from XML, developing ontologies for policy interoperability*
Directions: Combining UCON and Data Sharing with Semantic Web (UMBC-UTSA-UTD-MIT)

- UMBC Research – In particular, Lalana Kagal’s PhD research
  - REI (RDF-based) Policy Language and Reasoning system
- How can we extend UCON and REI to develop a security model and reasoning (possibly based on PELLET) for data sharing environment?
Where are we and where do we go from here?

• Status
  – Tremendous progress the past 6 years
  – Tim Berners Lee’s vision now includes security and policy management for the semantic web

• Future
  – Semantic web will never be complete; it will continue to evolve
  – We cannot buy a semantic web product; need to integrate several technologies
  – As we make progress with semantic web technologies, need to investigate CPT as well as tools for policy engineering
  – Need to share environment will be a major application; Network centric enterprise services and the Global information grid of the DoD
  – Specialized semantic webs for multiple domains and different types of data (e.g, geospatial, sensor-web etc.)