

Attributes, Anonymity, and Access: Shibboleth and Globus Integration to Facilitate Grid Collaboration

Von Welch,¹ Tom Barton,³ Kate Keahey,² Frank Siebenlist²

¹National Center for Supercomputing Applications, University of Illinois

²Mathematics and Computer Science Division, Argonne National Laboratory

³Networking Services & Information Technologies, University of Chicago

Abstract

In this paper we describe our work in progress to integrate the Shibboleth SAML-based framework and Globus Toolkit's PKI-based security infrastructure. The result will provide identity federation and attribute-based policy enforcement for Grids that leverages the Shibboleth system being deployed on campuses. We provide an overview of both Shibboleth and the Globus Toolkit, present our motivating use cases, and describe our planned integration work.

1 Introduction

As virtual organizations (VOs) [9] increasingly turn into distributed multi-institutional collaborations, secure authentication and authorization become a growing challenge. In the existing Grid [8] infrastructure to support VOs, these mechanisms are typically based on the identities of the interacting entities. While this approach is simple and intuitive, as VOs expand, it becomes impractical to administer. VO membership may change dynamically, rights may be granted to entities on a periodic basis, or a user's role in an organization might dynamically evolve. Such factors make it more practical to express users' rights based on their other attributes, such as institutional affiliation or role in a collaboration, rather than identity alone.

Indeed, it may be desirable to enable anonymous interactions between users, thus

protecting individual privacy while still providing basic security services to system owners.

In this paper, we present our work to address this issue by integrating two widely accepted technologies: Shibboleth [20], an Attribute Authority service developed by the Internet2 community for cross-organization identity federation, and the Globus Toolkit's [10] Grid Security Infrastructure (GSI) [26]. Our project, which is funded by the NSF National Middleware Initiative [16], is known informally as "GridShib" [11]. The objective is to provide mechanisms whereby a Grid service can authenticate a user using GSI, determining the address of the Shibboleth attribute service in the process, and then obtain from the Shibboleth service the select user attributes that the Grid service is authorized to see. Attributes obtained in this way can then be used by the Grid service in making authorization decisions.

In Section 2, we describe Shibboleth and the relevant security portions of the Globus Toolkit. Section 3 introduces the use cases we plan to address. In Section 4 we discuss our plans for the Globus-Shibboleth integration, describing modes of usage, technical details, and planned implementation. In Section 5 we compare related technologies. In Section 6 we summarize our plans and conclude the paper.

2 Background

In this section we provide an overview of the two software products relevant to our work: Shibboleth and the Globus Toolkit. A more detailed description of Shibboleth [20] and the Globus Toolkit [10] can be found on the individual Web sites. We also describe the authentication standards used by each of these software systems.

2.1 Shibboleth

Shibboleth is a system that asserts attributes about a user between organizations. More precisely, it asserts attributes between the user's home organization and organizations hosting resources that may be accessible to the user. By using an attribute-based authorization model, the Shibboleth architecture is able to protect user privacy better: identifying information may, but need not, be among the attributes presented about the user.

Shibboleth can be conceptually regarded as comprising three components:

- *Handle Service*: The Handle Service authenticates users in conjunction with a local organizational authentication service and issues to the user a *handle token*. The handle token (comprising a SAML authentication assertion [19]) is a bearer credential containing an identifier, or *handle*. The Handle Service is intentionally neutral to the choice of the organizational authentication mechanism and can function with almost any such service (e.g., LDAP [25]).
- *Attribute Authority*: When a user requests access to a target resource, he presents his handle token. The resource then presents the user's handle token to the attribute authority and requests attributes regarding the user. The attribute authority enforces privacy

policies on the release of these attributes, allowing the user to specify which targets can access which attributes. The Shibboleth Attribute Authority retrieves attributes from an organizational authority and provides them in the form of SAML assertions. As is the case with the Handle Service, the Attribute Authority is intentionally neutral to the specific implementation of the organizational attribute service; LDAP is typically used today

- *Target Resource*: The target resource includes Shibboleth-specific code to determine the user's home organization and hence which Shibboleth attribute authority should be contacted for the user, to retrieve attributes regarding the user, and to make authorization decisions based on those attributes.

