The Design, Development and Application of a Proxy Credential Auditing Infrastructure for Collaborative Research

Christopher Bayliss
National e-Science Centre
University of Glasgow
Email: c.bayliss@nesc.gla.ac.uk

Richard O. Sinnott
Melbourne eResearch Group
University of Melbourne
Email: rsinnott@unimelb.edu.au

Wei Jie
School of Computing
Thames Valley University
Email:wei.jie@tvu.ac.uk

Junaid Arshad
School of Computing
University of Leeds
Email: sc06ja@leeds.ac.uk

***Abstract*—Single sign-on and delegation of privileges are fundamental tenets upon which e-Infrastructures and Grid-based research more generally have been based. The realisation of single sign-on and delegation of privileges in accessing resources such as the UK e-Science National Grid Service (NGS - http://www.ngs.ac.uk) and other national facilities is typically facilitated by X.509-based Public Key Infrastructures (PKI) and exploitation of proxy certificates. This model can be categorised by authentication-oriented access and usage of resources. It is the case however that proxy certificates, can potentially be obtained and abused by a malicious third party without the knowledge of the holder. There is currently no method for end users to detect such misuse. In this paper we describe a novel proxy auditing solution that addresses this issue directly. We describe the design and implementation of this solution and illustrate its application in widely distributed and heterogeneous research environments. We focus in particular on the needs and requirements of such a facility in the ESRC funded Data Management through e-Social Science (DAMES - www.dames.org.uk) project, where secure access and monitoring of social simulations and associated data sets are required by the researchers and associated data providers.**

I. Introduction

In current e-Infrastructure environments, authentication and authorisation of users when accessing resources are essential functionalities that need to be supported. Authentication to facilities such as the UK e-Science National Grid Service (NGS) is predominantly achieved through public key infrastructure (PKI) and use of X.509 [1] certificates issued by the UK e-Science Certificate Authority (CA) http://www.grid-support.ac.uk/ca. Whilst other authentication models have also been explored including federated authentication models of access and usage based upon the Internet2 Shibboleth technology [2] in JISC funded projects such as SHEBANGS [3], ShibGrid [4] and SARONGS [5], the primary and most commonly adopted authentication model by the research community is still based upon X.509 PKI-based authentication where users acquire and maintain their own X.509 certificates and use them to create proxy credentials when submitted jobs or accessing data on resources such as the NGS. We note also that the UK e-Science CA also issues host certificates that can be used for similar purposes. Proxy certificates are commonly used to create a certificate with a minimal subset of the capabilities of the parent certificate, most commonly period of validity, making a certificate that is safer to delegate.

The primary middleware that is deployed on the NGS is the Globus Toolkit [6]. Globus has implemented a model of authentication based upon the Grid Security Infrastructure (GSI) [7]. GSI incorporates essential features to support single sign-on (SSO) and delegation of privileges (also often referred to as delegation of rights). In SSO, access to multiple distributed and autonomous resources, e.g. different NGS HPC clusters, is achieved with a single authentication, i.e. without repeated authentication challenge/responses from each cluster. With delegation of privileges, users are able to make their credentials available to Grid resources to act on their behalf. In realising this SSO and delegation of privileges, GSI relies on proxy certificates. In contrast to end user or host certificates which in the UK e-Science community are signed by the UK e-Science CA directly identity management to a local registration authority) , proxy certificates are signed using the private key of the user or host certificate itself. Proxy certificates can also be derived from other proxy certificates using the certificate's corresponding private key for signing. By signing each certificate with a predecessor's private key, a connection between derived proxy certificates is established that allows Grid resources to resolve the certificate chain up to the user/host certificate and eventually to the issuing CA. Such a chain is shown in Figure 1 Establishing this chain of trust ensures that proxy certificates are trustworthy, i.e. ultimately that they have been issued by the UK e-Science CA whose processes for issuance and revocation of certificates, for management of the underlying PKI etc are accepted both by the Grid users and the resource providers themselves.

