



SERAPIS Project

A Survey of Grid-Shibboleth projects

**Arts and Humanities Data Service
King's College London**

Author: Sanjay Vivek

Contact: Mark Hedges

13 March 2007

Table Of Contents

1.	Introduction	6
1.1	Motivation	6
1.2	Report Structure	6
2.	e-Science	8
2.1	Introduction	8
2.2	e-Science in the Arts and Humanities	8
3.	Grid Standards and Related Technologies.....	10
3.1	Introduction	10
3.2	OGSA	10
3.3	WSRF	10
3.4	OGSA-DAI	11
3.5	SRB	11
3.6	Globus Toolkit	12
3.7	OMII-UK.....	13
3.8	Condor.....	14
3.9	MyProxy.....	14
3.10	PERMIS	14
3.11	Grid Portals	15
4.	Grid Security.....	17
4.1	Introduction to Grid Security and PKI.....	17
4.2	Grid Security Requirements.....	17
4.3	Security Fundamentals	18
4.4	Symmetric Key Encryption.....	18
4.5	Asymmetric Key Encryption.....	19
4.6	The Certificate Authority	19
4.7	Digital Certificates	20
5.	Federated Identity Management with Shibboleth.....	22
5.1	Introduction	22
5.2	Related Standards.....	22
5.2.1	SAML	22
5.2.2	Liberty Alliance	23
5.2.3	WS –Federation	24

5.3	Shibboleth	24
5.4	Shibboleth architecture	24
6.	Shibboleth and Grid Computing.....	27
7.	Relevant Shibboleth based Grid Projects.....	29
7.1	Introduction	29
7.2	GridShib	29
7.2.1	GridShib for Globus Toolkit.....	29
7.2.2	GridShib for Shibboleth.....	29
7.2.3	GridShib Profile.....	30
7.2.4	Conclusions.....	31
7.3	GridShibPermis	32
7.3.1	Conclusions.....	33
7.4	ESP-Grid	33
7.4.1	Relevance of Shibboleth and PKI in a Grid infrastructure.....	33
7.4.2	Future users of the Grid	34
7.4.3	Conclusions.....	35
7.5	ShibGrid	35
7.6	SHEBANGS.....	38
7.6.1	ShibGrid vs SHEBANGS	39
7.7	BRIDGES.....	41
7.7.1	Portal Technology.....	42
7.7.2	Data Integration	42
7.7.3	Security	42
7.7.4	Conclusions.....	42
7.8	DYVOSE.....	43
7.8.1	Design and Implementation.....	43
7.8.2	Conclusions.....	44
7.9	VOTES	44
7.9.1	Architecture and Implementation	44
7.9.2	Conclusions.....	45
7.10	GLASS	45
7.11	SPIE.....	46
8.	Shibboleth in e-Humanities Projects	47
8.1	Current e-Humanities projects	47
8.1.1	DAM-LR.....	47
8.1.2	TextGrid.....	48
8.2	Future Projects	48
8.2.1	ASPiS.....	48
8.2.2	SARAH.....	50
9.	Conclusions.....	52
10.	References.....	53



1. Introduction

1.1 Motivation

To increase the uptake of Grid technologies within the Arts & Humanities domain, access to and the usage of Grid resources need to be simplified. Security is one aspect that needs to be made as simple as possible for Grid users.

As noted in research from the ESP-Grid project [16], current Grid middleware is intimidating for many users. This is especially true when it comes to handling security issues on the Grid. The complexity of digital certificates and PKI in general, may dissuade non-technical users from adopting Grid technology. Researchers need to be convinced that adopting Grid technology will simplify and quicken their daily research. Currently, by necessity, Grid users are mostly very technically proficient, and as such, the uptake of Grid technology is slow within the Arts and Humanities domain. Users should be able to access Grid services in a secure manner but via a security mechanism that is familiar to them, i.e. their local institution's username/password pair. This concept of authenticating the user at the user's local institution and not by some central authority, is commonly known as devolved authentication. Although devolved authentication can be done with PKI, it does not however scale well. Authorisation attributes management is another feature that does not scale well with PKI. Shibboleth, a federated identity management architecture developed by Internet2, shields the user from PKI by supporting devolved authentication and managing authorisation attributes very effectively. Shibboleth provides the user with a Single Sign-On architecture, which they are familiar with, and the user only has to authenticate once at their local institutions. This benefits the user in terms of having only to learn one Single Sign-On interface, and places the responsibility of managing identities and attributes with the most appropriate institution, which is commonly the user's local institution.

This report focuses upon several Grid and e-Science projects that have used Shibboleth to simplify the authentication and authorisation of Grid users. The use of Shibboleth, and other security technologies allow for dynamic establishment of virtual organizations with fine-grained security at their core.

1.2 Report Structure

The remainder of this document is organised in the following manner:

Chapter 2. e-Science

Justification and presentation of e-Science in the Arts & Humanities domain

Chapter 3. Grid Standards and Related Technologies

Discussion of the Grid standards and technologies that are appropriate for Grid computing, informed by existing grid computing activities.

Chapter 4. Grid Security

Description of the tools and techniques related to Grid security, which is pertinent to e-Science applications. This chapter feeds into chapters 5 and 7 in particular.

Chapter 5. Federated Identity Management with Shibboleth

Discussion of Federated Identity Management standards and technologies related to Shibboleth. This chapter feeds into chapters 6 and 7.

Chapter 6. Shibboleth and Grid Computing

Discussion about how Shibboleth handles devolved authentication very well in a Grid context.

Chapter 7. Relevant Shibboleth based Grid Projects

Description of projects that incorporate Shibboleth's attribute management and portability with Grid based applications.

Chapter 8. Shibboleth in e-Humanities Projects

Description of how Shibboleth has been applied in e-Humanities projects. This chapter also briefly describes two Grid based projects we have made bids for.

Chapter 9. Conclusions

A brief summary of the report in general.

2. e-Science

2.1 Introduction

The UK e-Science Core Programme was established in the late 1990s to help drive scientific research that was becoming increasingly reliant on collaborative and multidisciplinary efforts. The ever-increasing deluge of data from scientific processes mean that scientists need better computing and networking technology to generate, analyse, share and discuss their insights, experiments and results in a more effective manner. e-Science offers a promising vision of researchers that span across disciplines, laboratories, organizations and national boundaries collaborating together to support and enhance scientific processes. High-speed computing networks and greater processing power help researchers to collaborate on a variety of problems effectively and quickly. The underlying computer infrastructure that provides these facilities is commonly referred to as the Grid [24].

Currently, there are numerous e-Science projects that conduct research in various fields to support the growth of Grid computing. These new Grid technologies and methodologies enhance the research process by utilizing globally distributed data resources and sharing computational power. At the same time, new types of scholarly communications in the form of virtual organisations (VO) are being developed. For example, Access Grid provides tools to support structured meetings of researchers in group-to-group collaborations. This technology will be especially beneficial to Arts and Humanities (A&H) researchers as they move towards larger and more formal collaborations. The natural advantages of face-to-face meetings while being able to share digital resources instantly enhance collaborative research.

e-Science support for collaborative research leads to the globalisation of research whereby researchers from all over the world can work together and use each other's resources as if they were collocated. Digital knowledge objects can be created and re-used in virtual collaboration spaces. Essentially, e-Science is about collaborative research and not just about extra processing power and high-speed networking. It is about building pro-active relationships at various levels including machine-to-machine and researcher-to-researcher. This virtual collaboration effort at a global level is of key significance to A&H researchers in the future.

2.2 e-Science in the Arts and Humanities

The Arts and Humanities domain has not benefited fully from recent e-Science developments. This is despite the multi-fold increase in digital resources from these disciplines in the past decade. The Arts and Humanities Research Council commits roughly half its annual budget to projects, which produce some form of digital content, as did its predecessor, the Arts and Humanities Research Board. This deluge of digital material, which is often fuzzy, incomplete or inaccessible, provides the challenges that e-Science can address.

As in the science domains, increasingly new data and knowledge management tools are being employed in the A&H domain to deal with the new digital corpora. These tools help annotate data or resources with metadata in order to identify and describe the resource and their relationships with other resources. Metadata enabled technologies like the Grid are set up to overcome the limitations in access and interoperability of the digital corpora. As new ways of generating knowledge from data are explored, the A&H will find their place in a new data-driven research environment with shared resources and services.

The challenges faced by an A&H researcher when working with Grid technologies are probably not of the same scale as faced by researchers in other disciplines. The AHRC-JISC e-Science Initiative, a £1.8m national programme to promote and develop e-Science in the A&H, was formed to provide support for A&H researchers to work with Grid technologies. Among the main challenges faced by an A&H researcher are the complexities in obtaining and working with digital certificates and PKI. PKI provides secure and reliable access to Grid services through the management of keys and certificates. However, obtaining and the administration of digital certificates is a laborious process that might take several weeks. An A&H researcher also has to familiarise himself with PKI technology before being able to benefit fully from Grid technologies. The level of complexity involved might force A&H researchers to abandon the Grid in favour of less complex tools. A&H researchers have to be able to access Grid services in a secure manner but in a form they are familiar with. Shibboleth provides such a solution by allowing users to access Grid services by their local institutional usernames and passwords. As such, A&H researchers can be shielded from the complexity of PKI while being able to work with Grid resources themselves.

3. Grid Standards and Related Technologies

3.1 Introduction

Grid computing is made up of many concepts and can be defined in various ways. However, it essentially utilizes virtual distributed computing resources. Many diverse technologies can be used to implement such an environment, and as such, standards need to be defined and adopted to enable this wide range of hardware and software to interoperate. This chapter describes some of the key standards and evolving standards that apply to Grid computing.

3.2 OGSA

The Open Grid Service Architecture (OGSA) [40] was published by the Global Grid Forum (GGF) [21] to describe an architecture for a service-oriented Grid computing framework with an emphasis on business and research use. OGSA is based on several Web Service standards, primarily WSDL and SOAP, and it is a distributed interaction and computing architecture based on the concept of services. These services provide interoperability on heterogeneous platforms so that diverse services and resources can communicate and share information with each other.

OGSA as a framework requires a core set of interfaces, expected functionalities, resource models, and bindings. OGSA defines requirements for these core capabilities and thus provides a general reference architecture for Grid computing environments. It identifies the components and functions that are essential for a Grid based framework.

OGSA does not explicitly define programmatic interfaces or other aspects that would ensure interoperability between implementations. It is instead used to identify the products and functions that should be included based on a particular requirement.

3.3 WSRF

Web Services Resource Framework (WSRF) [66] defines conventions for managing *state* so that applications can reliably exchange information about changes in their state. In combination with WS-Notification and other WS-* standards, WSRF aims to host Grid services within a Web Services architecture. WSRF supersedes Open Grid Service Infrastructure (OGSI), which was the Grid community's initial effort to converge Grid and Web Services. Grid computing implementations was initially based upon the OGSI specification. OGSI was based on XML Schema Definitions and WSDL and provided a suite of useful tools for Web Services developers. However, OGSI was based on older Web Services standards and this was a hindrance to developers who were more accustomed to newer Web Services standards.

WSRF was a major advance over both OGSI and existing Web Services specifications as it supports stateful Web Service interfaces. However, Web Services interfaces are

usually stateless and OGSI overcomes this by including hooks for addressing stateful resources since every Grid computing node has an associated state. WSRF defines a new approach to accessing stateful resources, which supports the Grid computing infrastructure while also supporting Web Services in general. Essentially, OGSI supported a special-case solution for Grid computing while WSRF supports a generalised solution for all Web Services.

WSRF was adopted by the Globus Alliance's Grid computing tool kit, the Globus Toolkit 4.0 (GT4) in early 2005.

3.4 OGSA-DAI

OGSA-DAI [38] and its recently funded follow up project, Data Access and Integration 2 (DAIT), is a collaborative effort between Universities of Edinburgh, Manchester and Newcastle, the National e-Science Center, with IBM and Oracle as industrial partners. It was released in 2003 to facilitate the use of databases for developing Grid and Grid related applications. Its specifications include definition and development of generic Grid data services to enable access and integration with data held in relational databases, XML databases, and flat file structures.

OGSA-DAI aimed at making these data resources accessible within an OGSA compliant framework. The OGSA-DAI Grid services has basic functionalities to perform sophisticated operations like data federations and distributed queries in a Grid environment while protecting the developers from details like database driver technology, data formatting techniques and delivery mechanisms.

OGSA-DAI is made up of a number of co-operating Grid services. These Grid services form a middleware layer whereby clients can access remote data sources that hold the required data such as relational databases, XML databases or flat file structures.

OGSA-DAI supports both WSRF and WS-I [65] specifications and comes in two different implementations:

- OGSA-DAI WSRF 2.2 — for deployment with the Globus Toolkit and optionally Jakarta Tomcat
- OGSA-DAI WSI 2.2 — for deployment with Apache Axis 1.2RC3 or Apache Axis 1.2.1 on Jakarta Tomcat or with OMII 2.3.3

3.5 SRB

The Storage Resource Broker (SRB) [55], developed at the San Diego Supercomputing Center (SDSC), is a client-server middleware that enables unified searching of and access to heterogeneous, distributed, data resources on a very large scale. SRB is a core grid-enabling element within the UK e-Science policy and is used in a wide range of projects.

The SRB manages distributed and supports data grids that focus on data sharing. It enables unified access to data in heterogeneous data resources (different databases, different filesystems). The SRB system consists of the SRB client, the SRB server, the Metadata Catalogue (MCAT) database and the associated MCAT SRB server(s), which all sit above the archival databases (Oracle, PostgreSQL etc).

Data from various online databases in various formats can be stored, discovered, searched and accessed by researchers. A sample usage scenario would involve using SRB to uniformly search and process data using its metadata.

Although OGSA-DAI and SRB are both data integration and data access products, they differ in their approach. SRB puts more emphasis on managing and maintaining data files while OGSA-DAI's emphasis is on data retrieval. SRB is best suited when client applications are expected to receive data (or metadata) while OGSA-DAI is the better option when the client application is attempting to retrieve different formats of data from different sources and integrating it.

3.6 Globus Toolkit

The Globus Toolkit [59] is an open architecture, open source software toolkit developed by the Globus Alliance [58] project. It provides all the necessary libraries and components for running Grid jobs and supports the development of service-oriented Grid applications and infrastructures. Globus Toolkit is designed to address basic issues relating to security, data movement and management, resource access and management, and resource discovery. The latest version of the Globus Toolkit is version 4, Globus Toolkit 4 (GT4), and it combines Grid and Web Services through the adoption of the WSRF standards.

