Grid Authorization Based on Existing AAA Architectures

Manuel Sánchez[1], Gabriel López[1], Óscar Cánovas[2] and Antonio F. Gómez-Skarmeta[1]

[1] Department of Information Engineering and Communications
[2] Department of Computer Engineering
University of Murcia, Spain

Abstract. Grid computing has appeared as a new paradigm to cover the needs of modern scientific applications. A lot of research has been done in this field, but several issues are still open. One of them, the Grid authorization, is probably one of the most important topics regarding to resource providers, because they need to control the users accessing their resources. Several authorization architectures have been proposed, including in some cases new elements which introduce redundant components to the system. In this paper, we propose a new scheme which takes advantage of a previously existing underlying authorization infrastructure among the involved organizations, the NAS-SAML system, to build a Grid environment with an advanced and extensible authorization mechanism.

1 Introduction

In the last years, the computing and storage capacity required in scientific environments has exceeded the capacity offered by traditional computers. This problem has motivated the developmentof a new computer paradigm called Grid Computing [9], which defines the resource sharing among different organizations in a flexible, secure and coordinated way, conforming the so called Virtual Organizations (VO).

Nowadays, some aspects of the Grid computing such as resource sharing or discovery have been solved by projects as the Globus Toolkit [1]. However, other Grid aspects generally related to the security of the VO are still open, and one of the most important open issues in the Grid research field is user authorization. Indeed, authorization is a critical feature in Grid computing because when an organization offers its resources to users belonging to other domains, it wants to be sure that only authorized users are able to perform the set of allowed actions over each protected resource.

Authorization mechanisms in the Grid have evolved from a simple authorization file, listing the users who can access to each resource, to more complex schemes based on the use of authorization servers, access control policies or identity certificates. Several solutions, such as CAS [14] or VOMS [7], have been proposed, and some existing authorization mechanisms, as for example PERMIS [2], Akenti [16] or Shibboleth (GridShib) [18] have been adapted to provide authorization decisions to the Grid. However, these authorization systems introduce new elements in spite of the authorization process in the Grid environment could take advantage of existing ones.

Sánchez M., López G., Cánovas Ó. and F. Gómez-Skarmeta A. (2006).
Grid Authorization Based on Existing AAA Architectures.
In *Proceedings of the 4th International Workshop on Security in Information Systems*, pages 3-12

Although authorization is a key feature in Grid environments, this is not an exclusive topic of this field. Traditionally, organizations have protected critical resources, for example the communications network. In fact, the AAA architecture [5] was designed to solve this problem using different mechanisms to identify end users, such as login/password or identity certificates. Therefore, one of the most common network access control mechanism used by network providers is the one based on the AAA architecture. An example of them is the architecture Network Access Service based on the AAA architecture and SAML authorization attributes, NAS-SAML, described in [11].

Due to the fact that there are organizations using these kinds of architectures to control the network access, it would be desirable that this authorization information could be reused by other applications which also need to perform access control, for example the Grid. This paper proposes a new authorization mechanism for those Grid systems which takes advantage of an existing AAA infrastructure among two or more organizations to provide authorization decisions, and which makes use of XML-based standards such as SAML [13] and XACML [6] to manage the authorization data and access control policies in an extensible and distributed way.

The rest of this paper is structured as follows. In Section 2, an overview about Grid authorization is given. Next, Section 3 presents NAS-SAML. Section 4 describes the proposed architecture to provide authorization in a Grid environment, and the different design alternatives are shown in Section 5. In Section 6 other authorization proposals for Grid computing are described, specifying the main differences with the approach presented in this paper. Finally, conclusions and future work are presented in Section 7.

2 Authorization in the Grid

Grid computing is the response to a higher computer power and storage capacity demand. This technology tries to make use of the resources provided by several organizations to offer the sum of all of them to the users belonging to these organizations. This resource sharing implies several authorization issues, since an organization offering its resources wants to be sure that only allowed users perform the allowed operations over the protected resources.

Nowadays, there are several Grid implementations available, such as UNICORE [15] or Globus Toolkit (GT), being GT the most common one. In 2002, a Grid specification called OGSA [8] appeared in order to define a standard way to create Grid implementations. In OGSA, resources are offered by means of Grid services, which are web services with specific interfaces to address service discovery, dynamic creation, lifetime management and other features. This specification was redesigned in 2004 as WSRF [4], but the main idea remains unaltered.

