From Community to Resource Policies: A Tool for the Automatic Refinement of VO Policies

Benjamin Aziz[†], Alvaro Arenas[†], Shirley Crompton[‡], Brian Matthews[†]

[†]e-Science Centre, STFC Rutherford Appleton Laboratory, Oxford OX11 0QX
[‡]e-Science Centre, STFC Daresbury Laboratory, Warrington WA4 4AD

1 VO Security Policies

Virtual Organisations (VOs) represent a common abstraction for describing computations and storage in a Grid environment. A VO is considered as a collaborative environment in which real organisations combine to offer applications and resources to users. Such resources could be storage capabilities, software services or simply computational power represented as physical machines. A *VO policy* is a statement about the expected normal behaviour of applications, resources and users in the whole VO. A VO policy is usually written in terms of VO-wide concepts; possibly all the running applications, resources and users in the VO. VO policies are crucial to the correct operation of large-scale Grid-based VOs, dealing with issues of utilisation measurement, accounting, security etc [6].

Traditionally, a VO policy is seen as the collection of the individual resource policies. This view is both easily justifiable, in the sense that the VO consists in fact of the resources themselves, and passive, in the sense that no extra effort is required to manage and enforce VO policies that attempt to give a richer statement about the overall acceptable VO behaviour. More recently, research in security policies (for example,[1]) has moved on to the adoption of VO policies that express more than what the mere individual resource policies have been set up to express. This view is more interesting in the sense that it starts from the requirements of the VO itself, even prior to its population with resources, rather than starting from the policy constraints of resources themselves.

A problem, however, that arises with this latter view is regarding how the enforcement of the VO policy is achieved. There are at least two solutions: The first is to enforce the VO policy directly using a VO-wide Policy Enforcement Point (PEP). This case has the limitation of having a single centralised point of failure (such as operational or semantic failures). The second solution, which we adopt in this paper, is based on refining VO policies from their VO-wide representation down to their computational-level representation at individual resources [3]. This latter approach avoids the development of a centralised VO PEP but it is constrained by the limitations of the refinement algorithm and the fact that not all VO policies can be refined. We restrict our treatment in this paper to VO security policies, i.e. policies concerned with the access control of resources by users, that can be refined. Furthermore, we allow our refinement to be guided by predefined VO-to-resource hierarchies in a similar manner to [5], who assume the preexistence of a data refinement structure mapping high level VO concepts to low level computational ones.

2 The GridTrust Policy Refinement Tool

GridTrust[1] is a EU FP6 project, which provides a technical framework consisting of theoretical models, design tools and Globus-based software services for the enhancement of trust and security in Grid computing. One of the main research novelties in GridTrust is the adoption of a vertical view of trust and security in Grid systems, which starts at the level of requirements engineering and system design, and ends at the applications, middleware and foundation levels. For example, access and usage control VO policies generated from domain-specific security requirements can be refined to the computational levels (middleware and foundational) were they are enforced by the policy enforcement infrastructure.

To facilitate the refinement and deployment of VO policies to the computational level of individual resources, an Eclipse-based tool was designed and built in the project. The architecture of the tool is shown in Figure 1. The tool

[1] www.gridtrust.eu

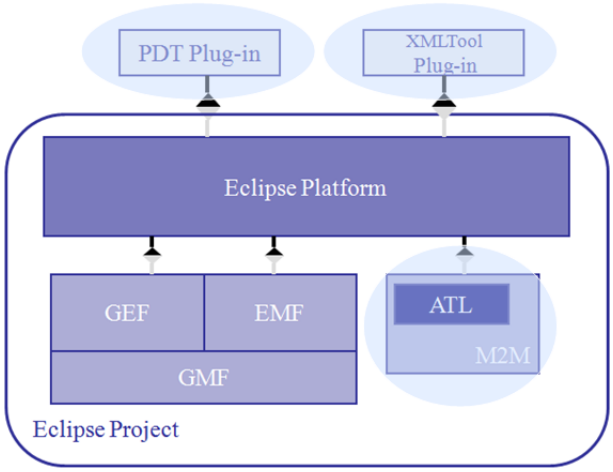

Figure 1: Architecture of the GridTrust Policy Refinement Tool

has the objective of providing a user-friendly environment for the specification and refinement of VO security policies written in XACML version 2.0 [4]. It provides three main functionalities in the form of Eclipse-based plugins:

- A Visual Editor for building VO-to-resource hierarchies. The idea behind this editor is to provide for the user a graphical palette, which consists of VO entities with attributes representing their equivalent in terms of computational entities. For example, a *VO process* called Edit could be expressed in terms of the *computational commands*, read, write and print.
- An XML Editor based on the Eclipse WTP Web Standard Tools[2] for writing XACML policies.
- A couple of transformation algorithms implementing VO-to-resource policy refinement and then deployment at the resource level. The algorithms are based on the Atlas Transformation Language (ATL)[3], an Eclipse-based modeling environment.

We believe that these functionalities will ease the task of VO policy administrators and system designers in specifying and refining VO security policies.

3 Case Study: Distributed Knowledge Management

The main application scenario driving the development of the GridTrust Policy Refinement Tool was inspired by one of the case studies in the project, namely *distributed knowledge management* [2], applied to the domain of large-scale geographic map processing. In this use case, the VO represents a business-level workflow consisting of the Edit, Merge, Validate and Publish processes. Users can edit individual aspects of the geographic maps according to certain conditions. Once all users have committed their versions, the administrator of the maps can then merge the different versions, send them to the reviewers for validation and final publication of the map. The case study utilised OGFs Storage Resource Management protocol [4] for linking the Alfresco [5] knowledge management application to the Grid.

In this scenario, the VO-to-resource hierarchy specifies that processes are mapped to computational commands, VO users are mapped to user identifiers and VO resources (geographic maps) are mapped to computational resources (directories or files). Then a single XACML-based VO policy stating which users can apply which processes on which geographic maps is refined using our tool (and guided by the VO-to-resource hierarchy) to:

1) first, a single equivalent policy in terms of the computational concepts (user identifiers, resources and commands), and

[2] http://www.eclipse.org/webtools/
[3] http://www.eclipse.org/m2m/atl/
[4] https://sdm.lbl.gov/srm-wg/
[5] www.alfresco.com

2) second, multiple policies each similar to the policy in 1) except that these refer to at most one resource.

The benefit of the result of 1) is a policy that can be reasoned on easily (e.g. it can be analysed for conflicts etc.), whereas the result of the second allows the original VO policy to be deployed at each individual resource.

References

[1] Benjamin Aziz, Alvaro Arenas, Fabio Martinelli, Ilaria Matteucci, and Paolo Mori. Controlling usage in business process workflows through fine-grained security policies. In *TrustBus'08: Proceedings of the 5th international conference on Trust, Privacy and Security in Digital Business*, pages 100–117, Berlin, Heidelberg, 2008. Springer-Verlag.

[2] The GridTrust Consortium. Specification of Applications and Test Cases. GridTrust Deliverable D5.1, 2007.

[3] Jonathan D. Moffett and Morris S. Sloman. Policy hierarchies for distributed system management. *IEEE Journal of Selected Areas in Communications, Special Issue on Network Management*, 11(9), 11 1993.

[4] Tim Moses. eXtensible Access Control Markup Language 3 (XACML) Version 2.0. OASIS Standard, 2005.

[5] L. Su, D. W. Chadwick, A. Basden, and J. A Cunningham. Automated decomposition of access control policies. In *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks, POLICY 2005*. IEEE, 2005.

[6] G. Wasson and M. Humphrey. Toward explicit policy management for virtual organisations. *4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY2003)*, 2003.