Governing Delegation of Authority within SOA Environments Using KAoS

ROBERT L. SEDLMEYER, IPFW
JIM JACOBS, RAYTHEON
ANDRZEJ USZOK, IHMC
JAMES MILLIGAN, AFRL
Objectives

- Design, develop, and demonstrate a delegation of authority access control service for Service Oriented Architectures (SOA)
  - Capture delegation policies in a semantic model from which delegation policies can be specified
  - Develop delegation policy enforcement mechanisms
  - Maximize interoperability
  - Minimize impact on existing services
  - Demonstrate capability in an operationally relevant scenario
Operational Context

Human-in-the-loop Service Orchestration

Delegation of Authority Using KAoS
Technical Approach

- Assumptions
  - Authentication is handled separately from authorization
    - ...but authentication info carried in X.509 certificate
  - Delegation of authority is role-based, but could have contextual constraints
- Capture delegation policies in a semantic model from which delegation policies can be specified
  - Define ontologies that model operational domain, access control policies and delegation of authority policies
  - Utilize KAoS policy language
- Develop delegation policy enforcement mechanisms
  - Utilize KAoS policy engine
Technical Approach

- Maximize interoperability
  - OWL for semantic modeling
  - WSDL for web service description
- Minimize impact on existing services
  - Where possible, restrict access control specifications to web service interface components
  - Place run-time code for policy enforcement in handlers
About KAoS
Semantic Model: Using a Foundational Ontology

Delegation of Authority Using KAoS
Semantic Model: Delegation Entity Relations
Demonstration Architecture

Delegation of Authority Using KAoS
Authorization Control Flow

Client

invoke

Handler

handleMessage

Guard

authorize

checkPermission

Service

Apply policy

Extract invocation elements

Convert to ontology terms

Delegation of Authority Using KAoS
Example

- The Delegation web service implements role delegation operations
- Objective is to ensure that only personnel serving in ‘Senior...Officer’ roles have access its operations
- KAoS policy created using KPAT

“Any SeniorIntelligenceDutyOfficer is permitted to delegate the Targeteer role to any IntelligenceOfficer.”
Future Work

- Develop tool for automatically adding SAWSDL annotations (prototype exists)
- Extend and refine micro-theory of delegation-of-authority
- Enhance usability of KPAT to aid in construction of more sophisticated policies
- Make architecture more flexible so other authentication mechanisms are easily integrated
Demonstration Scenario

Delegation of Authority Using KAoS
Semantic Model: Delegation Entity Types

Delegation of Authority Using KAoS
Unlike domain web services, the Delegation service modifies ontology instance data maintained by KAoS
SAWSDL (Semantic Annotations for WSDL and XML Schema) is a W3C Recommendation for mapping WSDL components to ontology models.

Here we use it to define an XSL file that contains the translation rules for mapping the DelegateRole service operation to our policy ontology.
Delegation Service Annotation

@WebService(
    endpointInterface = "com.ray.ont.delegation.DelegationService",
    targetNamespace = "http://ont.ray.com/Delegation/",
    serviceName = "DelegationService",
    portName = "DelegationServiceSOAP",
    wsdlLocation = "/WEB-INF/wSDL/delegation.wsdl"
)

@HandlerChain(file = "/resources/handlerchain.xml")
public class DelegationServiceImpl implements DelegationService
{
    
}

@HandlerChain used to associate web service with JAX-WS Handler that contains KAoS Guard for performing policy checking
Mapping rules specify how service operation parameters are mapped to ontological terms