CHAPTER 12 – Grid Security

Prof. Richard O. Sinnott
National e-Science Centre,
University of Glasgow

I. Introduction

Security is essential for inter-organizational collaborative e-Research. Without robust, reliable, easy to understand and manage e-Research security models and their implementations many communities and wider industry will simply not engage. To support inter-organizational, inter-disciplinary research it is essential e-Research security infrastructures support several key (defining) characteristics:

- ease of use for end users who should not have to learn complex new systems or adopt technological solutions that are not aligned with the fundamental reasons for engaging in e-Research - namely, to undertake research;
- single sign-on to distributed resources, i.e. once authenticated (and/or authorized) they are able to access and use a range of distributed resources without the need for further authentication;
- sites should be able to allow or deny access to their resources for given collaborators at their own discretion, i.e. they should be autonomous, and tools support should be available to help support this process;
- systems should scale to potentially support establishment and management of very large scale virtual organizations (VO) involving many collaborators from numerous institutions with different privileges;
- security infrastructures should incorporate or at least reflect existing (legacy) security infrastructures and policies of collaborators to ensure that e-Research specific security infrastructures do not violate/weaken existing policies on access and usage of resources;
- given the e-Research vision to support models where new collaborations can be rapidly supported, or where new institutions/users can be added, removed or have their privileges changed "on the fly" to existing collaborations, security infrastructures should be able to support such dynamic scenarios.

It is clear that no single security model or policy will suffice for all e-Research collaborations. Different domains have their own requirements on access to and use of resources by collaborators. However, it is essential that best practice in supporting collaborations is achieved. It is the case that currently, the predominant way in which security is currently addressed in the Grid community is through Public Key Infrastructures (PKI) [1] to support authentication. Whilst PKIs address some user identity issues, authentication in itself does not provide the fine-grained control over what users are allowed to do on remote resources (authorization). Furthermore, mainstream models of PKIs using centralized certificate authorities for authentication have their own associated problems. Instead what are required are finer grained security infrastructures that support e-Research and are aligned with the ways in which the e-Research community are comfortable with working. E-Research is – or should be! – about ***research*** and not learning about the nuances and/or complexities of different infrastructures, middleware or the associated mechanisms they require in place, e.g. X509 digital certificates [2].

In this chapter we present an overview of PKIs, their limitations and highlight how recent work in UK academia allows user-oriented security models that are aligned with access to internet resources more generally through the UK Access Management Federation based upon the Internet2 Shibboleth technologies [3]. We present a variety of case studies showing how the world of e-Research and non-Grid based access to and usage of secure internet resources can be aligned. Key to the success of this is shielding users from the underlying middleware, certificates, and features which are not directly supporting their primary reason for engaging in e-Research to begin with. Demonstrations of these

solutions across a variety of application domains is given to offer both a snapshot of state of the art security systems, as well as a vision for the integrated and secure Grid system of the future.

2. Authentication and Grid Systems

Fundamentally Grids are about sharing resources. With this in mind, it is essential that security is ensured, both of the underlying systems, and of the Grid infrastructures and applications running on top of them. This is especially the case as the Grid community moves from the academic, research-oriented background to more commercial arenas, and especially when one moves towards more security focused domains such as finance and e-Health. It is the case in computer security that the weakest link rule applies; this fact is magnified by Grid infrastructures due to their collaborative openness. Highly secure multi-million pound compute facilities can be compromised by inadequately secured remote laptops. Rigorous security procedures at one site can be made redundant through inadequate procedures at another collaborating site.

This problem is due in part to the lack of granularity in how security is currently considered. Grid security is still primarily based around Public Key Infrastructures (PKIs) which support validation of the identity of a given user requesting access to a given resource – so called *authentication*.

2.1 Public Key Infrastructures (PKI)

Cryptography is one of the main tools available to support secure infrastructures. Using cryptographic technology, confidentiality can be established by encrypting and decrypting messages and their contents. Encryption and decryption are done using keys. When these keys are the same, this is called symmetric-key cryptography.

Public-key cryptography uses different keys: private and public keys. Messages encrypted with a public key can only be read by an individual who possesses the private key. Any user can direct a message to a known destination, knowing that it can't be read by anyone else, simply by encrypting it using the public key of that destination. The owner of the private key can encrypt messages with that key, and the receiver of the message can be sure that it was sent by the owner of the private key. Both public key agreement and public key transport need to know who the remote public key belongs to, i.e. who has associated private key. The public key certificate is the mechanism used for connecting the public key to the user with the corresponding private key. Public key certificates include a Distinguished Name (DN) which can be used for identifying a given user.

A PKI is responsible for deciding policy, managing, and enforcing certificate validity checks. The central component of a PKI is a Certification Authority (CA). A CA is a root of trust which holders of public and private keys agree upon. CAs have numerous responsibilities including issuing of certificates, often requiring delegation to a local Registration Authority (RA) used to prove the identity of users requesting certificates. CAs are also required amongst other things to revoke older or compromised certificates through issuing Certificate Revocation Lists (CRL). A CA must have well documented processes and practices which must be followed to ensure identity management. Various PKI architectures are possible and the selection of which depends upon numerous factors. Whether numerous CAs are to be trusted? How important is it to be able to add new CAs? What kind of trust relationships exist between CAs?

The simplest PKI involves a single CA which is trusted by all users and service/resource providers. This model has been chosen for UK e-Science and is based on a statically defined centralised CA with direct single hierarchy to users. Getting a certificate can be seen as the starting point in accessing and using Grid-based resources such as the UK e-Science National Grid Service (NGS) [4]. The typical scenario for getting a certificate is as follows: researchers wishing to gain access to Grid resources such as the NGS apply to the centralised Certification Authority (CA) at Rutherford Appleton Laboratory (RAL) [5]. The CA will then contact their local Registration Authority (RA) who will in turn contact the user and request some form of photographic identification (such as a passport photo or university card) to

ensure that they are legitimate. Once the identity of the user has been ratified, the RA contacts the CA who subsequently informs the user (via email) that their certificate is available for download. The user downloads the certificate and associated certificate revocation lists into their internet browser. Once in their browser they are required to export it to forms appropriate to the Grid middleware.

The main benefit and reason for the widespread acceptance of PKIs within the Grid community is their support for single-sign on. Thus since all Grid sites in the UK trust the central CA at RAL, a user in possession of an X.509 certificate issued by RAL can send jobs or access resources more generally across all sites, or more precisely to all sites where a user has requested and been granted access to those sites. Typically with middleware solutions such as Globus [6], *gatekeepers* are used to ensure that signed Grid requests are valid, i.e. from known collaborators. When this is so, i.e. the DN of the requestor is in a locally stored and managed *grid-mapfile*, then the user is typically given access to the locally set up account as defined in the *grid-mapfile*.

2.2 Problems with PKIs

The above process is off-putting for many of the wider less-IT focused research community since it requires them to convert the certificate to appropriate formats understandable by Grid middleware using complex, cryptic openssl commands for example. Such requirements dissuade less IT-savvy researchers from engaging – especially as openssl is not commonly available on platforms such as Windows. It is possible for Windows-based PC users to install openssl-based solutions but this in turn requires them to install and configure additional software. In many cases, this is not possible, e.g. if they do not have sufficient privileges on their PC (root access etc) – a not uncommon practice in departments and faculties at many universities in the UK. In this case the researchers will instead have to refer to a local system administrator to help with the installation and configuration.

Assuming researchers have managed to obtain a certificate which they have converted into the appropriate format, they are then expected to remember strong passwords for their private keys with the recommendation to use upper and lower case and non-alphanumeric characters. The temptation to write down such passwords is apparent and an immediate and obvious potential security weakness. Problems also arise with researchers from institutions that do not have RAs in place.