In the early Grid implementations, authorization was performed by means of a file, called *grid-mapfile*, with a mapping between user's subject names and local account names. Only if the subject name appeared in this file, the user was allowed to access to the resource. This solution was not very scalable and several alternatives, such as CAS or VOMS appeared to address this problem. The requirements that a Grid security model must address are described in [20]. This document shows that an authorization service have to evaluate policy rules to take authorization decisions based on informa-

tion about the requestor and the target service, and must be transparent to the user and to the target service. Using this approach, once the user is authenticated, the hosting environment has to contact with this authorization service in order to obtain a decision about the user request. The Global Grid Forum (GGF), through one of its working groups, OGSA-Authz WG, defined an OGSA authorization service based on SAML for requesting and expressing authorization assertions, OGSA-SAML [19].

As Figure 1 shows, OGSA-SAML defines new SAML statements to carry the needed information between a Policy Enforcement Point (PEP) and a Policy Decision Point (PDP), those new introduced statements are the *ExtendedAuthorizationDecisionQuery* and *SimpleAuthorizationDecisionStatement* sentences. The first one includes a parameter to notify the PDP whether the PEP only needs a simple boolean authorization decision instead of a list of allowed rights. It also adds a mechanism to pass information about the requestor. The second statement contains a decision response to the first one as a whole, without enumeration of rights.

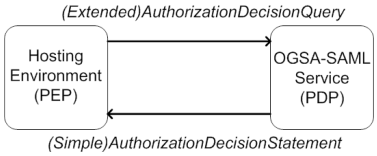

Fig. 1. OGSA-SAML Authorization Service.

This specification has been included in GT, providing a Grid service interface called *SAMLRequest port type*. In this way, the Grid service container, acting as PEP, can be configured to check the user's permissions from the PDP. Consequently, the PDP has to be a grid service which implements the interface *SAMLRequest*. In this proposal, as explained below, the authorization service used by the Grid system takes advantage of an underlying NAS-SAML infrastructure in order to manage the authorization process in a multi-domain scenario, which includes the retrieval of user attributes from his home domain and the authorization decision processes.

3 SAML-Based Network Access Control Architecture

During the last years, how to control the users that are making use of computer networks has become an increasing concern for network administrators. As a direct consequence, several security technologies have appeared in order to provide access control mechanisms based on the authentication of users. Traditionally, network access systems have been based on login/password mechanism. Other systems following a more advanced approach for mutual authentication are based on X.509 identity certificates. These systems are especially useful for organizations concerned about the real identity of the requestor. There are other organizations where the different users are classified according to their administrative tasks, the type of service obtained, or some others internal requirements. In those scenarios, the identity could not be enough to grant the access to

the resource being controlled, since we should know the role being played by the user in order to offer the right service. Therefore, a system able to assign to the different users the set of attributes specifying those privileges or roles is needed. This kind of systems is usually designed following the Role Based Access Control (RBAC) model.

In [11], a network access control approach based on X.509 identity certificates and authorization attributes is presented. This proposal is based on the SAML and the XACML standards, which will be used for expressing access control policies based on attributes, authorization statements, and authorization protocols. Authorization is mainly based on the definition of access control policies [10] including the sets of users pertaining to different subject domains which will be able to be assigned to different roles in order to gain access to the network of a service provider, under specific circumstances. The starting point is a network scenario based on the 802.1X standard and the AAA (Authentication, Authorization and Accounting) architecture, where processes related to authentication, authorization, and accounting are centralized.

The system operates as follows. Every end user belongs to a home domain, where he was given a set of attributes stating the roles he plays. When the user requests a network connection in a particular domain by means of an 802.1X connection, the request is captured by the AAA server located in the target domain, and it makes a query to obtain the attributes linked to the user from an authority responsible for managing them, located in the user's home domain. Alternatively, following a push approach, the user itself can present its attributes instead of letting the AAA server to recover them. Finally, the AAA server sends an authorization decision query to a local PDP entity, and that element provides an answer indicating whether the attributes satisfy the resource access control policy. Furthermore, that policy can also establish the set of obligations derived from that decision, for example some QoS parameters, security options, etc. This general scheme works both in single and inter-domain scenarios.

This scenario, although is focused on network access control, can be used as a basis to provide authorization services to higher level applications, such as the Grid. Moreover, due to NAS-SAML has been integrated [12] with other authorization systems, such as PERMIS [3], the Grid environment could be extended to those domains easily.

4 Architecture

This section describes the elements needed in the proposed solution to take advantage of the NAS-SAML infrastructure, an already implemented and tested system that can provide an authorization mechanism to organizations willing to collaborate by means of a Grid, specifically using the Globus Toolkit middleware.

