

# SVOPME: A Scalable Virtual Organization Privileges Management Environment

Gabriele Garzoglio<sup>1</sup>\*, Nanbor Wang<sup>2</sup>, Igor Sfiligoi<sup>1</sup>, Tanya Levshina<sup>1</sup>,  
Balamurali Ananthan<sup>2</sup>

<sup>1</sup>Fermi National Accelerator Laboratory, P.O. Box 500, Batavia, IL, 60510, USA

<sup>2</sup>Tech-X Corporation, 5621 Arapahoe Ave, Suite A, Boulder, CO 80303

E-mail: \* garzoglio@fnal.gov

**Abstract.** Grids enable uniform access to resources by implementing standard interfaces to resource gateways. In the Open Science Grid (OSG), privileges are granted on the basis of the user's membership to a Virtual Organization (VO). However, Grid sites are solely responsible to determine and control access privileges to resources using users' identity and personal attributes, which are available through Grid credentials. While this guarantees full control on access rights to the sites, it makes VO privileges heterogeneous throughout the Grid and hardly fits with the Grid paradigm of uniform access to resources. To address these challenges, we are developing the Scalable Virtual Organization Privileges Management Environment (SVOPME), which provides tools for VOs to define and publish desired privileges and assists sites to provide the appropriate access policies. Moreover, SVOPME provides tools for Grid sites to analyze site access policies for various resources, verify compliance with preferred VO policies, and generate directives for site administrators on how the local access policies can be amended to achieve such compliance without taking control of local configurations away from site administrators. This paper discusses what access policies are of interest to the OSG community and how SVOPME implements privilege management for OSG.

## 1. Introduction

The Grid computing environment has emerged as the leading technology for coordinated resource sharing among participating institutions and individuals. It enables execution of large-scale computation jobs by providing uniform access to distributed resources such as computational cycles and data storage, shared among participating institutes. The Virtual Organization (VO) is a key concept in grid computing. A VO manages members from different home institutes with common interests. Participating institutes of a VO typically contribute resources at one or multiple sites to be shared over a Grid by all members of the VO working toward the common interests. Multiple VOs can coexist within a Grid. Likewise, an institute can commit the same resources to multiple VOs in which it participates. Furthermore, individual members or institutes can join and leave VO's based on their interests and needs.

---

\* To whom any correspondence should be addressed.

Grid middleware aims to provide uniform access to all the resources made available at various distributed sites for members of a VO. The Grid Security Infrastructure (GSI) [5] provides the core security using X.509 certificates to support mutual authentication among users, resources, and secure communication. A VO is responsible for managing user membership according to its organizational structure of groups and group roles. Ideally, in modern Grids, members of these groups and roles should be granted specific privileges when accessing resources. At a Grid site, an individual's distinguished name (DN) and the attributes describing his VO membership are extracted from its X.509 certificate by the resource gateway. The DN and VO attributes are then mapped to a local user identity with specific user and group ID (UID/GID). Individual sites can then enforce the resource usage privileges through underlying OS' access control mechanisms.

### *1.1. Challenges in Reconciling VO and Site Policies*

Within a Grid body such as the Open Science Grid (OSG) [1] or European Grid for E-sciencE (EGEE) [2], a VO establishes resource-usage agreements with Grid resource providers to grant access of their resources to individual users of the VO. Recent advances in overall authentication and authorization infrastructures, such as those in OSG VO Services project, provide both the *mechanisms and tools* that enable the fine-grained, role-based access control over the Grid discussed above. However, these mechanisms and tools are not yet fully integrated together to provide a streamlined distributed user privilege management environment. In particular, there's a disconnect in propagating and reconciling VO privilege policy changes from VO membership registration systems, such as VOMRS/VOMS [7], to local Grid sites as local configurations, such as identity servers mappings [11], local account setups, and batch system configurations.

