Authorization Strategies for Virtualized Environments in Grid Computing Systems *

Xinming Ou Anna Squicciarini Sebastien Goasguen Elisa Bertino
Purdue University

Abstract

The development of adequate security solutions, and in particular of authentication and authorization techniques, for grid computing systems is a challenging task. Recent trends of service oriented architectures (SOA), where users access grids through a science gateway — a web service that serves as a portal between users of a virtual organizations (VO) and the various computation resources, further complicate the authorization problem. Currently, the security component developed as part of the Globus Toolkit, the de facto standard for grid infrastructures, is not fully equiped with the capabilities to meet those challenges. The main drawback of the current approach is that it relies on a low level identity-based authorization scheme. A low-level access control policy maps a user's identity (distinguished name) to a local account. This approach does not scale and is hard to manage in a distributed environment. The goal of this paper is to make a first step towards new authorization solutions that better fit novel grid infrastructures characterized by virtual organizations and science gateways. We review and analyze several solutions proposed, in particular GridShib and the VO Privilege Project, as they represent the most innovative techniques currently under development to achieve attribute-based authorization. We then propose several solutions for grid authorization through science gateways, and discuss how those existing projects can be leveraged to implement the solutions we propose.

1 Introduction

Grid computing [15, 16] represents an important infrastructure that makes it possible for multiple institutions to pool their computing resources and to collaborate in order to solve computationally intensive problems. As more organizations and users are becoming interested in using grid computing systems in a variety of application domains, security becomes a key issue. Designing a scalable authorization strategy for such a context is a complex problem. Although authorization in distributed systems has been extensively investigated, not much work has been done to address authorization problems facing real large distributed systems such as grids. The current de facto solution, as represented by the GSI component [7] included in the Globus toolkit [2], adopts a simple low-level approach based on user identities. An access control list (grid-mapfile) that maps a user's global identity (distinguished name, or DN) to a local account has to be set up at *every* grid site. Users whose DN appears on the list is authorized to use machines in the site, with privileges associated with the local account. This simple approach is in essence the same authorization mechanism used for a single machine, for example the Unix "/etc/passwd" file. In a distributed system like a grid, there are thousands of users and it is not realistic to base authorization decisions on individual users' identities. Not only the size of the grid-mapfile will far exceed human managibility, but also

*This work was supported by the National Science Foundation through nanoHUB NMI integration and deployment grant SCI-0438246 and through the TeraGrid Resource Provider grant SCI-0503992.

it unnecessarily replicate many information across a wide domain. In reality, access decisions in large systems are typically based not on user's unique identity, but on his *attributes*, such as being a member of an insitution or involved in a particular computation project. Thus, it has been widely acknowledged in the grid-computing community that attribute-based authorization should be the direction of development for grid security. A number of projects are going on in this direction, such as the VO Privilege Project [10], GridShib [4], and PERMIS [13]. However, there are quite a few decision dimensions when it comes about designing an attribute-based authorization scheme for grid computing. Recent years have seen grid computing moving towards a more "virtualized" environments. The usage of computational resources is less divided by institutional boundaries, and virtual communities are formed that often include members from multiple organizations. It is also becoming more popular to access grid computing resources through a web-service based front-end, sometimes called science gateways. In this paper we would like to explore design options for attribute-based authorization in grid that will better suit the need in such virtualized environments.

The rest of the paper is organized as follows. Section 2 provides an overview of the main requirements grid systems currently need to face due to the emerging trend of virtualized environment. We then focus on the authorization requirement, and present in Section 3 the existing approaches and main challenges. In Section 4 we propose three possible authorization operation modes for accessing grid through science gateways. In Section 5 we elaborate on the case study of the NanoHub science gateway. We conclude the paper in Section 6.