As Figure 2 shows, this architecture might be used when two or more organizations share an AAA infrastructure with NAS-SAML support. One one hand, each organization has its own AAA server, the key element in this scenario since it is responsible for performing the authorization process in every domain. Two modules help the AAA server, the Source Authority (SA), which produces the authorization attributes, and the Policy Decision Point (PDP) entitled to obtain authorization decisions. When the AAA server needs some information about a foreign user, it will ask the home AAA server, located in the user's home domain, in order to obtain this information. The communica-

tion between AAA servers is made through the DIAMETER protocol, specifically the DIAMETER-SAML extension [11]. The authorization process is guided by access control policies, represented in XACML, provided to the PDP. In this way, by adding the suitable service policies, the infrastructure can be extended to support other application-level services willing to obtain authorization decisions about users.

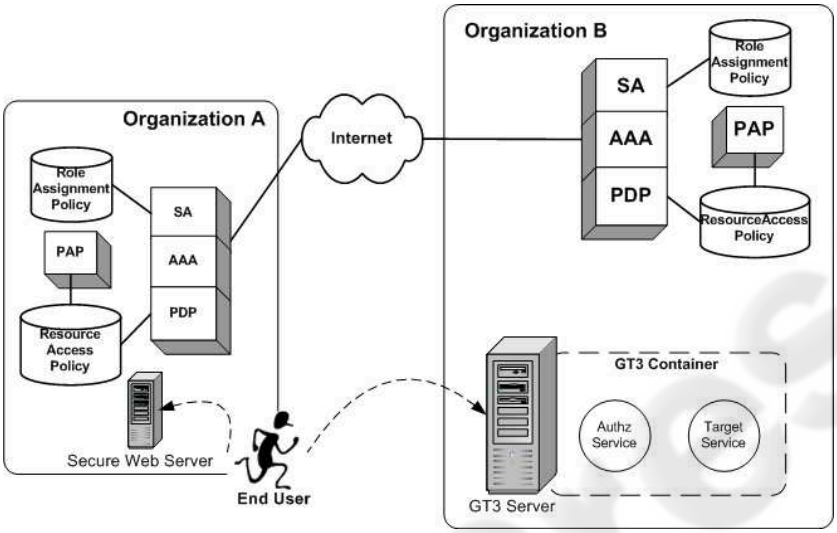

Fig. 2. Architecture elements.

On the other hand, an authorization system in a Grid environment needs the set of elements shown in Figure 2. In this scenario, when an end user wants to access to a Target Service, the GT3 server may check the user's rights to perform the requested action asking an Authorization Service. To take the right decision, this service may use the information about the user contained in the authorization request or, if necessary, it can contact with external entities.

Trying to integrate those two scenarios, the main aim of this work is to provide to those organizations a way to cooperate using a Grid infrastructure by means of an advanced and extensible authorization model, trying to reuse already existing authorization data and elements whenever is possible. This goal can be achieved taking advantage of the extensibility of the OGSA-Authz authorization interface and the flexibility of the NAS-SAML infrastructure to process Grid authorization requests.

It is necessary to define how those two architectures can be integrated, that is, to define the communication interfaces between the Grid and NAS-SAML entities, and the different design alternatives depending on the user requirements and the application scenarios. Therefore, we need to define how the Authorization Service will interact with the local AAA server, in order to follow the authorization process as explained in Section 3.

5 Design Alternatives

Four different scenarios are possible in NAS-SAML, depending on the use of the pull or push approach to access to the network, and whether the user is accessing from his home domain or from a foreign one. In this section we are going to focus on inter-domain scenarios since a Virtual Organization only makes sense among several institutions. Therefore, this section describes two scenarios based on the pull and push models involving, at least, two administrative domains. In the first one, the user gains access to the Grid resources in the traditional way since every authorization task is performed by the authorization service. In the second one, the user preselects the set of attributes he wants to use, and then he presents them to the authorization service.

5.1 Pull Model

In this model the authorization process is transparent to the user, so the Globus client software needs no modification. As Figure 3 shows, when the user wants to access to the Target Service, the GT3 server sends an *ExtendedAuthorizationDecisionQuery* to the Authorization Service, following the OGSA proposal. This message contains the user's identity, the target resource and the action to be performed on that resource. In this scenario, the resource is the Grid Service invoked by the user, and the action is the service method to be executed. Once the Authorization Service gets this information, builds a standard SAML *AuthorizationDecisionQuery* and sends it to the local AAA server, using the DIAMETER-SAML protocol, as described in NAS-SAML [11].