The Globus Toolkit uses the Grid Security Infrastructure (GSI) [70] to address security issues that arise from the sharing and coordinated use of resources over an open network. GSI provides a number of useful services for Grids, including mutual authentication and single sign-on. GSI is based on public key cryptography, transport-level and message-level security mechanisms. GSI authentication and authorization framework is based upon the use of X509 server certificates [68] and X509 proxy certificates [61], which asserts user's identity expressed as a unique X.500 Distinguished Name (DN).

Proxy certificates allow a user with a valid X.509 certificate to temporarily delegate his identity and user privileges (some or all user privileges) to another entity since the DN of a proxy certificate must be subordinate to the DN of the user, and this is done by issuing it a proxy certificate, and thus easing the authentication and authorization process. This enables a Single Sign-On mechanism since a Grid job is given its own PKI key pair and proxy certificate, and it can authenticate to various systems on the Grid without additional input from the user. PKI and Grid security will be explained in greater detail in Chapter 4.

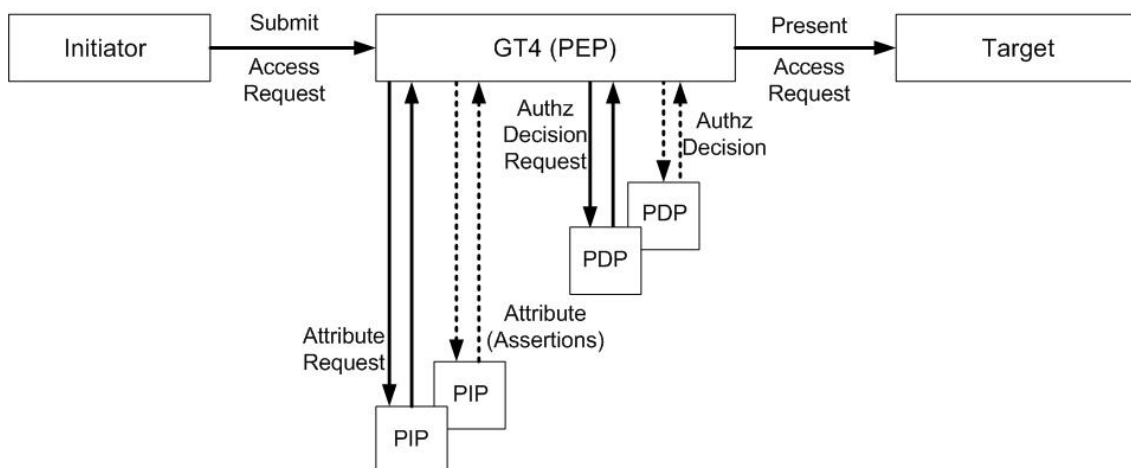


Figure 1. Globus Toolkit 4 Authorisation Architecture [6].

As shown in Figure 1, the GSI server-side authorization determines the access rights of a user who attempts to access a Grid resource by implementing a decision engine, which evaluates a chain of authorization schemes called Policy Decision Points (PDPs). This authorization chain may also consist of Policy Information Points (PIPs), which are merely used to collect information such as attributes or attribute assertions, which are vital in the decision making process. Globus classifies PDPs and PIPs as interceptors. GT4 is seen as the Policy Enforcement Point (PEP), which enforces the decisions made by the PDPs, and passes the decisions to the PDPs via the PIPs. However, by default, authorization in GSI is based on Access Control Lists (ACL), which is located in a Grid-mapfile, and this Grid-mapfile details the DN's of the users who are allowed to access each Grid resource. However, this approach is not scalable and inconvenient for large-scale distributed systems. The main drawback to this approach is that a user is granted all rights or no rights at all, depending if the user's DN is in the Grid-mapfile.

As of version 4, Globus Toolkit GSI uses SAML to access external third party authorization decision services, through either the SAML AuthorizationDecision protocol callout [69] to PDPs or the SAML Attribute Request protocol callout to PIPs, thus enabling specialised PDPs and PIPs to be included in the GT4 authorization chain. These callouts are used in GridShibPermis (explained in greater detail in 7.3), which implements Shibboleth as the PIP and PERMIS as the PDP.

3.7 OMII-UK

Open Middleware Infrastructure Institute UK (OMII-UK) [41] is a collaborative effort between the School of Electronics and Computer Science at the University of Southampton, the OGSA-DAI project at the National e-Science Centre and EPCC, and the myGrid [33] project at the School of Computer Science at the University of Manchester. The OMII-UK mission is to provide software and support to the UK e-Science community and its international partners. They provide Grid software distributions to researchers and industrial developers who wish to develop applications

for a Grid infrastructure. OMII-UK also invests in community developers to provide easy to use and open-source software that provides a secure Web Service hosting environment, Web Services and the necessary tools and environments to access these services. They also ensure that this software is supported through comprehensive training and documentation.

The OMII distribution 3.1.0 is available from <http://www.omii.ac.uk/downloads/>. It is an open source middleware that host applications on an OMII server, and these applications can be accessed securely by OMII client software. The middleware is built on a Web Services container, with WS-Security and additional OMII-specific services. Applications either run directly on the server or with a local job management system like Condor (explained in greater detail in Section 3.8).

3.8 Condor

Condor [9] is batch queuing system for managing compute-intensive jobs and was developed at University of Wisconsin Madison. It does this by providing a High Throughput Computing (HTC) environment. Essentially, an HTC environment makes efficient use of available resources while providing high throughput for jobs. Like other full-featured batch systems, Condor supports traditional queuing, scheduling functionalities, resource monitoring, and resource management, alongside newer technologies like resource classifications. A typical usage scenario consists of a user submitting a job to Condor, which then queues and monitors the job, before presenting the results to the user once the job has been completed. Traditionally, batch systems use a dedicated set of machines owned by a single enterprise or organization but Condor extends this functionality by managing non-dedicated resources when they are not in use.

3.9 MyProxy

MyProxy [34] is an online credential repository for the Grid and has been used widely in numerous Grid based projects for the past 6 years. Users can retrieve their delegated X509 credentials via the Grid Security Infrastructure (GSI). Storing Grid credentials in a MyProxy repository allows a user to retrieve a proxy credential whenever and wherever the user might be, and without having to worry about managing private key and certificate files. A user can also connect to a Grid portal and retrieve a proxy credential, which accesses a Grid resource on the user's behalf. A user can also allow trusted servers to renew their proxy credentials using MyProxy to ensure that any long running jobs do not fail in the event of an expired proxy credential.

3.10 PERMIS

The Privilege and Role Management Infrastructure Standards Validation (PERMIS) [44] software is an authorization system that can implement Role Based Access Control (RBAC) authorisation systems [10]. PERMIS can either be implemented as a Java based API or as SAML request/response messages, which in turn makes authorization decisions about the accessibility of a particular resource. Rules governing access rules

and rights are defined using XML policies [11]. Additionally, these policy files also define role hierarchies (or privilege inheritance), Sources of Authority (SOA) who are trusted to allocate which roles to whom, and delegation of duties (or privilege delegation). The resulting XML policies are defined in the form of X509 Attribute Certificates (ACs) and stored in one or more LDAP databases. These XML policies are then retrieved by the PERMIS access control decision mechanism when performing authorisation checks on a user.

A PERMIS user accesses resources via the application gateway, which is comprised of an Application dependent Enforcement Function (AEF) and an Application independent Decision Function (ADF). A user's access request comprises of the user's name, the target resource name, and the requested action. Optional parameters such as time of day or resource availability may also be included. The decision is made based on the VO policy and the roles of the user retrieved from the AC at decision time. The PERMIS policy is comprised of a Role Allocation Policy (RAP) and a Target Access Policy (TAP). ACs are checked against the RAP by the AEF and valid attributes are passed to the ADF, which returns a granted or denied response according to the enforced policy.

PERMIS implements authorization on top of existing X509 authentication systems (PKI) and is also compatible with Shibboleth. A Grid/Globus based project that incorporates the functionalities of PERMIS and Shibboleth called GridShibPermis is explained in greater detail in Section 7.3.

3.11 Grid Portals

Grid computing portals is a major component of many large-scale Grid computing projects. A portal is a Web application that runs in the Tomcat server while a portlet is an application that is accessible via the portal, e.g. a Grid service. Portals have standard interfaces for accessing Grid services, resources, applications, tools or collaboration services for federations of researchers. Portals also deliver complex grid solutions to users without having to worry about pre-installed Grid software. As a result, a researcher is protected from the complexities of accessing a Grid resource.

A typical scenario may involve a user navigating to the portal page and being shown the portlets the user is authorized to access based on their identity and the authorization policies in place. This can help in the formation of virtual organisations (VOs) by composing separate Grid services into one front-end Web page.

A user is usually granted entry to a portal by a username/password pair. In a Grid portal, this username/password pair is used to associate a user's identity with a DN, before generating a proxy certificate for future Grid job submissions. As a result, the portal has to carry an Access Control List of its valid users so that only authorized users are given the necessary privileges.

Several Grid portal solutions are available including IBM's WebSphere Portal Server [26] and GridSphere [23]. WebSphere Portal Server is used in BRIDGES (explained in



greater detail in Section 7.7) and is a highly robust and reliable solution although it is difficult to integrate with other applications because it makes use of proprietary software. GridSphere is an open source solution that runs on Tomcat and can be easily adapted to integrate with other Grid based applications. Other significant portal technologies include PURSE [43] and GAMA [22].

4. Grid Security

4.1 Introduction to Grid Security and PKI

Security is a major feature in Grid computing. It covers a wide range of issues including authentication, data integrity, authorization, and auditing. These issues have to be properly addressed if Grid computing is to be more widely adopted by corporate and government IT departments. Without a proper security policy in place, the integrity and confidentiality of the data processed within the Grid would be at risk. To effectively secure a Grid environment, there are numerous tools and technologies available. This chapter examines several of these technologies.

Understanding basic Grid security requirements and security fundamentals are necessary to better evaluate Grid security. Grid security is based upon established security standards. This chapter looks at the fundamentals of Grid security and the underlying technologies that enable Grid security.

4.2 Grid Security Requirements

A virtual organisation is among the fundamental concepts in Grid computing. A virtual organization (VO) can be defined as a dynamic group of individuals, groups, or organizations, which define the conditions and rules (business objectives and policies) for sharing resources [17].

In a Grid VO environment, the entities that participate in Grid transactions can be distributed in different trust and management domains, which can span governmental, industrial and academic organisations. These entities are also related to each other in an ad-hoc manner. As such, a comprehensive Grid security framework that adheres to local domain-level security policies and VO-defined policies is required. To fulfil this requirement, the Grid security framework needs to interoperate with multiple administrative domains while maintaining a clear separation of the security policies deployed by the virtual and real (local) domains. The main security challenges faced in a Grid environment can be summarized as follows [35]:

- **Integration** - The Grid security infrastructure is required to integrate with existing security frameworks, and which could span different platforms and hosting environments. The underlying Grid security infrastructure has to be implementation independent and be able to extend new security services or patches when available.
- **Interoperability** - The Grid services that span across multiple administrative domains has to be able to exchange messages with each other without any restrictions, enable separate security policies for each domain, and authenticate and differentiate a user from one domain to the next.

- Trust Relationships - A Grid service request can span multiple domains and these domains have to establish trust with each other. The dynamic nature of a Grid environment makes it unfeasible to implement trust mechanisms prior to runtime. This issue of trust is further complicated with transient Grid services.

4.3 Security Fundamentals

Security consists of three fundamental services; Public Key Infrastructure (PKI), Authentication, and Authorization. PKI defines the comprehensive system required to provide public-key encryption and digital signature services. The primary aim of PKI is to manage keys and certificates. PKI helps an organisation to maintain a secure networking environment through the management of keys and certificates. PKI uses a combination of encryption and digital signature services to provide secure access to a wide variety of applications. Public-key encryption and digital certificates will be explained in greater detail in the latter parts of this chapter.

It is important to differentiate between authentication and authorization. Authentication is the process of establishing if the user is who claims to be, and this is typically a permanent attribute. Authorization is the process of establishing what a user is allowed to do, and this attribute can vary over time since the user's position in the organization might change over time and he may have less or more rights. PKI helps solve authentication by delivering user credentials provided the initial enrolment is securely done and assuming the signing Certificate Authority (explained in greater detail in Section 4.6) is trustworthy. Authentication can be verified by PKI if the user can prove his certificate is valid. Authorization, which is based on a set of user attributes, is less applicable with PKI. A PKI certificate may contain a set of user attributes used by an application to determine a user's authorization but these attributes are static for a given user and might not have been verified, which causes a number of problems. For example, if a user's role has changed, then some attributes might have to be updated as well, thereby invalidating the certificate. Another problem that might arise is that not every application requires the same set of attributes so the user might need different certificates for different applications, or a super certificate which contains the entire subset of attributes, which invariably means that many services will receive more information about the user than necessary. Consequently, PKI is not a scalable solution for exchanging user attributes for authorization. Shibboleth offers a more distributed model for authorisation and is explained in greater detail in Chapter 5.

4.4 Symmetric Key Encryption

Symmetric key encryption involves using a single key to perform both the encryption and decryption of data. The key must only be distributed to the two parties involved in the transaction, the sender and the receiver, to ensure that the data is only read by these two parties. The integrity of the data is compromised if a third party accesses the key. In general, symmetric key encryption is fast and secure but requires additional care and administration of the shared key. It's commonplace to use asymmetric key encryption with symmetric key encryption for the management of keys.

4.5 Asymmetric Key Encryption

Asymmetric key encryption is also known as public key encryption or cryptography. In public key cryptography, an asymmetric key pair (public key and private key) is used. The key used for encryption is different from the one used for decryption. Public key encryption requires the private key to be secured while the public key can be made available to the public. Generally the public key is made available in the digital certificate that is issued by a Certificate Authority.

The cryptographic algorithm based upon the public key and private key pair is designed so that an encrypted message can only be decrypted with the corresponding key of that key pair. This means that an encrypted message cannot be then decrypted with the same encryption key.