In normal Shibboleth usage, a new handle token is acquired each time the user accesses a different resource. A handle token can be reused on returning to a resource for a relatively short period of time. Each handle token has a different unique identifier for the user that is meaningful only to the Shibboleth attribute authority. As a result, a target resource cannot rely on the handle to learn the true identity of a Shibboleth user, nor can it correlate subsequent accesses as coming from the same user.

The current implementation of Shibboleth (version 1.2) is primarily designed to function with Web applications (i.e., standard Web servers and browsers). Plans for future implementations (starting with version 1.3) include support for non-Web applications.

Figure 1 shows the typical operation of Shibboleth. A number of steps not relevant to this paper are omitted for clarity.

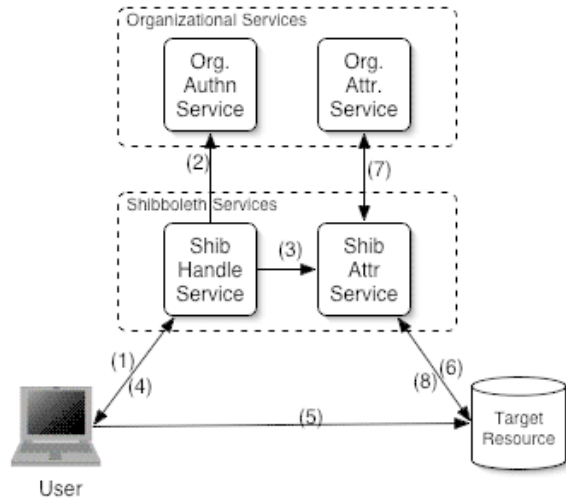


Figure 1: Simplified Shibboleth architecture and typical usage. Steps are described in the text.

The steps shown in Figure 1 are as follows:

1. The user connects and authenticates to the Handle Service.
2. The Handle Service uses a local organizational authentication service to verify the user's authentication information.
3. The Handle Service creates a new handle to identify the user. This handle is registered with the attribute authority so that it can be mapped to the user's attributes when a request from a resource arrives.
4. The handle is placed into a handle token and returned to the user.
5. The user sends a request to a target resource and presents the handle token.
6. The resource examines the handle token to determine which Shibboleth service can provide attributes about the user. It contacts that Shibboleth service and requests attributes, providing the handle token to identify the user.
7. After validity checks have been performed on the handle token and the handle has been mapped to the user's identity, the applicable attribute release policy for that resource is checked

whether communication of the requested user attributes is allowed. If so, the requested attribute values are retrieved.

8. The Shibboleth attribute authority casts the attributes in the form of a SAML attribute assertion and returns the assertion to the target resource.
9. (Not shown) After receiving the attributes from Shibboleth, the target resource makes an authorization decision regarding the user's request based on those attributes.

2.2 Globus Toolkit

The Globus Toolkit provides basic functionality for Grid computing [8], with services for data movement and job submission, and a framework on which higher-level services can be built. The Grid in general has been adopting Web services technologies, and this trend is reflected in recent versions of the Globus Toolkit in following the Open Grid Services Infrastructure [24] and now the Web Services Resource Framework [29] standards. This convergence of Grid and Web services was part of our motivation for adopting Shibboleth in our project (which uses the SAML standard).

The Grid Security Infrastructure, on which the Globus Toolkit is based, uses X.509 identity certificates [12] and X.509 proxy certificates [23, 27]. In brief, these certificates allow a user to assert a globally unique identifier (i.e., a distinguished name from the X.509 identify certificate).

We note that in Grid scenarios there is often a clear separation between the certificate authorities (CAs), which are the authorities of identity, and the authorities of attributes or authorization. For example, in the case of the DOE SciDAC program [18], a single CA, the DOE Grids CA [3], serves a broad

community of users, while the attributes and rights for those users are determined by their individual projects (e.g., National Fusion Grid, Earth Systems Grid, Particle Physics Data Grid).

Authorization in the Globus Toolkit is based on access control lists for each resource that specify the identifiers of the users allowed to access the resource. Higher-level services to provide richer authorization exist; we discuss these, and their relationship to this work, in Section 5.

2.3 GridLogon/MyProxy

GridLogon is a credential service being developed at NCSA as an extension to the popular MyProxy service [15]. MyProxy is a credential management service and is the de facto mechanism used to provide security to Grid portals worldwide.

