An Authorization Scenario for S-OGSA

Pinar Alper, Oscar Corcho, Michael Parkin, Ioannis Kotsiopoulos, Paolo Missier,
Sean Bechhofer, Carole Goble
School of Computer Science. University of Manchester
Oxford Road. M13 9PL. Manchester. United Kingdom
+44 (0)161 2756821
<penpecip, ocorcho, parkinm, ioannis, pmissier, seanb, carole>@cs.man.ac.uk

ABSTRACT
The Semantic Grid initiative aims to exploit knowledge in the Grid to increase the automation, interoperability and flexibility of Grid middleware and applications. To bring a principled approach to developing Semantic Grid Systems, and to outline their core capabilities and behaviors, we have devised a reference Semantic Grid Architecture called S-OGSA. We present the implementation of an S-OGSA observant semantically-enabled Grid authorization scenario, which demonstrates two aspects: 1) the roles of different middleware components, be them semantic or non-semantic, and 2) the utility of explicit semantics for undertaking an essential activity in the Grid: resource access control.

Keywords
Semantic Grid, architecture, authorization, S-OGSA.

1. Grid and Semantic Grid
The Grid vision is defined as the next generation infrastructure that will enable coordinated, well-controlled sharing of resources through dynamic, transient confederations known as Virtual Organizations (VOs). The roadmap for the realization of this vision is elaborated in the Open Grid Services Architecture (OGSA) [3]. The OGSA view of the Grid is comprised of a 3-tiered service-oriented architecture, where applications are brought together with Grid resources through a layer of middleware services. In the middleware layer OGSA defines certain service categories as core capabilities that Grids should have: Security, Resource Management, Execution Management, Optimization, Data, Information and Self-Management.

The Semantic Grid initiative aims to foster the progress in the realization of the Grid vision by extending it so that resource metadata is exposed and handled explicitly, and shared and managed via Grid protocols. To date, the application of Semantic technologies to the grid has been through exploratory experimentation where pioneering applications combining Grid and Semantic technologies were built.

In order to provide a systematic approach to building Semantic Grid systems and to outline their architectural organization and interaction patterns we have developed S-OGSA [1], a reference Semantic Grid architecture. S-OGSA extends the core capability set of OGSA by adding another category of services: the **Semantic Provisioning Services**. This category is decomposed at least into 4 sub-categories, namely Ontology Services, Metadata Services, Annotation Services and Reasoning services. All together these services are responsible for generating, managing and exploiting semantically-encoded metadata in the Grid. Furthermore, S-OGSA defines the **Semantically Aware Grid Services** as middleware services that provide an OGSA enumerated capability but differ from others by being capable of operating over explicit semantics.

Demos and Posters of the 3rd European Semantic Web conference (ESWC 2006), Budva, Montenegro, 11th-14th June, 2006

2. A Scenario in the Insurance Domain
We have implemented a role based access control system for an International Insurance Settlement Grid. The scenario requires that Customers should be allowed to make Insurance Policy Applications based on an evaluation of their previous car insurance and accident history. Close investigation of this scenario has revealed that it can be cast as a Grid authorization scenario. Prior to detailing the implementation we will briefly cover the background technologies that have been used in it.

2.1 Background on Authorization
Authorization falls in the scope of the Security category in OGSA. It is normally needed after the authentication of a client, so as to decide whether or not it can access a specific resource. The OGSA-AuthZ framework[i] describes different authorization models, architectures, components and systems that are currently used to support authorization in Grid applications.

Authorization decisions are based on the information available from the client and on the list of rules in a particular expression language that govern whether or not access requests will be approved, namely the authorization or access control policies. Among the languages used to represent authorization request/response messages and access control policies the most complete is XACML[ii].

Our implementation conforms to the OGSA-AuthZ framework and uses XACML to deliver request/response messages.

2.2 Declarative Approach to Authorization
Access control policies can be expressed in different ways and with different languages, and are usually distributed among the organizations belonging to a VO, so that we can talk about central and local policies. One common example of an access control policy is an access control list, which may control the access to

specific resources from individual users as well as from the groups they belong to and/or from the roles they play in the VO.

Access control lists and similar specifications are useful and work in many contexts, but they may not be sufficient when it comes to expressing more complex access control policies or when users, groups or roles cannot be easily expressed by enumeration, due to the existence of a large number of users or to the dynamicity of the user base. This is the situation in our insurance case study: we cannot pre-determine the eligibility of each customer for insurance application at the time the access control policies are made. We can, however, specify access control rules based on the roles that a customer plays. These roles are defined in terms of certain restrictions on the customer properties, and are obtained at run-time taking into account the customer's properties.