Whilst proxy certificates allow for SSO and delegation of privileges to be achieved, they are also a potential danger to the overall security of the Grid infrastructure itself and to the disparate end users themselves. Thus whilst the private key of a user credential is normally encrypted and requires a strong password to use, private keys of proxy credentials

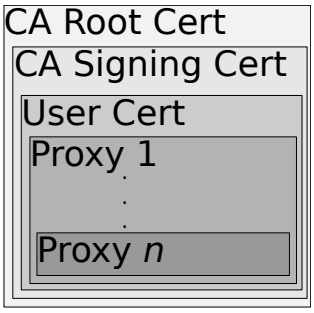

Fig. 1. A certificate chain for a proxy certificate of depth n

are generally unencrypted and stored on the local file system of the Grid resource protected only by file permissions. This model is not a design mistake, but a key requirement that is used to support SSO and delegation of privileges, i.e. since SSO implicitly demands that users only enter their password once and not every time that their proxy credential is used or delegated. To minimise the threats of proxy credentials, most Grid middleware (including Globus and GSI) set a default proxy certificate validity to a much shorter time-span than the life of the X.509 credential itself the default for proxy credential lifetimes is set to 12 hours.

We note that while attempting to create a proxy with a lifetime beyond that of its parent should render it invalid, a suitably prepared attacker may only need a few minutes by using approaches described in [8].

When selecting a lifetime for a proxy certificate it is important to ensure it will remain valid throughout its usage. When submitting a job to a queue of indeterminate length this can be problematic and results in users setting lifetimes significantly longer than required to compensate for unpredictable latency. Therefore, proxies may well have a significant lifetime remaining at the end of the task they were created to perform increasing the window of opportunity an attacker has for using a stolen certificate. While [9] suggests that proxies be invalidated by adding a CRL distribution point to their proxy certificates this is not a viable option in a Grid environment where most certificates are managed by individual users for two reasons. Firstly, most, if not all, the proxy certificate generating tools available do not offer CRL generation as an option and, secondly, CRLs are not checked regularly enough to ensure the revocation was distributed before the proxy expired.

A further challenge is with delegation of privileges an essential component for successful e-Infrastructures. To understand this, consider the follow representative scenario of Grid usage. A user submits a computationally intensive job to run on an NGS compute cluster but their associated input data exists on a different NGS data cluster. The results themselves are required to be written to a local campus Grid resource associated with the NGS, e.g. in the case of Glasgow this might be the ScotGrid resource http://www.scotgrid.ac.uk which itself is a full partner of the NGS. To support SSO and delegation of privileges, the initial NGS compute cluster resource may be presented with a proxy certificate from the user who uses a command such as *grid-proxy-init*, *voms-proxy-init*, or exploits a credential repository such as *MyProxy*. Irrespective of how the proxy credential is created, it is subsequently made available to a particular cluster worker node through a Grid mapfile mapping to a local HPC account. As part of the job execution, this proxy credential can then be used to create a further proxy credential used to access and securely copy data from the NGS data cluster, e.g. through *gridFTP*. Once this data is returned and job execution proceeds and completes, a final proxy credential can be created that is used to return the final resultant data sets to the local campus Grid resource, e.g. ScotGrid.

As seen in this scenario, resource-oriented delegation of privileges of user credentials is supported that allow jobs to act on behalf of the end user (represented by their original proxy credential). The main issue with this model however is that multiple proxy credentials now exist on multiple distributed clusters. Should one of these clusters become compromised then the proxy credential can subsequently be used to create further proxy credentials and used to access other remote resources, masquerading as the original user. This whole process of masquerading as the user can occur without any knowledge of the user themselves who created the initial proxy credential when submitting their job. They may well (quite rightly!) assume that local NGS and/or ScotGrid garbage collection activities take place after running their jobs which will automatically remove proxy credentials and/or temporary files that have been created in executing the compute/data-oriented tasks. This assumption may well be naive however, and as a result security threats and dangers on wider use of their proxy credentials may well exist.

It is emphasised that the proxy credential SSO and delegation of privileges model is especially open to the weakest link security paradigm. That is, should *any* resource in the Grid be compromised by a malicious third party and they manage to gain elevated operating system privileges on that resource, they also gain access to all proxy credentials that are delegated to that resource at the time of the attack and can subsequently attack other resources under the guise of a valid user with valid proxy credentials, in so doing garnering further delegated proxy credentials. These can then be used to explore and exploit potential system vulnerabilities on other resources and to launch distributed denial of service attacks (amongst other worst case scenarios).