Currently, this lack of automatic policy instantiation/reconciliation mechanism is handled manually via verbal discussion between VO administrators and site administrators. Manually synchronizing the preferred VO privilege policies with the supported site-local privilege policies is a brittle and time-consuming process. This is especially true when privilege policies change dynamically, which is not uncommon for large VO's. For example, new privilege policies, groups, and/or roles can be added to a VO. It often takes a long time for sites to be configured properly in order to support the new policies/groups/roles. Furthermore, a Grid site can participate in collaboration with many VO's, which may make the local policies, expressed by system configurations and identity mappings, hard to manage and track.

All these changes can make reconciling VO privilege policies a non-trivial task for both VO administrators and Grid site administrators. Conversely, without supporting the VO privilege policies, jobs submitted to a Grid site may produce unexpected errors or results, since individual users may not have the VO-specific privileges in accessing resources. Such Grid site may therefore be deemed "unsupportive" of the VO and thus render its resources unusable for the VO's members. This lack of VO privilege support can result in lower resource utilization of resources which are otherwise perfectly usable. With the Grid looking to attract more VOs and institutions to provide shared resources, the disconnection between preferred and enforced policies is becoming a significant issue.

### *1.2. The Need for Managing VO User Privileges*

In order to maintain and continue the growth of Grid deployment and to realize the vision of Grid Computing of providing uniform access to distributed resources, there is an urgent need to bridge the

gap between VO privilege policy management and local Grid site configurations. VO user roles and privilege policies must be able to propagate to Grid site automatically, yet allowing site administrators to retain full control over site policies. Furthermore, robust tools need to be made available for VO and Grid site administrators, to define and edit VO privilege policies, distribute these policies to Grid sites, and convert them into local privilege policies and local configurations, which include local identities maintenance and mappings. Finally, with the constant changing of number of VOs, organizations, and privilege policies, there need to be tools to help VO and site administrators alike to verify all policies are consistent with each other.

## **2. Related Work**

SVOPME project is synergistic to many projects on authorization management. For example, the GPBox [13] project is a policy management framework for the Grid environment to globally modify the execution priorities at sites of jobs submitted from VOs. Compared to GPBox, SVOPME project does not attempt to configure site policies directly. Instead, SVOPME produces reports about the compliance of local configurations and hints on how the configurations could be modified for the site to provide better support. We believe making sure the local site administrators retain the full control of site configuration will give them peace of mind and reduce the suspicion regarding the eventual adoption process of SVOPME project.

Another synergistic effort to the SVOPME project is the EGEE Authorization Service [12]. Similar to SVOPME, the EGEE Authorization Service aims to provide consistent authorization decisions for distributed services over the Grid. It provides software components for defining privilege policies at services. These policies are then used to answer queries about whether a particular action is permissible by certain users. Although the EGEE Authorization Service also aims at providing a set of consistent authorization policies over the Grid, unlike SVOPME, the new Authorization Service does not focus on the VO policies. The two projects will be able to leverage the work done by each other.