2 Virtual organizations and service oriented architectures

A virtual organization (VO) is a community of individuals that transcends traditional organizational boundaries. Many computational projects conducted in the grid infrastructures span multiple institutions. It is not always feasible to designate any one of the physical institutions as having the administrative rights over the VO; or the members of the VO do not want to bother with the physical organization's administration to help them maintain the VO membership information. The implication of this on grid security is that information needed for authorization decisions may not always be available within the administration of a particular physical organization. For example, in typical settings, to determine whether a user has certain rights to use some resource, the resource provider can query a user database maintained by the user's home institution (e.g. an LDAP server) to retrieve the user's relevant information. However, if VOs are involved, it is not always clear where to retrieve relevant user information for authorization purposes. The authorization policy thus must be flexible enough to specify different trust relationships: for certain kinds of user information the resource provider trusts the user's home institution; but for user's membership regarding a VO, the resource provider may trust a database maintained by the VO's members.

In parallel, recent years have seen trends of grid computing moving towards a *service oriented architecture* (SOA), where a user does not directly interact with grid infrastructures but rather access the grid through a *service provider*. There are several reasons why this architecture has become popular, the most important one being that the service provider can maintain a collection of applications of particular interest to a user community. The users of the service provider can directly use those applications without having to obtain the application code and upload them to the grid sites. We distinguish service providers and resource providers even though it is understood that resource providers who actually operate and maintain the hardware resources necessary to execute services could also be service providers. We intentionally separated the SP and RP to tackle the case when a scientific community may build its own infrastructure and offer tailored services to its user community. These services in turn may use other services offered by the RP such as job execution, file transfer and so on. Indeed, there is a trend to move service offerings one level up in the hands of the actual scientific communities who know best what their needs are and how they want to interact with their services to enable more science. This trend is illustrated by the TeraGrid science gateway program [9] whose aim is to outreach to communities as a whole and build generic access to

TeraGrid resources so that a large number of scientists in the nation may benefit from the TeraGrid resources.

In some sense the challenges caused by the addition of VO and science gateways to grid security are related. Both require the authorization system to handle a large group of users. This requires the back-end grid infrastructures to have the ability to authorize a community of users in an effective and scalable way. However, the current authorization mechanisms in place in grids do not properly address this requirement yet, as we illustrate in the next section.

3 Authorization in grid

Grid is inherently a federated environment, where every local site wants to retain its authority on determining who can use its resource. In Globus toolkit's GSI component, authorization is done by consulting an access control list called "grid-mapfile". The file contains mapping from a DN, a globally unique name assigned to a grid user, to a local account. Only DN's that has an entry on a grid site's grid-mapfile can use the site's computational resource. Once the user is authenticated, it is assigned to a local account according to the grid mapfile and runs its job as the local account. Further access control can be done through that account in the local system.

Grid authorization based on grid-mapfile is not scalable, because it requires the resource provider to maintain authorization state for every potential user. In the emergence of virtual organizations and science gateways, this is hard to do not only because the number of users associated with a VO or service provider may be huge, but also because the membership of users in a VO or service provider may change dynamically and it is not realistic to require the resource provider to keep track of this membership change. Currently, a number of projects are going on to address this problem [4, 10] and they all adopt *attribute-based authorization*.

3.1 Attribute-based authorization

In real life, authorization decisions are typically based on a user's role in an organization, rather than his unique identity. For example, a policy regarding the usage of a computer may contain a statement like "every faculty member of the university can use the machine", instead of listing the names of every faculty member. "Faculty member" is an attribute associated with a user describing the user's role in an organization. Policies written in terms of attributes rather than individual user's identity more accurately capture the high-level access control intention. For example, when John Smith is no longer a faculty member, the policy does not need to be modified to revoke his privileges. Similarly, when a new faculty joins, the policy does not need to change to allow the new faculty to use the resource. Certainly, an authority needs to provide the attributes for every user that may use the resource. Such attributes should be provided by an attribute server trusted by the resource provider (for example, a server maintained by the university's HR department can provide attributes regarding faculty membership). There are a number of dimensions to choose when building an attribute-based authorization system. Different options result in different attribute collection process.

push mode vs. pull mode. Attribute-based authorization can be implemented according to two different strategies: *push* and *pull*. The push strategy requires the user to contact an attribute authority service to obtain attribute certificates and "push" them to the target service when submitting a request. This approach allows the user to select the specific roles he wants to be authorized with. The pull strategy, on the other hand, does not require the users to present any attribute. The attributes are directly retrieved by the resource provider on behalf of the user. The pull strategy makes attribute retrieval transparent to users.

