

XACML Profile and Implementation for Authorization Interoperability between OSG and EGEE

G. Garzoglio,^{1*} I. Alderman², M. Altunay,¹ R. Ananthakrishnan,³ J. Bester,³ K. Chadwick,¹ V. Ciaschini,⁴ Y. Demchenko,⁵ A. Ferraro,⁴ A. Forti,⁴ D. Groep,⁶ T. D. Hesselroth,¹ J. Hover,⁷ O. Koeroo,⁶ C. La Joie,⁸ T. Levshina,¹ Z. Miller,² J. Packard,⁷ H. Sagehaug,⁹ I. Sfiligoi,¹ N. Sharma,¹ S. Timm,¹ F. Siebenlist,³ V. Venturi,⁴ J. Weigand¹

¹*Fermi National Accelerator Laboratory, Batavia, IL, USA*

²*University of Wisconsin, Madison, WI, USA*

³*Argonne National Laboratory, Argonne, IL, USA*

⁴*INFN CNAF, Bologna, Italy*

⁵*University of Amsterdam, Amsterdam, The Netherlands*

⁶*NIKHEF, Amsterdam, The Netherlands*

⁷*Brookhaven National Laboratory, Upton, NY, USA*

⁸*SWITCH, Zürich, Switzerland*

⁹*BCCS, Bergen, Norway*

E-mail: garzoglio@fnal.gov

Abstract. The Open Science Grid (OSG) and the Enabling Grids for E-science (EGEE) have a common security model, based on Public Key Infrastructure. Grid resources grant access to users because of their membership in a Virtual Organization (VO), rather than on personal identity. Users push VO membership information to resources in the form of identity attributes, thus declaring that resources will be consumed on behalf of a specific group inside the organizational structure of the VO. Resources contact an access policies repository, centralized at each site, to grant the appropriate privileges for that VO group. Before the work in this paper, despite the commonality of the model, OSG and EGEE used different protocols for the communication between resources and the policy repositories. Hence, middleware developed for one Grid could not naturally be deployed on the other Grid, since the authorization module of the middleware would have to be enhanced to support the other Grid's communication protocol. In addition, maintenance and support for different authorization call-out protocols represents a duplication of effort for our relatively small community. To address these issues, OSG and EGEE initiated a joint project on authorization interoperability. The project defined a common communication protocol and attribute identity profile for authorization call-out and provided implementation and integration with major Grid middleware. The activity had resonance with middleware development communities, such as the Globus Toolkit and Condor, who decided to join the collaboration and contribute requirements and software. In this paper, we discuss the main elements of the profile, its implementation, and deployment in EGEE and OSG. We focus in particular on the operations of the authorization infrastructures of both Grids.

* To whom any correspondence should be addressed.

1. Introduction

The Open Science Grid (OSG) [1] and Enabling Grids for E-science (EGEE) [2] are two large international collaborations of research institutions and national laboratories that provide distributed computing infrastructures for e-Science projects and experiments. Both Grids are independently financed and managed. Therefore, despite the many shared middleware components, the development of different solutions for the two Grids formed technological gaps in different service domains. These gaps do not serve well the common user communities and the attempts to standardize inter-Grid policies. It is to bridge these gaps that EGEE and OSG have started, throughout the years, a range of joint interoperability projects.

The Authorization Interoperability project [3] was initiated and charged with mitigating the differences between the access authorization infrastructures of both Grids. The project had three main goals:

1. Standardize the authorization call-out protocol between resource gateways and policy decision points;
2. Implement the standard protocol in order to enable the reuse of authorization code between Grids;
3. Enable the deployment of middleware developed for the OSG in the EGEE authorization infrastructure and vice versa, by changing only middleware configuration.

The project was started in Feb 2007 and involved groups from OSG, EGEE, and major middleware providers such as the Globus Toolkit [4] and Condor [5]. At the time of this writing, the interoperable authorization infrastructure is being deployed on both Grids.

This paper is organized as follows. Chapter 2 discusses the OSG and EGEE authorization infrastructures. Chapter 3 summarizes the main elements of the authorization interoperability profile. Chapter 4 discusses the implementation of the profile and how the architectures of the authorization infrastructures are simplified by the adoption of a common set of modules. Chapter 5 discusses the operations of the authorization infrastructures. Chapter 6 concludes the paper with a brief summary.

2. OSG and EGEE Authorization Infrastructures

OSG and EGEE have similar authorization models. Both Grids use X509 proxies and end-entity certificates for single sign-on and delegation. Through inter-Grid organizations, such as the Joint Security Policy Group (JSPG) [6], OSG and EGEE trust a common set of Certificate Authorities, which enable user authentication across Grids.