In short, this whole process does not lend itself to the wider research community which the e-Science and Grid community needs to reach out to and engage with. It is a well known adage that the customer is always right. Usability and addressing researcher requirements is crucial to the uptake and success of Grid technology. End user scientists require software which simplifies their daily research and not make this more complex. Given the fact that the initial user experience of the Grid currently begins with application for UK e-Science certificates, this needs to be made as simple as possible, or potentially removed completely.

There are other issues with PKIs and Grid certificates as currently applied in the e-Research community. The fundamental issue with PKIs is trust. Sites trust their users, CAs and other sites. If the trust between any of these is broken, then the impact can be severe, especially since users are in principle free to compile and run arbitrary code. Thus with PKIs there is no mention of what the user is allowed to do once they have gained access to the resource. For example, users can in principle run arbitrary applications, starting a variety of local processes. In reality, a set of existing applications and infrastructure are often pre-deployed across the Grid nodes, hence the issue and risks of uploading executables is diminished. However, given the fact that compilers are commonly available on these resources, the possibility to compile arbitrary code and run executables spawning arbitrary processes exists. There is typically no security middleware enforcement on what processes can be started, by whom and in what context, other than the local enforcement given by the privilege associated with the local account. As the Grid community moves towards more security focused domains such as e-Health, such a model will never be supported. Instead practices and solutions which help make Grid infrastructures safer are required. Authorization base systems offer one approach to improve this security model.

3. Authorization and Grid Systems

Once a user has had their identity validated at a remote resource, it is essential that users actions are restricted based on who they are, what they are trying to do, and in what context etc. There are various methods of enforcing this restriction, the simplest method being the use of an Access Control List (ACL), which lists what users have access to a privilege. Essentially, a user presents their credentials at a gatekeeper to a resource, which consults a known list of users. This basic authorization structure extends the concept of authentication and no more. If the user cannot authenticate to the satisfaction of the gatekeeper then the resource request will be denied. The Globus GSI [GSI] software is an example of the classic ACL used to enforce authorization and provides a relatively coarse-grained approach to implementing security through the *grid mapfile* mapping of DNs to local user accounts.

A problem that arises when trying to apply this method to a dynamic Grid environment is that only one list exists, where there could be many privileges that require different ACLs. For example, a user might need access to a given resource for different purposes within a given VO. Having a single list with a predefined set of accounts and user DNs does not support this multi-role approach. This is a solution that would not scale well in a large VO. A more sophisticated method of applying authorization controls is through use of Role-Based Access Control (RBAC) mechanisms, which allow Privilege Management Infrastructures (PMI).

There are several RBAC middleware solutions available that support authorization that have been explored within the Grid community. We provide an overview of some of the more prominent of those here.

3.1 Globus Security Infrastructure (GSI)

The Globus toolkit [6] supports GSI-based authentication and authorization. This includes:

- WS Authentication with support for both message level and transport level security. Message level security is achieved through an implementation of the WS-Security standard that supports message protection at the Simple Object Access Protocol (SOAP) [9] message level. Transport level security is achieved through use of X.509 certificates to establish Transport Layer Security (TLS) connections.
- WS-Authorization through an authorization framework based upon the OASIS Security Assertion Markup Language (SAML) [10] authorization application programming interface. Through this SAML AuthZ API, a generic PEP can be achieved which can be associated with arbitrary services. Thus rather than developers having to explicitly engineer a PEP on a per application basis, the deployment information associated with the service is used. Authorization checks on users attempting to invoke “methods” associated with Globus services are then automatically raised and forwarded to the PDP which in the simplest case will respond with an allow/deny. However we note that in recent versions of the Globus infrastructure it is now possible to configure a chain of authorization mechanisms together. We note that one issue that has been encountered with the SAML AuthZ profile is the lack of granularity in how users might invoke actions. For example, different actions may or may not be allowed depending upon the data that they wish to access and potentially change. The SAML AuthZ profile does not currently allow actions to be distinguished based upon the parameters that might be associated with them. The Grid standards community is working on addressing this deficiency.
- Credential Management through MyProxy [11] is a credential storage and management system which has widespread acceptance as the way in which credentials should be managed within a Grid environment. Instead of users managing their own private keys and credentials, they can delegate them to a MyProxy repository. Through username and password access to MyProxy repositories, short lived proxy certificates can be created. MyProxy solutions are now being used in combination with portals for example, where users accessing a portal through a username and password will automatically have short lived proxy certificates created which can subsequently be used for Grid based job submission. Of all of the authorization infrastructures, GSI is

arguably the most straightforward to establish and use. Unsurprising since GSI has been developed as an integral part of the Globus development. That said and as noted, the ACL based approach offered by *grid-mapfile*s is a limited form of authorization however recent enhancements such as through the SAML authorization API now offer richer possibilities for finer-grained access control.

3.2 Community Authorization Service (CAS)

CAS [12] implements access control using a centralized authorization model. The main idea behind CAS is that a resource owner delegates allocation of authorization rights to a community administrator and lets the community administrator determine who can use this allocation. This is achieved by the administrator by having a CAS server, which acts as a trusted intermediary between VO users and resources. The CAS server decides whether a given user has sufficient privileges depending upon the community policy and if so, gives the user the right to perform the requested actions depending on the user's role in the community.

To achieve this, CAS keeps track of its community membership information. It also contains the access control policy statements which define policies along the lines of "who is allowed what type of access on which resources". To help manage this, CAS introduces the concept of rights in the form of a *capability*. Through possession of a particular capability (which is itself stored in a database associated with the CAS server), a user can show that they are allowed to access and use a particular resource.

To access a CAS-managed resource, a user has to first request a capability to use that resource. If this is the case, the CAS server responds with an appropriate capability. This capability corresponds to the intersection of the set of rights granted to the community by the resource provider and the set of rights defined by the capabilities granted to the user by the community. Following this, user presents the capability to the resource provider responsible for that resource. The resource provider verifies the rights for both the community and the capability to grant access to the user to the resource.

For finer grained access control and ensuring site autonomy, local resource providers can additionally apply their own local policies to determine the amount of access granted to users with presenting particular capabilities. This substantially reduces the work of resource administrators. The CAS architecture itself builds on the authentication and delegation mechanisms provided by the Globus GSI. In using CAS, a user will generate a proxy credential signed by his/her won user credential. The proxy credential is presented to the CAS sever, which returns a new credential, known as CAS proxy credential. This credential contains the CAS policy assertions to represent the user's capabilities and restrictions as an extension. SAML authorization decision statements are used to express the CAS policy assertions. The CAS proxy credential is presented to the resource provider. The resource provider then verifies the validity of the proxy credential and parses the CAS policy assertions to obtain the restrictions imposed by the CAS server. Thus, the CAS credential facilitates the mapping of the user to a local account, and the restrictions determine the operations the user is allowed to perform.

CAS provides scalability in terms of the number of users and VOs. Each user needs to be known and trusted by the CAS server (but not by each provider). Similarly, each resource provider needs to be known and trusted by the CAS server (but not by each user). However, the centralized model of a single CAS server as with many other distributed system examples leads to scalability and fault tolerance limitations. Many users requesting access to CAS will result in potential bottlenecks. Furthermore, the failure of the CAS server implies that no VO-wide resources enforcing access control based upon CAS capabilities will be available. This is further exacerbated since the VO administrator may need to maintain all VO-wide users' capabilities.