For example, if a public key encrypts a message, the message can then only be decrypted with the corresponding private key for that public/private key pair. One of the keys is designated as the public key because it is made public by the owner of the public/private key pair, and a trusted Certificate Authority guarantees the ownership. The corresponding private key is kept securely by the owner and never made available to the public.

Public key encryption can be used to securely send a message between two parties. The sender first encrypts the message using his private key and then encrypts the message again with the receiver's public key. The receiver in turn firstly decrypts the message using this private key and then the public key of the sender. In this instance, both parties can be assured that an intercepted message cannot be read by anyone else.

Although public key encryption improves security, it does however require a long encryption time, especially for large amounts of data. As such, public key encryption is often used to securely transmit a symmetric encryption key between two parties, and all further encryption is performed using this symmetric key.

4.6 The Certificate Authority

In the current Grid environment, a Grid resource needs a certificate signed by an authorised Certificate Authority before it can communicate with another Grid resource. A fully implemented Certificate Authority (CA) has many responsibilities and these responsibilities should be diligently followed to provide a secure and complete security framework. The primary responsibilities of a CA include:

- Correctly identifying the entities that request for certificates
- Issuing, revoking, and archiving certificates
- Protecting the CA server
- Providing unique namespaces for certificate owners
- Serving signed certificates to those needing to authenticate entities
- Complete logging functions

Within certain PKI environments, a Registrant Authority (RA) works with the CA to help with the user verification duties. The RA is responsible for validating a request for a certificate of public keys and forwarding any relevant user information to the CA. The RA is generally responsible for ensuring that user's information is valid before issuing a signed digital certificate. The Globus Toolkit provides a simple CA for testing purposes and functions as both the RA and CA. However, in a production Grid environment, commercial PKI solutions are recommended.

Among the primary concerns in a Grid PKI environment is ensuring its trustworthiness. A CA has to issue a certificate verifying its own identity before it can sign and issue certificates for others. This is done with the following steps:

1. The CA generates a random public/private key pair.
2. The CA protects its private key.
3. The CA issues its own certificate verifying its identity.
4. The CA signs this certificate with its private key.

The CA's private key is among the most important components in the whole PKI. Once compromised, the CA's private key can be used to impersonate anyone in the Grid network.

4.7 Digital Certificates

A digital certificate is a digital certificate that associates a Grid resource with its specific public key. A certificate contains a public key and relevant details about the owner of the public key. A certificate is seen as an unforgeable and tamper-proof electronic ID once it has been signed by a CA for a particular Grid VO. In essence, a digital certificate certifies that the enclosed public key belongs to the entity listed in the certificate.

Digital certificates, also known as X509 certificates, generally do not contain any confidential information, and as such does not pose a security risk if made available to public.

A digital certificate typically consists of a unique distinguished name (DN) and certificate extensions that contain further information about the user of host that is being certified. The information in this section may include the user's email address, physical location or address, and organization unit as shown in Figure 2.

When a Grid user wants to initiate a Grid session with a Grid service or application, the user attaches his digital certificate with the request. Upon receiving the request with the certificate, the Grid service checks the signature of the CA within the certificate. The Grid service can be assured of the authenticity of the user if an authorised CA signed the user's certificate

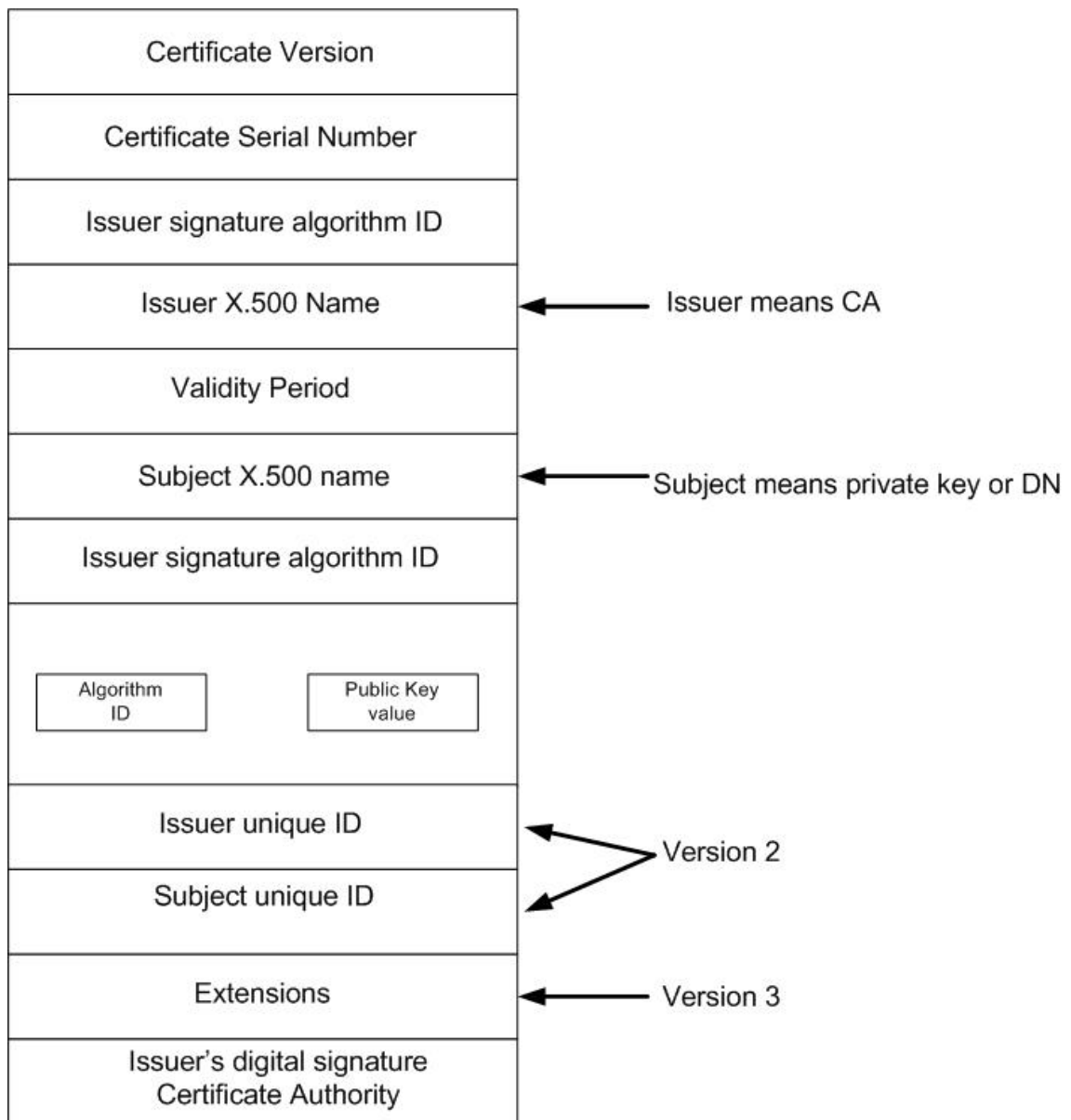


Figure 2 Graphical depiction of a digital certificate.

When communicating in a Grid environment, the Grid service will use the user's public key (contained in the digital certificate) to decrypt the SSL session ID, which is used to encrypt all data communication between Grid computers.

5. Federated Identity Management with Shibboleth

5.1 Introduction

A federation is a group of organizations or domains (universities, corporations, content providers) that agree on a common set of rules and standards for the sharing of resources and services. The primary purpose of a federation is to cooperate in inter-organizational authentication, authorization and accounting. In essence, federation enables identities to be portable.

Federated Identity Management (FIM) enables an authenticated identity to span across multiple domains within a federation. It allows users to associate their identity information with different domains while negating the need for a centralized storage of personal information. The primary goals of federated identity services are:

- Reducing the cost of identity management by ensuring an individual's identity is only managed at their home institution and not duplicated elsewhere.
- Ensure that only relevant parties are allowed to access identity information.
- Build on open standards to ensure secure and reliable transactions
- Guaranteeing user privacy by enforcing user preferences regarding the release of identity information.
- Respect an organization's existing trust relationship by ensuring an organization does not have to establish a new trust relationship unless it wants to.

Technology standards for identity federation include SAML (Security Assertion Markup Language), the Liberty Alliance framework, and WS-Federation. These standards will be explained in greater detail in the next section. Shibboleth is an open source implementation of the federated identity management model for higher education. It triggers authentication within an institution and supports the secure exchange of identity information between institutions in a federation. Shibboleth is explained in greater detail in Section 5.3.

5.2 Related Standards

Standards are vital to federation deployment because they provide the common suite of protocols, semantics and business logic that enable the various parties and their identity frameworks to interoperate. This section describes some of the relevant standards associated with federation and how they are interconnected.

5.2.1 SAML

The SAML (Security Assertion Markup Language) [46] framework is an OASIS [42] approved standard that facilitates the exchange of authentication and authorisation information across heterogeneous systems. This enables Service Providers (the owners of resources) to query Identity Providers (owners of user data) for information about

their users and to grant access based on this information. SAML is an XML based framework and uses XML to define a standard set of message exchanges between systems to enable the sharing of user identity information. The framework is also used to describe the content of these messages and how they should be exchanged. SAML is more than just a markup language because it also consists of protocols, bindings, and profiles. SAML specifies a request-response protocol, which can be used by a Service Provider to request assertions from the Identity Provider. A binding defines how SAML protocol messages are to be transmitted using SOAP over HTTP, and profiles determine how SAML can be used by standard web browsers.

SAML is increasingly becoming the standard way of exchanging security information and is rapidly being adopted by many commercial and non-commercial products. The main advantage in using SAML is that the exchange of messages is platform and application independent. This allows different applications running on heterogeneous platforms to interoperate with each other in a secure manner.

The messages exchanged using SAML are either in the form of assertions or requests for assertions. Requests for assertions initiated by Service Providers and contain information about the Service Provider. Assertions are initiated by Identity Providers, containing information about a user, and can be in the form of authentication assertions or attribute assertions. Authentication assertions contain information such as “*John Doe has logged into X University using a username/password pair at 12.34pm*”. Attribute assertions contain information such as “*John Doe is a postgraduate and is a member of the Y Department*”, and Service Providers can use this information for authorisation decisions.

SAML allows extensions, known as SAML profiles, to the basic protocol. Shibboleth is an example of this profile. Extensions specific to Shibboleth include the concept of anonymity, federations, and a common set of attributes as defined by the eduPerson attribute schema. Shibboleth is currently based upon SAML 1.1 but will support SAML 2.0 by early 2007.

5.2.2 Liberty Alliance

The Liberty Alliance Project [60] is a global identity consortium made up of almost 150 members, with the goal of developing federated identity, interoperable strong authentication and identity-enabled Web services. Liberty Alliance consists of two specifications, Liberty Federation and Liberty Web Services.

Liberty Federation [31], which consists of ID-FF 1.1, 1.2 and SAML 2.0 specifications, enables Internet users to authenticate and sign-on only once and then browse multiple Web sites in a federated network without having to repeat the authentication process.

Liberty Web Services [32], which consists of ID-WSF 1.0, 1.1 and 2.0, is an open framework for deploying and managing a variety of identity-based Web services.

5.2.3 WS –Federation

WS-Security (now known as Web Services Security (WSS)) [67] is a set of specifications originally developed by Microsoft, IBM, and VeriSign to provide security to Web Services. One of the specifications in WSS is WS-Federation. WS-Federation defines model for federating trust and identity related functions. Some of the features and functions in WS-Federation overlap with SAML, Liberty ID-FF, and Shibboleth. However, WS-Federation is currently only supported by Microsoft and IBM and without any existing implementations, and is therefore of less importance to any current needs.

5.3 Shibboleth

Shibboleth [50] is an architecture that supports federated single sign-on, which allows users to access Web based resources using a single login. It is based on SAML and was developed by the Internet2 middleware group. JISC [30] sees Shibboleth as the next generation access management solution for the UK Higher Education community and the successor to Athens [1]. To this end, JISC has invested almost £6.6 million in its Core Middleware Programmes.

Organisations need to be part of a federation before it can access resources using Shibboleth. A federation creates a *circle of trust* for organisations that want to access a set of resources. Each federation has its own set of policies that a prospective member organisation has to adhere to, and pre-defined levels of trust for access to resources.

EDINA's SDSS [49] is one such federation and it is a development federation for managing access to UK academic online resources. The UK Federation [62] was launched on November 30 2006, and is the predecessor to the SDSS Federation. Other federations include the Swiss SWITCH AAI [56], InQueue [28], and InCommon [27]. The next section describes a high level Shibboleth implementation.

5.4 Shibboleth architecture

Shibboleth forms part of an organisation's federated identity management system by providing access to protected resources. Shibboleth enables the exchange of authorisation and authentication information in the form of SAML assertions between organisations and service providers. Shibboleth does not have an authentication system of its own and as such, uses an organisation's authentication system and user information databases to send identity information to a service provider. The service provider then uses this identity information to make authorisation decisions.

This section describes how the Shibboleth architecture operates by passing user attributes securely between an organisation and a service provider.

The Shibboleth framework consists of three major software components:

Identity Providers (IdP)

- Allows authentication and Single Sign On (SSO) within the institution and federation.
- Holds the user's attributes and defines how these attributes should be released in the form of the Attribute Release Policy (ARP).

Service Providers (SP)

- Hosts the protected resource and determines whether a user is allowed to access the resource depending on the attribute information it received from the IdP.
- The SP is also responsible for defining the set of attributes required to access a particular resource and publicising it to the other members of the federation.

Federation

- This is a set of trusted IdPs and SPs, which agree to work together within a given set of policies and legal agreements.
- Provides the Where Are You From (WAYF) service. The WAYF holds a list of participating organisations and is made available when users attempt to access a protected resource within the federation.

