# ACLs in OWL: practical reasoning about access

## [Extended abstract]

Norman Gray
VOTech Project, University of Leicester, UK; University of Glasgow, UK
norman@astro.gla.ac.uk

## ABSTRACT

The problems of describing, and once described determining, access to resources is one which maps easily to an ontology reasoning problem. We describe a flexible and dynamic access-control system, which naturally supports access-control federation, plus the prototype implementation of a practical system which helps disparate authorities manage and reason about user access to resources.

## Keywords

Authorization, OWL

## 1. INTRODUCTION

Even after the authentication problem has been solved, and it becomes straightforward to identify individuals reliably, we are still left with the authorisation problem, of reasoning about what a given individual is allowed access to. An individual might be allowed access to a resource in their own right, or because they are a member of a collaboration, because their institution is a member of a consortium, because they are located in a particular country, or for some other more elaborate reason. Management of the logic of access is typically distributed, so that the assertion that a particular group has access might be made by the owner of the resource, distinct from the authority who places a specific individual in that group. Add to this the observation that different categories of user might be given partial or otherwise limited access to a resource, and it is clear that managing access control lists (ACLs) is both logically intricate and of considerable interest for a distributed system such as the Semantic Web.

OWL is a very good match to this problem, more so than a rule-based system, since the question of whether a given user should be allowed access to a resource reduces very naturally to, firstly, a question of class subsumption, and secondly the question of whether the user can be deduced to be a member of the class of individuals allowed access. That class can be defined by the owner of the resource, in terms of a variety of other classes expressing institutional affiliations or membership of collaborations. An individual's membership of one of these latter classes can be asserted by a separate authority, and communicated to the resource owner as OWL assertions. Thus the important and taxing problem of federation [1] is more naturally approached from this direction than with other methodologies.

Setting up an ACL ontology in OWL is not a major challenge, though it would need to be as small as possible, and rigourously modular. The more interesting challenge is the practical question of how this functionality may be made available in such a way that asserting authorities may access the reasoning services and manage the sets of assertions conveniently, without necessarily having experience with, or much interest in, the Semantic Web.

Current approaches to this problem depend on the Shibboleth or PERMIS architectures (see [1] for a useful summary). Though carefully designed and implemented, these are designed with a rather static and hierarchical context in mind, and are therefore ill-suited to the more dynamic and fluid relationships of the Semantic Web. Articulating an access policy using an OWL ontology, on the other hand, has the following advantages:

- It is flexible: a very broad range of access policies may be expressed in logical form, since the expression as an OWL ontology is essentially (mobile) code.
- It is secure: it does not have the disadvantage of fully flexible mobile code, since it is a small restricted language, which may be reasoned about reliably.
- The approach can easily build on existing data sets, since an ACL ontology can add semantics to existing LDAP, SAML or other registration sources, reexpressed in RDF.
- Sets of assertions can be composed in a natural and controlled fashion.

In this poster we describe such a system, which we are currently prototyping as a component of the International Virtual Observatory Alliance's (IVOA [2]) security infrastructure.

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

## 2. IMPLEMENTATION

We have developed a prototype system which implements this approach.

A resource owner expresses their access policy by defining a class of individuals who are allowed a given access to the resource, such as reading from or writing to it. The owner then defines membership of that class in terms of concepts in this or other ontologies. For example, a university library might allow access to its electronic serials to staff members in that university, plus individuals who have borrowing rights in a partner university. Or a database might be available to researchers in institutions within EU countries. Crucially, the sets of assertions from the partner library, or the geographical information about institutions, can be made available from existing data sources re-expressed in OWL; they are available in discrete packets, so that the trust issues concerning the assertions' provenance and integrity are orthogonal and modular, and can be managed using existing techniques; and the architecture is flexible, requiring only limited coordination between actors, since the resource owner can decide what concepts in the 'foreign' ontology they wish to use to define their allow/disallow classes.

Central to this architecture is a reasoner (which in X.812 terms is a 'Policy Decision Point'). When an individual requests access to the resource (at an X.812 'Policy Enforcement Point'), the reasoner is consulted to determine whether the individual is provably in the class permitted access. In principle this would be an OWL-DL reasoner, but because the relevant ontology would be relatively stable in practice, it could be transformed off-line into a hierarchy which a simpler (and faster) reasoner could use. Confirming the feasibility of this is one of the remaining problems.

We have implemented an initial version of the required functionality in a REST-ful web service called Quaestor, available through a convenient and completely language-neutral API. This generic service manages multiple knowledgebases, composed of sets of client assertions, with the resulting merged ontology queriable through SPARQL. The service is implemented using the Jena and ARQ frameworks, and runs inside the Tomcat servlet engine.

At present (May 2006), the implementation is at a prototype stage. Possible future developments include:

1. embedding the resulting service in a production system;
2. creating simple client applications which assist authorities' authoring of the relevant assertion sets, without obliging users to learn OWL or learn to use SW tools;
3. further refactoring of the access-control ontology to separate generically useful concepts from ones specific to a particular resource;
4. persisting the uploaded models, perhaps using the Manchester Instance Store [3];
5. signing ontologies, so that only certain authorities may update authentication information.

Experience in the coming months, plus confrontation with the use-cases and security infrastructure of the IVOA, will help us determine whether these are indeed in roughly priority order. We acknowledge that task 2 would potentially be a large and challenging task (though it is a path already trodden by the developers of the Gene Ontology [4]), but we expect that there will be a rather large class of simple cases which will need only basic automation, so that it may turn out reasonable for the more complicated, rarer, logic programming tasks to be engineered by hand; finding out how true this is in practice is one of the important goals of our project. Crucially, such clients are only for convenience, and any authority which can in fact generate RDF can interact with the service naturally and directly.

Task 5, though part of a large and important problem in general [5], will be postponable for us, given our overall system design. We expect in any case that it can be factored out from the reasoning aspects of the design.

By the later part of this year we expect to have demonstrated the integration of a service providing SW-style reasoning to a large non-SW architecture.

## 3. REFERENCES


[1] J Watt, Richard O Sinnott, and A J Stell. Dynamic privilege management infrastructures utilising secure attribute exchange. In *Proceedings of the UK e-Science All Hands Meeting*, 2005.

[2] International virtual observatory alliance [online]. Web: `http://www.ivoa.net`.

[3] Sean Bechhofer, Ian Horrocks, and Daniele Turi. The OWL instance store: System description. In *Proceedings CADE-20*, Lecture Notes in Computer Science. Springer, 2005.

[4] Gene Ontology Consortium. Creating the gene ontology resource: design and implementation. *Genome research*, 11(8):1425–1433, 2001.

[5] XML-Signature requirements [online]. 1999 [cited February 2006]. Web: `http://www.w3.org/TR/xmldsig-requirements`.