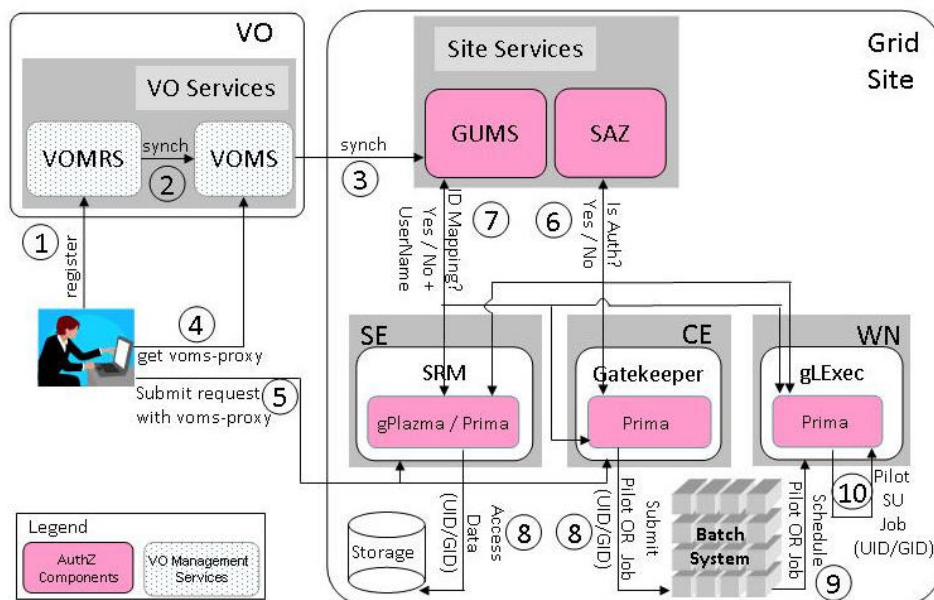
Another effort closely related to SVOPME is the Authorization Interoperability project [3], which defines an attribute and obligation profile for authorization interoperability across Grids as described in Section 3.

## **3. VO Policies and Grid Sites**

Figure 1 illustrates the security model in OSG. Other modern Grid software stacks, such as EGEE, also adopt similar security models [4]. The figure depicts the procedures for performing authentication on the VO side and authorization on Grid sites. The Authorization Interoperability project has standardized the terms and formats used in authorization process between security components implementing Policy-Enforcement-Point (PEP), such as gLexec [14], and Policy-Decision-Point (PDP), such as GUMS [10]. This profile is based on the eXtensible Access Control Markup Language (XACML) [8] and the Security Assertion Markup Language (SAML) [9].

As we mentioned earlier, a VO may want to control user privileges on resources made available on Grid sites. Similarly, a VO may define privilege policies to better meet users with specific missions. However, as highlighted also by Figure 1, the existing OSG security model does not provide support for policy-administration-point (PAP), i.e., how a VO can define its privilege policies. In order to facilitate the definition, propagation, and verification of VO privilege policies, we need to codify VO

privilege policy definitions. This project has adopted XACML as its VO privilege policy definition language. Using XACML allows us to leverage the existing effort in the Authorization Interoperability project and provides a common platform for propagating, comparing, and reconciling privilege information between VO's and sites.



**Figure 1.** The OSG Security Model

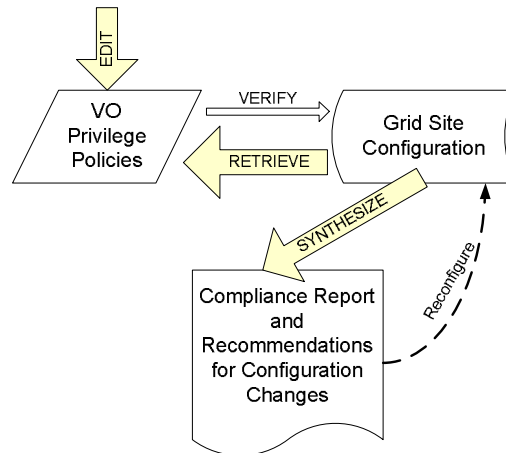
Specifically, the following is a list of the common VO privilege policies supported by SVOPME.

- Account mapping policies allow a VO to define how users of a particular group or role should be mapped to local accounts. One policy is for all users of a VO group or role to be mapped to a single shared local “group account”. Alternatively, a pool account policy maps users of a specific VO group or role into one of a pool of local accounts, granting members of the same group the inherent OS-level protections.
- Relative priority policies allow a VO to specify that jobs submitted by a particular VO group or role should be executed with higher priority than those submitted by another group or role. For example, a VO may want to grant the highest execution priority to jobs from members of the production team to ensure highest throughput in producing science.
- Pre-emption policies define if jobs submitted from a group or role should be allowed to run for consecutive hours without pre-emption to ensure quick turn-around time of these tasks.
- Permission policies define if users from a specific group or role are allowed to access certain storage areas. For example, a VO may want to grant only users playing the “software administrator” role the permission to install software into the \$OSG\_APP area for the VO.
- UNIX group sharing policies allow a VO to define finer grained permission management. For example, a VO may want to specify that local accounts of two groups share the same group ID on a site to ensure that they can freely exchange data, if needed.
- Job suspension policies let a VO to specify if jobs submitted from a particular group or role are not to be suspended.
- Disk quota policies allow a VO to specify the maximum disk usage by a group of users.