IdP for institution vs. IdP for VO In an attribute-based authorization system, an attribute server is also referred to as an IdP (Identity Provider). Some IdP's are associated with a physical organization such as a university. IdP's may also be associated with a VO. An important question raised in this context is how a user's attributes should be

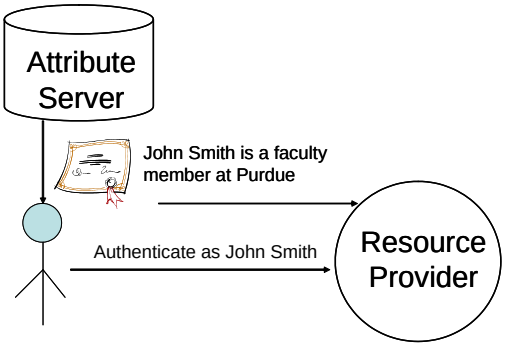

Figure 1. Push mode

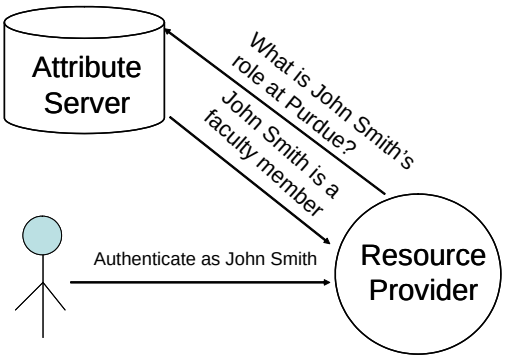

Figure 2. Pull mode

maintained given the diversity of attributes. A user's home institution may seem to be the only entity entitled to provide a user's attributes. However, when VO's are involved, it is often the case that a user's attributes may not be related to his home institution at all. Thus, it is necessary to support flexible attribute maintenance schemes. The resource provider must be able to specify which IdP it trusts for certain kinds of attributes. Requiring the local site to subscribe to a fixed trust relationship in regard to attribute retrieval, (for example by only allowing a user's home institution to provide his attributes) is not likely to be adequate with the emergence of the virtualized environment in grid computing.

3.2 Enforcement of policies

In an identity-based authorization system, an access control policy is typically a list consisting of entries of (subject, object, operation), where "subject" is the user's identity and "object" represents the requested resource. For example, a grid-mapfile is such an access control list, where subject is a user's DN, object is a local account, and the implicit operation is "submit a job on behalf of the local account". In an attribute-based authorization system, user's attributes, instead of an id, are used in the "subject" field. The resource provider's policy must also include statements on who has the authority to assert a user's attributes. This is called the *trust management policy*. This is useful especially in environments in which the same attributes could be provided by different authorities. It is the resource provider's responsibility to decide who to trust for specific attributes. Without a formal trust management policy, such trust will have to be hardcoded in the implementation in an ad hoc way, which may result in security breaches.

Once the policy is defined, a *policy decision procedure* on the resource provider side consumes users' attributes from appropriate authorities and evaluate the policy. Since attribute assertions are from different sources, it is

important that these assertions be transformed into a uniform format that has well defined semantics, for example in the XCAML [17] language. The policy decision point can then evaluate the policy with the retrieved attributes and render a decision. An example of authorization mechanism evaluating such policies is the Shibboleth [8] project, which is explained in the following section.

3.3 Current projects investigating attribute-based grid authorization

A number of projects are on going to implement different attribute-based authorization schemes for grids. We now briefly describe two of them: the VO Privilege Project [10] and GridShib [4].