Access to resources depends on user identity as well as on user membership to a community, also called a Virtual Organization (VO), associated to the Grid. These VO-specific attributes are asserted by VO Membership Services (VOMS) [7] and always pushed by the users to the resources in the form of an Attribute Certificate (AC), together with the user certificate. This bundle of user certificate and AC is often called the extended-credentials.

VOs are organized according to a hierarchical structure of user groups and group roles. For example, a VO might define a “simulation” group, with an “administrator” role, comprising different subgroups for different simulation domains. User registration systems, such as the Virtual Organization Registration Management Service (VOMRS) [8] or the administrative interface of VOMS (VOMS-Admin) [9], implement the membership registration workflow, allowing a hierarchy of VO administrators to grant a user membership in a group or group role. This membership information is then maintained and published by the VOMS systems.

At sites, the extended credentials are mapped to local privileges. These privileges encompass access to several different resources, such as batch systems, local file systems, and storage systems, etc. Since many of these systems do not natively handle Grid credentials, privileges are typically enforced through standard operating system mechanisms, such as Unix ID (UID) / Group ID (GID). In order to maintain the consistency of such mapping within related resources, such as a cluster of computers, resource gateways delegate the Grid-to-local credentials mapping to a site-wide Policy Decision Point (PDP). The standardization of the communication protocol between PDPs and the

Resource Gateways, also called Policy Enforcement Points (PEPs), was the focus of the Authorization Interoperability project. Figure 1 shows a diagram of the OSG and EGEE authorization infrastructures.

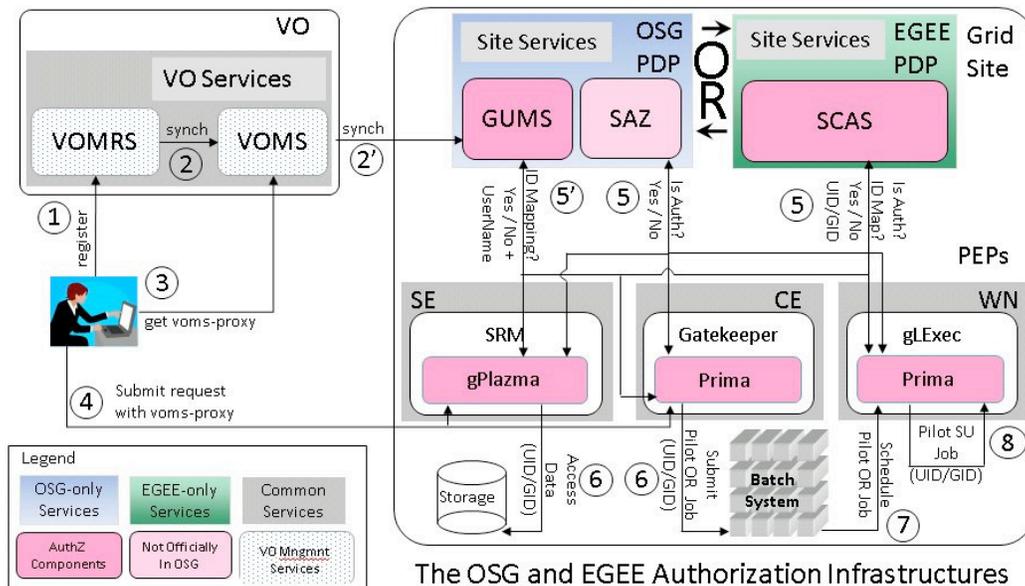


Figure 1. The OSG and EGEE authorization infrastructures. A user registers with a VO (1). The VO membership information is synchronized with a VOMS server (2) and, in the OSG case, with site PDPs (2'). The user extends her credentials with VO membership attributes (3), before submitting an access request to a resource gateway (4). In this diagram, the gateways shown are SRM to access a Storage Elements (SE), Globus Gatekeeper to access a Computing Element (CE), and gLExec to access a Worker Nodes (WN). Resource gateways call-out to the site Policy Decision Point (SCAS for EGEE, GUMS and SAZ for OSG) using specific authorization modules (Prima, gPlazma, SCAS client) (5 and 5') to get the mapping of Grid credentials to a local identity. The standardization of this call-out protocol is the subject of this paper. If authorization is granted, the resource gateway forwards the request to the appropriate underlying resource (6, 7, 8).