- File retention policies defines for how long files owned by users of a specific group or role should be kept in the storage.
- Network policies allow a VO to request outbound network connections for jobs submitted by a group of users.
- Policies on job resubmission semantics instruct the underlying batch system to execute jobs from a specific group or role at most once. This is particularly important in some high energy physics data processing, where the workflow must guarantee that output data is not duplicated.
- Data privacy policies allow a VO to specify that files created by a group of users should be private to the group by default.

#### 4. The SVOPME Architecture

We are developing the Scalable Virtual Organization Privilege Management Environment (SVOPME) to address the challenges in administering and maintaining user privileges over multiple VOs and grid sites. Built on top of the existing OSG privilege infrastructure, the SVOPME project is an extension to the OSG VO Services project [3]. The proposed work fills a gap common to other Grid privilege management infrastructures, as we pointed out in Section 1. The remainder of this section presents the technical approach for designing and implementing SVOPME and how it addresses the needs of modern VO Services.



**Figure 2.** SVOPME helps distribute and realize VO privilege policies

**Figure 2** illustrates how SVOPME fills the gap between VO administrators and grid site administrators. The proposed environment will provide a scalable and consistent privilege management framework and greatly reduce the cost and overhead for managing user privileges across a Grid.

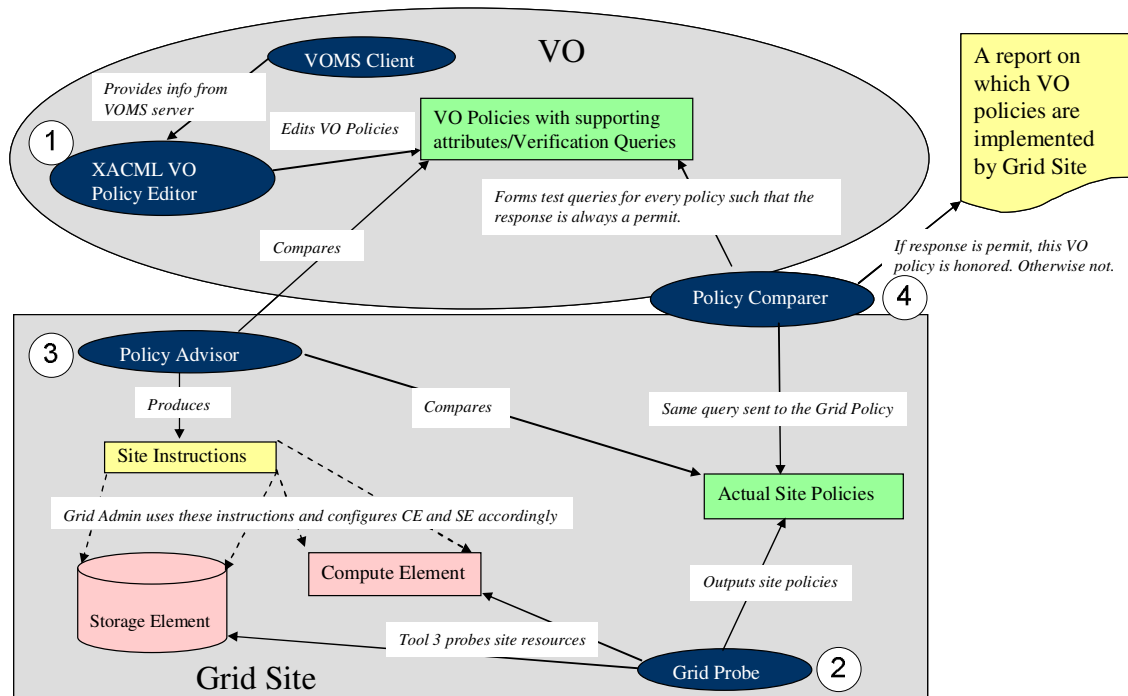
Key data entities maintained by SVOPME contain information necessary for distributing VO privilege policies to Grid sites and transforming them into appropriate site configurations. They include:

- **VO Privilege Policies:** A set of VO's organizational privilege policies for users of different groups and roles. A VO administrator will define this document. SVOPME will codify how to document these policies using XML schema to enable automatic operations.
- **Site Configurations:** A collection of all relevant Grid site system and software configurations that enforce the local privilege policies. These configurations include the local user account

settings, local user group setting, identity mapping (GUMS) configurations, batch system configuration, Storage element configurations, etc. Like the VO Privilege Policies, SVOPME will also need to codify and transform these configurations into a document that can be reasoned and compared against VO policies.

- **Compliance Report and Recommendations for Site Configurations:** This is the document that will be generated by SVOPME on each Grid site to provide detailed evaluation of how many of the VO policies are supported and to generate recommendations for site administrators how to modify the local configurations to fully support VO privilege policies. Note that the site administrators maintain full control over their site configurations.

The SVOPME project will provide the services and tools that automate the creation and transition among this information. **Figure 3** illustrates the architecture of the prototype SVOPME tools that we developed thus far. Four main tools provide the key functionality of the SVOPME prototype implementation. The prototype tools include support for VO administrators to generate VO privilege policies. They can also generate local site privilege policies based on existing site configurations. The prototype tools also allow VO administrators to verify the degree of support a Grid site offers. Furthermore, Grid site administrators can also use the tools to generate directives that not only report how many VO privilege policies a Grid site support, but also suggestions on how to modify the site configurations to add support to the VO policies the site fails to offer. These tools are labelled as 1-4 in **Figure 3**. The remainder of this Section describes these tools in more details.



**Figure 3.** SVOPME Architecture

#### 4.1. VO Components

On the VO side, we implemented the **XACML VO Policy Editor**. As we mentioned earlier, the VO Policy Editor uses XACML as the internal representation of privilege policies. This provides a generic

mechanism for describing, combining, and reasoning with policies. However, XACML is a language too complex and verbose for VO administrators to express their policies. Furthermore, since XACML is a generic language for defining privilege policies, it defines a very limited set of standard attributes about resources, actions, etc. Each community, therefore, needs to define a vocabulary (i.e., XACML profile) to frame the concepts in its domain and VO administrators may not be expert in it.

In order to address these issues, we developed a “domain-specific” GUI-based VO policy editor. The editor enables its users to create individual privilege policies as separate XACML files. We design the editor to be an editor for “domain-specific” policies for Grid use. As opposed to GUI-based editor for generic XACML document, our VO Policy Editor predefined a set of VO policy types that users can generate and edit.

VO administrators can then use the editor to create new VO privilege policies by selecting from a list of pre-defined policy types and filling in the key information for the type of policy being created. For example, when defining an account mapping policy, an administrator only needs to specify the kind of users (their group and role attributes) to which the policy will be applied and the kind of account to which these users should be mapped. A utility tool called “VOMS Client” [6] in **Figure 3** contacts the VO's VOMS server to retrieve the VO structural information and offers the information to the editor. This alleviates users from having to remember and type out all the group and role combinations.