A high level Shibboleth implementation is shown in Figure 3. A circle of trust has to exist within the federation whereby Service Providers (SP) and Identity Providers (IdP) trust each other. A typical scenario of a user accessing a protected Shibboleth resource is shown below:

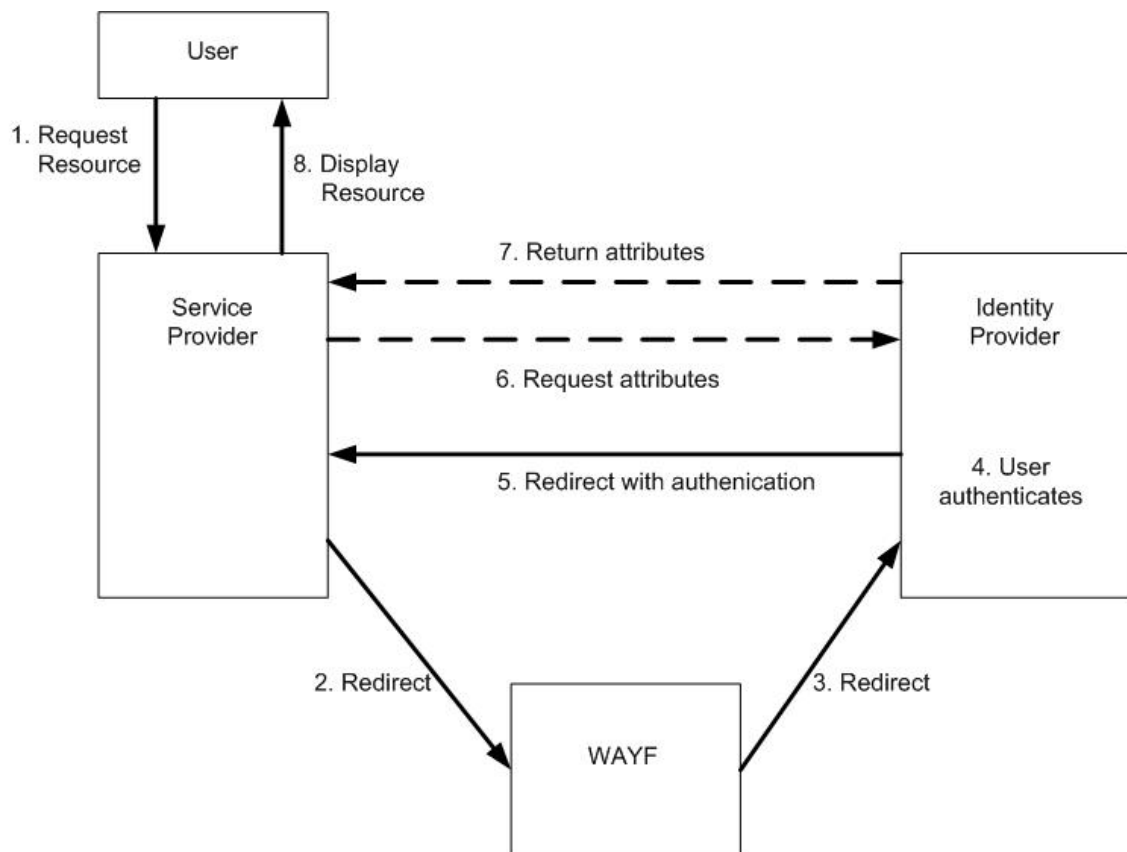


Figure 3 Shibboleth authentication and authorisation process.

1. A user attempts to access a protected resource on a SP using a standard browser.
2. Since the SP does not know about the user, the user is redirected (by HTTP 302) to the Federation's Where Are You From (WAYF) service.
3. At the WAYF, the user is asked to select his home institution (IdP) from the list. The user might be able to record this preference if the Federation's policies allow it.
4. The user is redirected to his IdP and asked to authenticate himself if he has not already done so. As part of the redirection, the target SP is conveyed to the IdP.
5. The IdP generates a SAML handle (an opaque handle associated with the user's identity) and redirects the user back to the SP with the handle.
6. The SP uses this handle to request attribute information from the IdP about the user.
7. The IdP allows or denies the attribute information (like role, email address, affiliation) to be send to the SP depending on the Attribute Release Policy (ARP).
8. Based on the attribute information, the SP then either allows or denies the user access to the protected resource.

6. Shibboleth and Grid Computing

The continuing success of the UK e-Science program, and Grid computing in general, is highly dependent on the availability and the ease that current and new users can access the underlying infrastructure. To encourage more users from non-traditional Grid user domains such as Arts and Humanities (A&H) to integrate their applications with the Grid, users should be able to easily run their Grid-based jobs or access large-scale resources such as the UK National Grid Service (NGS).

The current process of submitting a job to the NGS is long and laborious. Firstly, the user has to get a UK e-Science X.509 Certificate issued by the UK e-Science Certificate Authority (CA), which is based at Rutherford Appleton Laboratory (RAL) in Oxfordshire, UK. This CA issues certificates that bind a user's identity to the public key contained in the issued X509 certificate. Presently, the user has to initiate a certificate request from the CA web interface. This certificate request is then sent to a local Regional (or Registration) Authority (RA), which is recognized within the user's Grid community. The RA could be located in a remote institution if the user's local institution is not a registered RA. The user has to personally meet the RA and provide a form of standard identification (driving licence, staff card, passport) to prove their identity. Once the RA establishes the identity of the user, the certificate request is then approved, and the CA is given the go ahead by the RA to issue the certificate.

This process may take several days or even weeks if there is no nearby RA. This process is further complicated when nationally recognized digital certificates are involved because the user has to adhere to various protocols and best practices to ensure the integrity of the CA. For example, measures such as strong password encryption of the private key, appropriate backup of issued certificates, and the conversion of certificates to appropriate formats have to be undertaken. The process of obtaining a certificate, and then familiarising themselves with PKI technology is very likely to dissuade less technical researchers from engaging with Grid technology.

Scientists require technology that will simplify and quicken their daily research. However, initial user experiences of the Grid, especially for non-technical users, make it more complex to access Grid resources. This learning curve when it comes to certificates and understanding PKI is forcing researchers to abandon the Grid in favour of applications they are more familiar with. This is especially true of A&H researchers who are probably not as technically proficient as their engineering colleagues. The extra level of complexity that comes with obtaining and administering digital certificates will in most circumstances force A&H researchers to continue working with applications that they are more familiar with.

However, this situation can be remedied by using existing security technologies that can provide authentication and authorisation services to end users in a form familiar to them, e.g. based on their local institutional usernames and passwords. Shibboleth as an architecture that supports devolved authentication is one such solution. Devolved

authentication allows the authentication of the user to be done at the institution best placed to perform identity checks, the user's local institution. Identity management performed centrally will not scale when the number of Grid users increase because current security policies are aimed at maintaining an active and trusted relationship between a Service Provider and an individual user. As such, devolved authentication is required to make the Grid scale. Although devolved authentication can be done with certificates and PKI, it is more easily supported using Shibboleth. Shibboleth not only helps with Grid scalability, it also provides the end user with a security mechanism that they are familiar with. Users are shielded from the complexities of PKI and they can access Grid services securely in a form they are familiar with, a username/password pair.

The next chapter describes various Shibboleth based projects that aim to solve the scalability and other issues commonly faced when deploying Grid applications to a wider range of users.

7. Relevant Shibboleth based Grid Projects

7.1 Introduction

This chapter describes several Shibboleth based Grid projects that look at leveraging Shibboleth's cross-organisation identity federation and attribute management with Grid based applications.

7.2 GridShib

GridShib enables interoperability between Globus and Shibboleth, and is a collaboration between NCSA (National Center for Supercomputing Applications), University of Chicago and Argonne National Laboratory [2]. It was a one-year project that ran from 1/12/2004 – 6/12/2005. It consists of separate plug-ins for the Globus Toolkit (GT) and Shibboleth. Once these plug-ins are properly configured, a GT Grid service provider is able to securely request attributes about a user from a Shibboleth IdP (the Attribute Authority (AA) of the IdP to be specific) through the Shibboleth protocol. The next two sections describe these plug-ins in greater detail.

7.2.1 GridShib for Globus Toolkit

GridShib for Globus Toolkit is a plug-in for Globus Toolkit 4.0. It gets SAML attribute assertions about a requesting Grid user from a Shibboleth AA and makes the appropriate access control decision based on these attributes. The user's attributes are collected by the Policy Information Point (PIP) in the form of SAML assertions before the assertions are parsed, extracted and delivered to the configured Policy Decision Point (PDP). The PDP makes an authorization decision based on the user's attributes. There is a clear separation between the PDP and PIP and this enables the plug in to be used with external decision or authorization making tools. This flexible feature of the plug-in is exploited by the GridShibPermis project.

7.2.2 GridShib for Shibboleth

GridShib for Shibboleth is a name mapping plug-in for a Shibboleth 1.3 IdP. Its primary purpose is to process attribute queries from Grid SPs based on the user's X.509 Subject distinguished name (DN). The main challenge in integrating the Globus Toolkit with Shibboleth is that different IdPs identify the same user in various formats and schemas. The other challenge is that GT4 identifies users by X.500 distinguished names (DNs) while Shibboleth identifies users by opaque handles. The GridShib for Shibboleth plug-in overcomes this incompatibility by enabling the Shibboleth AA to map the user's DN to a local Shibboleth handle. Upon receiving an attribute query, the Shibboleth attribute authority uses this plug-in to map the DN and utilizes the resulting principal name to resolve attributes. When a Grid SP issues an attribute query, the Shibboleth AA uses this plug-in to map the DN and the resulting principal name is used to resolve attributes. Currently, mapping entries are read from an ordinary text file and is similar to the

gridmapfile used by the Globus Toolkit. However, this file-based name mapping does not scale and causes GridShib software management issues. The GridShib project plans to resolve this in future releases by using a database implementation.

7.2.3 GridShib Profile

The GridShib Profile is an extension of the Shibboleth Attribute Exchange Profile [5] with the primary difference being the usage of X.500 distinguished names (DNs) to identify principals instead of opaque handles. The primary user case in GridShib assumes that the Grid Client already has a valid PKI certificate. This valid PKI certificate will contain the name of the institution (the Identity Provider or IdP) in the form of X.500 DN. In most Grid-based scenarios, this PKI certificate is used to generate a short-term proxy certificate. This proxy certificate is forwarded to the Grid SP, authenticated and the IdP information is extracted. The Grid SP subsequently forwards the IdP information to the AA, and receives a SAML assertion containing the user's attributes. This typical GridShib user case consists of four steps as depicted in Figure 4, and is described in greater detail below:

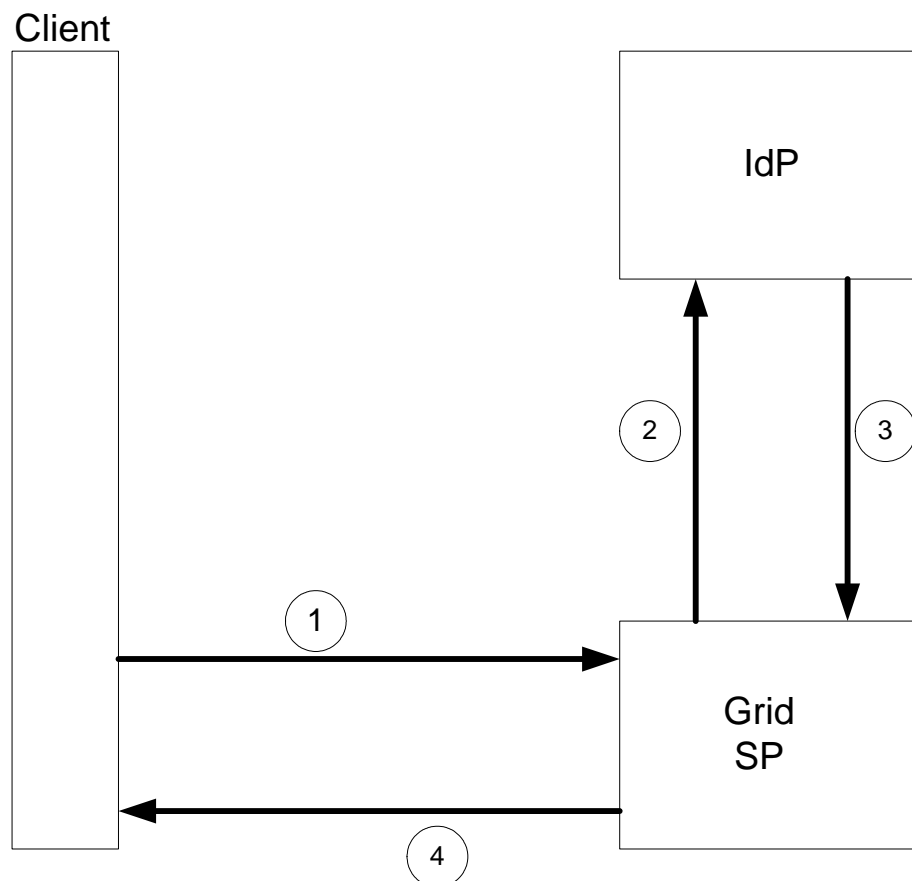


Figure 4 GridShib Protocol Flow [53]

Step 1 initiates the grid request/response cycle. The Grid Client authenticates to the Grid SP using their PKI certificate. After authentication, the Grid SP extracts the Client's DN from the PKI certificate.

In step 2, the Grid SP constructs a SAML attribute query containing the NameIdentifier element, which is the DN extracted from the client's certificate in the previous step. The Grid SP uses its X.509 credential to authenticate to the AA.

In step 3, AA component of the IdP authenticates the attribute request, maps the DN to a local principal name with the GridShib plug-in, and retrieves the attributes after being filtered by the Shibboleth ARP. These attributes are then send to the Grid SP in the form of a SAML attribute assertion.

Lastly, the Grid SP parses and extracts the attribute assertion before caching the attributes. An access control decision is made based upon these attributes and the client request is processed provided access is granted before returning a response to the Grid Client.

7.2.4 Conclusions

The primary aim of GridShib was to utilize the attribute management infrastructure of Shibboleth by transferring Shibboleth attributes in the form of SAML attribute assertions between the Shibboleth IdP and any Globus Toolkit based PDP.

However, one of the drawbacks to the GridShib approach is that the end user still requires a valid PKI certificate from a CA. Obtaining and the administration of PKI certificates is not a trivial task and this may deter non-technical users from using Grid services. Replacing the PKI certificate or lessen the usage of it are the primary goals for projects like ShibGrid, SHEBANGS, and BRIDGES aim to take away the complexity of using PKI certificates by integrating Shibboleth's decentralised authorization infrastructure with Grid services. Taking away the complexity of using PKI certificates by integrating Shibboleth's decentralised authorization infrastructure with Grid services is the primary goal of projects like ShibGrid, SHEBANGS, and BRIDGES, and will their approach will be explained in greater detail in the coming sections.