The problem of consistency of the access within a set of resources was particularly acute in EGEE before this work, in the case of “pilot-based” job submissions. This is a technique used by workload management systems [10, 11] to procure a pool of Grid resources and make them part of a virtual batch system. In this scheme, pilot jobs reserve a worker node and user jobs make use of it. Because the identity of a user job is typically different from the identity of a pilot job, in order to enable accounting of resource usage and proper privilege setting, the worker node must act as a resource gateway and appropriately set local identities for pilot and user jobs. These requests can come from thousands of worker nodes in a cluster; and if the mapping policies are local to each node, as in the case of EGEE before this work (Sec. 5.2), the consistency of the information is potentially a problem. Centralizing the authorization decision at the PDP is the technique used to enforce policy consistency. The drawback of this approach is that the high number of requests to the PDP makes the operations of the infrastructure more challenging (Sec. 5.1).

3. Authorization Interoperability Profile

In order to enable interoperability of the communication between PEP and PDP in OSG and EGEE, our collaboration decided to adopt common standards whenever possible. The authorization interoperability protocol is based on the Security Assertion Markup Language (SAML) [12] profile of the eXtensible Access Control Markup Language (XACML) [13], both standards from the Organization for the Advancement of Structured Information Standards (OASIS). While XACML

defines the XML-structures that are exchanged between PEP and PDP to communicate the security context and the rendered authorization decision, SAML defines the on-the-wire messages that envelope the XACML PEP/PDP conversation.

In the XACML model, the PEP requests an authorization decision to the PDP and receives a response. The request includes information on the identity of the requestor (“Subject”) and the “Action” to be performed on a given “Resource” for a certain “Environment” (e.g. at a certain time). The response includes an authorization “Decision” and, if it Permits access, a series of “Obligations” that the PEP must fulfill to comply with the access policies.

To allow our community to express domain-specific authorization policies, we have augmented the XACML standard with an interoperability profile [14] that standardizes names, values, and semantics for common attributes and obligations. These define the vocabulary used by OSG and EGEE to express properties of a Subject, of a Resource, etc. Details of this vocabulary are described elsewhere [3, 14]. In summary, every attribute defined in this profile is scoped within the URI namespace <http://authz-interop.org/xacml>. In turn, attributes for each XACML context (e.g. Subject, Resource, Obligation, etc.) are defined within a context-specific namespace (e.g. <http://authz-interop.org/xacml/subject>, <http://authz-interop.org/xacml/resource>, etc.).

These attributes define characteristics of

- the **subject**, such as subject identity (subject-x509-id), membership to a VO (subject-vo) and VO groups / roles (voms-fqan);
- the **resource**, such as resource type (resource-id as CE, SE, WN) and other characteristics to uniquely identify the resource, such as its identity (resource-x509-id) and host name (dns-host-name);
- the **action**, such as action type (action-id as queue, execute-now, access file) with possible additional details (rsl-string);
- the **environment**, such as the obligations supported by the PEP or the pilot job subject context;
- the **obligations**, such as UID/GID or Username, defining the appropriate privileges for the access, or storage-related obligations, such as path restrictions (rootpath and homedir), request priority, etc.

Similar profiles are being defined by other groups, such as the Open Grid Security Architecture Authorization Working Group (OGSA-AuthZ WG) [15] of the Open Grid Forum (OGF). This group in particular, has the goal of standardizing authorization call-out across several different security models, ours being one of them. As the profiles worked on by this group become standard, we can envision their integration with our implementations.

4. Implementation

The authorization call-out infrastructure has been implemented as a layered architecture. Figure 2 shows a diagram of the architecture for EGEE and OSG. The resource gateways (bottom green components) receive access requests from users and act as Policy Enforcement Points of the authorization infrastructure. Authorization modules (“call-out” yellow components) implement the specific authorization call-out interface of a resource gateway. Some gateways support a common authorization interface and reuse the same call-out module. This is the case, for example, of the resource gateways from the Globus Toolkit (pre-Web Services Gatekeeper and GridFTP). Gateways implemented in different languages (e.g. C or Java) instead have by necessity different call-out module implementations. The call-out module implementations are typically different for OSG and EGEE; however, in the case of OSG, for example, these modules are thin layers built on top of common authorization libraries (“XACML lib” yellow components). These libraries implement the authorization interoperability profile and rely on underlying libraries that implement the SAML v2 profile of XACML v2. Through these libraries the XACML authorization message is composed and sent over the wire to the Policy Decision Points (top pink components). Since the content of the message is built according to the specifications of the authorization interoperability profile, these messages are understood by PDPs built initially for OSG / US sites (e.g. the GUMS mapping service

[16] or the Site Authorization Service (SAZ) [17]) or for EGEE (Site Central Authorization Service (SCAS) [18]).