Simply put, GridLogon acts as an online-CA, authenticating the user through a variety of mechanisms and issuing (short-lived) X.509 identity credentials suitable for use with Grid resources. GridLogon will provide a pluggable authentication mechanism to allow for the use of different local authentication systems, such as username/password, One-Time-Password, Kerberos, and Public Key.

3 Motivating Use Cases

In this section, we describe the use cases we wish to support with our work.

3.1 Project Leveraging Campus Attributes

The first scenario, shown in Figure 2, resembles the basic model in which Shibboleth is used today, except that the target resource is a Grid service instead of a Web-based application.

In this scenario, authorized users of the Grid service are located at one or more campuses and can be described by some campus-oriented attribute (e.g., chemistry professor). Verifying this attribute at their home institution authorizes user access to the Grid services.

An example of where this service could be applied in a Grid context is TeraGrid [1]. Each site on TeraGrid could operate a Shibboleth service in order to identify their staff and user community. This would enable TeraGrid resources to be easily available to the entire TeraGrid community without having comprehensive access control lists maintained on the resource.

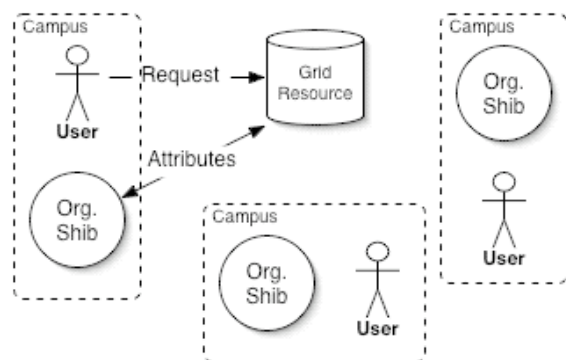


Figure 2: Scenario showing users being authorized based on their campus-assigned attributes.

3.2 Project-Operated Shibboleth Service

Figure 3 depicts a different scenario, where the project deploys and operates its own attribute authority. This scenario has the benefit that the project can freely assign attributes and add users as it wishes, without involving campus administration staff. However, the project must itself operate the Shibboleth service, a critical requirement from the perspective of both security and reliability. This approach is beyond the scope or capabilities of many projects.

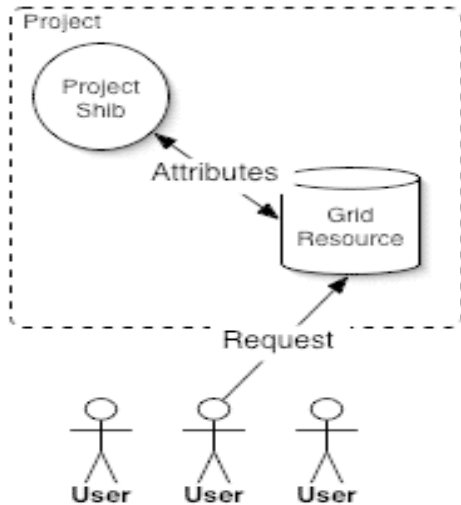


Figure 3: Scenario showing Shibboleth service operated by a project.

3.3 Campus-Operated, Project-Administered Approach

The scenario shown in Figure 4 is a hybrid of the two preceding scenarios. It empowers the project to administer its own attribute space while allowing the Shibboleth service to be maintained by campus staff who are expert in running such services.

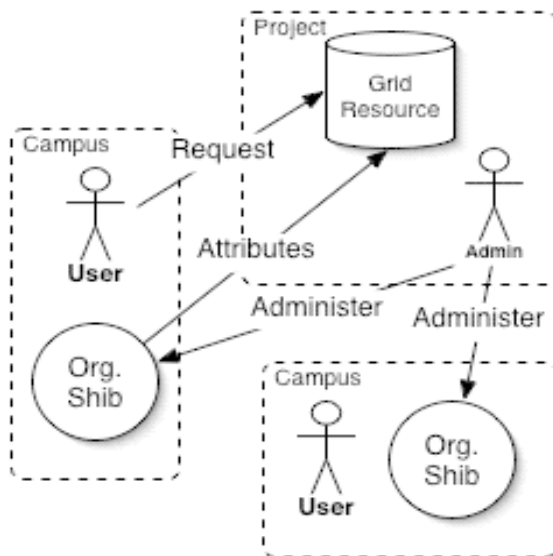


Figure 4: Scenario showing hybrid mode of operation, with the campus operating the Shibboleth service and the project administering its portion of the attribute space.