To allow easy extension of the policies supported, we have designed the editor to use policy templates. The policy template design allows us to support new policy types by simply adding new templates without modifying the core editor code. The templates also define the XACML policies the editor generates to conform to the future VO privilege profile easily. Other than the VO privilege policies, the editor also creates corresponding XACML queries that other tools can use to validate the correct enforcement of the corresponding policies.

## *4.2. Site Components*

In this Section, we describe the 3 site-specific components in SVOPME, namely, Grid Probe, Policies Advisor, and Policies Comparer.

### *4.2.1. Grid Probe*

Mechanisms for enforcing Grid site privilege policies currently are scattered at different locations on a Grid site. There is no centralized entity to manage and configure existing Grid middleware infrastructure. For example, the VO group and role to local user ID mapping is managed by Grid site's GUMS database. Setting up of local user ID and group ID, which are the basic subject entities for enforcing local policies, has to be done via OS tools. Batch system and storage element, too, have their own configuration points to control privileges such as priority and permissions.

To try to compare and reason on VO policies directly with all these configuration points will result in *ad hoc* software tools that are very complicated and hard to maintain and expand. To address this issue, we developed the Grid Probe tool that scans and gathers configuration information from various tools and mechanisms. The Grid Probe then analyzes this information and generates the effective local privilege policies in XACML.

#### 4.2.2. Policy Advisor

The Policy Advisor runs on Grid sites. It verifies if the VO privilege policies are supported at the site by comparing the VO policies, retrieved from a VO service, against the local privilege policies, generated by the Grid probe. The Policy Advisor performs this comparison by running the corresponding test queries, retrieved from a VO service, over the local policies and see if the requested action is allowed on the Grid site. If it is, then the grid site does support the corresponding policy correctly. If not, the Policy Advisor will recommend a way to modify the site configuration to correct the problem.

For example, the following output generated by the Policy Advisor shows two policies that the Grid site fails to support and how to correct the problems.

```
[java] VO/Grid Account Mapping Policy Advices
[java] -----
[java] /TECHX/Role=VO-Admin is group mapped on the Grid
site.
Should be pool mapped.
[java] No Account Mapping Policies for /TECHX/VISITORS were
found on the Grid site. /TECHX/VISITORS needs to be pool mapped
in the GUMS config
[java] VO/Grid Grid Accounts Policy Advices
[java] -----
[java] No matching Grid Accounts Policy was found for
/TECHX/VISITORS on the Grid site. Create a mapping in GUMS
config such that /TECHX/VISITORS be mapped to at least 3
account(s)
[java] /TECHX/Role=VO-Admin mapped to 1 account(s)
(techxVOadmin) on the Grid site, is not sufficient enough. Need
to be mapped to at least 3 accounts.
```

**Figure 4.** An example output from the Policy Advisor tool. In this example, the mapping and account policies are checked. For each policy there are two problems in the site configuration, as compared to the desired VO policies.

#### 4.2.3. Policy Comparer

The Policy Comparer behaves similarly to the Policy Advisor described in the previous section. It can be run by either VO or site administrator to verify the degree of support that a site offers to a VO. Test queries are issued against site policies to check if the policies are supported at the site. However, Policy Comparer purposely limits the information produced, in order to protect the privacy of a site in its configuration details.

### 5. Discussion

As we alluded in Section 1.2, SVOPME project aims to address the scalability problems in providing consistent resource usage over the Grid. This translates directly into lower costs in managing both VOs and grid sites. Specifically, SVOPME helps alleviate VO administrators' workload as they no longer need to submit *ad-hoc* jobs to individual sites to figure out which privilege policies are



enforced at the site and which are not. SVOPME also provides a set of commonly used privilege policy templates that VOs can use to put together their own VO-specific policies. Moreover, the VO and site policy services automate the communication between VOs and grid sites. This greatly reduces the efforts needed from VOs and sites alike, to support opportunistic usage of resources.