Even more complex arrangements are possible when a credential management service such as the SARONGS system is used. This allows for the dynamic creation of short lived, low assurance X.509 certificates to allow users without a certificate to access the NGS via translation of SAML assertions from Internet2 Shibboleth-based Identity Providers.

Obviously many of these issues are caused by allowing e-Researchers access to resources such as the NGS to compile and run or simply execute arbitrary code based solely upon authentication through GSI. In the GSI model, a locally maintained Grid mapfile is used to map the distinguished name

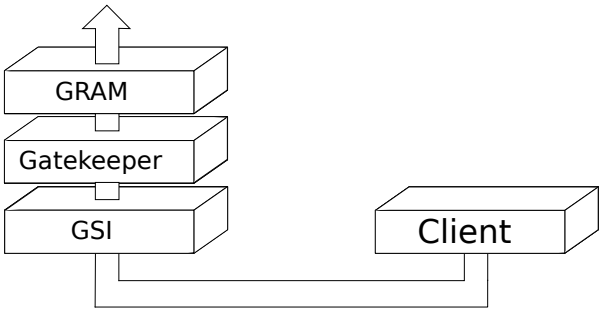

Fig. 2. A simple Globus stack.

(DN) of the user as included in their X.509 certificate with a local user account on the Grid resource as shown in Figure 2.

The GSI-based model provides users with flexibility and is relatively easy to implement and support for resource administrators, but also represents a clear danger in that users can execute arbitrary codes. A more secure model would be to support finer grained authorisation where the users themselves do not get access to local accounts to do *stuff*, but access to services that are fixed and targeted at their needs. Thus for example the NGS is currently exploring GT4 based hosting of services, e.g. a *BLAST* service for biologists, a *Gaussian* service for chemists, as well as support for portals where a predefined set of applications is made available. However it is still the case that the vast majority of people accessing and using the NGS are using GSI models of access and usage (including *GSISSH*) and this seems likely to remain the case for the foreseeable future.

In this context, it is thus highly desirable to reduce the overall risk associated with proxy credentials ideally through transparent extensions to mainstream Grid middleware as deployed on national facilities such as the NGS, i.e. Globus and its use of GSI. We note that it is not realistic to simply deprecate use of authentication-only models since for many researchers, this ability to compile and execute their own simulation codes is essential. To tackle this , one model which has been put forward in the JISC funded Proxy Credential Auditing Infrastructure for the UK e-Science National Grid Service (PCA – http://pca.nesc.gla.ac.uk/) and described in this paper, is to extend the Grid infrastructure capabilities of the NGS and similar resource providers with monitoring and auditing services for proxy credential usage and tracking. The primary requirements for this facility were that it should allow *audit-enabled* GSI resources to automatically capture proxy credential usage and send this and related information to one or more targeted secure, online services for tracking proxy credential usage. These services should allow proxy credential usage information at the:

- individual user level — so that individual users are generally aware of their credential usage and can identify when their credentials are potentially being misused;
- virtual organisation (VO) usage level — so that VO administrators and the members involved are able to track the access to and usage patterns associated with their VO and by its members. Thus it might be the case that a particular VO has been set up to only use particular resources, e.g. particular NGS nodes. When a proxy credential is used to access other resources not identified as part of the VO, then this might highlight that a potential misuse of a proxy credential is taking place (or has taken place).
- at the resource provider level — so that service providers can themselves monitor the usage of their own facilities by their user communities and detect ”as early as possible” any abuses or misuses of credentials, and where appropriate revoke proxy credentials; update certificate revocation lists and update Grid mapfiles, e.g. remove the DN and account information for compromised certificates.

In this work we acknowledge that the approach we are taking represents a pragmatic and realistic model for improving overall security rather than a guarantee of overall security. We recognise that GSI-based access and usage is likely to continue to be the mainstream approach in accessing resources such as the NGS and similar international facilities for some time to come. Our aim is thus to provide a mechanism for rapid detection of credential abuse that address key stakeholders demands.