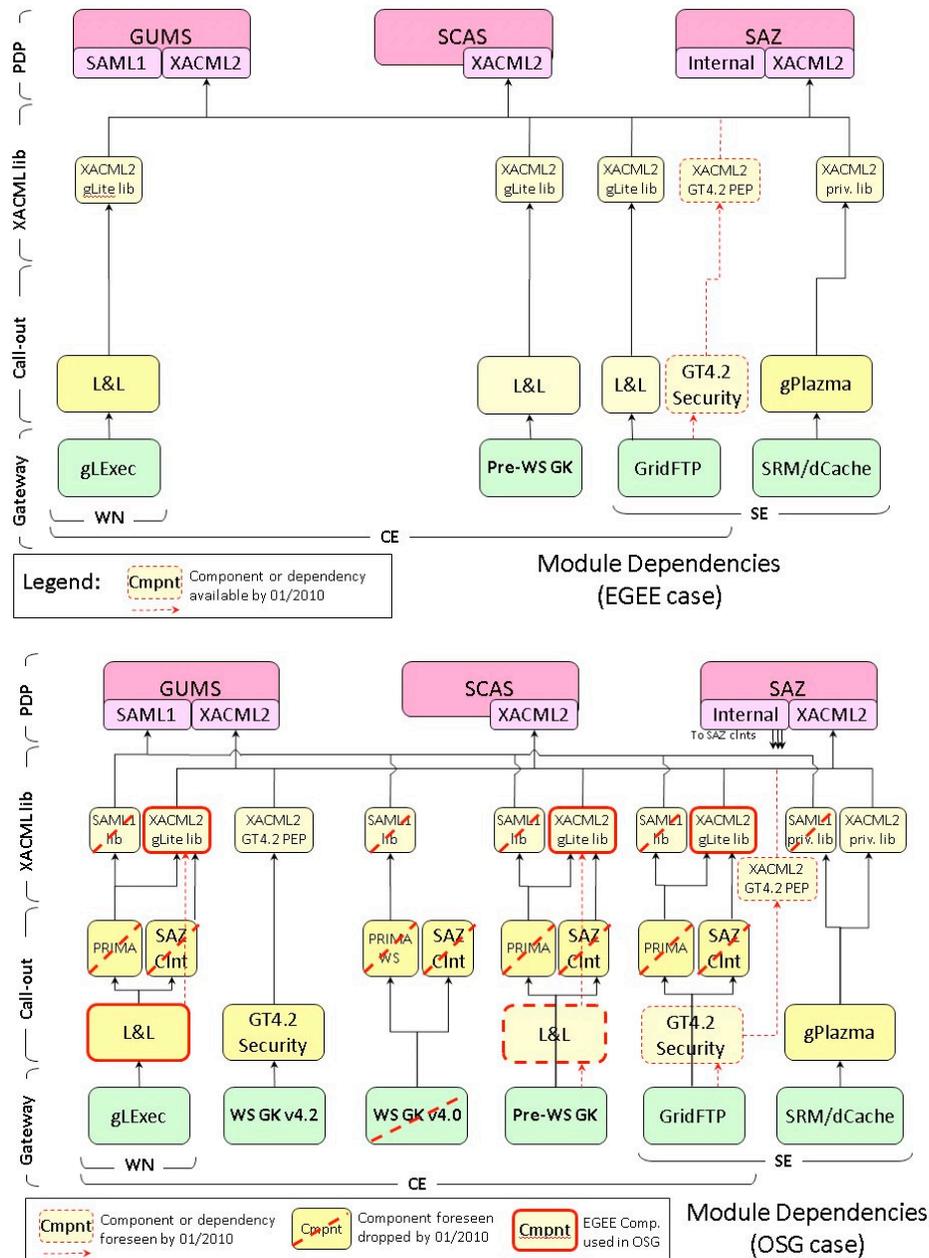


Figure 2. Architectural diagram of the authorization call-out infrastructure for EGEE (top) and OSG (bottom). The implementations of the call-out modules are different, but the libraries implementing the authorization interoperability profile (XACML lib) are common. In 2010, since most resource gateways for OSG and EGEE are common, call-out modules can be shared as the XACML libraries implementations.

The middleware that implements most resource gateways for OSG and EGEE is shared between the two Grids. This includes the pre-Web Services Globus Gatekeeper and GridFTP, SRM/dCache storage elements, and the gLExec user-switching utility for the worker nodes. In the future, there is the

possibility, therefore, to share the implementation of not only the XACML profiles libraries, but also of the call-out modules. Especially for OSG, this would greatly simplify the infrastructure. Figure 2 (bottom) shows with dashed lines the components of the OSG architecture that could be dropped or adopted in 2010, should call-out modules be shared with EGEE.