- *Privilege Project.*

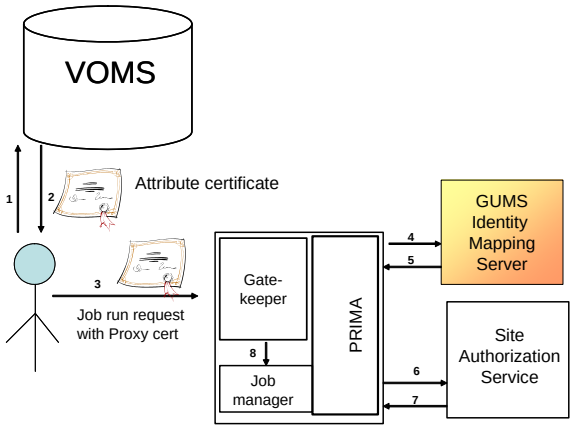

Figure 3. VO Privilege Project

The VO Privilege Project [10] has been developed by US CMS and US ATLAB in order to implement fine grained authorization for access to grid-enabled resources and services. The implemented access control mechanism is illustrated in Figure 3[1]. As shown, the VO Privilege Project implements a push mode mechanism. VOMS is an IdP that is associated with a VO. A user, before accessing the grid service, first obtains an attribute certificate from VOMS and embeds it into his grid proxy certificate[2]. The user then presents the proxy certificate to the site when submitting a job or initiating a file transfer. Operatively, when the proxy is forwarded to a gatekeeper, instead of consulting the grid mapfile, the gatekeeper contacts the site-wide GUMS [5] service to map the request to a local account. The PRIMA module retrieves the attribute information from the user's proxy certificate and communicates it to GUMS, and GUMS uses the attribute information to decide whether the request is authorized and if so which local account to use. As a last step, the gatekeeper contacts the Site Authorization Service to enforce other site-specific access control policies.

- *GridShib.*

GridShib [4, 19, 11] is an example of a pull mode authorization system. The idea in GridShib is to develop an authorization system for grids by integrating the technologies in Shibboleth [8]. Shibboleth is an infrastructure for cross-identity authentication, and it exploits the concept of federated identity information to federated user attributes. In Shibboleth, when a user at one institution tries to use a resource at another, Shibboleth sends attributes about the user to the remote institution, rather than making the user log in to that

[1] This is a simplified version of the diagram at http://computing.fnal.gov/docs/products/voprivilege/

[2] A proxy certificate is used to authenticate a user with a grid site and is valid for a relatively short period of time.

institution. The receiver can check whether the attributes satisfy its access control policy. The IdP in the Shibboleth architecture has all the user attributes and user privacy preferences which are taken into account when this IdP gives information to a service provider.

In its current version, GridShib is implemented as a plug-in for Globus Toolkit 4.0 (GT4). The plug-in implements a policy decision point based on attributes obtained from a Shibboleth attribute authority. In a nutshell, the underlying idea in GridShib is to authenticate grid users using the existing GSI module in Globus, determining the address of the Shibboleth attribute server in the process, and then obtain from the Shibboleth service the selected user attributes that the Grid service is authorized to access. Differently from other approaches, in GridShib the clients of services are not directly affected and do not even need to know Shibboleth is involved in their decision-making. Unlike the full-fledged Shibboleth deployment, which adopts handle-based authentication, GridShib uses X.500 distinguished names to identify principals, thus the current PKI infrastructure in the grids does not need to be touched. The service provider receives a proxy certificate in place of the handle typically issued by a Shibboleth IdP. Like Shibboleth, GridShib's plan is to associate an IdP with a user's physical institution. This scheme may be too restrictive as users will have multiple affiliations and some of those affiliations will be virtual organizations. An approach that supports flexible IdP association is more desirable.

As illustrated, the goals of the two projects are very similar as they both aim to develop a flexible and efficient attribute-based authorization mechanism in grid systems. The approach to the problem is different mainly with respect to the strategies used to collect user attributes and whether the attribute server is associated with a VO or a user's physical institution, two design dimensions mentioned in Section 3.1. In both cases user authentication is based on X.509 [18] certificates. The possibility of a user to push his attributes as in the Privilege Project, let him select the role he wants to use for executing the request, though it requires additional operations at the user's side. The pull mode adopted by GridShib is however easier to deploy, since clients of services are not affected and can submit jobs regardless the underlying authorization system used. Currently it is not possible to determine which of them will better address the access control requirement we have outlined, as both projects are still work in progress. In particular, GridShib is at an infant stage and it still needs to be tested on real case scenarios. GridShib in its final version will include both push and pull mode.