Another drawback is that GridShib is unable to make access control decisions based on a combination of attributes. For example, a researcher who is also an IEEE fellow cannot combine his departmental role and IEEE role to access a Grid resource. Although it is possible to overcome this limitation by having a separate PKI certificate for each role, this increases the complexity of setting up trust relationships with the AA, and also the level of detail when formulating Attribute Release Policy (ARP) files. The GridShibPermis project is an extension of GridShib and it attempts to provide greater authorization control to dynamically changing conditions or when multiple attribute authorities are in the framework by integrating PERMIS with the GridShib PDP.

7.3 GridShibPermis

The GridShibPermis project based at the University of Kent, integrates the identity federation and attribute assignment functions of Shibboleth with the policy-based enforcement function of PERMIS, in order to provide a flexible fine-grained authorization system for Grid jobs running under Globus Toolkit [6]. This is achieved by using the GridShibPermis Context Handler, which acts as a bridge between GridShib and PERMIS.

GridShibPermis contains a Context Handler that interfaces to PERMIS, and functions as a PDP in the Globus Toolkit authorization framework. As mentioned in Section 3.6, custom PIPs and PDPs can be included directly in the Globus Toolkit GSI authorization framework through callouts to external authorization services. This means that third party services, like the GridShibPermis Context Handler, can be incorporated into the authorization chain. This authorization chain can be configured at resource, service or container level. Once the request has been authenticated, the authorization chain is invoked in the order that each PIP and PDP is specified in the chain. In this authorization chain, the GridShib PIP is seen as a PIP while the GridShibPermis Context Handler is seen as a PDP.

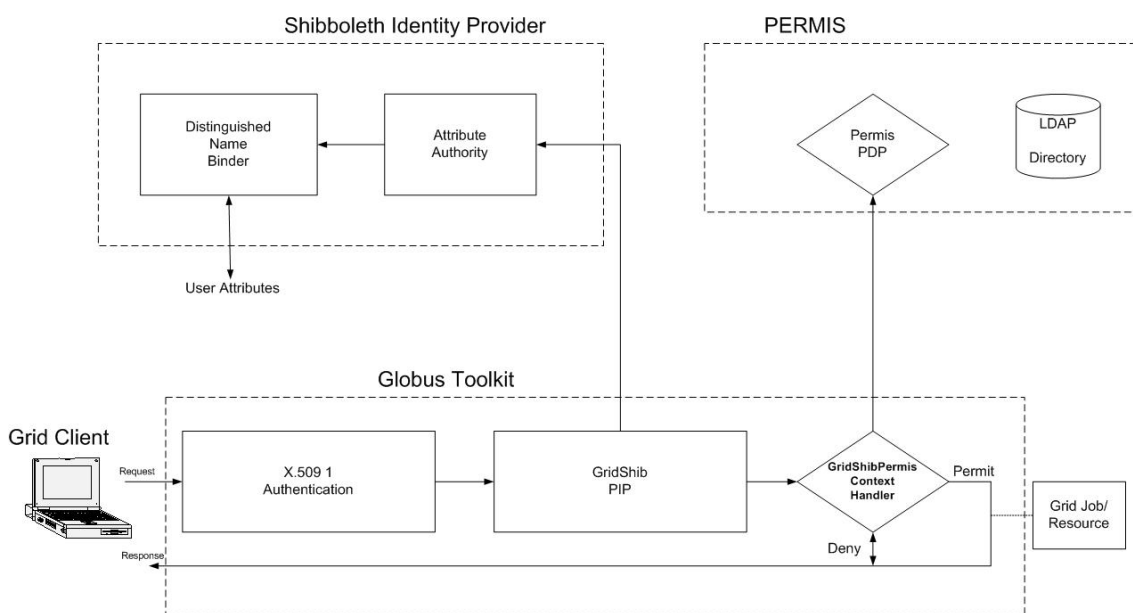


Figure 5 GridShibPERMIS Integration Scheme.

As shown in Figure 5, the GridShibPermis Context Handler, which serves as a PDP, is invoked after the GridShib PIP. The attributes obtained from the GridShib PIP is forwarded to the Context Handler, which then converts them to a Java format that is readable by PERMIS. These attributes are then forwarded to the PERMIS CVS, which in turn ensures that the IdP is trusted to release those attributes in the first place. These attributes are then send along with information about the user's request and targeted

resource to the PERMIS PDP. The PERMIS PDP makes an access control decision based on the configured PERMIS policy, and the Context Handler forwards this decision to the GT4 PEP. This entire authorization chain is based upon a deny-override mechanism [18], which means the chain processing stops immediately if any PDP in the chain returns a deny decision.

7.3.1 Conclusions

However, GridShibPermis still requires the end user to be a *poweruser*. A poweruser in Grid terminology is someone who is proficient with computers at a high level and especially proficient in the administration of PKI certificates. A GridShibPermis user/administrator probably has to be even more proficient than the normal Grid user because dealing with PERMIS role policies is fairly complicated.

In summary, the GridShibPermis project uses existing Shibboleth and PERMIS infrastructures to provide policy driven role-based privilege management for Grid applications although the complexity of administering PKI certificates still remain.

7.4 ESP-Grid

The ESP-GRID [16] is a JISC project, which ran from July 2004-June 2006, with the primary aim of exploring how Shibboleth offers solutions to issues of grid authentication, authorisation and security. It also explored the usage of PKI in UK e-Science Grid projects and Grid projects in general and whether it is appropriate although it has already been implemented. Furthermore, it also looked at how the access management regime between the e-Science Grid and the JISC IE interoperate.

The ESP-Grid is a comprehensive project that hopes to understand the relevance of Shibboleth to the Grid and PKI. ESP-Grid is a pioneering project and has collaborations with various other projects and is also the basis of numerous other JISC Shibboleth based projects. As such, a complete review of this ESP-Grid is beyond the scope of this document. However, this document will detail the main findings and the conclusions gathered from this project. The three primary objectives and findings of ESP-Grid are detailed below:

7.4.1 Relevance of Shibboleth and PKI in a Grid infrastructure

ESP-Grid argues strongly for the case of devolved ID management and identity assertion, which can be ably supported by Shibboleth. PKI could support devolved ID management but it generally does not. Shibboleth does this by devolving authentication and most of the management of authorisation attributes to the home institution. This would benefit a Grid user since he only has to familiarize himself with one single-sign on interface or portal, and it places the responsibility of managing user attributes and identities with the home organization, which in most cases is the most appropriate organization. Devolved ID management with Shibboleth is valid for both security and scalability reasons as long as Federation security policies are followed by the home organization.

ESP-Grid also argues that current grid policies will not scale since the policies advocate one-on-one relationships between a Grid service provider and each user. A scalable Grid solution would require devolved authentication and authorization and this is more easily achievable via Shibboleth.

ESP-Grid also postulates that the Customer-Service Provider (C-SP) model needs to be initiated for the Grid to scale [16]. In a C-SP model, the Service Provider is responsible for authenticating and authorising users. The authentication point can be at the Service Provider portal or be devolved using Shibboleth. In this model, the Service Provider runs jobs on the Grid on behalf of the user and the user is generally protected from having to deal with PKI certificates.

ESP-Grid identifies Shibboleth as a possible solution to the issues of scalability when it comes to managing authentication and authorisation information. Shibboleth is especially beneficial to users who are not Grid-savvy and they do not require deep technical knowledge in PKI to access a Grid resource.

7.4.2 Future users of the Grid

Among the core findings from ESP-Grid was Shibboleth is more applicable to less Grid –savvy users in certain situations while client based PKI is more applicable to more technical proficient users in other situations [16]. From this study, a summary of future Grid users was concluded and is given below:

Type of User	Typical characteristic	Main Role
SEUD	Service End-User (data). Little or no computing expertise.	User of applications served by SPs. Uploads data or runs queries.
SEUX	Service End User (executables). Some understanding of code creation.	As SEUD, but runs either executable code or scripts via SPs.
PUA	Power User Agnostic of grid resource node. High degree of computing expertise.	Develops programs and data but does not care where processing takes place.
PUS	Power User requiring Specific grid resource nodes. High degree of computing expertise.	As PUA but may have more platform etc. dependent expertise and some sysadmin expertise.
PUDS	Power user Developing a Service. High degree of computing expertise.	As PUA/PUS but developing expertise like SP.
SP	Service Provider. High degree of	As PUA/PUS but has expertise in authorisation

	computing expertise.	And possibly identity management.
Grid-Sys	Infrastructure sys-admin. High degree of computing expertise.	System administration of grid nodes, possibly with infrastructure delivery and security expertise.

Table 1 Grid Users of the future [37].

ESP-Grid also concluded that most users would require simple and secure applications to access Grid resources. They also concluded that many of the future grid users are likely to access Grid services via portals that run jobs on behalf of the user. The portals can be used by majority of users and shields them from the complexities of PKI technology. The portal controls the users access to the Grid and devolved authentication and authorisation can be enabled by Shibboleth. Although current Grid users are more likely to use client based PKI, this is likely to change in the future with Shibboleth enabled portals.

7.4.3 Conclusions

Among the outcomes from the ESP-Grid included that PKI is good at establishing identities but poor at providing authorisation and role privileges to users. They also questioned the need to assert the user's identity in every grid transaction. ESP-Grid believes that presenting and logging user's identity *up front* only gives an illusion of security because the security credentials being presented could be old or invalid. The more important requirement is the swift suspension or revocation of rights when an improper transaction is initiated.

ESP-Grid concludes that Shibboleth is an ideal way of devolving authentication and authorisation to the appropriate people to manage identities and user attributes. They also believe that the C-SP model is necessary to allow new users to take up Grid technology and Shibboleth is best placed to achieve this.

ESP-Grid maintains that client based PKI is most applicable to Grid Power Users (the current majority of users) but that Shibboleth with devolved authentication and authorisation will benefit the majority of future Grid users.

7.5 ShibGrid

ShibGrid [52] and SHEBANGS [47] are projects funded by JISC in 2006 as prototype systems that will allow National Grid Service (NGS) users to securely access the NGS portal via Shibboleth. SHEBANGS will be explained in greater detail in the next section and a comparison between the two projects will also be given. ShibGrid is a one-year project and will run from 1/2/2006 – 31/1/2007.

The primary aim of ShibGrid is to develop a prototype system that allows NGS users with or without UK e-Science certificates to access NGS resources securely through the

integration of Shibboleth, Globus's GSI and MyProxy. It aims to provide a Shibboleth based authentication infrastructure to Grid users to lessen the complexity of using Grid tools and technology. Users can access the NGS portal with or without a valid UK e-Science certificate and both these scenarios are described below. Both these scenarios are adapted from [52].

Accessing the NGS portal: user with or without a UK eScience certificate

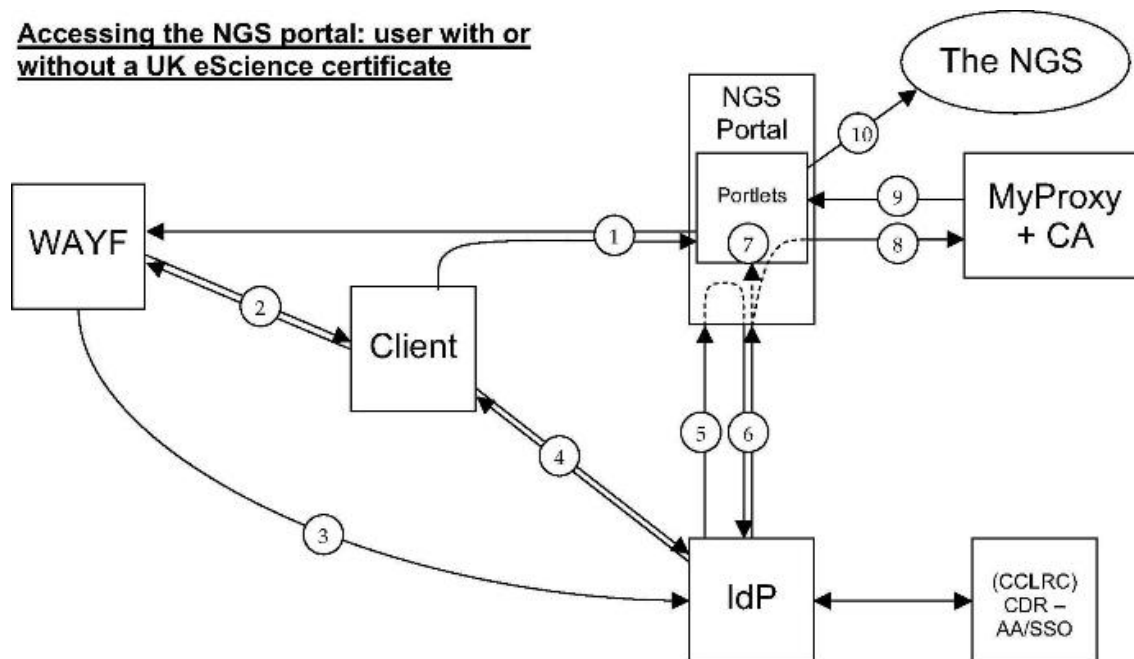


Figure 6 Scenario 1 - User with or without a UK eScience certificate [52].

Accessing the NGS portal: Users with or without a UK e-Science certificate

1. User requests access to the NGS portal (the SP) through a Shibboleth logon, the user's browser is redirected to the WAYF.
2. The user chooses the appropriate IdP in the form returned by the WAYF.
3. The user's browser is re-directed to the correct IdP.
4. The user is authenticated at the IdP end with Shibboleth AA.
5. The IdP redirects the user's browser back to the NGS portal (SP). The signed authentication SAML assertions are passed in this redirect, via use of AA.
6. The NGS portal calls out to the IdP's Attribute Authority for attributes about the user.
7. The user is authorised to use the NGS portal through these attributes.
8. In addition the signed attribute assertions are used as the password for access to the MyProxy server. These credentials are used as either the authorisation to release a proxy (if the user has an eScience certificate and has already uploaded a proxy) or as authentication to allow the auto-generation of a low assurance proxy, from the built in MyProxy-CA, if they have not uploaded a proxy.
9. The proxy credential is returned; optionally other attributes can be added to the proxy (through the use of certificate extensions). A future development would

- be to retrieve VO attributes from a VOMS server and write them to the proxy.
This can be used by gatekeepers to restrict access to resources.
10. The user can access the NGS.