The infrastructure would be simplified, first, because the GUMS and SAZ PDPs, typical of an OSG deployment, would be called through a single module per gateway, as opposed to one module for GUMS and one for SAZ, as today. Second, in 2010, we expect to initially deprecate and then stop supporting the legacy call-out interfaces to the PDPs (marked as “SAML1” and “Internal” in the diagrams). Third, because of the participation of the Globus Toolkit team in the Authorization Interoperability project, we expect to be able to deploy natively supported call-out modules for the Web Services Gatekeeper and GridFTP (“GT4.2 Security” and “XACML2 GT4.2 PEP” yellow components).

This simplification and the sharing of the code would highly reduce the cost of maintaining the authorization infrastructure for our relatively small OSG and EGEE developer communities.

5. Operations of the Authorization Infrastructures

Both EGEE and OSG have had years of experience operating authorization infrastructures from user registration to resource access authorization. The new Authorization Interoperability infrastructure has undergone independent integration tests and achieved certification on both Grids. As the new Authorization Interoperability protocol is being deployed on OSG and EGEE, we discuss hereby the current practices in operating the authorization infrastructures and comment on the foreseen minor impact of the new protocol on these operations.

5.1. OSG Operations

Access control relies on the authentication infrastructure for cryptographic validation of the authorization assertions. In the early years of OSG and EGEE, the public certificates of the trusted CAs and VOMS servers had to be distributed to all running Grid services. VOMS certificates were mainly used by resource gateways to validate the AC embedded in the extended user credentials. This validation ascertains that a user can rightfully claim membership to a certain VO, thus use resources on its behalf. However, maintaining and distributing the dozens of trusted VOMS certificates was operationally challenging: given their numbers, statistically, VOMS service certificates would require renewal / redistribution almost weekly. This prompted OSG to implement a validation approach different from cryptographic validation. The GUMS authorization server hourly synchronizes with the membership database of all supported VOMS servers. When the user submits her credentials to access a resource, the GUMS server checks that the user’s claim of membership to a group is reflected in its copy of the appropriate VOMS database. This way, even if AC were forged, only legitimate group memberships could be claimed and be authorized. Clearly, cryptographic validation of AC has additional advantages with respect to this membership synchronization mechanism and they are discussed in Sec. 5.2. Despite recent improvements of the VOMS server, which effectively lift the requirement of distributing the VOMS certificates, OSG still adopts this alternative validation mechanism.

In OSG, the Grid-to-local identity mapping rules are typically managed by a site-central GUMS server¹, the equivalent of SCAS for EGEE (Sec. 5.2). This server is controlled by a configuration file, which lists all VOs supported by the Grid, together with the VOMS addresses, VO groups and roles, and the mapping policy preferred by the VOs². This list is maintained by the Grid operations center and released periodically as a template for sites to update their GUMS instances. The “update”

¹ Some sites still rely on gridmap-files for access control; however, the number of these sites is shrinking, as they cannot support role-based access to resources, an important authorization paradigm for most large VOs.

² An example of such policy is that users within a group should be all mapped to the same accounts (*group account*); another example is that users within a group should be mapped to individual accounts (*pool account*).

operation and, in particular, the merging of GUMS configurations is currently done primarily by hand and will benefit from a future planned automation. The VO information in the template is available from VOMS servers, but it is typically updated by the Grid operators upon explicit request by the VO administrators. The template release process and, in particular, the communication between VO and Grid operations will also benefit from automation. In addition, the updating of the mapping policies can be particularly complicated for Storage Elements, which still rely on local mapping files to augment the GUMS decisions with storage-specific attributes, such as allowed Root Path for a user. To help improve the propagation of the desired VO policies to the sites, reducing the role of Grid operations, we are working on the Scalable Virtual Organization Policy Management Environment (SVOPME) [20]. While the operations of the current infrastructure could be streamlined, they will not be affected by the adoption of the new authorization interoperability protocol.

The current lack of automation in these operational processes results in delays in the adoption of new templates by sites. This can pose security risks, in particular when the Grid removes support for a VO. To mitigate these risks, sites like Fermilab use user banning tools, such as the Site Authorization Service (SAZ), the equivalent of LCAS in EGEE (Sec. 5.2). This service can ban access to users, to entire VOs, VO groups and roles, and to all certificates signed by a given Certificate Authority. This mechanism gives site administrators full control to user access and it is faster than relying on standard Certificate Authority banning mechanisms, such as Certificate Revocation Lists. In addition, it is a convenient tool to temporarily remove access to a user in good standing with security, but that accidentally took disruptive actions with her jobs towards the infrastructure. The ability for the central Grid operations to ban users on the Grid automatically is a future goal for both OSG and EGEE.