3.3 Virtual Organization Membership Service (VOMS)

VOMS [13] is a system for managing authorization data within VOs. VOMS has been developed as part of the European DataGrid project (http://edg-wp2.web.cern.ch/edg-wp2). VOMS provides a centralized database of user roles and capabilities and a set of tools for accessing and manipulating the database and using the database contents to generate Grid credentials for users when needed.

The centralized VOMS model requires all sites to agree upon the roles and privileges that are to be used throughout a particular VO. In this model, all sites agree in advance on the definition and names of the roles that are applicable to their particular VO, and the privileges that will be assigned to them. A single VO administrator is then appointed who will typically assign these roles to individuals on a case by case basis when users ask to be granted particular roles or permissions in the VO. The VO administrator may appoint other administrators to help him in this task, but all administrators are conceptually equal, in that each can in principle over-ride the decisions made by the others. The VOMS model has gained widespread acceptance due to the simple model for defining the roles specific to a particular VO and how they can be used/enforced. Sites themselves are responsible for configuring their resources to use these roles. With VOMS, this is implemented with tools such as the Local Centre Authorization Service (LCAS) and the Local Credential Mapping Service (LCMAPS) [14] which map the user role information into group identities (*gid*), user identities (*uid*) and associated local pool accounts established on the local cluster for that particular VO. Refinements can be made to this model in order to allow more local control over the use of resources, e.g. applying file store limits to a particular VO. We note that this local enforcement is not explicitly defined within the VO policy (given by the definition of the roles in the VOMS server). Rather, this is left up to local administrators to decide how the particular roles and privileges associated with that VO should be interpreted when accessing the resource.

VOMS offers tools that allow users to generate local proxy credentials based on the contents of the VOMS database and embed these within X.509 proxy credentials. This credential includes the basic authentication information that standard Grid proxy credentials contain, as well as role and capability information from the VOMS server. One of the benefits of VOMS is that Grid applications can use the credential without using the VOMS data. Alternatively, VOMS-aware applications can use the VOMS data to make both authentication and authorization decisions regarding user requests. Given the background and history to VOMS, the focus of authorization has primarily been at the level of mappings to local groups and accounts on clusters, but it is quite possible to use VOMS credentials to make finer grained access control decisions as we shall see in the case studies of section 5. One way in which such finer grained access control can be supported is through the Privilege and Role Management Infrastructure Standards Validation (PERMIS) technology [15,16].

3.4 Privilege and Role Management Infrastructure Standards Validation (PERMIS)

The PERMIS project (www.openpermis.org) was an EC project that built an authorization infrastructure to realize a scalable X.509 Attribute Certificate (AC) based privilege management infrastructure. Through PERMIS, an alternative and more scalable approach to centrally allocated X.509 public key certificates can be achieved through the issuance of locally allocated X.509 ACs.

The PERMIS software realises a RBAC authorization infrastructure. It offers standards-based APIs that allows developers of resource gateways (gatekeepers) to enquire if a particular access to a resource should be allowed. The PERMIS RBAC system uses XML-based policies defining rules, specifying which access control decisions are to be made for given VO resources. These rules include definitions of: subjects that can be assigned roles; Source of Authority (SoA), e.g. local managers trusted to assign roles to subjects; roles and their hierarchical relationships; what roles can be assigned to which subjects by which SOAs; target resources, and the actions that can be applied to them; which roles are allowed to perform which actions on which targets, and the conditions under which access can be granted to roles.

Roles are assigned to subjects by issuing them with X.509 Attribute Certificate(s). Once roles are assigned, and policies developed, they are digitally signed by a manager and stored in one or more LDAP repositories. The process to set up and use PERMIS can be split into two parts: *Administration* and *Use*. To set up and administer PERMIS requires the use of a LDAP server to store the attribute certificates and reference the SoA root certificate. A local CA is required to be set up which designates that the SoA and all user certificates created from this CA must have a DN that matches the structure of the LDAP server. The DN of the user certificate is used to identify the client making the call on the Grid service.

From the user's perspective, once the administrator has set up the infrastructure, the PERMIS service is relatively easy to use. Unique identifiers are placed as parameters to services when they are deployed.

These are the Object Identification (OID) number of the policy in the repository; the URI of the LDAP server where the policies are held and the SoA associated with the policy being implemented. Once these parameters are input and the service is deployed, the user creates a proxy certificate with the user certificate created by the local CA to perform strong authentication. The client is run and the authorization process allows or disallows the intended action.

The PERMIS infrastructure offers very fine grained authorization capabilities both in terms of policy expression and enforcement. The policy editing tools allow for easy development of the XML based policies. With support for the SAML Authorization API, PERMIS allows direct linkage between Grid services and authorization infrastructures.

PERMIS is perhaps the most advanced authorization infrastructure with software that meets the needs of the wider e-Research communities. It provides tools to support the definition and seamless of enforcement of authorization policy. Recent enhancements to PERMIS and associated Grid standards now allow PERMIS to work with a variety of Grid middleware and other authorization technologies including VOMS and XACML [17].

Such authorization technologies are essential for site administrators in providing secure access to their resources. However the purpose of such authorization technologies is not solely on protecting access to systems, but in allowing access to systems. That is, the end users must be able to access and use protected resources. Furthermore, the vast majority of researchers are unaware of X509 attribute certificates and their use in supporting privilege management infrastructures. Rather they are more focused upon research. Thus technologies are required which hide these technological solutions as much as possible from the end user and ideally are aligned with the way in which researchers themselves wish to access distributed and heterogeneous resources. The Internet2 Shibboleth technology [Shib1, Shib2] is currently being rolled out across UK academia and provides the opportunity for such hiding of authorization technology from end users whilst providing common way in which resources can be accessed and used more generally.

4. Shibboleth and Grid Security

With the Shibboleth model of resource access, sites are expected to trust remote security infrastructures for example in establishing the identity of users (authentication) and their associated privileges (authorization). To support this, the Shibboleth architecture and associated protocols identify several key components that should be supported including federations, Identity Providers, Service Providers and optionally Where Are You From (WAYF) services. Through these components, end users ave single usernames and passwords from their home institutions (which they are more familiar with than PKIs!) which will provide for seamless access to a range of resources at collaborating institutions and service providers. Local security policies at service provider sites can then be used to restrict (authorise) what resources authenticated users are allowed access to.

To support this federations are established which are used to agree and enforce common policies and technical standards to provide a common infrastructure for managing access to resources and services in a uniform way. Numerous international Shibboleth-based federations exist including: InCommon (http://www.incommonfederation.org), the federation formed by the Internet2 community in the United States, InQueue (http://inqueue.internet2.edu/) for sites wishing to test and explore the Shibboleth federated trust model, the SWITCHaai federation of the higher education system in Switzerland (http://www.switch.ch/aai/), the HAKA federation developed by the Finnish universities and polytechnics (http://www.csc.fi/suomi/funet/middleware/english/) with more in the pipeline such as the Meta Access Management System (MAMS) in Australia (https://mams.melcoe.mq.edu.au/zope/mams/kb/shibboleth/). In 2006, the UK established the UK Access Management Federation for Education and Research (http://www.ukfederation.org.uk). Through the UK Federation common access to a wide range of resources is now possible covering a wide spectrum of the research community: from the arts, social sciences, education, to the physical, engineering and life sciences.

To understand the impact of Shibboleth technologies on Grid security it is first necessary to have an

appreciation of the interactions that typically arise with Shibboleth. When a user attempts to access a Shibboleth protected service or Service Provider (SP) more generally, they are typically redirected to a WAYF server that asks the user to pick their home Identity Provider (IdP) from a list of known and trusted sites. The service provider site already has a pre-established trust relationship with each home site, and trusts the home site to authenticate its users properly.