Accessing the NGS portal: User with a UK e-Science certificate

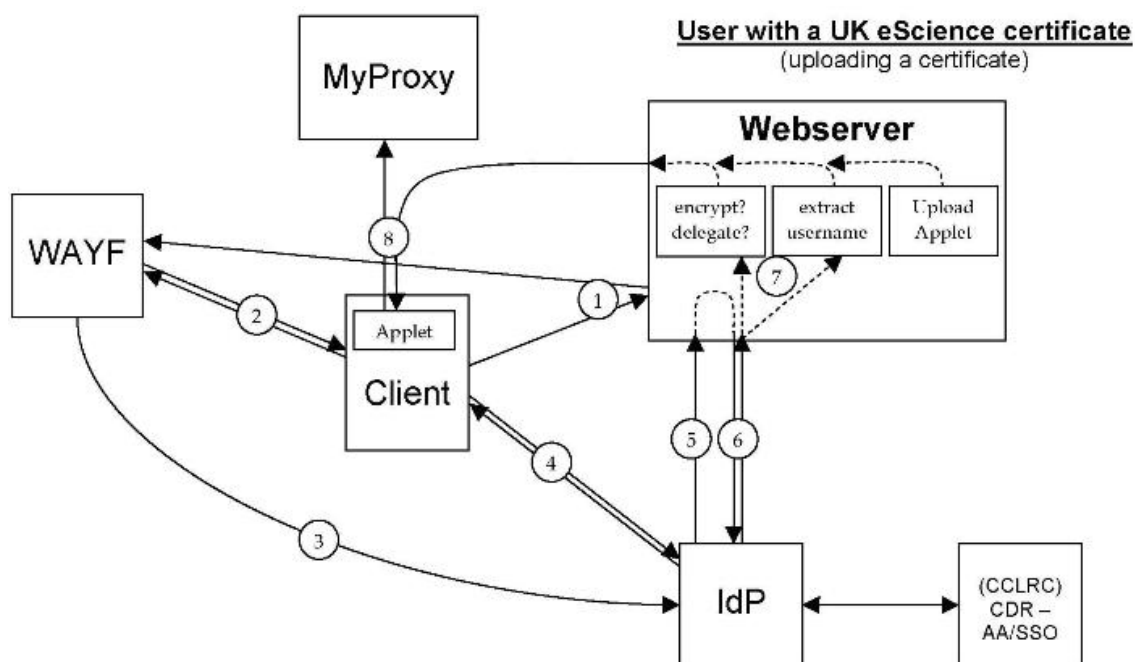


Figure 7. Scenario 2 - User with a UK eScience certificate (Uploading Certificate) [52].

1. User goes to the MyProxy upload page (SP), the user's browser is redirected to the WAYF.
2. The user chooses the appropriate IdP in the form returned by the WAYF.
3. The user's browser is re-directed to the correct IdP.
4. The user is authenticated by the IdP.
5. The IdP redirects the user's browser back to the MyProxy upload page (SP). The signed authentication SAML assertions are passed in this redirect.
6. The MyProxy upload page calls out to the IdP's Attribute Authority for attributes about the user.
7. The username is extracted and the attributes are encrypted and delegated to the user. These are then passed as the parameters of the upload tool itself, which must run on the user's machine to have access to their certificates.
8. The user unlocks and uploads their certificate using the applet. It is envisioned that they can obtain their certificate from a browser, 2*.pem files, one PKCS12 file or an already existing proxy. This tool will also allow them to destroy their proxy (if they want to start using generated proxies).

7.6 SHEBANGS

SHEBANGS [47], which runs from 21/11/2005 – 21/2/2007, is similarly funded by JISC to enable academic users to access NGS through Shibboleth, in particular, using the NGS portal. SHEBANGS aims to help researchers use existing Grid resources without having to deal with the complexities of using UK e-Science certificates.

SHEBANGS aims to serve the users who do not have pre-installed Grid middleware and can only access NGS services via existing portals after authenticating themselves at their home institution via Shibboleth. SHEBANGS uses the attributes fetched from Shibboleth to generate short-term credentials, which are then uploaded to a MyProxy server. SHEBANGS has a separate component called the Credential Translation Service (CTS), which manages the process of requesting user attributes, generating short lived certificates, and uploading these certificates to a MyProxy server.

Grid proficient users also benefit because they are able to pre-register their Grid credentials and access NGS through a portal. The figure below shows the movement of authentication data in the scenario where the user contacts the Credential Translation Service (CTS) first.

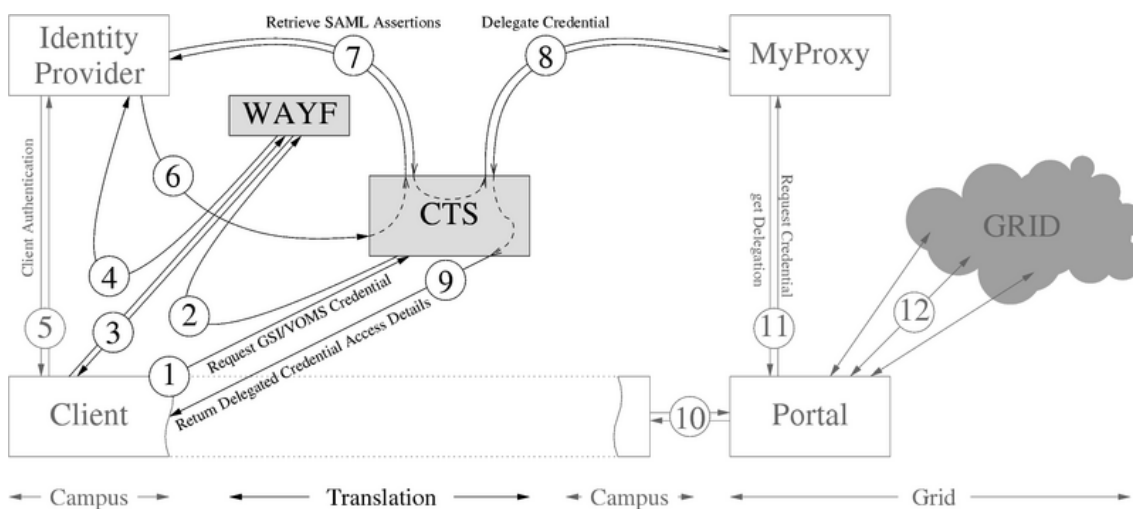


Figure 8 SHEBANGS authentication data flow [47]

Figure 8 shows the flow of authentication data when a user makes initial contact with the CTS. The various steps involved during this authentication are described below and is adapted from [47].

1. A user, perhaps referred by an HTML link from the portal, points their browser at the CTS.
2. The user's browser is redirected to a trusted Where Are You From (WAYF) service.
3. The WAYF server presents a form, which the client completes and posts back to the server.

4. The user's browser is now redirected to the appropriate institutions IdP.
5. Out of bands authentication takes place between the user and the IdP.
6. The IdP redirects the browser back to the CTS (passing Shibboleth artefacts in the URL).
7. The CTS uses the artefacts to obtain SAML Assertions from the IdP on a secure back channel. The CTS evaluates the SAML Assertions and issues a GSI Credential.
8. The CTS delegates this credential to a MyProxy Server.
9. The CTS returns a web page over HTTPS to the client, which contains a MyProxy username:password:server triplet.
10. The user logs into the portal using the MyProxy triplet, and is now able to use the Grid.
11. The portal obtains a GSI credential (out of bands).
12. The portal accesses the NGS (out of bands).

7.6.1 ShibGrid vs SHEBANGS

Table 2 details the differences between ShibGrid and SHEBANGS.

ShibGrid	SHEBANGS
The NGS portal acts as a Shibboleth SP, which means that the portal is Shibboleth aware. It is also the portal's responsibility to request user attributes and pass these attributes to the Shibbolized MyProxy server, which in turn will generate a short-termed certificate and delegate it to the portal.	The NGS portal is unaware of Shibboleth. A separate component called Credential Translation Service (CTS) is implemented, which acts as a Shibboleth Service Provider to request user attributes, generate short-lived certificates, and then pass these certificates to a standard MyProxy server.
The MyProxy server is Shibbolized by enabling it accept SAML assertions to authenticate a user, and then generating a short-term credential using its in-house CA.	The MyProxy server is a standard credential repository and is Shibboleth unaware. The CTS takes care of all the short-term certificate generation.
Due to architectural differences, SHEBANG users are required to access CTS first, before being redirected to their local institution (IdP) for authentication. If successful, users are given a username/password/Server URL triplet, which can then be manually typed into NGS portal Login form for Portal/NGS access.	The NGS portal will redirect users back to the IdP for authentication. If authentication is successful, users are able to login to the portal and access NGS resources.

SHEBANGS is trying to incorporate Virtual Organization Membership Service (VOMS) assertions created by CTS into GSI credentials as extensions for authorization purposes.	Currently, ShibGrid only focuses on authentication via Shibboleth and it is not concerned with authorization. It uses PERMIS to decide what privileges can be granted to an authorized user based on his DN.
---	--

Table 2 Differences between ShibGrid and SHEBANGS

7.7 BRIDGES

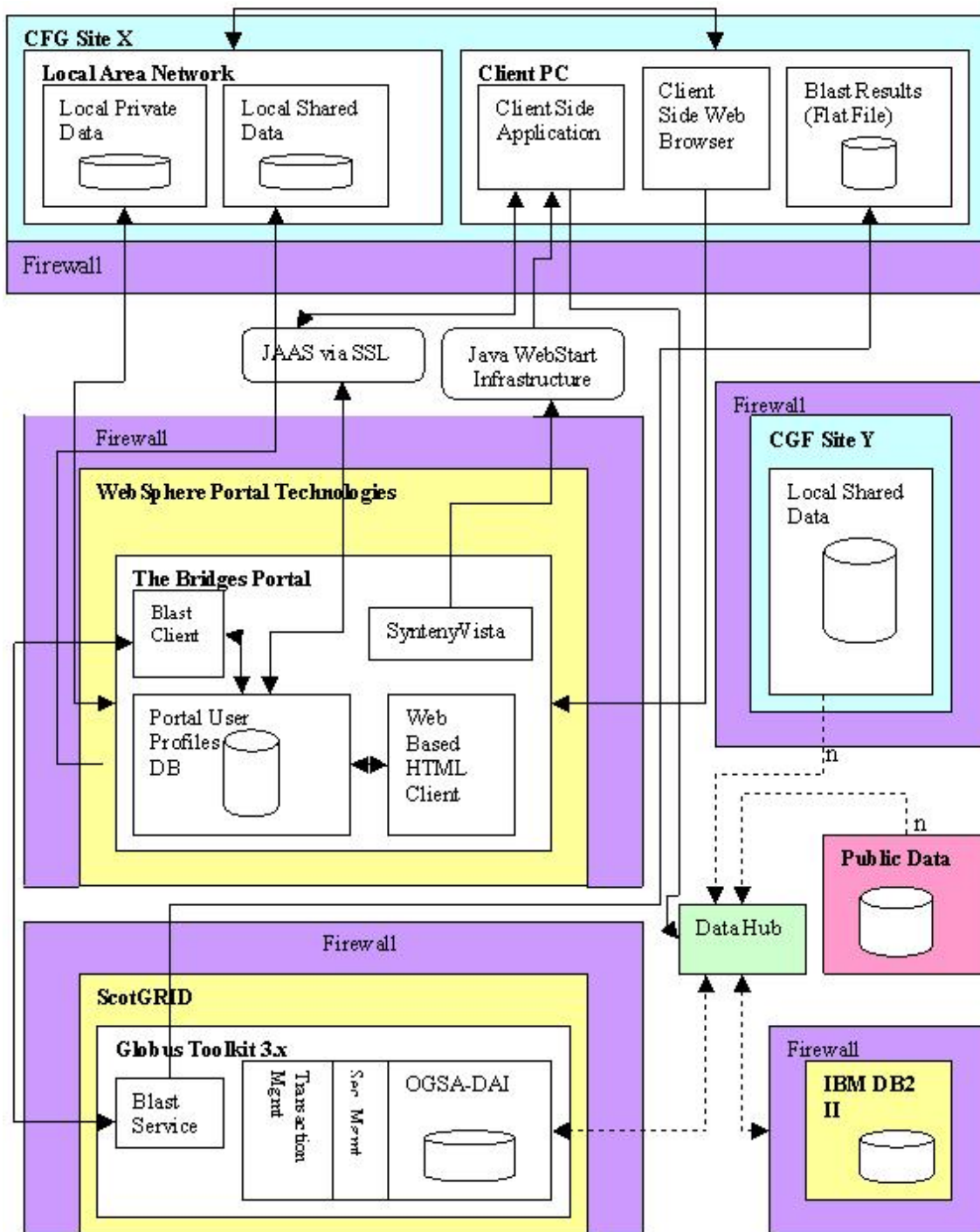


Figure 9 Overall BRIDGES architecture [40]

BRIDGES (Biomedical Research Informatics Delivered by Grid Enabled Services) [3] focuses on delivering a Grid infrastructure to Cardiovascular Functional Genomics (CFG) scientists. It is a collaborative effort between National e-Science Centre,

University of Glasgow, and IBM UK Life Sciences. It uses OGSA-DAI and IBM's Information Integrator [25] to deal with the federation of distributed biomedical data. The BRIDGES system architecture is shown above and is explained in greater detail in the following sections.

7.7.1 Portal Technology

Web portal technology was used to provide BRIDGES services as it can be personalized to the user's requirements and it also protects an inexperienced user from the complexities of Grid technology. IBM's WebSphere Portal Server was used to develop the BRIDGES portal due to its versatility and robustness.

7.7.2 Data Integration

The lack of programmatic access to live biological databases was among the major problems encountered in the BRIDGES project. This was overcome by implementing a data warehouse in DB2, which was populated by data fetched from public domain databases. For the purpose of evaluation, BRIDGES used two different technologies to access and integrate data from heterogeneous public data sources. IBM's Information Integrator is used in BRIDGES to access heterogeneous data sources by interacting with them using native wrappers for that specific data source. OGSA-DAI does the same by using grid services as a middleware layer between client applications and the data source. The current version of the public data federation tool supports both Information Integrator and OGSA-DAI. The GeneVista visualisation tool, which is integrated in the BRIDGES portal can be used to view and browse through data.