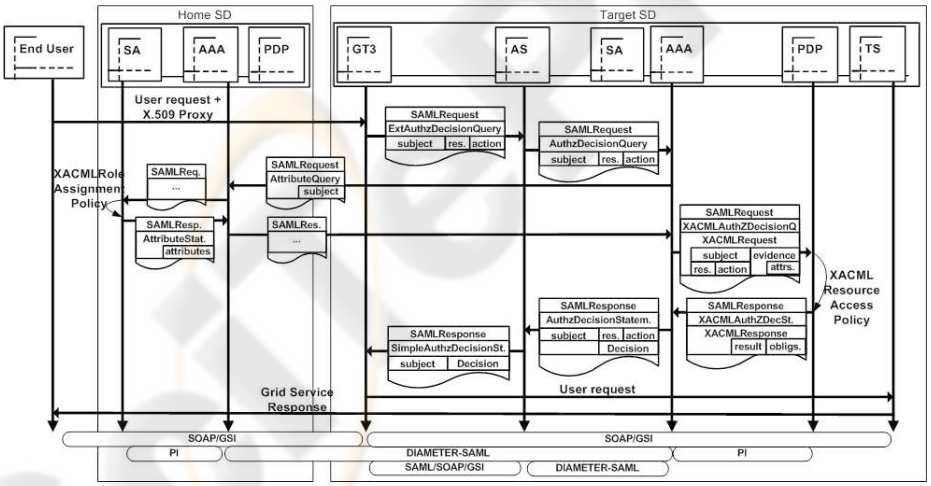

Fig. 3. Pull Model.

Once the local AAA server receives the *AuthorizationDecisionQuery* message, it sends an *AttributeQuery* request to the user's home AAA server, asking for the user's

attributes. The local AAA server is able to discover the home AAA server location from the user's subject, as described in [11]. The home AAA server gets those attributes from its local SA and responds with an *AttributeStatement* sentence, containing those attributes. When the local AAA server receives this message, it uses the attributes and the information received from the Authorization Service to obtain an authorization decision consulting the local PDP. The AAA server sends an *AuthorizationDecisionQuery* message to the PDP, and the *AuthorizationDecisionStatement* sentence received is sent to the Authorization Service, which forwards the decision to the GT3 server as a *SimpleAuthorizationDecisionStatement* sentence.

The advantage of this alternative is that a Virtual Organization making use of the NAS-SAML scenario can make use of the authorization mechanism without user knowledge. In this way, users continue accessing to GT3 in the traditional way, and organizations can manage attributes and its mapping to permissions in a transparent way. The drawback is that the user cannot control the parameters related to the authorization process, such as the set of attributes used to obtain the decision.

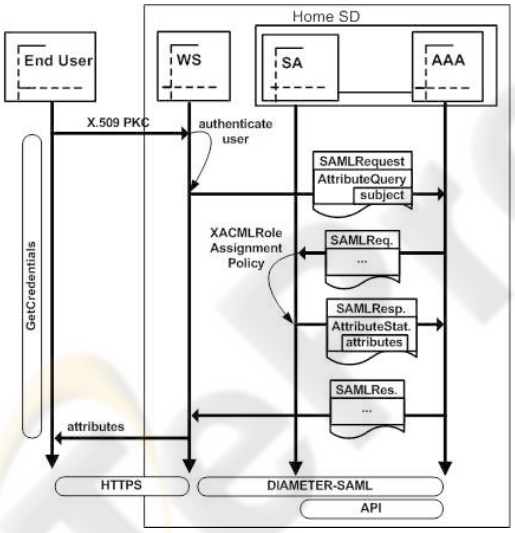

Fig. 4. Attribute Recovery.

5.2 Push Model

In the push model, the user selects in his home domain the attributes he wants to present to the target domain when requesting access to the Grid service. As Figure 4 shows, this process is done by means of a secure web server (WS), located in the user home domain, which returns the attributes to the user. First, the user and the WS authenticate mutually, using X.509 certificates. Then, the WS requests the user attributes to the home SA using the underlaying AAA infrastructure. It sends an *AttributeQuery* request to the AAA server asking for all the user attributes and this server obtains the requested information from the SA. The attributes are returned to the WS also through the AAA server in an

AttributeStatement response message. Finally the user selects the attributes he wants to disclose and the WS provides them to the user as digitally signed SAML *Assertions*.