While we believe this hybrid approach to be the best of the three scenarios, it does require some form of administration delegation capabilities that is not present today. We address this issue in Section 4.3.

4 GSI-Shibboleth Integration

Our work comprises five major components:

- *Assertion Transmission* – to enable the transmission of assertions from the Shibboleth service to the Grid software and ultimately the Grid runtime authorization decision-making component.
- *Attribute Authority* – to enable discovery of the appropriate attribute authority for a user in the Grid context. Since Grid resources serve users from multiple organizations, a mechanism is needed to determine which organization's Shibboleth service is authoritative for a particular user.
- *Distributed Attribute Administration* – to manage subsets of an attribute space served by a Shibboleth service by projects outside the domain operating the Shibboleth service (as described in Section 3.3).
- *Pseudonymous Interaction* – to extend to Grids the pseudonymous interaction provided by Shibboleth.
- *Authorization* – to provide a mechanism that is integrated with the Globus Toolkit and can take advantage of the attributes.

4.1 Assertion Transmission

The fundamental engineering problem that must be solved is how to transmit user attributes from a Shibboleth attribute service to the Grid resource so that an authorization

decision can be made utilizing those attributes.

Two fundamental modes of operation address this problem:

- *Pull mode*: The target Grid service, after authenticating the user, will contact the appropriate Shibboleth service to obtain attributes regarding the user. This is analogous to normal Shibboleth operation today, as described in Section 2.1.
- *Push mode*: The Grid user, before contacting a target service, will contact an appropriate Shibboleth service to obtain attributes and then convey those to the target Grid service at the time of making a request. This is analogous to how other attribute authority systems in the Grid context work today, described in Section 5.

The pull mode of operation has the advantage of being more easily deployed; since clients of services are not affected and do not even need to know Shibboleth is involved in their decision-making. However, as we describe in the subsequent section on Attribute Authority discovery, the push mode has the advantage of allowing a user to select a particular role. Hence we plan on implementing both, to allow for flexible deployment to meet the requirements of different projects.

Regardless of the mode chosen, there exists the issue of federating the Grid identities, which consist of X.509 distinguished names (DNs), with the local identities used by the organization operating the Shibboleth service. This federation will require that a mapping be performed between the DN and the local site identifier. Shibboleth, as described in Section 2.1, already performs a similar mapping from the handle issued by the Handle Service to the local identity, and

the upcoming release of Shibboleth (version 1.3) will support a generalized version of this mapping feature capable of supporting DN, which will solve the basic problem of mapping identifiers.

4.2 Attribute Authority Discovery

One issue in distributed systems that serve users from multiple communities is determining which organization a particular user is from and hence the organization whose attribute authority that can provide attributes regarding the user. This is often referred to as the “Where are you from?” (WAYF) problem.

Shibboleth currently addresses this problem by asking users to identify their home organization when they attempt to access the target resource. In its current model of supporting interactive Web browser-based applications, this approach is acceptable. In the Grid context, however, where the user may be using client software that does not support this level of interactivity or the client may be an unattended batch job, we need a different approach.

We will explore the following possible solutions:

- Use the push mode of operation, described in Section 4.1 and have the user select the attribute authority to contact. This approach has been taken by other systems, such as VOMS described in Section 5.1. The main drawback is that it requires modification of client behavior or software, which can present a deployment challenge.
- Place a pointer (e.g., a hostname) to the attribute authority to contact in the user’s X.509 identity certificate. This solution requires cooperation of the CA issuing the user’s identity credentials, which

may not always be available, and also binds attribute information to the user's identity credential, which may raise problems if the lifetimes of these two elements are not in synch.

- Place a pointer to the attribute authority's location in the user's proxy certificate. Since the user can create proxy certificates fairly easily and with short lifetimes, this approach solves a number of problems with having the issuing CA place information in longer-term identity certificates. The actual placing of the information could probably be automated, and users could select from different attributes authorities, and even multiple authorities, depending on the specific role or roles they want to adopt.