Similarly, SVOPME allows sites to advertise and prove the degree a VO is supported. For a site to support a new VO and its privilege policies there are now semi-automatic mechanisms to amend the site configurations. Equally important is that Grid sites do not relinquish the privilege enforcement to the VOs. Rather, SVOPME informs the site administrators with a formal VO policy assessment.

## **6. Work Plans**

We have implemented the core set of proof-of-concept tools in SVOPME to demonstrate the effectiveness of the overall project. For the project to be successful and provide a set of tools that are truly applicable to the actual daily operations of the Grid, new tools and services need to be implemented and enhancements are needed for the existing tools. Specifically, we need to provide a more comprehensive set of VO privilege policies that our tools supports. We are making steady progress on adding new privilege policies and enhance site SVOPME components to support these policies.

The current prototype implementation only covers the core enabling features for defining VO policies and verifying site configuration. Communications between VOs and sites still remain to be defined and implemented before the prototype can be turned into a product. Although SVOPME adopts a very modular design and can easily be extended, we need to provide a set of comprehensive documents to describe how users can add support for new VO policies in various SVOPME components. Likewise, because sites may have their own unique configurations, sites may need to come up with customized tools to better synthesize site-equivalent policies and configuration advises specific to the site.

## **7. Conclusions**

To address the scalability issues in ensuring consistent resource access over the grid, we have developed a prototype implementation of the SVOPME project. The proof-of-concept prototype tools demonstrate the feasibility of the overall project to provide an environment for VOs and sites to communicate the VO privilege policy needs and the degree of site support automatically. Fully implemented, the SVOPME project can greatly reduce the costs in running and maintaining VOs and sites alike. We are working to turn the prototype implementation into a product that can be incorporated into standard Grid middleware distributions.

## **8. Acknowledgments**

The SVOPME project is partially funded by DOE OASCR SBIR Grant #DE-FG02-07ER84733, the Fermi National Accelerator Laboratory, and the Tech-X Corporation. 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.

## 9. References

- [1] Pordes R et al. 2007 The Open Science Grid *Journal of Physics: Conference Series* **78** 15
- [2] Laure E et al. 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 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] Garzoglio G et al. 2009 An XACML profile and implementation for Authorization Interoperability between OSG and EGEE *Proceedings of Computing in High Energy Physics and Nuclear Physics 2009, Prague, Czech Republic*
- [5] Foster I and Kasselmann C 1997 Globus: A Metacomputing Infrastructure Toolkit *International Journal of Supercomputer Applications*, **11**(2) 115-128
- [6] 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
- [7] 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*
- [8] Moses T et al. 2005 Extensible access control markup language (xacml) version 2.0 *Oasis Standard*
- [9] Cantor S, Kemp J, Philpott R, Maler R 2005 Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2. 0 *OASIS SSTC*
- [10] Garzoglio G et al. 2008 An XACML Attribute and Obligation Profile for Authorization Interoperability in Grids *Fremilab White Paper* CD-doc-2952-v2
- [11] Lorch M, Kafura D, Fisk I, Keahey K, Carcassi G, Freeman T, Peremutov T, Rana A S 2005 Authorization and account management in the Open Science Grid *The 6th IEEE/ACM International Workshop on Grid Computing, 2005*
- [12] The EGEE Authorization Service:  
<http://twiki.cern.ch/twiki/bin/view/EGEE/AuthorizationFramework>  
Accessed on May 13, 2009
- [13] Cesini D, Ciaschini V, Dongiovanni D, Ferraro A, Forti A, Ghiselli A, Italiano A, Salomoni D 2008 Enabling a priority-based fair share in the EGEE infrastructure *Journal of Physics: Conference Series* **119** 062023 DOI:10.1088/1742-6596/119/6/062023
- [14] Sfiligoi I, Quinn G, Green C, Thain G 2008 Pilot job accounting and auditing in Open Science Grid *Proc. 9<sup>th</sup> IEEE/ACM International Conference on Grid Computing* 112 - 117 DOI 10.1109/GRID.2008.4662789