After the user has picked their home site, their browser is redirected to their site's authentication server, e.g. an LDAP repository, and the user is invited to log in. After successful authentication, the home site redirects the user back to the SP and the message carries a digitally signed SAML authentication assertion message from the home site, asserting that the user has been successfully authenticated (or not!) by a particular means. The actual authentication mechanism used is specific to the IdP.

If the digital signature on the SAML authentication assertion is verified and the user has successfully authenticated themselves at their home site, then the SP has a trusted message providing it with a temporary pseudonym for the user (the handle), the location of the attribute authority at the IdP site and the service provider URL that the user was previously trying to access. The resource site then returns the handle to the IdP's attribute authority in a SAML attribute query message and is returned a signed SAML attribute assertion message. The Shibboleth trust model is that the target site trusts the IdP to manage each user's attributes correctly, in whatever way it wishes. So the returned SAML attribute assertion message, digitally signed by the origin, provides proof to the target that the authenticated user does have these attributes. We note that later versions of the Shibboleth specification have introduced a performance improvement over the earlier versions, by allowing the initial digitally signed SAML message to contain the user's attributes as well as the authentication assertion. Thus the two stages of authentication and attribute retrieval can be combined.

We note that the connection from the IdP to the service provider can also be optionally protected by SSL in Shibboleth. Here SSL is used to provide confidentiality of the connection rather than message origin authentication. In many cases a confidential SSL connection between the IdP and SP will not be required, since the handle can be opaque/obscure enough to stop an intruder from finding anything out about the user, whilst the SAML signature makes the message exchange authentic. However the message exchange should be protected by SSL if confidentiality/privacy of the returned attributes is required. The attributes in this assertion may then be used to authorise the user to access particular areas of the resource site, without the service provider ever being told the user's identity. Shibboleth has two mechanisms to ensure user privacy. Firstly it allows a different pseudonym for the user's identity (the handle) to be returned each time, and secondly it requires that the attribute authorities provide some form of control over the release of user attributes to resource sites, which they term an attribute release policy. Both users and administrators should have some say over the contents of their attribute release policies.

Shibboleth offers numerous possibilities and potential advantages in the context of the Grid. Single sign-on via authentication at a home site and subsequent acceptance and recognition of the authentication and associated attributes released to remote sites is the most obvious advantage. Thus users need not remember X.509 certificate passwords but require only their own institutional usernames and passwords. Institutions can establish their own trust federations and agree and define their own policies on attribute release, and importantly SPs can decide upon what attributes and attribute values are needed for authorization decisions.

The uptake and adoption of Shibboleth technologies within a Grid context is not without potential concerns however. Sites need to be sure that collaborating sites have adopted appropriate security policies for authentication. Strength of user passwords and unified account management across sites is needed. Shibboleth is also by its very nature much more static that the true vision of the Grid, where VOs can be dynamically established linking disparate computational and data resources at run time. Instead Shibboleth requires agreed sets of attributes that have been negotiated between sites. The UK Federation for example is based around the exchange of a small agreed set of eduPerson attributes [18] between IdPs and SPs in the federation.

It is important to note that these attributes are typically statically defined and agreed upon between

the institutions prior to joining the federation, and hence before any formulation of VOs or requests to access Grid resources, i.e. they are based upon statically defined PMIs. This is often sufficient to allow access to certain resources, e.g. a given e-journal for example requires the SP only to know that the individual accessing the resource is from an institution that has paid their subscription for that journal. In the context of the Grid, membership of an institution will not typically be sufficient information for a decision on access to a specific Grid service hosted and managed by a given VO. Rather, VO-specific attributes are needed. This requires more dynamic models of attribute creation and assignment.

The JISC funded DyVOSE project [19] developed solutions which allow for the dynamic creation and acceptance of attributes targeted to the specific needs of different VOs. This is more aligned with the dynamic creation of VOs across Grid infrastructures where dynamic delegation of privilege is supported. As the complexity and number of security policies increases, the ability of a given SoA to delegate responsibility to others is necessary. Through extensions to the PERMIS software, the DyVOSE supported dynamic delegation of authority whereby Grid sites were able to allow an attribute authority controlled by an external SoA to be delegated the ability to assign roles meaningful to a home SoA. Through this, a remote Grid user could hold a role based in the home institution that will allow access to a potentially remote service provider Grid resources.

We note that in static delegation, the roles at the remote institution would need to be hand written into the policy at the home institution. Dynamic delegation factors away the role assigning powers to subordinate authorities, which may delegate the ability to assign local roles to remote attribute authorities, and vice versa. Thus a Glasgow “Student” role may be assigned to an Edinburgh Computing Science user, so they may access the Glasgow resource without the Glasgow SoA knowing about any Edinburgh roles. This trust relationship is agreed beforehand, where it is implicit that the role of Student at Glasgow and Trainee say at Edinburgh are equivalent. Complex delegation allows new intermediate roles with less privilege than their superior role to be defined and assigned to remote attribute authorities. The DyVOSE Delegation Issuing Service supported such dynamic creation and recognition of attribute certificates and is described in detail in [20].

One of the key issues that have still to be resolved with attributes for the Grid community is related to the attribute release policy. At present an SP will request the attributes associated with the potentially opaque identifier (handle) that is returned from an IdP. If a user from the University of Glasgow is involved in numerous Grid projects and VOs however, and all of this information on what VOs this person is involved in, and what their role is in that VO etc are encoded in the core set of attributes, then it is difficult to restrict the information being released. Thus SP may receive more information than they might actually need to make an authorization decision, e.g. if this SP was just one of the many VOs that the user was involved in, then this SP would know more about all VOs the user was involved in. Of course these attributes will be encoded, however, the SP will be able to decode the attributes due to the trust relationships and certificates previously put in place. It is possible to have a richer array of attributes other than the core set of eduPerson attributes, but for interoperability and simplicity, having a core set is beneficial. Given that the focus of much of the Grid community as being represented by the NGS does not focus upon privacy or confidentiality, such issues are not immediately important. Once more security focused groups are involved however, attribute release policies will become more important and only those attributes absolutely needed, will be released.

To support the definition and enforcement of attribute release and acceptance policies, the Open Middleware Infrastructure Institute (OMII) Security Portlets simplifying Access to and Management of Grid Portals (SPAM-GP) project [21] is developing tools (portlets) that support attribute release and attribute acceptance policies, as well as portlets that support the configuration of content within portals appropriate with the attributes it receives from different IdPs. Initial results in applying these portlets across a range of VOs are described in [22].

Alternative models, e.g. based upon agreeing upon a centralized attribute authority for a specific VO such as VOMS, and using those attributes for access decisions specific to resources across the VO is another model. As noted previously, VOMS has traditionally been used for access to and usage of HPC resources using LCMAPS/LCAS and not specifically for fine grained security access to services. The

JISC funded Integrating VOMS and PERMIS for Superior Grid Authorization (VPman) project [23] is exploring these issues. Through enhancements to the Grid standards and implementation of technologies allowing for the pushing or pulling of attributes needed for access to a range of Grid services (including services using the Globus/OMII middleware), a range of authorization scenarios have been demonstrated. These are described in detail in [24] along with the advantages and disadvantages of centralized vs decentralized security models. These scenarios show for example how VOMS attributes passed (pushed) and used to enforce access control (by PERMIS) on access to and usage Grid services providing clinical data. Other scenarios focus on showing how VOMS attributes can be pulled and used to decide upon access control (by PERMIS) on access and usage of restricted electronics applications.