3.4 Attribute-based authorization in a virtualized environment

The attribute-based authorization discussed in this section is better suited to provide security solutions for the virtualized environment described in Section 2. For example, the VO Privilege Project is particular aimed at managing authorization at the VO level. GridShib, by incorporating Shibboleth authorization mechanism which has been widely deployed for web-based access control, is well positioned to meet the security problems brought about by the science gateways. In the next section, we propose several scenarios for enabling attribute-based authorization for accessing grid through science gateways, and discuss how the existing projects may be leveraged to implement them.

4 Proposed Grid authorization solutions for science gateways

Currently, command line access is the most common operation mode for grid applications. A user first authenticates himself to a grid site using his X.509 certificate. The site then makes its access control decisions either based on the grid-mapfile or the user's attributes and its local policy. When accessing grids through a science gateway, a user does not necessarily authenticates himself directly to the grid site. Often times the authentication is between a user and a service provider, which uses one or more existing grid infrastructures as its computational backend. This architecture poses a challenge to authorization: without the user's identity, how can the grid site know who

is using its resources and whether to grant access or not? In this section we propose three possible authorization modes for this service-oriented architecture. The different scenarios are all characterized by three main entities: a user, a service provider (the science gateway), and a resource provider (the grid site). Since the user does not interact directly with the resource provider (RP hereafter), the RP will have to rely on the service provider (SP hereafter) to perform some of the authorization tasks. The difference among the three scenarios largely arise from the different levels of trust between SP and RP.

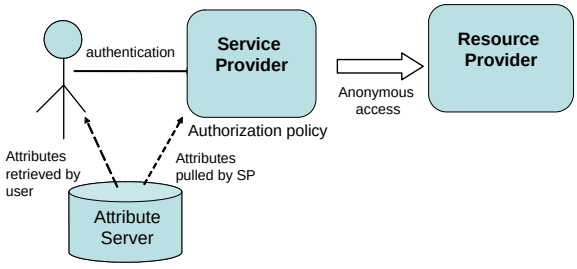

Figure 4. Complete trust

- **Complete Trust.** In this case, the SP is trusted completely by the RP. The analyzed scenario is sketched in Figure 4. The SP enforces both authorization and authentication, while the RP is not involved at all in the process. The user is essentially anonymous at the RP side — the only information the RP gathers from the request is related to the specific job to execute. The SP negotiates resource allocations with the grid site and its users do not need to be aware that they are using the grid as the computational backend. Ideally, authorization at SP should be based on users' attributes. Logically the attribute server and the SP are two different entities, but physically they could either be at the same site or belong to different organizations. Once authentication and authorization is completed, the job request is then forwarded to the RP. The request is typically forwarded as one single grid DN, and from the resource provider's point of view it is not possible to identify who the real user is. Actually many users of the SP may not even have a grid account. They may not even be aware that the SP is using one of the grids to perform computation. The advantage of this approach is that a user does not need to go through the lengthy application process to obtain a grid account allocation while still able to enjoy the computational power provided by the distributed infrastructure. The disadvantage is that the grid site cannot control its own access policy (e.g. differentiated service levels) based on users' credentials. For example, the resource provider cannot provide priority scheduling for members involved in a particular project if the job request is issued through the service provider, since such job requests will be uniformly mingled with all the other requests from the same SP and the RP has no way to know which request is from the particular group of users. In order to provide such priority scheduling, the RP will have to instruct SP to differentiate its service levels for that particular group.

- **Medium trust.** In this case, the SP still performs authentication, but the final authorization decision is made by the RP. A sketch of the process is given in Figure 5. The identity token passed from SP to RP does not necessarily contain the user's unique identity, but rather a user's attributes or a handle that can be used later to retrieve a user's attributes. These attributes should be provided either by the SP or another trusted third party. For example, a SP may provide attributes reflecting a user's paid premium and request the RP to schedule jobs on different priorities based on the premiums. Or the SP can provide attributes reflecting a user's trustworthiness — an anonymous user is less trustworthy than a registered user. The RP can then apply different access control policies based on a user's trustworthiness. The advantage of this approach, compared with the complete-trust model, is that the RP can have its own policy to differentiate its service based on users' attributes and can have better control on how to allocate its computation resource. Note also