7.7.3 Security

BRIDGES portal users are authenticated by X.509 Distinguished Name (DN), which is embedded in their browsers. This X.509 certificate alongside PERMIS is used to make authorization decisions. This PERMIS based authorization system is used to limit the jobs the users can run and invoke depending on their roles.

BRIDGES has also integrated with Shibboleth so that user attributes, including the X.509 DN and user's various roles are securely requested from an AA of a Shibboleth IdP. These attributes are cached locally and subsequently used to determine authorization decisions by using PERMIS authorization infrastructure.

7.7.4 Conclusions

The BRIDGES project began in October 2003 and is engaged in the evaluation of a wide variety of Grid technologies applied to the life science domain

The BRIDGES project looked at the evaluation of a wide range of Grid technologies applied to the life science domain, and in particular to CFG researchers. The BRIDGES portal provides access to various Grid services to CFG researchers in a personalized and integrated manner.

Security is an important aspect of the portal and researchers have X.509 certificates imbedded in their browsers. In conjunction with PERMIS, this means that researchers using the portal are protected to a certain extent from complexities of obtaining and protecting their PKI certificates. However, this does mean that a researcher can only access the portal from a machine already embedded with an authenticated PKI certificate.

The BRIDGES project allowed biomedical researchers to access and integrate Grid resources with a familiar username/password mechanism. Researchers were able to upload nucleotide (or protein) sequences and match them against a vast number of local and remote genomic databases. Among the main findings from the BRIDGES project was that providing X.509 user certificates to BRIDGES researchers for authentication purposes were largely unsuccessful. Alternative solutions including using X.509 server certificates were adopted instead since researchers were more comfortable with username/password solutions. The BRIDGES also addressed issues such as re-engineering the client side UI to make them accessible and user friendly, e.g. to make them more Google-like. In short, researchers were more comfortable working in a familiar environment, which shielded them from the complexities of using PKI technology.

7.8 DYVOSE

DyVOSE (The Dynamic Virtual Organisations in e-Science Education) [14], which ran from 1/5/2004 – 30/4/2006, is a JISC funded Core Middleware Program project aimed at exploring the use of Grid technology in the education domain. The first part of the project focussed primarily at static Privilege Management Infrastructures (PMIs) while the latter part of the project focussed on dynamic PMIs.

7.8.1 Design and Implementation

In the first phase of DyVOSE, Advanced Computing MSc students at the University of Glasgow were asked to implement a Globus service, which searched and sorted a large text file (The Complete Works of Shakespeare) by submitting jobs to a local Condor pool. They were then asked to secure this Globus service by using PERMIS to restrict access based on the roles played by the students in the assignment.

In the second phase of DyVOSE, the focus was on implementing a dynamic delegation of authority in a VO where students were asked to implement a bioinformatics application, which initially accessed a remote database in Edinburgh containing nucleotide/protein sequences. The various roles and associations, which the Glasgow students needed to access the database, were configured dynamically using extensions to PERMIS.

In conjunction with ESP-GRID, Shibboleth was used to enable federation wide authentication and this identity is used to allocate user access based on the attributes fetched from a Shibboleth IdP. In this part of the project, a Globus service deployed to a

GridSphere portal is protected by Shibboleth. The user attributes from Shibboleth are then used by PERMIS to provide role-based access to the service.

7.8.2 Conclusions

DyVOSE has shown that Shibboleth is an effective authentication mechanism in a Grid infrastructure and it can be used with PERMIS to provide simple dynamic delegation between two sites in a VO.

7.9 VOTES

The VOTES project (Virtual Organisations for Trials and Epidemiological Studies) [64] is an MRC funded project exploring how Grid technologies can be used to support clinical trials and epidemiological studies. The main aim of VOTES is to establish a re-usable Grid framework that supports the three key stages of any clinical trial or observational study: (1) recruitment of potentially eligible participants, (2) data collection, and (3) study management. This framework will consist of adaptive Grid services that can be tailored to meet the needs of any particular clinical study.

Enforcing traditional Grid security requirements in the clinical and healthcare domain is a complex issue because it deals with more than just the traditional principles of Grid security: Authentication, Authorization and Accounting (AAA). Higher levels of privacy and integrity are required without compromising on the level of complexity of the framework. The framework needs to be secure and effective but yet easy to use by non-technical users like healthcare personnel. A term commonly used to describe a system with this particular set of requirements is Clinical Virtual Organization (CVO).

7.9.1 Architecture and Implementation

VOTES supports federated queries in a user oriented, but secure, environment, as depicted in Figure 10. This infrastructure is hosted on a trial test bed at the National e-Science Centre (NeSC) at the University of Glasgow. VOTES is implemented with a variety of Grid based technologies and these include GridSphere, Globus, and OGSA-DAI.

VOTES uses a GridSphere portal as a front-end and it communicates with the Globus Toolkit v4.0 Grid service. This Grid service in turn accesses an OGSA-DAI data service. The OGSA-DAI data service accepts queries from a driving database via standard SOAP messages while also accepting queries from the subsidiary databases, using direct JDBC connections

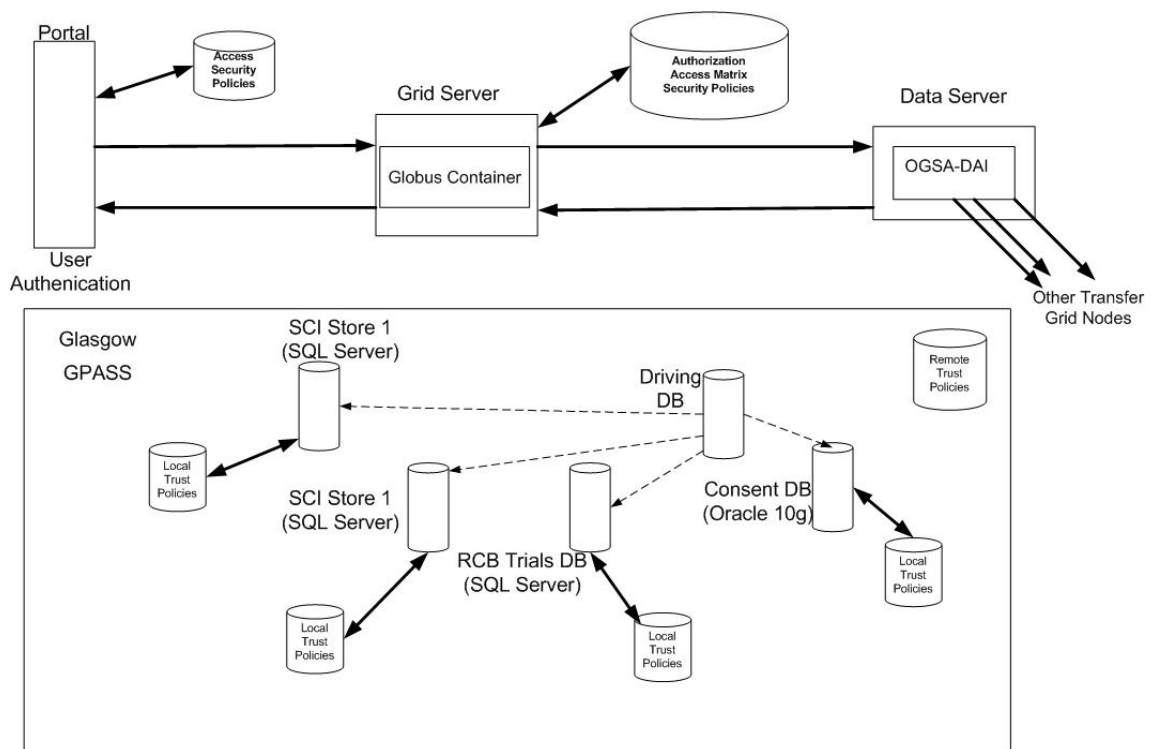


Figure 10 Glasgow CVO node architecture [54].

VOTES uses Shibboleth to associate an authenticated user identity along with the user's attributes to the role-based allocation of privileges within the VOTES portal. These attributes determine the level of access and privilege rights that the user has to a requested clinical data set.

7.9.2 Conclusions

VOTES aims to address the problems of implementing a Grid framework in a clinical and healthcare domain. The security issues faced are very specific to this domain and require a greater level of privacy and security. Shibboleth alongside role-based authorization was among the technologies used to extend greater level of authorization control to the clinical data sets in the VOTES portal. It should be noted that the Shibboleth model is inherently more static than the Grid vision of data and applications being found and invoked on the fly. However, this static model is ideal for the clinical domain since it is highly unlikely that new people, new data sets or new services are continually, dynamically added or removed from the clinical environment.

7.10 GLASS

The GLASS (GLASgow early adoption of Shibboleth) [20] project, which runs from 1/3/2006 – 28/2/2007, is investigating how Shibboleth technology can be adopted in the University of Glasgow medical and educational domains. GLASS will focus on how the integrated directory infrastructure for unified user account management currently being rolled out across the University of Glasgow can be utilized in a Shibboleth environment.

The project will also be exploring how Shibboleth can be integrated into Grid based scenarios to provide better security when accessing Grid resources and medical data sets in particular. GLASS is still in its infancy and further details about the project have yet to be delivered.

7.11 SPIE

SPIE (Shibboleth-aware Portals and Information Environments) [51] was funded by JISC and ran between 1/7/2004 – 30/0/2006. It was a collaborative effort between the University of Bristol and Eduserv (Athens).

The primary aim of SPIE was to showcase the effectiveness of Shibboleth in integrating institutional and national information environments by enhanced portals and portlets.

SPIE fulfilled this aim by addressing the following objectives [51]:

- Integrating Shibboleth within an institutional information environment, including both a portal environment and single sign-on authentication system.
- Deploying Shibboleth-related components and the PERMIS decision engine at Oxford University as a 'real life' testbed.
- Enabling institutional users to access remote resources seamlessly by integrating local and remote Shibboleth domains.
- Evaluating the usability of the Shibboleth approach from the perspective of end-users and resource managers.
- Disseminating the results of each project deliverable widely within the UK and beyond.

8. Shibboleth in e-Humanities Projects

This chapter describes how Shibboleth has been used in e-Science projects for humanities disciplines. Shibboleth was integrated into the Grid components of these projects to enable user authentication. Shibboleth is also fairly widespread among the Digital Library community and is seen as a common authentication and authorisation tool. The projects described below are still under development, and as such concrete implementation details are not available as yet. However, a high level overview of how Shibboleth is used in these projects is given instead. This chapter also briefly describes two Grid based proposals we have submitted.

8.1 Current e-Humanities projects

8.1.1 DAM-LR

The Distributed Access Management for Language Resources (DAM-LR) [13] project was initiated in 2005 under the 6th Framework Program of the European Commission (DG Research). Its main objective was to integrate the language resource archives (LRAs) of the partner institutions that form DAM-LR. These archives are virtualised in a data grid environment and appear to users as one large repository. DAM-LR established a formal federation made up of the partner institutions in 2006. The partners in the federation agreed on a number of policies pertaining to the federation, including a unified metadata domain and various legal issues.

8.1.1.1 Architecture

The DAM-LR architecture is a fully distributed and is based upon Grid, Digital Library, and Language Resource technologies. Each resource in the federation is identified by a unique resource called a URID, and a Handle System is used to process the URIDs.

User administration is managed locally with LDAP by each partner institution. The LDAP repository contains a commonly agreed upon set of user attributes and uses Shibboleth to securely exchange these attributes between a user's home institution and the institution with the protected resource. DAM-LR uses Shibboleth to simplify user authentication by supporting Single Sign-On. A local user can access resources freely at a partner site by authenticating at his local institution only once. Each partner site has a resource management tool that decides if a user is authorised to access a particular resource based on the user's attributes.

One of DAM-LR's objectives is to implement a federation wide unique identifier that can be used in authorization records in a similar fashion to the usage of user IDs in Apache webserver htaccess files [4]. They are currently developing new attribute based from a concatenation of an institute identifier and the user's ID. This feature is possible to a certain extent with the eduPersonTargetedID attribute. As described in the Shibboleth Glossary [48]:

Persistent Identifier (*eduPersonTargetedID*): *This special identifier type allows an IdP and SP to preserve a single identifier for one principal across all current and future transactions involving that principal without revealing or using that principal's identity. These identifiers are opaque to SP's and vary per SP, protecting against collaborative Denial of Privacy attacks, while preserving a 1:1 guaranteed mapping for liability and preference management purposes.*

However, as mentioned above, the value of the `eduPersonTargetedID` differs from SP to SP. This hinders the usage of using the `eduPersonTargetedID` attribute as a federation wide unique identifier.

8.1.2 TextGrid

TextGrid [57] is a three year project that was launched in February 2006, and is publicly funded by the German Federal Ministry of Education and Research (BMBF). It is a part of BMBF's continuing support for Grid infrastructure and the convergence of e-Science technologies in particular.

TextGrid aims to establish a generic, virtual workbench consisting of tools for scholarly text processing. This virtual workbench is based on Grid technologies and e-Science methodology. The software tools provide support for annotation, edition, text processing, and publication. TextGrid aims to integrate text collections from various partners seamlessly into TextGrid. Various Grid middleware data management tools are used to fulfil tasks like integrating heterogeneous storage architectures, data replication, or brokering of virtual files.

In a similar fashion to DAM-LR, TextGrid aims to exchange user attributes between partner institutions and these attributes are then used to decide if the user is allowed to access a protected resource.

8.2 Future Projects

This section describes two Grid based proposals we have submitted.

8.2.1 ASPiS

The proposed ASPiS (Architecture for a Shibboleth-Protected iRODS System) project will be a collaboration between the Arts and Humanities Data Service (AHDS) and the Council for the Central Laboratory of the Research Councils (CCLRC), with the aim of achieving this integration in the case of iRODS (Rule-Oriented Data Management System) [29], an open source data grid middleware system being developed at the San Diego Supercomputer Center (SDSC) as the successor to the widely used SRB (Storage Resource Broker).

The proposed project will focus on two complementary aspects of this integration:

- Controlling access to resources in an iRODS data grid.
- Creation of provenance metadata for resources in an iRODS data grid.