5. Case Studies in User-Oriented Grid Security

The vision of the Grid in seamlessly accessing and using a range of resources is a compelling one, but one that depends on supporting technologies. Single-sign on to resources is one of the fundamental requirements to the realisation of this vision. As noted, different domains will have their own requirements and needs on how this is achieved and in turn on the kinds of security infrastructures and associated policies that need to be enforced. Arguably, the domain that places greatest emphasis on security is the life sciences, and especially when dealing with personal clinical and genetic data sets.

In the post-genomic era, data is growing exponentially. Numerous public genomic, proteomic and metabolomic resources are arising for researchers interested in different organisms (human, rats, arabidopsis,…), different diseases (cancer, diabetes, …), different biochemical and cell signaling pathways amongst numerous other areas. Linkage of individual genetic data with clinical data including for example a given patient or family medical history can be used to detect early onset of hereditary diseases; suggest to patients ways to decrease the likelihood of certain diseases arising, or even to treat patients in a personalized manner where different drugs can be targeted – not just to demographic/phenotypic descriptions of people where for example a young male might be given a different drug than an older grandmother for a similar disease, but actually on the genetic difference of the individuals themselves.

The life sciences in the post-genomic age offer huge potential benefits to mankind. At the same time however there is considerable concern regarding the use of these data sets for other, potentially non-clinical uses. Insurance agencies or employers would be keen to know the likelihood of those they are offering policies/jobs to are to developing a chronic disease. Public association with certain diseases or medical histories more generally leaves individuals rightly concerned that their data is not accessible to the masses.

Within this context, the National e-Science Centre (NeSC) [25] at the University of Glasgow has been exploring the development and management of a wide range of e-Research infrastructures. At the heart of these infrastructures are fine grained security; usability for the end user and support of inter-disciplinary e-Research. Development of a new drug or exploration of a given diseases requires a wide array of researchers and research domains need to be involved. This can include biologists, bioinformaticians, clinicians, statisticians, chemists, physicists, pharmacologists, epidemiologists amongst numerous others. Only through their successful inputs can the whole research process be supported. To explore how we have developed infrastructures supporting inter-disciplinary research we describe several completed and on-going projects including: the MRC funded Virtual Organisations for Trials and Epidemiological Studies (VOTES) [26] project; the BBSRC funded Grid Enabled Microarray Expressions Profile Search [27] project, and the DTI funded Biomedical Research Informatics Delivered by Grid Enabled Services (BRIDGES) project [28].

For demonstration purposes we outline how this infrastructure can support research into cancer however the same infrastructure can (and is) being used across a wide range of other clinical areas. We also note that the following are based upon representative data sets only.

5.1 VOTES Project

The VOTES project was funded by the MRC to develop a Grid framework through which a multitude of clinical trials and epidemiological studies could be supported. Thus rather than engineering bespoke solutions for a given trial or study, VOTES focused on providing an infrastructure where a multitude of trials and studies could be developed and supported, each with their own particular nuances in terms of the data that is being accessed, by whom, and the security policies that apply etc.

At the heart of clinical trials are three key processes: patient recruitment; data collection and study management. Recruitment is primarily concerned with identifying the patients that are *potentially* suitable for a clinical trial, or even whether sufficient patients exist that meet the criteria for a given trial to take place as might be the case when conducting a feasibility study. Once identified it is essential that patients are advised about all matters related with the clinical trial including the potential benefits, the potential dangers and importantly how the collected information will be used within the trial, and potentially any plans for future use of the data. One of the challenges here is that it is essential that patient consent is obtained before any access to identifying patient data is made.

Once a set of patients have been identified, invited to join a particular trial and have subsequently accepted, the next phase is typically focused on that actual undertaking of the trial itself. This includes collecting data on the patients throughout the course of the trial. If the trial is concerned with drug evaluation say, then it is typically necessary to randomize patients with some patients being given the drugs to be evaluated and others a placebo: double blinding is often used here where neither the patients nor those running the trial are aware of who is receiving the drug/placebo to avoid potential bias. Instead a trusted third party is used to keep this information. Tracking and monitoring the patients throughout the trial is typically essential to ensure that all necessary information is collected. In some circumstances this monitoring and follow up can run for many years – and can often lead to insights on the long term effects of particular treatments for example.

Throughout the whole trial process it is essential that the trial is conducted according to a strictly defined protocol. This will typically describe what information is being collected, for what purpose, as well as how it will subsequently be used within the trial and any potential follow on trials or studies. In all of these phases it is essential to ensure that the different people with different roles within the trial can only access and use the data sets and services associated with their particular role in the trial.

The Grid infrastructure developed within VOTES has been described in [29,30]. In brief, the VOTES infrastructure is based upon a portal which provides access to a range of distributed services through which various clinical data sets can be accessed and used. The services themselves have been implemented using Grid middleware such as Globus toolkit version 4 and the OGSA-DAI software. The current implementation combines the access to and usage of a range of software and data sets in widespread use across the NHS in Scotland. These include access to Scottish Morbidity Records (SMR) - one of the most comprehensive clinical data repositories in the UK. The SMR data sets are constructed in conjunction with the General Register Office (GRO) for Scotland. For the purposes of the VOTES project, a subset of over 4 million records of the data has been provided by the NHS covering over 30-years of patient data and care related to hospital discharges, psychiatric admissions and discharges, cancer registrations and deaths. It is worth noting that the NeSC employees working on this project have been granted NHS Honorary accounts. Close collaboration with the NHS is essential in supporting the development of services providing access to clinical data.

The access to the portal is through Shibboleth following the interactions described in section 3. Figure 1 (left) shows the interface to one study where information on cancer patients is being returned (the names are for demonstration purposes only and not real identities). The results of this query are returned on the right of Figure 1. Key to this is that the various attributes that are returned (right hand side of browser interfaces). These are delivered via Shibboleth interactions and used to both personalise the access to different services through the portal, i.e. the portlets that are accessible are based upon having the roles to see them. This model corresponds to security models along the lines of “what you can see is what you can do”.