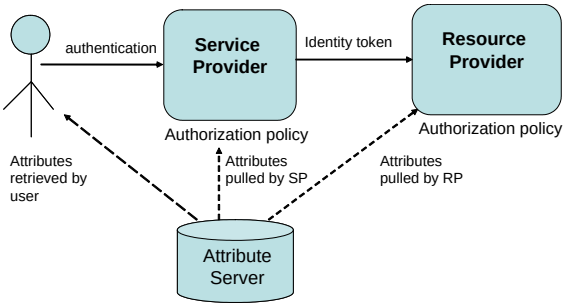

Figure 5. Medium trust

that under this mode, the SP can also have its own authorization policy. A request may already be filtered out by the SP's policy before reaching the RP. The SP can also depend its authorization policy on users' attributes and retrieve attributes from trusted IdP's.

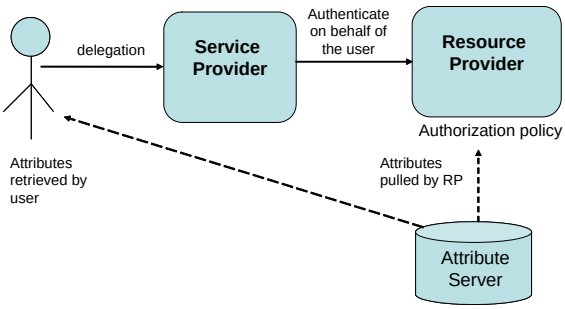

Figure 6. No trust

- **No trust.** In this scenario both authentication and authorization are executed by the RP. A user who already has a resource allocation at RP can access the grid through the SP using his own grid credentials. The advantage of this approach is that the RP can independently track individual users using their resources. In the previous two cases the RP needs the coordination and information from the SP to trace back who is behind a job running on its machines. The disadvantage is that only users with allocation on the RP can use the service provided by the SP, which may be too inflexible in practice. To implement this approach, a mechanism to securely delegate a user's grid authentication credential to the SP is necessary. Users are in this case aware that they are using certain grid resources and the service provider forwards job requests to the proper resource provider.

4.1 Implementation schemes

All the three cases discussed above adopt an attribute-based authorization scheme. Since the VO Privilege Project and GridShib are two projects being actively developed for enabling attribute-based authorization in grid, we would like to explore ways that one can use them for science gateways.

In case 1 the SP replaces the resource provider for the role of performing authentication and authorization. If the SP uses GSI for this purpose, then both the VO Privilege Project and GridShib can be directly used here. However, it may not be the case that the SP adopts GSI as its security mechanism (e.g. the NanoHUB example that will be introduced in Section 5). In this case the SP can still implement its own attribute-based authorization but will not be able to directly leverage the techniques developed by these projects.

Both the push mode and the pull mode could be used in attribute retrieval, and Shibboleth can play a role. For example, the SP may rely on a third-party Shibboleth IdP to provide the identity service and base its authorization decisions on the attributes retrieved from the IdP.

Case 2 has the most flexibility in terms of how to leverage the existing projects for implementation. One natural solution is to use GridShib and here again we have a number of possibilities. One is that the service provider is also an IdP. The SP can serve as a weak CA — a certificate authority entitled to issue short-term user credentials for authentication purposes. The DN's in such certificate could be meaningless, but they allow the resource provider to recognize different users. This kind of short-term certificates are sometimes called "junk certificate". A junk DN certificate can then serve as the identity token in Figure 5. A Shibboleth IdP will be running at SP site and the RP invokes the GridShib plug-in to retrieve attributes by presenting the junk DN to the IdP at SP site. Since the junk DN is issued by the SP's CA, the SP can link it to one of its users and retrieve his attributes. The other possible way of applying GridShib is that the SP relies on another IdP to provide authentication and attribute service. One of the developing projects in the GridShib use cases has this flavor [3]. On the other hand, the VO Privilege Project could also be applied to implement case 2. The service provider can obtain an attribute certificate from VOMS on behalf of the user and forward it to the RP. Currently the VOMS service uses a DN to identify a user, which means the user needs to establish a globally meaningful grid DN. This may be less flexible in terms of usability, but it allows greater flexibility in terms of attribute retrieval. The attribute servers do not need to be tightly associated to the authentication service (for example, the service provider), as long as both recognize the same set of DN's.