The operations of GUMS and SAZ have been challenged in recent years by the introduction of gLExec at the worker nodes (Sec. 2). Because of the large number of worker nodes in a cluster, the peak number of authorization requests per second to these PDP has dramatically increased with respect to the original infrastructure with only computing and storage gateways. PDPs must be robust against peaks of authorization requests, typically when synchronized events occur in the cluster. For example, when a user decides to remove hundreds of running jobs, the batch system issues the cancel command almost concurrently. This results in a peak of authorization requests, which PDPs must be able to withstand, using caching and queuing techniques. To give a sense of scale, the GUMS server of a large OSG site (FermiGrid) can support a steady rate of authorization requests of at least 500,000 calls per day (~ 6 Hz). The system has been tested to withstand peaks of 10 Million calls per day (116 Hz). To achieve these rates, FermiGrid uses two load-balanced High Availability GUMS servers.

The introduction of the authorization interoperability infrastructure changes the communication on the wire for authorization assertions and decision and will not affect the operational properties of the system.

5.2. EGEE Operations

Prior to this project, the EGEE deployment was based on access controls deployed locally to the service being protected. The services involved in access control (L&L in Fig. 2) were the Local Centre Authorization Service (LCAS), for authorization decisions, and the Local Credential Mapping Service (LCMAPS), for translating 'Grid' credentials to Unix domains users and groups [21]. LCAS makes binary authorization decisions and provides banning capabilities that can be based on subject DN and VOMS FQANs – the latter through the use of Global Access Control List policies. LCMAPS procures local credentials, such as AFS tokens and POSIX credentials ('acquisition') and subsequently applies these to the running process, for example through *setuid* and *initgroups* system calls ("enforcement"). Both systems are "pluggable frameworks", with a state machine invoking a sequence of 'plug-ins' such as attribute extraction or Unix credential enforcement.

Both EGEE and OSG use the same technology and processes to manage VO membership as described in Sec. 5.1, but the assertions are processed differently. In EGEE, VOMS embedded Attribute Certificates (ACs), signed by the VO's VOMS server and bound to the end-entity certificate of the subject, are fully validated and subsequently used to determine VO membership. The resource

owner makes access control and mapping decisions based on the VOMS FQANs sent with the subject proxy credentials, and holds its policies only in the form of FQANs and subject DNs. In particular, contrary to the OSG model described in Sec. 5.1, the site does not synchronize any content from the VOMS database, as it infers VO attributes from the signed assertions. This also means that, in the EGEE case, the VOMS AC validity must be checked as part of the proxy chain validation. Before the VOMS implementation lifted the requirement for distributing VOMS end-entity certificates, this system relied on the EGEE configuration infrastructure to distribute the certificates for the validation. This configuration infrastructure greatly simplified the operations of the authorization system. Authorization decisions and VO membership are logged when a resource is actually accessed.

The site configuration to support specific VOs in EGEE is facilitated through the use of the YAIM configuration tool [22] and the “VO information cards”, managed on the CIC Portal [23]. The CIC portal acts as a central coordination point for all VOs in use within EGEE and most national-Grid based VOs in Europe. An on-line tool [24] on the portal allows sites to select the VOs that they wish to support and generate the corresponding site-local configuration based on the VOMS group and roles in use within these VOs. Other mechanisms to configure VO support are also in use within EGEE, in particular VO-specific configuration being integrated in the fabric management system Quattor [25].

Specific bans can be applied any time by the site at the authorization stage by LCAS, based on subject DN or specific VOMS groups and roles. Similarly to SAZ in OSG (Sec. 5.1), these bans take precedence over other site policies with regard to VOMS-based authorization and mapping. Each site in EGEE and OSG autonomously makes banning decisions because there is no currently deployed central distribution of infrastructure-wide ban lists or any central banning service that can be dynamically incorporated in site policies – although such a service could be created using the existing components discussed in this paper.