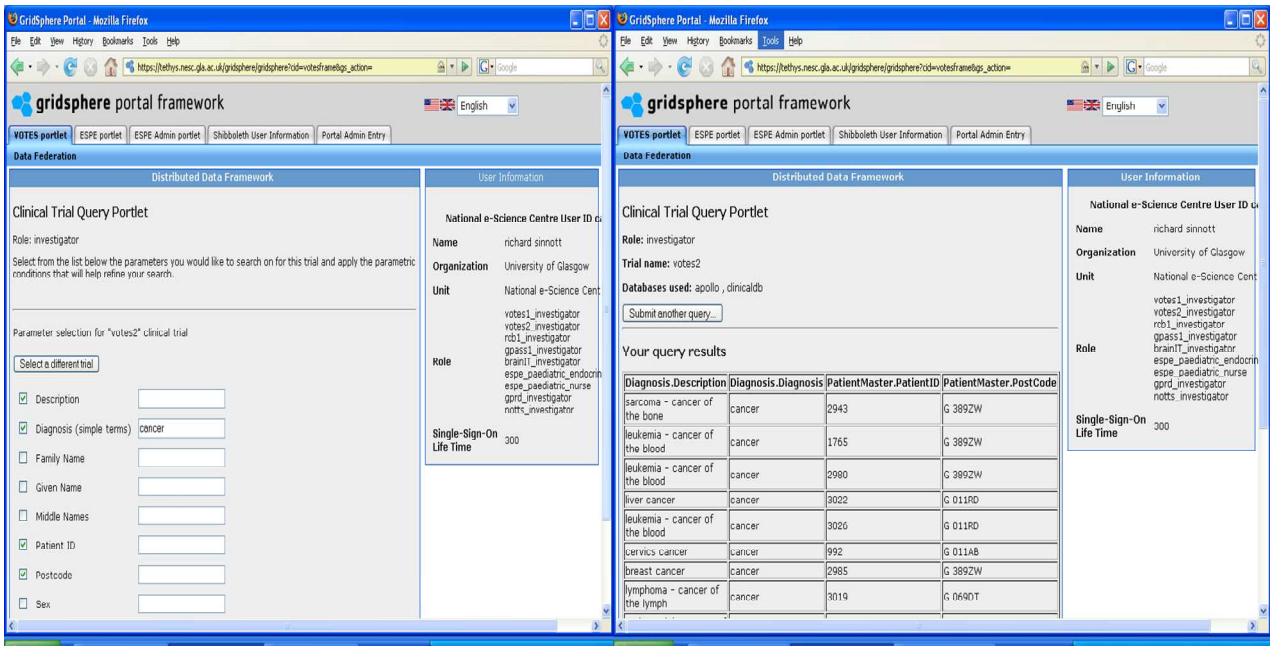

Figure 1: VOTES portal showing cancer clinical data query

However, remote data providers are unlikely to simply allow access to their data sets for someone who has authenticated and provided the right roles to a remote portal. They will want to make their own authorisation decisions. To support this, the remote services providing access to data are also protected with PERMIS and have their own local security policies on access and usage. When a user issues a query which is federated to a remote service provider, their authorisation infrastructure (PERMIS) is configured to pull the X509 attribute certificate associated with that user request to make their own local authorisation decision. In supporting this, the attributes delivered via Shibboleth are kept in a VO-specific attribute authority (LDAP server) associated with the portal. Thus all service providers know where to go to obtain the attributes that they need and have agreed upon when requests for secure access to their data sets are made. When pulled, these attributes are checked for authenticity, validity and if ok, the query is run and resultant data sets returned. We note that the infrastructure does not allow arbitrary querying. Rather, the queries are agreed *a priori* and configured in the portal so that a limited form of querying is possible. Thus a nurse or investigator can only query data sets based upon the forms in the portal – which can be parameterised.

The results of the given query shown here depict the cancer patients; the specific forms of cancer that they have and other information such as their postcode. The geospatial clustering of diseases is often an important research area in its own right. Similarly, when undertaking a feasibility study for a given trial it may be important to know the number of patients with a given condition for example. Such scenarios are being explored in the recently funded ESRC project Data Management through e-Social Science (DAMES) project [31].

Assuming that a set of individuals have been found with a particular condition and they have been recruited to a given trial. Understanding the genetic similarities and differences of these individuals is often required. Do they have the same gene mutations? Are their genetic differences which can help shed insight into why this individual developed this particular disease? To address these kinds of questions requires access to services that deal with individuals genetic data sets. One way in which genetic data is established and analysed is through microarray analysis. The BBSRC GEMEPS project developed various services that allowed secure access to both non-public and public microarray data sets.

5.2 *GEMEPS Project*

The GEMEPS project developed a Grid infrastructure for discovery, access, integration and analysis of

microarray data sets. Through the GEMEPS infrastructure scientists were able to ask the following kinds of questions and obtain appropriate results based upon their privilege: who has run a microarray experiment and generated similar results to mine?; who has undertaken experiments and produced data relevant to my own interests, e.g. for a particular phenotype, for a particular cell type, for a particular pathogen, on a particular platform or microarray chip set?; show me the results from a particular collaborator; show me the conditions and analysis associated with experimental results similar to mine.

In all of these scenarios, the premise was that sites should keep and maintain their own data and define their own security policies on access and usage. Since scientists are often reluctant to publish their data in public repositories until they have published results in recognised journals which can, depending on the journal be a long and protracted affair many data sets remain inaccessible to the wider research community. These issues are discussed in detail in [32].

Scientists in the first instance would like to be able to query across a range of experiments based on any one or more of a variety of search terms. Thus scientists are unlikely to be interested comparing experimental results from *homosapiens* and *barley* for example. We note that at the gene name level however, it is often the case that common gene name clashes do exist across species. To support this basic metadata querying, the GEMEPS project implemented a simple user oriented portlet that allows for a variety of these kinds of information to be used for querying over available (subject to authorisation privileges) data sets as indicated on the left of Figure 2. This was compliant with the bioinformatics standards in this area. The portal itself was Shibboleth protected. Thus when the user from the VOTES portal wished to access and use the GEMEPS portal they could simply redirect their browser to the GEMEPS portal. Access to and usage of this portal and the services that it makes available are done without the need for further re-authentication – one of the key characteristics identified for users: "single sign-on". Figure 2 (left) shows a user searching for experiments containing information on *homosapiens* related microarray experiments conducted on the *GPL570* platform targeting the specific condition *cancer* with *standard Affymetrix procedures* used for hybridisation. The resultant experiments that have been conducted and are available to that user are shown on the right of Figure 2. We note that this is just one example of a query that can be supported.

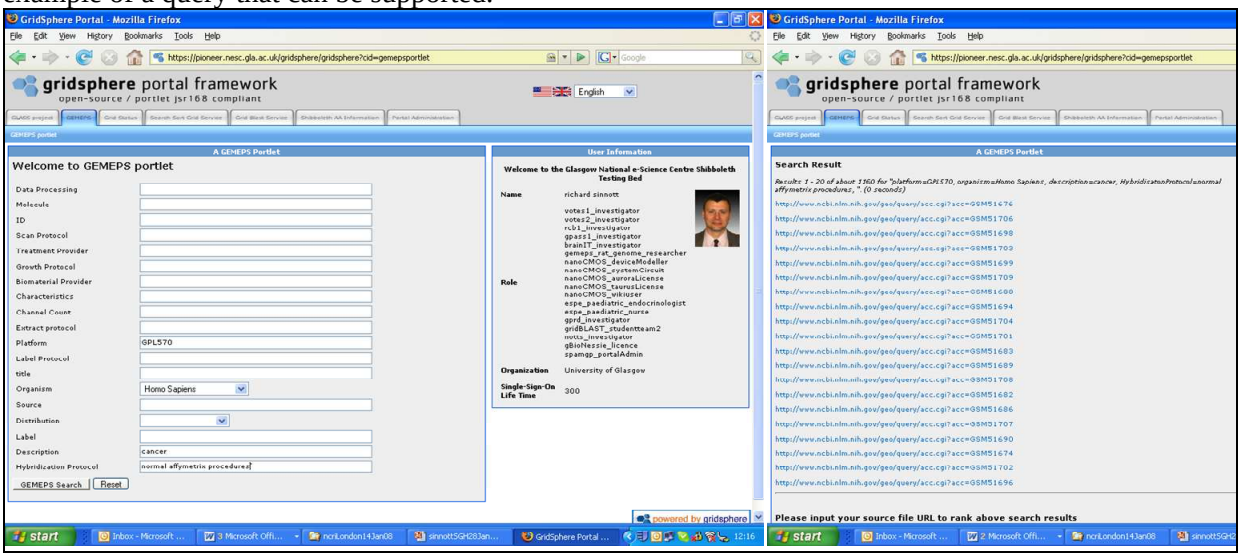

Figure 2: GEMEPS showing cancer microarray experiment query

As well as querying over metadata of experiments, GEMEPS developed services that allowed quantification of the similarity of experiments themselves through comparison of the gene expression levels across experiments. Different statistical similarity models were used for this purpose and are described in [33].