Once the user has obtained the desired attributes, the X.509 Proxy Certificate [17] used to identify the user in GT, can be used to carry and present them to the GT3 server. The reason for using a X.509 Proxy Certificate instead of using a X.509 Attribute Certificate, is that the proxy is a short lived certificate created by the user from its own certificate, such as a ticket, which is currently supported by the Globus Toolkit since it is the mechanism used to identify users. In this way we do not need to incorporate additional functionality related to X.509 ACs. These attributes are added to the X.509 Proxy as non critical extensions which can be recovered from the GT3 server. Then, they can be used to obtain the authorization decision, as Figure 5 shows.

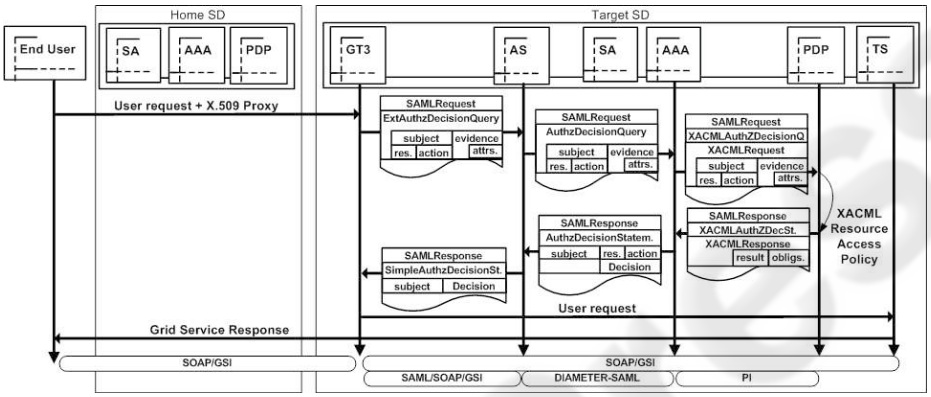

Fig. 5. Push Model.

When the GT3 server receives the user's request, it extracts the attributes and adds them as evidences to the *ExtendedAuthorizationDecisionQuery* message sent to the Authorization Service. Using the received data, this service builds an *AuthorizationDecisionQuery* which is sent to the local AAA server. From the user identity, the resource, the service method, and the evidences, the PDP responds to the AAA server using an *AuthorizationDecisionStatement* sentence, indicating the authorization decision.

6 Related Work

As we previously mentioned, several Grid authorization mechanisms have been proposed during the last years. In this section we analyze five of the most important solutions, outlining the main advantages and drawbacks related to each one of them.

Two authorization systems specifically designed for Grid environments are CAS [14] and VOMS [7]. They try to solve the problems of scalable and flexible representation and enforcement of access policies in a Virtual Organization using a server which maintains the community policies. Both systems only supports the push access mode, and in CAS the element which enforces authorization is the own target service.

In the Virtual Organization Membership Service (VOMS) the enforcement of the authorization is made by a gatekeeper, eliminating the need for modifying every service.

PERMIS and Akenti are two existing authorization systems which have been adapted to the Grid. PERMIS is a RBAC mechanism based on the use of X.509 Attribute Certificates and its own XML-based policy language. This system has been integrated into GT3 as authorization method by means of the PERMIS Authorization Service [2]. On the other hand, Akenti represents the resource access policy as a set of distributed signed certificates which are gathered up when necessary to take an authorization decision. In this system, resources are accessed via a resource gateway which contacts the Akenti server. The use of a plug-in to handle the interface between the job manager in GT2 and the Akenti server is described in [16], but a solution to integrate Akenti into OGSA Grids is not implemented. These two authorization systems have been designed to work in single organizations, so they are not suitable to be used in multi-domain environments, such as the grid. For example, they have not defined a protocol to exchange authorization information between the different organizations. Besides, they only permit the pull access mode in the Grid.

GridShib [18] is a project which main goal is the integration of Shibboleth and Globus to provide identity federation and attribute-based policy enforcement for Grids. Furthermore, Shibboleth offers pseudonymous interaction with the resources, which GridShib expects to incorporate into the Grid. This system also offers the possibility to access to the Grid using the push and pull models. GridShib allows a similar result that our proposal. The main differences between both alternatives are due to the underlying architectures, NAS-SAML and Shibboleth. In NAS-SAML the involved organizations take advantage of an already existing AAA infrastructure which had been previously deployed for other purposes, in this case, the network access control. On the other hand, Shibboleth is a more recent architecture mainly focused on web applications.