The iRODS is an open source project being developed by the SDSC as the successor to SRB, building on the experience of SRB but with significantly enhanced functionality. The iRODS system allows an administrator to create a set of rules defining how the data are to be managed and by which users. Each rule corresponds to a particular micro-service capable of executing the rule and the iRODS server is constructed only with the rules necessary for that system. iRODS, like SRB, provides a metadata catalogue holding information necessary to access the data and the iRODS system will have the capability to federate and create an iRODS-based data grid. At the moment, iRODS handles authentication by means of usernames and password, although it is currently being enhanced to handle X.509 certificates as well.

We plan to integrate Shibboleth with iRODS, and thus enabling the authentication of a user to be devolved onto the user's home institution. Middleware will be implemented to extract a user's Shibboleth attributes (which can include a unique user identifier and multiple roles) and make them available within iRODS, allowing finely grained access control to data resources. We will investigate how these attributes/roles can be used to make authorisation decisions within the iRODS context. One line of investigation will be to use the iRODS system's built-in rule execution architecture, whereby an attempt at accessing a resource results in the execution of a "micro-service", which could be used to determine access rights. We will also investigate the use PERMIS alongside Shibboleth, to make authorisation decisions.

Data provenance is a key issue in a dynamic Grid environment, where numerous users and virtual organisations can execute a wide variety of services and workflows that create and modify data. Provenance metadata may be regarded as representing the steps by a particular piece of data was derived, and it is fundamental when assessing the quality and accuracy of information, and also provides added value to grid users who subsequently publish, cite or further process the data. Provenance metadata typically incorporates a variety of elements, but an important component will be the identities or roles of the users that played a part the data's derivation, information that Shibboleth is well placed to contribute. In this part of the project, we will implement software components that allow iRODS to create and record provenance metadata for data when it is created or modified, including but not limited to the Shibboleth attributes.

Our design approach throughout will be to develop modular service-based components that encapsulate key functionality and conform to common interfaces. The project represents the community's first steps in integrating Shibboleth and provenance with the iRODS system, and we consider that our approach will greatly facilitate subsequent changes and extensions to the software, as well as allowing the future integration of different service implementations within the same framework. All software produced will be founded on detailed and validated use cases, user-centric iterative development, rigorous testing, and close liaison with actual and potential users throughout the project.

8.2.2 SARAH

The fundamental goal of the SARAH (Service-oriented Architecture for Research in the Arts and Humanities) will be to develop a prototype grid infrastructure for the A&H, to enable researchers to dynamically discover and retrieve resources relevant to their research across collections whose content and implementation are heterogeneous, which are subject to independent management regimes, and which are widely distributed geographically. These data sources must be seamlessly integrated, the heterogeneity indiscernible to the users. Discovery will function at deep levels of granularity, looking at content as well as metadata.

We see two aspects in particular to the development of the A&H Grid: the technical and the semantic. Previous investment by the e-Science community has provided a selection of mature Grid middleware with which to implement the technical infrastructure. Consequently, the key aspect of our proposal is the application of semantic web technologies to integrate diverse and heterogeneous resources. These technologies explicitly represent knowledge about collections and services in a flexible and extensible manner, complementing the interpretative nature of A&H research. Data-mining and other automated analysis tools will be integrated as modular services within a Grid architecture to facilitate the generation of machine-readable metadata. Following the service-oriented approach of the Grid framework, retrieved information will be delivered according to well-defined service interfaces that can be passed to automated services for further processing to support new research.

The project aims to demonstrate the validity of this approach, developing a working prototype that integrates grid-enhanced digital collections with discovery, data integration and text-mining services. The prototype will be extensible, enabling other collections and services to be incorporated, and could form the kernel of a future e-Humanities infrastructure. The prototype will greatly enhance researchers' access to digital resources, and will promote inter-disciplinary collaboration in virtual laboratories, both within the AHRC's domain and with researchers in other disciplines.

In the field of Grid technologies, SARAH will investigate:

- Grid middleware, e.g. GT4 and OMII. Given the project's interest in data services, a particular interest will be taken in middleware with a data integration emphasis, such as OGSA-DAI, OGSA-DQP and myGrid/Taverna.
- The experiences of other data grid implementations, e.g. the NERC DataGrid [36] and the European DataGrid [15] project.
- Semantic web technologies, including work on Data Webs [12], and ontologies such as CIDOC-CRM [7] for cultural heritage objects, PROTON [45] for knowledge management, and the history-specific VICODI [63] ontology.
- Semantic grid projects, such as S-OGSA, ComparaGrid [8] and OntoGrid [39].
- Projects integrating grids with digital libraries and textual studies, e.g. DILIGENT, SIMILE, TextGrid.

The proposed semantics-based discovery and browsing will rely on generating appropriate metadata or annotation. Although the infrastructure will allow the incorporation of diverse annotation services, we propose for definiteness to focus on a well-defined toolset. An initial survey of the state of the art has led us to GATE (General Architecture for Text Engineering) [19], a leading open source system for building large-scale semantic annotation software. GATE has an extensive and multidisciplinary user base, having been applied in both scientific and humanities research to disparate data, including text, image and audiovisual material. It has demonstrated its practical effectiveness in empirically evaluated trials, and conforms to international standards, e.g. it is the reference implementation for the ISO annotation standards group (ISO/TC37/SC4). Another advantage is GATE's proven compatibility with grid middleware like Taverna.

The advantage of the proposed approach for A&H research is that it will provide new ways of discovering and processing information, bringing together resources in unanticipated combinations. It will enable researchers to forge links that would not have been possible without SARAH's capabilities for deep discovery and integration of disparate information, thereby opening new areas for research.

9. Conclusions

This report has outlined how Shibboleth can simplify authentication and authorisation of Grid users through devolved authentication. Shibboleth enables a user to access a Grid resource securely in a form that is familiar to them, i.e. via their local institution's username/password pair. Devolved authentication in the form of Shibboleth also helps with Grid scalability because the current approach to Grid authorisation via Access Control Lists will not scale in a dynamic virtual organisation. This report also provides details about various projects that incorporate Shibboleth's attribute management functionality with Grid based applications. Most of these projects conducted research on how to access Grid resources without using PKI and digital certificates. It is hoped that the research from these projects will allow for more users from non-technical backgrounds to access Grid resources.

10. References

1. Athens Access Management System, <http://www.athens.ac.uk/>
2. Barton, T., Basney, J., Freeman, T., Scavo, T., Siebenlist, F., Welch, V., Ananthakrishnan, R. Baker, B., and Keahey, K. (2006). *Identity Federation and Attribute-based Authorization through the Globus Toolkit, Shibboleth, Gridshib, and MyProxy*. 5th Annual PKI R&D Workshop.
3. Biomedical Research Informatics Delivered by Grid Enabled Services (BRIDGES), <http://www.brc.dcs.gla.ac.uk/projects/bridges/>
4. Broeder, D., Veenendaal, R., Nathan, D., and Stromqvist, S. (2006). *A Grid of Language Resource Repositories*. Second IEEE International Conference on e-Science and Grid Computing (e-Science'06), 2006.
5. Cantor, S. et al., (2005). *Shibboleth Architecture: Protocols and Profiles*. Internet2-MACE. <http://shibboleth.internet2.edu/docs/internet2-mace-shibboleth-arch-protocols-latest.pdf>.
6. Chadwick, D.W., Novikov, A., Otenko, O. (2006). *GridShib and PERMIS Integration: Adding Policy-driven RBAC to Attribute-based Authorisation in Grids*. Presented at TERENA TNC 2006, Catania.
7. CIDOC-CRM, <http://cidoc.ics.forth.gr/>
8. ComparaGrid, <http://www.comparagrid.org/>
9. Condor, <http://www.cs.wisc.edu/condor/>
10. D. W. Chadwick, A. Otenko, E. Ball. (2003). *Role-based access control with X.509 attribute certificates*. IEEE Internet Computing, March-April 2003, pp. 62-69
11. D.W.Chadwick, A. Otenko. (2002). *RBAC Policies in XML for X.509 Based Privilege Management*. Appeared in Security in the Information Society: Visions and Perspectives: IFIP TC11 17th Int. Conf. On Information Security (SEC2002), May 7-9, 2002, Cairo, Egypt. Ed. by M. A. Ghonaimy, M. T. El-Hadidi, H.K.Aslan, Kluwer Academic Publishers, pp 39-53
12. Data Webs, <http://www.rin.ac.uk/data-webs>
13. Distributed Access Management for Language Resources (DAM-LR), <http://www.mpi.nl/DAM-LR/>
14. *Dynamic Virtual Organisations in e-Science Education project (DyVOSE)*, <http://www.nesc.ac.uk/hub/projects/dyvose>
15. European DataGrid, <http://eu-datagrid.web.cern.ch/eu-datagrid/>
16. Evaluation of Shibboleth and PKI for Grids (ESP-Grid), <http://wiki.oucs.ox.ac.uk/esp-grid>
17. Fellenstien, C., Joseph, J., and Ernst, M. (2004). *Evolution of grid computing architecture and grid adoption models*. IBM Systems Journal.
18. Freeman, T., Ananthakrishnan, R. (2005). *Authorization Processing for Globus Toolkit Java Web Services*. IBM Developer Works, <http://www-128.ibm.com/developerworks/grid/library/gr-gt4auth/>.
19. GATE, <http://gate.ac.uk>
20. GLASgow early adoption of Shibboleth (GLASS), <http://labserv.nesc.gla.ac.uk/projects/glass/>

21. Global Grid Forum (GGF), <http://www.ggf.org/>
22. Grid Account Management Architecture (GAMA), <http://grid-devel.sdsc.edu/gama>
23. GridSphere, <http://www.gridisphere.org>
24. Ian Foster and Carl Kesselman (eds), *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, July 1998. ISBN 1-55860-475-8.
25. IBM Information Integrator, <http://www3.ibm.com/solutions/lifesciences/solutions/InformationIntegrator.html>
26. IBM WebSphere, <http://www-306.ibm.com/software/websphere>
27. InCommon, <http://www.incommonfederation.org/>
28. InQueue, <http://inqueue.internet2.edu/>
29. iRODS, http://irods.sdsc.edu/index.php/Main_Page
30. Joint Information Systems Committee (JISC), <http://www.jisc.ac.uk/>
31. Liberty Federation, http://www.projectliberty.org/index.php/liberty/strategic_initiatives/federation
32. Liberty Web Services, http://www.projectliberty.org/index.php/liberty/strategic_initiatives/web_services
33. myGrid, <http://www.mygrid.org.uk/>
34. MyProxy, <http://grid.ncsa.uiuc.edu/myproxy/>
35. Nagaratnam, et. al.(2002). *The Security Architecture for Open Grid Services*.
36. NERC DataGrid, <http://ndg.badc.rl.ac.uk/>
37. Norman, M. (2006). *Types of grid users and the Customer-Service Provider relationship: a future picture of grid use*. Presented at the 2006 UK e-Science All Hands Meeting.
38. OGSA-DAI, <http://www.ogsadai.org.uk/>
39. OntoGrid, <http://www.ontogrid.net/ontogrid/index.jsp>
40. Open Grid Services Architecture (OGSA), <http://www.globus.org/ogsa/>
41. Open Middleware Infrastructure Institute UK (OMII-UK), <http://www.omii.ac.uk/>
42. Organization for the Advancement of Structured Information Standards (OASIS), <http://www.oasis-open.org/home/index.php>
43. Portal based User Registration Service (PURSE), (<http://www.gridcenter.org/solutions/purse>)
44. Privilege and Role Management Infrastructure Standards Validation (PERMIS), <http://sec.isi.salford.ac.uk/permis/>
45. PROTON, <http://proton.semanticweb.org/>
46. Security Assertion Markup Language (SAML), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security
47. Shibboleth Enabled Bridge to Access the National Grid Service (SHEBANGS), <http://www.mc.manchester.ac.uk/research/projects/shebangs/#motiv>

48. Shibboleth Glossary, <https://spaces.internet2.edu/display/SHIB/ShibbolethGlossary>
49. Shibboleth Support and Support Services, <http://sdss.ac.uk/>
50. Shibboleth, <http://shibboleth.internet2.edu/>
51. Shibboleth-aware Portals and Information Environments (SPIE), <http://www.oucs.ox.ac.uk/rts/spie/>
52. ShibGrid, <http://www.oesc.ox.ac.uk/activities/projects/index.xml?ID=ShibGrid>
53. Sinnott, R. et al., (2005). Security Infrastructure for Grid-Enabled Biomedical Services. Presented at UK e-Science All Hands Meeting 2005.
54. Stell, A., Sinnott, R., and Ajayi, O. (2006). *Secure, Reliable and Dynamic Access to Distributed Clinical Data*. To be presented at the Third International Life Sciences Grid Workshop, 13th-14th October, Yokohama, Japan.
55. Storage Resource Broker (SRB), <http://www.npaci.edu/DICE/SRB/>
56. SWITCH AAI, <http://www.switch.ch/aai/>
57. TextGrid, <http://www.textgrid.de/>
58. The Globus Alliance, <http://www.globus.org/>
59. The Globus Toolkit, <http://www.globus.org/toolkit/>
60. The Liberty Alliance Project, <http://www.projectliberty.org/>
61. Tuecke, S., Welch, V., Engert, D., Pearlman, L., Thompson, M. *Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile*, RFC3820, June 2004.
62. UK Federation, <http://www.ukfederation.org.uk/>
63. VICODI, <http://www.vicodi.org/>
64. Virtual Organisations for Trials and Epidemiological Studies (VOTES), <http://labserv.nesc.gla.ac.uk/projects/votes/index.html>
65. Web Services Interoperability (WS-I), <http://www.ws-i.org/>
66. Web Services Resource Framework (WSRF), <http://www.globus.org/wsrf/>
67. Web Services Security (WSS), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss
68. Welch, V., Foster, I., Kesselman, C., Mulmo, O., Pearlman, L., Tuecke, S., Gawor, J., Meder, S., and Siebenlist, F. (2004). *X.509 proxy certificates for dynamic delegation*. Proceedings of 3rd Annual PKI R&D Workshop.
69. Welch, V., Siebenlist, F., Chadwick, D., Meder, S., and Pearlman, L. (2004). *Use of SAML for OGSA Authorization*, Global Grid Forum.
70. Welch, V., Siebenlist, F., Foster, I., Bresnahan, J., Czajkowski, K., Gawor, J., Kesselman, C., Meder, S., Pearlman, L., and Tuecke, S. (2003). *Security for Grid Services*. 12th IEEE International Symposium on High Performance Distributed Computing.