Assuming that a researcher has identified a set of genes that they believe play a role in cancer for patients identified through the VOTES portal, they may often want to understand the protein and

nucleotide sequences associated with those genes to compare how similar/dissimilar they are. The bioinformatics application Basic Local Alignment Search Tool (BLAST) [34] is often used for this purpose. This can be a computationally expensive task. The BRIDGES project developed services that support this directly.

5.3 BRIDGES Project

The BRIDGES project successfully completed at the end of 2005. Its' remit was to provide a Grid infrastructure to support research into genetic causes of hypertension – one of the main causes of cardiovascular mortality. Before BRIDGES, many of the activities that the scientists undertook in performing their research were done in a time consuming and largely non-automated manner. This was typified through "internet hopping" between numerous life science data sources. To address this, BRIDGES developed a security focused data and a compute Grid infrastructure. The data Grid that was developed within BRIDGES is described in [35,36].

The BRIDGES compute Grid used PERMIS to make/enforce distinctions between different privileged and non-privileged users. In particular, the policies defined and enforced with PERMIS were:

- If they are *unknown* users the job would only be submitted to the local "free" Condor pool at NeSC Glasgow;
- If we recognised the users but they do not have a local account on HPC resources at Glasgow, the job would only be submitted to the Condor pool and the NGS;
- If we recognised the users and they have an account Glasgow HPC resources then the job would be sent to the Condor pool, the NGS and to ScotGrid.

These decisions on user identity used the DN of the individuals returned by Shibboleth from their IdP although it was is equally possible to user some other eduPerson attributes instead. We note that this raises issues in the application of Shibboleth itself in the Grid domain. Shibboleth has been developed to support user anonymisation and privacy when accessing and using resources across a federation. However, with the Grid model, knowing which user is accessing a resource, especially in the biomedical domain is crucial. We also note that whilst Shibboleth supports user anonymisation and privacy it is not mandatory and free text strings containing information such as the DN of the user from an IdP to an SP can be returned. The policies on what information and attributes an SP can ask for and what information an IdP is prepared to release will form part of the overall federation contract. There is no obligation on an IdP releasing potentially sensitive information about a given user. However if an SP requests certain attributes to be returned for example which the IdP refuses to release then the SP is completely free to refuse to grant access to their own resource. SP autonomy is thus assured.

The actual selection of where to submit a user jobs was based on availability of resources (which was established dynamically). In accessing and using the BRIDGES portal, users could simply redirect their browser from the GEMEPS or VOTES portal to the BRIDGES portal and automatically access and use services without the need for further re-authentication. The front end to the Grid BLAST service is accessible as shown on the left of in Figure 3. This provided access to a range of genomic and microbial data resources pre-deployed on major HPC resources. To support large scale BLAST usage, users were able to select options that allowed them to be emailed the results when the jobs completed, or they could interactively see the status of the jobs across the various Grid resources (whether they are queued, completed, running) shown in the right of Figure 3.

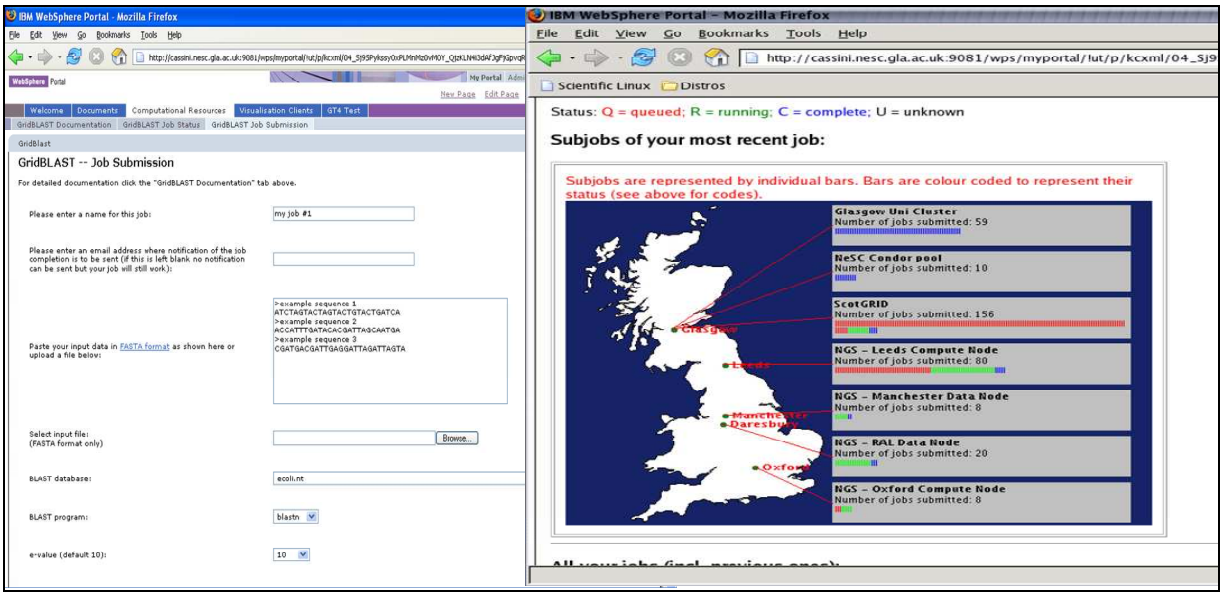

Figure 3: BRIDGES Grid BLAST front end and running/monitoring jobs

6. Conclusions and Recommendations

In this chapter we have explored the current limitations of authentication only based solutions with PKIs as the basis for Grid security. Lack of granularity of authorization will dissuade large groups of researchers from engaging. Perhaps more of an issue is the initial steps through which researchers are asked to proceed before they are able to do anything "on the Grid". X.509 certificates and the process of acquiring one and converting it to appropriate formats is a hurdle that a large swathe of the non-Grid research community will not overcome. It needs to be made simpler and ideally removed completely. Shibboleth offers one possibility through which the usability factor for end users can be addressed.

There are numerous authorization frameworks available today and we have tried to give an overview of their functionality and suitability of some of the most prominent of these. Of those we have listed here, the PERMIS middleware is arguably the most mature solution with tools available for security policy specification and enforcement; for linkage to Grid services in a generic manner; and for linkage to Shibboleth. It is still the case that wider uptake and application across a range of different scenarios is needed before they solutions can be hardened into real products however. For example, considerable effort is still required for deployment and configuration of PERMIS and its inter-working with Globus, OMII and Shibboleth solutions. This will no doubt resolve in time but requires more community effort in exploring and practical experiences of these solutions and whether they meet critical needs of the research community.

Issues not discussed here but essential to consider include fabric management. Grids will always be seen as a threat if they ignore the issue of fabric management. A unified treatment and associated framework for analysing the security of Grid applications, Grid middleware and the underlying OS is needed. If VOs are to be truly secure, then blindly trusting partners to take necessary steps is naïve. Instead, tools are needed to assess the security infrastructures and software across sites before VOs are established. Will site X want to collaborate with site Y if they allow telnet access, or they are using an older version of software with known security holes? Configuration management needs to be brought to the fore in understanding the establishment, management and monitoring of VO fabrics. This may well include patterns of usage for example. Exploratory work in this area is presented in [37].

Shibboleth represents a clear opportunity to overcome the current issues with PKI based security. Trust federations at an institutional level where users can authenticate at their home site and have

appropriate attributes released to service providers (which will use them to make authorization decisions) changes the dynamic of security. There has always been a large degree of trust in the Grid community: trust of users, trust of sites, trust of CAs etc. Hence Shibboleth does not add a new trust requirement especially. Instead trust is moved to IdPs (and ensuring that they have appropriately strong authentication and authorization schemes) and WAYFs (which ensure that the "correct" IdPs are identified and matched with SPs).