The Site Central Authorization Service (SCAS) allows ‘remote’ invocation of the EGEE authorization and mapping subsystems, with capabilities similarly to GUMS in OSG. The SCAS service itself is a wrapping of LCAS and LCMAPS, and basically provides a ‘recursive’ implementation thereof, with configuration changes scoping it to access control and credential acquisition only. Actual enforcement of the credential mappings (e.g. the *setuid* call) must remain local to the invoking service, since these have to be done in-process. The SCAS service employs a secure channel with client-side authentication, using either a full host credential or a proxy certificate. The SCAS client credential is verified and the client itself (optionally) authorized by the SCAS service before the request is processed. The SCAS service supports only the XACML profile described in this paper.

The accompanying *scas-client* LCMAPS plug-in provides the capability to communicate with any service using the interoperable XACML profile described in this paper, such as SCAS and GUMS. Alongside potentially other acquisition plug-ins and the required enforcement plug-ins, it both returns the authorization decision and populates the credential store of the LCMAPS framework with the returned obligations. In order to support the EGEE premise of accepting signed VOMS ACs and expressing access control based on FQANs, while at the same time, ensuring interoperability with OSG, the actual validation of the credential chain with its VOMS ACs should be retained at the ‘client’ (business service) end, through the *verify-proxy* LCMAPS plug-in. The client has a built-in round-robin and fail-over mechanism, although the LCMAPS policy language also can be used to fail-over to node-local policies. In this way, the introduction of the SCAS service in the production infrastructure is a deployment-time configuration choice, and can be used in addition to the existing authorization systems. The plug-in can also be used with all services that use LCMAPS as the credential mapping system.

Certification of the SCAS server and the associated client for use in EGEE was completed in April 2009 based on the “gLExec on worker node” scenario, although deployment in conjunction with other services is largely equivalent.

The interaction from a “*scas-client*” LCMAPS plug-in to a SCAS service is using SSL, without the “session reuse” feature. Each SCAS service connection starts with a full chain mutual

authentication. Each interaction also needs to perform the marshalling and unmarshalling of the SAML v2 and XACML v2 message parts and the content needs to be handled by the LCAS and LCMAPS systems on the server side and enforced at the client side. Table 1 shows the current SCAS performance.

	# clients	cum. server load	cum. server peak	# interactions per sec.
1 SCAS daemon	15	10,00%	13,00%	~24Hz
1 SCAS daemon	30	15,00%	18,00%	~31Hz
2 SCAS daemons	30	20,00%	25,00%	~42Hz
2 SCAS daemons	60	40,00%	48,00%	~66Hz
4 SCAS daemons	60	60,00%	65,00%	~80Hz

Table 1. SCAS performance: clients run on dual quad core 3GHz Xeon class machines; servers run on: double single-core 2.2 GHz Opteron server.

The service side forks off a set of worker daemons to handle the client's requests, processes them and sends out the results. The amount of worker daemon is tuneable to the available hardware. The performance scales linearly with the hardware's processing power.

6. Conclusions

The OSG and EGEE have joined forces with major middleware providers, such as Globus and Condor, to standardize the authorization call-out protocol from Grid resource gateways to policy decision points. This work produced an authorization interoperability profile, based on the SAML v2 profile of XACML v2, and a layered library implementation for both C and Java. The paper discusses operational practices of the authorization infrastructure as the system is being deployed on OSG and EGEE.

Acknowledgments

Fermilab is operated by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the United States Department of Energy. This work was partially funded by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy, under Contract DE-AC02-06CH11357.

This work is part of the research program of the Dutch Foundation for Fundamental Research on Matter (FOM), which is financially supported by the Netherlands Organisation for Scientific Research (NWO).

References

- [1] Pordes R, Petravick D, Kramer B, Olson D, Livny M, Roy A, Avery P, Blackburn K, Wenaus T, Wurthwein F, Foster I, Gardner R, Wilde M, Blatecky A, McGee, J, and Quick R. 2007. The Open Science Grid *Journal of Physics: Conference Series*, **78** 15
- [2] Laure E, Hemmer F, Aimar A, Barroso M, Buncic P, Di Meglio A, Guy L, Kunszt P, Beco S, Pacini F, Prezl F, Sgaravatto M, Edlund A, Mulmo O, Groep D, Fisher SM, and Livny M. 2004. Middleware for the next generation Grid infrastructure *Proceedings of Computing in High Energy Physics and Nuclear Physics 2004, Interlaken, Switzerland* 826
- [3] Garzoglio G, Alderman I, Altunay M, Ananthakrishnan R, Bester J, Chadwick K, Ciaschini V, Demchenko Y, Ferraro A, Forti A, et al. 2009. Definition and Implementation of a SAML-XACML Profile for Authorization Interoperability across Grid Middleware in OSG and EGEE *Journal of Grid Computing* DOI: 10.1007/s10723-009-9117-4
- [4] Foster I and Kesselman C. 1997. Globus: A Metacomputing Infrastructure Toolkit *International Journal of Supercomputer Applications*, **11**(2) 115-128