For case 3, the only difference from the current standard grid access method is that the service provider acts on behalf of a user. Both GridShib and VO Privilege Project can be readily applied at the RP side. The only additional requirement is to put a credential delegation service (such as MyProxy [12]) on the SP.

4.2 Grid interoperability

As more grid infrastructures emerge, there is an increasing need for different grids to interoperate seamlessly. Currently, user credentials for one grid cannot be directly used to access resources at another. A user will need to maintain multiple credentials if he wants to use multiple grids.

The emerging trends of virtual organizations and science gateways provide an opportunity for more seamless grid interoperability. A VO or science gateway can negotiate resources with different grids and users can be saved the burden of requiring grid credentials by themselves. To achieve this in practice requires the attribute retrieval protocols be standardised among the various grids. For exmaple, the VO Privilege Project uses ASN.1 format to embed an attribute certificate in a user's X.509 certificate, wheareas GridShib uses SAML assertions to retrieve user attributes. This means a grid infrastructure that adopts one of these technologies will not be able to understand attribute assertions from another grid that uses the other technology. A single standard format for expressing attributes and a single standard protocol for retrieving them must be defined to achieve grid interoperability.

5 Case Study: NanoHUB

One example that demonstrates the move towards service oriented architectures in grid computing is the nanoHUB project [6], which is becoming the de facto cyberinfrastructure for the computational nanotechnology community with 1,600 annual users accessing the nanoHUB simulation service and a total of 9,000 users accessing other services such as collaboration tools and educational modules. NanoHUB is a TeraGrid science gateway and is also an Open Science Grid VO.

It is necessary to study the authentication and authorization solutions for the nanoHUB cyberinfrastructure and how it will interoperate with the TeraGrid and OSG resources and policies. The nanoHUB infrastructure is based on the InVIGO middleware [14] and has integrated the use of Condor-C and Condor-G [1] to submit jobs

to grid resources. InVIGO's innovation lies in the use of virtualization techniques. The nanoHUB makes use of virtual machines based on Xen 3.0 and VMware. Virtual machines can provide execution jail but also offer a level of isolation from the physical infrastructure. Virtualization is the solution to build a cyberinfrastructure for a community on top of one organization administrative domain. VMs together with virtual networking and virtual file system are the basis of a virtual environment where users of a community can be isolated. At a first order the nanoHUB is only computational oriented and does not face data transfers, replication and publication issues. The default nanoHUB user is loosely known with only a valid email address to identify them and no X.509 certificate therefore it differs from the model of standard TeraGrid users and OSG VO members. TeraGrid users have received an allocation on the resource providers and can use GSI authentication to access the resources, authorization is then done thorugh grid mapfile and local unix accounts. This approach won't scale for the science gateways as the number of users will be several thousands like in the nanoHUB case. It is unrealistic not only to maintain a grid mapfile with thousands of entries but it is also unrealistic to have thousands of grid accounts created for the nanoHUB: one per user. A more realistic approach is to use a VO account like in the OSG model but that approach breaks because it still requires every user who will access the resource to have its DN mapped to the VO account on the remote resource. Additionally the nanoHUB users are not issued a trusted X.509 certificate nor does the nanoHUB operates its own CA and vouches on the identity of its users. When a user accesses a nanoHUB service he is being authenticated on the nanoHUB portal through standard login/password mechanism, then his roles are looked up in an LDAP database which will give or deny access to simulation tools.