A related challenge that we will explore is a scenario where a user may be acting in a role that combines attributes from multiple organizations or from the project and an organization. In this scenario the user's attributes would come from multiple Shibboleth services. It remains unclear at time whether this is a true requirement for a user community, so our exploration of this problem may be minimal.

4.3 Distributed Attribute Administration

The current Shibboleth attribute management paradigm assumes that the complete attribute space asserted by a particular attribute authority is managed within a single administrative domain. This model makes sense when all the attributes are concerned with the user's role in the single domain; for example, if the administrator works for the user's university and the attributes all concern the user's position at the university.

This attribute management model does not support resource targets wanting to use attributes that are asserted by other authorities. One example is an issue that is already being faced by the Shibboleth community and is known as the "IEEE problem": having universities provide IEEE membership status to allow resource targets to authorize based on their IEEE membership rather than on their campus affiliation. While the authoritative party for the attributes, IEEE in this case, could establish its own Shibboleth service, this approach may not always be desirable because some organizations may not have the resources or skills to operate a highly available secure attribute authority service.

A new privilege management system called Signet [21], which is being developed by a working group of the Internet2 Middleware Initiative, supports the distributed administration of privileges. Shibboleth-enabled access to Signet is planned, which will enable authorities outside of the administrative domain in which a Signet instance is operated to be delegated the ability to manage a portion of the attribute space that can be asserted by that domain's attribute authority. This arrangement has the potential to support the use case described in Section 3.3.

In collaboration with the Signet development team, we will explore the possibility of allowing administrative delegation of the attribute space in a single attribute authority service among multiple organizations as a means to solve this problem.

4.4 Pseudonymous Access

Shibboleth allows for pseudonymous access as part of its normal operation. To provide anonymity in the Grid context, we will integrate the GridLogon service with

Shibboleth and the Globus Toolkit. As we described in Section 2.3, the GridLogon service issues X.509 certificates to authenticated clients. We will implement an extension to GridLogon module that issues an authenticated client a set of credentials with a pseudonym identifier, which will make the GridLogon service essentially act as the Shibboleth Handle Service normally does. GridLogon will register the pseudonym with the Shibboleth attribute service, such that subsequent queries can be mapped to the user's attributes.

4.5 Authorization Mechanism

To allow for the use of attributes in the Globus Toolkit runtime, we need to extend the current ACL-based authorization mechanism to encompass attribute-based policy. We intend to leverage existing standards and implementations here to the greatest extent possible. Our implementation will most likely be very simple at first, for example, using attributes in place of identities to map users to local accounts.

We will explore the integration of XACML [6] with the Globus Toolkit security runtime, with a mapping of SAML attribute assertions to XACML attribute assignments as described in [14]. The result will be a Web services runtime that can make authorization decisions about the user's invocation request of the Web service operations based on the Shibboleth user's attribute and XACML policy rules. The aim is to make the attribute retrieval and the evaluation and enforcement of the authorization policy transparent to the application.

The Globus Toolkit currently supports an authorization callout [28], which allows external services to provide authorization decisions for Globus deployments as described in Section 5.3. Our goal is to

provide attributes received from Shibboleth to those external authorization services in order to allow them to incorporate those attributes in their decision-making process.

In parallel with our GridShib effort, the Globus Toolkit team has also started work on a more ambitious authorization-processing framework. As the toolkit is used by many different Grid applications and projects worldwide, it cannot mandate specific security technologies and mechanisms, and has to adopt a modular approach to accommodate the choices made by those responsible for deployment. For example, identity and attribute assertions have to be supported in X.509 Identity and Attribute Certificate, Kerberos, and SAML Identity and Attribute Assertion formats. Furthermore, all these statements can either be available in local storage within the trusted computing base of the relying party, be pushed by other parties via SOAP headers or Proxy Certificate embedding, be pulled from online services, or external attribute and authorization services can be queried through Shibboleth/SAML call-out interfaces.

In the first step of this authorization processing, the received and collected assertions with their associated issuers are verified and validated. The resulting attribute statements with their issuer-subject information are subsequently translated and mapped by mechanism specific Policy Information Points (PIPs) into a common format that is presented to the Policy Decision Point (PDP). Our GridShib effort should be able to leverage this authorization framework development work.