- [5] Thain D, Tannenbaum T, and Livny M. 2005. Distributed Computing in Practice: The Condor Experience *Concurrency and Computation: Practice and Experience* **17**(2-4) 323-356
- [6] The Joint Security Policy Group (JSPG): <http://proj-lcg-security.web.cern.ch/proj-lcg-security> Accessed May 2009
- [7] Alfieri R et al. 2004. VOMS, an authorization system for virtual organizations *Proceedings of European across Grids conference No1, Santiago De Compostela, Spain* **2970** 33-40
- [8] Levshina T. 2006. The Virtual Organization Management Registration Service *Proceedings of Computing in High Energy Physics and Nuclear Physics 2006, Mumbai, India*
- [9] Ceccanti A, Ciaschini V, Dimou M, Garzoglio G, Levshina T, Traylen S, Venturi V. 2009. VOMS/VOMRS Utilization patterns and convergence plan *Proceedings of Computing in High Energy Physics and Nuclear Physics 2009, Prague, Czech Republic*
- [10] Sfiligoi I. 2008. glideinWMS-A generic pilot-based Workload Management System *Journal of Physics: Conference Series* **119** 062044
- [11] T Maeno T. 2008. PanDA: distributed production and distributed analysis system for ATLAS *Journal of Physics: Conference Series* **119** 062036
- [12] Cantor S, Kemp J, Philpott R, Maler R. 2005. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2. 0 *OASIS SSTC*
- [13] Moses T et al. 2005. Extensible access control markup language (xacml) version 2.0 *Oasis Standard*
- [14] Garzoglio G et al. 2008. An XACML Attribute and Obligation Profile for Authorization Interoperability in Grids *Fremilab White Paper CD-doc-2952-v2*
- [15] The OGF OGSA-Authorization Working Group: <http://forge.gridforum.org/projects/ogsa-authz>, Accessed May 2009
- [16] Lorch M, Kafura D, Fisk I, Keahey K, Carcassi G, Freeman T, Peremutov T, Rana AS. 2005. Authorization and account management in the Open Science Grid *The 6th IEEE/ACM International Workshop on Grid Computing, 2005*
- [17] Chadwick K, Sharma N, Timm SC, Yocum DR. 2009. FermiGrid – Site AuthoriZation (SAZ) Service *Proceedings of Computing in High Energy Physics and Nuclear Physics 2009, Prague, Czech Republic*
- [18] Groep D. 2008. gLExec, SCAS and the way forward *EGEE08 Conference - the Middleware Security Group, Istanbul, Turkey*
- [19] Chadwick K, Berman E, Canal P, Hesselroth T, Garzoglio G, Levshina L, Sergeev V, Sfiligoi I, Sharma N, Timm S, Yocum DR. 2008. FermiGrid – experience and future plans *Journal of Physics: Conference Series* **119** 052010
- [20] Garzoglio G, Wang N, Ananthan B. 2009. SVOPME: A Scalable Virtual Organization Privileges Management Environment *Proceedings of Computing in High Energy Physics and Nuclear Physics 2009, Prague, Czech Republic*
- [21] Röblitz T, Schintke F, Reinefeld A, Barring O, Lopez M B, Cancio G, Chapeland S, Chouikh K, Cons L, Poznanski P, et al. 2004. Autonomic Management of Large Clusters and Their Integration into the Grid *Journal of Grid Computing* **2**(3): 247-260
- [22] YAIM <https://twiki.cern.ch/twiki/bin/view/EGEE/YAIM> Accessed May 6, 2009
- [23] Aidel O, Cavalli A, Cordier H, L'Orphelin C, Mathieu G, Pagano A, Reynaud S. 2007. CIC portal: a collaborative and scalable integration platform for high availability grid operations *Proc. 8th IEEE/ACM International Conference on Grid Computing 2007* DOI 10.1109/GRID.2007.4354124
- [24] YAIM Tool <https://cic.gridops.org/yaimtool> Accessed May 6, 2009
- [25] Childs S, Poleggi ME, Loomis C, Muñoz Mejias LF, Jouvin M, Starink R, De Weirdt S, Meliá GC. 2008. Devolved management of distributed infrastructures with Quattor *Proc. of the 22nd conference on Large installation system administration (LISA08)* pp. 175-189

The submitted manuscript has been created in part by UChicago Argonne, LLC, Operator of Argonne National Laboratory ("Argonne"). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.