When a job is being run on the local virtual machines, it runs as the user using standard UNIX AA techniques through the PAM module. If a grid job submission is needed, then Condor-C is being used to forward the job to a machine on which users cannot login and where a nanoHUB proxy certificate is created to talk to RPs on TeraGrid or OSG using Condor-G. Condor-C is used to map job requests from nanoHUB users to a trusted certificate on remote grid resources. This nanoHUB certificate has been issued by a TG-trusted CA and is also trusted by OSG. The difference stems from the fact that no individual users DN will be mapped to this service account on the remote sites. The identity of the user submitting the job is only known from the nanoHUB, i.e the SP. Additionally the user does not know where the job is running either. This is essentially the complete-trust scenario described in section 4. However some nanoHUB users may already have valid certificates that they may want to use to run nanoHUB services on TG or OSG. These users may actually have their own allocation on TeraGrid and may decide to use nanoHUB services on their allocation. This is the situation discussed in the third scenario of Section 4. Managing all these different use cases and authorization criteria could become quite complex. Therefore we see that the nanoHUB and most probably more Virtual community cyberinfrastructures are in the need for an authentication and authorization model that offers hybrid models and delegated trust models.

Future efforts in the nanoHUB will focus on deployment of an authentication/authorization model. The model will be a hybrid of the three use cases described in section 4. Default users will be given limited privileges, i.e limited access to services. In the short term nanoHUB may maintain its own attribute server while in the long term it may also leverage other Shibboleth deployment at U.S. institutions relying on a distributed authentication system to grant access to nanoHUB services. Currently we are planning to base our implementation on the GridShib technology. But other approaches such as VO Privilege Project are also alternatives.

6 Conclusion

In this paper we discussed the need for attribute-based authorization technology in grid computing to accomodate the emerging trend of virtualized environment, where users access grid resources as a virtual community through a service provider. We presented several design dimensions in building an attribute-based authorization system for grids and discussed their perspective advantages and disadvantages with regard to this trend. We proposed several solutions for attribute-based authorization for accessing grids through science gateways, and discussed how existing/onging grid authorization projects in this area could be leveraged to build such systems. As

a case study, we described nanoHUB, a science gateway for the computational nanotechnology community, the authorization challenges faced by it, and how the framework outlined in this paper can solve the problems.

References

[1] Condor, High Throughpout Computing. http://www.cs.wisc.edu/condor.

[2] Globus Toolkit. http://www.Globus.org.

[3] Grid portal use case for gridshib. https://authdev.it.ohio-state.edu/twiki/bin/view/GridShib/GridPortalUser.

[4] The Gridshib project. http://gridshib.globus.org/.

[5] The Gridshib project. http://grid.racf.bnl.gov/GUMS/.

[6] NanoHUB. http://www.nanohub.org.

[7] Overview of the grid security infrastructure. http://www.globus.org/security/overview.html.

[8] The Shibboleth. http://shibboleth.internet2.edu/.

[9] The Teragrid project. http://www.teragrid.org.

[10] The VO Privilege Project. http://computing.fnal.gov/docs/products/voprivilege/.

[11] T. Barton, J. Basney, T. Freeman, T. Scavo, F. Siebenlist, V. Welch, R. Ananthakrishnan, B. Baker, and K. Keahey. Identity federation and attribute-based authorization through the globus toolkit, shibboleth, gridshib, and myproxy. In *5th Annual PKI R&D Workshop*, October 2005.

[12] J. Basney, M. Humphrey, and V. Welch. The MyProxy Online Credential Repository. In *Software: Practice and Experience*, volume 35(8), 2005.

[13] D. Chadwick. Authorisation in Grid Computing. *Information Security Technical Report*, 10(1):33–40, unknown 2005.

[14] J. Fortes, R. Figueiredo, and M. Lundstrom. Virtual computing infrastructures for nanoelectronics simulation. *Proceedings of the IEEE*, 93(10), August 2005.

[15] I. Foster and C. Kesselman, editors. *The grid: blueprint for a new computing infrastructure*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[16] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the Grid: Enabling scalable virtual organizations. *Lecture Notes in Computer Science*, 2150:1–??, 2001.

[17] OASIS. Xacml 2.0 approved as oasis standard.

[18] W. F. R. Housley, W. Polk and D. Solo. Internet X.509 Public Key Certificate and certificate revocation list (CRL). RFC, 3280, network working group, April 2002.

[19] V. Welch, T. Barton, K. Keahey, and F. Siebenlist. Attributes, anonymity, and access: Shibboleth and globus integration to facilitate grid collaboration. In *4th Annual PKI R&D Workshop*, April 2005.