Understanding what attributes are needed in the Grid community is essential. Many solutions may only require that the DN is passed over for example, so that accounting and logging of the resource usage for that individual can be achieved, i.e. not further attributes are needed to make an authorization decision. Other more prescriptive VOs may require more information such as VO membership, role of the user etc. Mapping such attributes into a form that Shibboleth can make use of is needed. Once such scenarios can be supported, more understanding of the attribute release policies, attribute acceptance policies and how they might be implemented can be achieved.

In addition to these efforts the web service standards community OASIS, IETF, W3C amongst numerous others are producing a plethora of specifications which "in principle" could help simplify Grid security, however these is still considerable fluidity in these developments with partial/draft specifications, full specifications, and a variety of implementations existing. Single sign on solutions to services at numerous sites, and complementary efforts within the Liberty Alliance consortia offer potential solutions of direct relevance to the Grid community in its move towards web based solutions and service oriented architectures. Similarly, the Web 2.0 community through efforts such as OpenId [38] is proposing yet more security solutions. The security future thus remains in considerable flux.

6. References

[1] R. Housley, T. Polk, *Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructures*, Wiley Computer Publishing, 2001.

[2] ITU-T Recommendation X.509 (2001) | ISO/IEC 9594-8: 2001, Information technology – Open Systems Interconnection – Public-Key and Attribute Certificate Frameworks.

[3] Internet2 Shibboleth Initiative, http://shibboleth.internet2.edu/

[4] UK e-Science National Grid Service, http://www.ngs.ac.uk

[5] UK e-Science Certification Authority, http://www.grid-support.ac.uk/ca

[6] Globus project, http://www.globus.org

[7] Globus Grid Security Infrastructure (GSI), http://www.globus.org/toolkit/docs/4.0/security

[8] R. Sandhu, E.J. Coyne, H.L. Feinstein, H.L. and C.E. Youman, *Role-Based Access Control Models*, *IEEE Computer* 29 (2): 38-47. IEEE Press.

[9] OASIS, Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) v1 September 2003, http://www.oasis-open.org/specs/#samlv1.0

[10] W3C Simple Object Access Protocol, http://www.w3.org/TR/soap/

[11] MyProxy Credential Management Service, http://grid.ncsa.uiuc.edu/myproxy/

[12] L Pearlman, et al., A Community Authorization Service for Group Collaboration, in Proceedings of 3rd IEEE International Workshop on Policies for Distributed Systems and Networks, 2002.

[13] R. Alfieri,et al, *Managing Dynamic User Communities in a Grid of Autonomous Resources*, CHEP 2003, La Jolla, San Diego, March, 2003.

[14] Local Centre Authorization System, http://hep-project-grid-scg.web.cern.ch/hep-project-grid-scg/lcas-lcmaps.html

[15] D.W.Chadwick, A. Otenko, *The PERMIS X.509 Role Based Privilege Management Infrastructure*, Future Generation Computer Systems, 936 (2002) 1–13, December 2002. Elsevier Science BV.

[16] D.W.Chadwick, A. Otenko, E.Ball, *Role-based Access Control with X.509 Attribute Certificates*, IEEE Internet Computing, March-April 2003, pp. 62-69.

[17] OASIS eXtensible Access Control Markup Language (XACML), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml

[18] eduPerson Specification, http://www.educause.edu/eduperson/

[19] Dynamic Virtual Organisations in e-Science Education (DyVOSE) project, http://www.nesc.ac.uk/hub/projects/dyvose
[20] R.O. Sinnott, J. Watt, D.W. Chadwick, J. Koetsier, O. Otenko, T.A. Nguyen, *Supporting Decentralized, Security focused Dynamic Virtual Organizations across the Grid*, 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, Dec 2006.
[21] Open Middleware Infrastructure Institute (OMII) Security Portlets simplifying Access to and Management of Grid Portals (SPAM-GP) project, http://www.nesc.ac.uk/hub/projects/omii-sp
[22] J. Watt, R.O. Sinnott, J. Jiang, G. Stewart, A. Stell, D. Martin, T. Doherty, *Federated Authentication and Authorisation for e-Science*, in Proceedings of APAC 2007 conference, Perth, Australia, September 2007.
[23] Integrating VOMS and PERMIS for Superior Grid Authorization (VPman) project, http://sec.cs.kent.ac.uk/vpman/
[24] R.O. Sinnott, D. Chadwick, T. Doherty, D. Martin, A. Stell, G. Stewart, L. Su, J. Watt, *Advanced Security for Virtual Organizations: Exploring the Pros and Cons of Centralized vs Decentralized Security Models*, 8th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2008), May 2008, Lyon, France.
[25] National e-Science Centre, http://www.nesc.ac.uk
[26] Virtual Organisations for Trials and Epidemiological Studies (VOTES) project, http://www.nesc.ac.uk/hub/projects/votes
[27] Grid-Enabled Microarray Expression Profile Search (GEMEPS) project, http://www.nesc.ac.uk/hub/projects/gemeps
[28] Biomedical Research Informatics Delivered by Grid Enabled Services (BRIDGES) project, http://www.nesc.ac.uk/hub/projects/bridges
[29] R.O. Sinnott, O. Ajayi, A.J. Stell, *Supporting Grid Based Clinical Trials in Scotland*, Health Informatics Journal Special Issue on Integrated Health Records, Vol. 14 (2), June 2008.
[30] A.J. Stell, R.O. Sinnott, O. Ajayi, *Supporting Nationwide e-Clinical Trials and Studies*, Proceedings of 15th Mardi Gras Conference, Baton Rouge, USA, February 2008.
[31] Data Management through e-Social Science (DAMES) project, www.dames.org.uk
[32] R. O. Sinnott, M. M. Bayer, J. Koetsier, A. J. Stell, *Grid Infrastructures for Secure Access to and Use of Bioinformatics Data: Experiences from the BRIDGES Project*, 1st International Conference on Availability, Reliability and Security, (ARES'06), Vienna, Austria, April, 2006.
[33] R.O. Sinnott, C. Bayliss, J.Jiang, *Security-oriented Data Grids for Microarray Expression Profiles*, HealthGrid 2007 conference, Geneva, April 2007.
[34] S.F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, *Basic Local Alignment Search Tool*, J. Mol. Biol. 215:403-410 (1990).
[35] R.O. Sinnott, D. Houghton, *Comparison of Data Access and Integration Technologies in the Life Science Domain*, Proceedings of UK e-Science All Hands Meeting, September 2005, Nottingham, England.
[36] R. O. Sinnott, M. M. Bayer, J. Koetsier, A. J. Stell, *Advanced Security on Grid-Enabled Biomedical Services*, Proceedings of UK e-Science All Hands Meeting, September 2005, Nottingham, England.
[37] R.O. Sinnott, J. Muhammad, Y. Wu, *Deployment of Grids through Integrated Configuration Management*, Proceedings of 26th International Conference on Parallel and Distributed Computing and Networks (PDCN), Innsbruck, Austria, February 2008.
[38] Open Id, http://openid.net/

Author/s:
Sinnott, Richard O.

Title:
Grid security

Date:
2009

Citation:
Sinnott, R. O. (2009). Grid security. In L. Wang, W. Jie & J. Chen (Eds.), Grid computing: infrastructure, service, and applications (pp. 307-334). Boca Raton, USA: CRC Press.

Publication Status:
Published

Persistent Link:
http://hdl.handle.net/11343/28888

File Description:
Grid security