7 Conclusions and Future Work

This paper demonstrates that when two or more organizations wants to collaborate by means of a Grid, they have to introduce new authorization elements to manage their user's rights and the resource access control. Besides, in some cases the elements introduced are redundant because each organization has its own authorization mechanism, or collaborating organizations share a previous authorization infrastructure deployed with another intention. This paper proposes to take advantage of the underlying AAA infrastructure to provide an extensible and scalable authorization mechanism based on the use of SAML statements to represent the authorization data.

Due to NAS-SAML allows both pull and push access modes, this authorization mechanism offers these two kind of access to the Grid.

As a statement of direction we are integrating NAS-SAML in other high level applications, such as an admission control infrastructure for multimedia systems.

Acknowledgments

Partially supported by IST-2004-026600 project and TIC2003-08154-C06-03 project.

References

1. Globus alliance home page. `http://www.globus.org`.
2. D. Chadwick, S. Otenko, and V. Welch. *Using SAML to link the Globus Toolkit to the PERMIS Authorisation Infrastructure*, September 2004. Proceedings of Eighth Annual IFIP TC-6 TC-11 Conference on Communications and Multimedia Security.
3. D.W. Chadwick, O. Otenko, and E. Ball. Implementing role based access controls using x.509 attribute certificates. *IEEE Internet Computing*, pages 62 – 69, March – April 2003.
4. K. Czajkowski, D. Ferguson, I. Foster, J. Frey, S. Graham, T. Maguire, D. Snelling, and S. Tuecke. *From Open Grid Services Infrastructure to WS-Resource Framework: Refactoring & Evolution*, 2004.
5. C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, and D. Spence. *Generic AAA Architecture*, August 2000. RFC 2903.
6. A. Anderson et al. *EXtensible Access Control Markup Language (XACML) Version 1.0*, February 2003. OASIS Standard.
7. R. Alfieri et al. *Managing Dynamic User Communities in a Grid of Autonomous Resources*, 2003. Conference for Computing in High Energy and Nuclear Physics.
8. I. Foster, C. Kesselman, J.M. Nick, and S. Tuecke. *The Physiology of the Grid. An Open Grid Services Architecture for Distributed Systems integration*, 2002. Globus Project.
9. I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the Grid. enabling scalable virtual organizations. *Supercomputer Applications*, 2001.
10. O. Cánovas G. López and A. F. Gómez. *Use of XACML policies for a Network Access Control Service*, 2005. 4th International Workshop for Applied PKI, IWAP 05.
11. R. Marín G. López, A. F. Gómez and O. Cánovas. *A Network Access Control Approach based on the AAA Architecture and Authorzation Attributes*, 2005. First International Workshop on Security in Systems and Networks.
12. Antonio F. Gómez-Skarmeta Sassa Otenko Gabriel López, Óscar Cánovas and David Chadwick. *A Heterogeneos Network Access Service based on PERMIS and SAML*, 2005. 2nd European PKI Workshop.
13. Eve Maler, Prateek Mishra, and Rob Philpott. *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML)v1.1*, September 2003. OASIS Standard.
14. Laura Pearlman, Von Welch, Carl Kesselman, Ian Foster, and Steve Tuecke. *The Community Authorization Service: Status and Future*, 2003. Conference for Computing in High Energy and Nuclear Physics.
15. A. Streit, D. Erwin, Th. Lippert, D. Mallmann, R. Menday, M. Rambadt, M. Riedel, M. Romberg, B. Schuller, and Ph. Wieder. *UNICORE - From Project Results to Production Grids*, 2005.
16. M.R. Thompson, A. Essiari, K. Keahey, V. Welch, S. Lang, and B. Liu. *Fine Grained Authorization for Job and Resource Management using Akenti and the Gloubs Toolkit*, March 2003. Proceedings of Computing in High Energy and Nuclear Physics.
17. S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson. *Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile*, June 2004. RFC 3820.
18. V. Welch, T. Barton, K. Keahey, and F. Siebenlist. *Attributes, Anonymity and Access: Shibboleth and Globus Integration to Facilitate Grid Collaboration*, 2005. 4th Annual PKI R&D Workshop.
19. Von Welch, David Chadwick, Sam Meder, Laura Pearlman, and Frank Siebenlist. *Use of SAML for OGSA Authorization*, June 2004.
20. Von et all. Welch. *Security for Grid Services*, 2003. Twelfth Internation Symposiumon High Performance Distributed Computing.