4.6 Planned Timeline

The GridShib project officially began in December 2004. Prior to that date we had identified requirements and made

preliminary project plans. We are now focusing on implementing the pull mode as described in Section 4.1; we expect to have a first release by the summer of 2005 based on the upcoming release of Shibboleth (version 1.3) and a post-4.0 version of the Globus Toolkit. Enabling the push mode and pseudonymous access will follow in 2006.

5 Related Work

Our work is distinguished from related work mainly through the Shibboleth support for pseudonymous interaction, its fine-grained attribute release policy, and its existing broad support base in the Internet2 community.

5.1 VOMS

The virtual organization management service (VOMS) [5] was developed by the European Data Grid project to allow for attribute-based authorization to the Globus Toolkit job management services. It uses X.509 attribute certificates [7] in a push mode to assert attributes in a modified version of the Globus Toolkit.

We believe that Shibboleth, with its use of SAML, will be more easily interoperable with Web services-based technologies emerging in the Grid community. VOMS also does not support a pseudonymous mode, nor does it have any other provisions for privacy support.

5.2 CAS

The Community Authorization Service (CAS) [17] is similar to Shibboleth in its use of SAML assertions. However CAS operates at the level of capabilities rather than attributes; that is, instead of expressing abstractly what someone is, CAS expresses explicitly what actions they are allowed to take. CAS also does not support a

pseudonymous mode or have any other provision for privacy.

5.3 Akenti and PERMIS

Akenti [22] and PERMIS [2] are authorization systems that have been integrated with the Globus Toolkit through the use of authorization callouts [13,28]. Both Akenti and PERMIS allow for the use of X.509 attribute certificates to make attribute-based authorization decisions. We envision our work as being complementary to these systems. Our focus falls on the technology to transport SAML assertions from the Shibboleth attribute authority to the Globus Toolkit-based services, whereas these systems are designed primarily as authorization decision makers. We envision a mode of operation in which these systems can be used to provide rich authorization capabilities using Shibboleth-issued SAML assertions in addition to the X.509 attribute certificates they use today.

5.4 Signet

The Signet privilege management system [21] is being developed by a working group of the Internet2 Middleware Initiative. Signet can manage privileges that are expressed as attributes asserted by Shibboleth. Signet itself is planned to be “Shibbolized” to support delegation of privilege management beyond the bounds of a single organization.

As discussed in Section 4.3, we plan to collaborate with the Signet team, in order to enable Signet to manage the access policy to Grid resources.

5.5 ESP-Grid Project

The ESP-Grid project [4] is evaluating how Shibboleth could be used to benefit Grid authentication. We have met with the members of the ESP-Grid project and will

stay in contact, sharing experiences and results of our work.

6 Conclusion

We have described the motivations for identity federation and for attribute-based authorization in Grids. We have described our plans for addressing these motivations through integration of Shibboleth and the Globus Toolkit in order to produce a system capable of enabling attribute-based authorization in Grids, leveraging existing campus Shibboleth infrastructure, and allowing for pseudonymity.

Acknowledgments

This work is funded by the NSF National Middleware Initiative, under award SCI-0438424.

Our work would not be possible were it not for our collaboration with the Internet2 Shibboleth development team chaired by Steve Carmody.

We also thank Ian Foster and Ken Klingenstein for their advice and guidance.

“Globus Toolkit” is a registered trademark of the University of Chicago.

The submitted manuscript has been created by the University of Chicago as Operator of Argonne National Laboratory (“Argonne”) under Contract No. W-31-109-ENG-38 with the U.S. Department of Energy. The U.S. Government retains for itself, and others acting on its behalf, a paid-up, nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