To define complex roles declaratively we have decided to use and extend the KAoS suite of ontologies[iii]. This ontology set contains descriptions about actors, groups, actions, resources, policy types, etc., and are extended with concepts related to the insurance domain (accidents, insurance companies, customers, etc.). Furthermore, we define customer roles that will be used to express the access control policies. Examples or such roles are GoodReputationDriver (a driver whose accident record contains at most one claim and who has been registered with an insurance company), BadReputationDriver (a driver whose accident record shows three or more claims), etc.

3. System Architecture

Figure 1 shows the component interactions: *1)* a Grid-enabled **Ontology Access Service**, WS-DAIOnt [2], responsible for hosting and managing the VO ontologies, *2)* a set of **Metadata Services,** powered by the Atlas P2P RDF storage and querying system [4], which store insurance customer metadata (Policy Information Point-PIP), *3)* the **Reasoning Service**, which is a description logic classifier used to infer customer roles based on their properties, *4)* the (XACML compliant) **Authorization Service**, which evaluates the access control function in the system (Policy Decision Point-PDP) and 5) the **CarFraud Service**, through which customers make their insurance policy applications (Policy Enforcement Point-PEP).

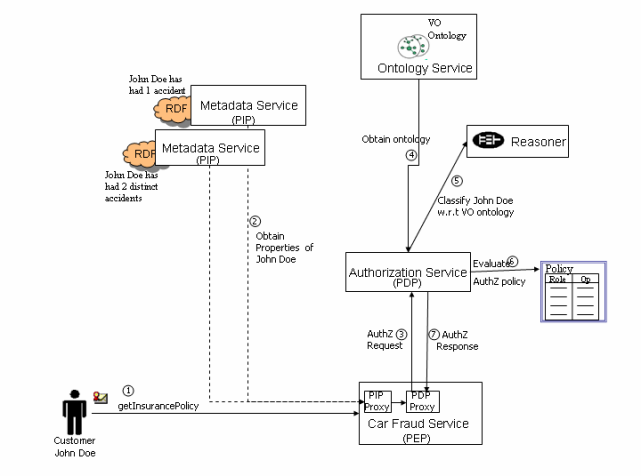

Figure 1. Authorisation scenario architecture.

The system operates as follows: An Insurance Customer makes a policy application by calling the associated method of the CarFraud Service (Step 1). The Car Fraud service delegates the eligibility evaluation of this person to the local authorization proxies. These proxies first contact the Metadata Service to obtain the properties of the customer (Step 2). Once customer metadata is gathered, an XACML Authorization request is generated for the Subject with an attribute containing the RDF based metadata regarding this subject (Step 3). Upon receiving the request the Authorization service contacts the ontology service to obtain the VO ontology containing the role definitions (Step 4). The ontology together with the customer metadata is passed onto the Reasoner to infer the role of the insurance customer (Step 5). Once the customer's roles are inferred the Authorization services evaluates the access control function using this information (Step 6) and returns a Permit/Deny/Indeterminate result to the Car Fraud Service's authorization proxy (Step 7).

4. Conclusions

With our implementation we have aimed to demonstrate:

The Semantic Grid Ecosystem of Services. The scenario demonstrates how a Semantically Aware Grid Service, namely the Authorization Service, uses some S-OGSA Semantic Provisioning Services, namely Metadata, Ontology and Reasoning, to deliver enhanced functionality via exploiting semantic metadata.

Grid Compliant Semantic Middleware. The services in the scenario are WS-RFiv compliant Grid services running on the Globus Toolkit 4v container. We believe it is important for the Semantic technologies and tools to be Grid enabled so as to enable their uptake by the Grid community.

Flexibility of Declarative Approaches for Authorization. The role-based authorization mechanism is based on dynamically inferring customer roles using a reasoner over OWL concept descriptions and instance data.

5. Acknowledgements

This work has been done as part of the EU IST Project OntoGrid (FP6-511513) and of the EU Marie Curie Fellowship RSSGRID (FP6-2002-Mobility-5-006668).

6. REFERENCES

[1] Corcho et al. An Overview of S-OGSA: a Reference Architecture for the Semantic Grid. Journal of Web Semantics. 2006. To appear

[2] Esteban et al. WS-DAIOnt: Ontology Access Provisioning in Grid Environments. GGF16 Semantic Grid workshop. 2006.

[3] Foster et al. The Open Grid Services Architecture Version 1.0. Specification, GGF OGSA Working Group, 2005.

[4] Koubarakis et al. Semantic Grid Resource Discovery using DHTs in Atlas. GGF16 Semantic Grid workshop. 2006.

[i] https://forge.gridforum.org/projects/ogsa-authz/
[ii] http://www.oasis-open.org/committees/xacml
[iii] http://ontology.ihmc.us/kaos.html
[iv] http://www.oasis-open.org/committees/wsrf
[v] http://www.globus.org/toolkit