Author Contact Information

Tom Barton

tbarton@uchicago.edu

Kate Keahey

keahey@mcs.anl.gov

Frank Siebenlist

franks@mcs.anl.gov

Von Welch

vwelch@ncsa.uiuc.edu

References

1. Catlett, C. The TeraGrid: A Primer, 2002. www.teragrid.org.
2. Chadwick, D.W. and Otenko, A., The PERMIS X.509 Role Based Privilege Management Infrastructure. 7th ACM Symposium on Access Control Models and Technologies, 2002.
3. DOEGrids Certificate Service, <http://www.doegrids.org>, 2004.
4. ESP-GRID – Evaluation of Shibboleth and PKI for GRIDS. http://e-science.ox.ac.uk/oesc/projects/index.xml.ID=body.1_div.20, 2004.
5. EU DataGrid, VOMS Architecture v1.1. 2003. http://grid-auth.infn.it/docs/VOMS-v1_1.pdf.
6. eXtensible Access Control Markup Language (XACML) 1.0 Specification, OASIS, February 2003. <http://www.oasis-open.org/committees/xacml/>
7. Farrell, S., and Housley, R., An Internet Attribute Certificate Profile for Authorization. RFC 3281, IETF, April 2002.
8. Foster, I., and Kesselman, C. (eds.). *The Grid 2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 2004.
9. Foster, I. Kesselman, C., and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International J. Supercomputer Applications*, 15(3), 200–222, 2001.

10. Globus Toolkit. <http://www.globus.org/>, 2004.
11. GridShib Project Web site, <http://grid.ncsa.uiuc.edu/GridShib>
12. Housley, R., Polk, W., Ford, W., and Solo, D., Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. *RFC 3280*, IETF, April 2002
13. Keahey, K., Welch, V., Lang, S., Liu, B. and Meder, S.. Fine-Grain Authorization Policies in the Grid: Design and Implementation. In 1st International Workshop on Middleware for Grid Computing, 2003.
14. Lorch, M., Proctor, S., Lepro, R., Kafura, D., and Shah, S., First Experiences Using XACML for Access Control in Distributed Systems, ACM XML Security Workshop, October 2003.
15. Novotny, J., Tuecke, S., and Welch, V., An Online Credential Repository for the Grid: MyProxy. In *Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10)*, IEEE Press, August 2001
16. NSF Middleware Initiative (NMI). 2004. www.nsf-middleware.org.
17. Pearlman, L., Welch, V., Foster, I., Kesselman, C. and Tuecke, S., A Community Authorization Service for Group Collaboration. IEEE 3rd International Workshop on Policies for Distributed Systems and Networks, 2002.
18. Scientific Discovery through Advanced Computing (SciDAC), <http://www.scidac.org>, 2001.
19. Security Assertion Markup Language (SAML) 1.1 Specification, OASIS, November 2003.
20. Shibboleth Project, Internet2, <http://shibboleth.internet2.edu/>
21. Signet, <http://middleware.internet2.edu/signet/>, 2004.
22. Thompson, M., Johnston, W., Mudumbai, S., Hoo, G., Jackson, K., and Essiari, A., Certificate-based Access Control for Widely Distributed Resources. 8th Usenix Security Symposium, 1999.
23. Tuecke, S., Welch, V. Engert, D., Thompson, M., and Pearlman, L., Internet X.509 Public Key Infrastructure Proxy Certificate Profile, *RFC 3820*, IETF, 2003.
24. Tuecke, S., et. al. Open Grid Services Infrastructure (OGSI) Version 1.0, Global Grid Forum, 2003.
25. Wahl, M., Howes, T., and Kille, S., Lightweight Directory Access Protocol (v3), RFC 2251, IETF, 1997.
26. Welch, V., Siebenlist, F., Foster, I., Bresnahan, J., Czajkowski, K., Gawor, J., Kesselman, C., Meder, S., Pearlman, L., and Tuecke, S., Security for Grid Services. In *12th IEEE International Symposium on High Performance Distributed Computing*, (2003).
27. Welch, V., Foster, I., Kesselman, C., Mulmo, O., Pearlman, L., Tuecke, S., Gawor, J., Meder, S. and Siebenlist, F.. X.509 Proxy Certificates for dynamic delegation. In *Proceedings of the 3rd Annual PKI R&D Workshop*, 2004.
28. Welch, V., Ananthakrishnan, R., Meder, S., Pearlman, L., and Siebenlist, F., Use of SAML for OGSA Authorization (work in progress), Global Grid Forum, May 14, 2004.
29. WS-Resource Framework. <http://www.globus.org/wsrf/>, http://www.oasis-open.org/committees/tc_home.php?wg_aabbrev=wsrf, 2004.