Furthermore, this work also underpins the areas of granularity in supporting n-tier based approaches. In terms of granularity of access, finer-grained authorisation approaches such as those based upon Role Based Access Control (RBAC) depend upon authentication. Knowing the identity of an individual requesting access to a particular resource is the first step in deciding what roles this person might have which can subsequently be used for finer-grained authorisation decisions, e.g. using technologies such as XACML, OAuthZ or PERMIS. If a proxy credential has been compromised, then a masquerader attempting to access a remote resource may well be indistinguishable from a legitimate user since the Policy Enforcement Point (PEP) — Policy Decision Point (PDP) that might well support finer-grained RBAC models, may well be configured to pull further attribute certificates from a remote attribute authority to make a local access control decision. Similarly, n-tier systems function primarily based upon passing of credentials for authentication and authorisation. Compromised authentication tokens given as proxy certificates, are indistinguishable between tiers unless other challenge/responses are demanded, which violate the intrinsic model and benefits of SSO. In short, if authentication systems are compromised then more granular n-tiered authorisation systems may well be redundant!

A. Related Work

Initial work on proxy credential auditing was described in [10] and [11]. A Globus incubator project has been established to support enhancements and refinements to this work. This work was explored in the course of the PCA project indeed it formed the initial starting point for the work, however

a different architecture and system design has since been undertaken for reasons discussed below.

More generally a body of work has been undertaken on auditing of stack based systems that support message passing paradigms. Typically in this model, an incoming message can pass through several different application layers as it traverses the stack making monitoring individual calls problematic since calls are often logged independently between them making identifying events caused by a specific call problematic. Systems like DTrace [12] and SystemTap [13] have been developed to address run-time logging information. In the absence of a modified application these systems use kernel level services to monitor the target. Monitors are then bound to components throughout the system which generate events when triggered which are subsequently sent to a central monitoring system which filters and processes them in accordance with a script supplied by the user. However, these systems were designed to monitor single systems and have access to robust methods of associating events with their cause via process identifiers. Furthermore approaches such as X-Trace [14] have also been developed to supporting logging of network applications more generally but would have required the introduction of a modified stack which is discussed later. None of these address the fundamental problem of proxy credential usage in collaborative and loosely coupled research environments such as Grids.

II. PCA Software Architecture

The basic model of authentication to an NGS resource through the Globus software stack is illustrated in Figure 2. The initial work on the PCA project explored the prototyping work described in [15] and [11] where the focus was upon implementation of an audit-enabled enhancement to GSI, i.e. replacing the lower level of Figure 2 completely. Whilst supporting the basic auditing capabilities, the work described there had issues in its widespread deployment. Most importantly, it required development and roll-out of a new version of GSI to resource providers such as the NGS. There are numerous pragmatic aspects which make this non-trivial to achieve and other models were thus required.

An improved model of auditing is to provide a transparent auditing layer to the GSI software stack as shown in Figure 2. This is the approach that has been taken in this work. The audit enabled version for proxy credential auditing is displayed in Figure II.

This architecture was adopted for several reasons. Firstly, obviously and most importantly the basic requirements of the system were to allow appropriate parties to observe activity associated with a proxy credential in order to allow both appropriate and inappropriate activity to be identified. Another design requirement was due to the understandable reticence of system administrators of facilities such as the NGS to want to install applications which may worsen the performance, complexity, stability or security of their resources. Therefore, there was the need to integrate with existing software stacks with as little impact as possible. These requirements ultimately

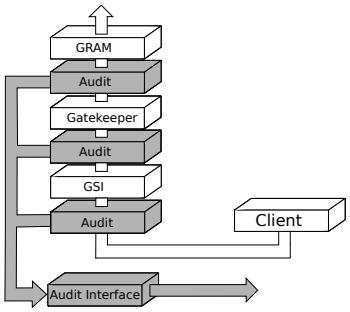

Fig. 3. A simple Globus stack incorporating PCA-based auditing capabilities.

precluded several of the methods used by other similar systems as they require the modification of key components which have performance degradation consequences and/or issues of complexity in system-wide deployment.

The original design of [10] was based upon embedding the sink of the auditing information, i.e. a secure web service that should be notified when the certificate was used or more precisely an event was raised when the proxy certificate (or certificates derived from the proxy certificate) was received by a GSI-audit enabled service. This single sink of information model had several limitations especially when propagating data to third parties. A different model is for data to be collected at the resource on *all* requests without requiring special modifications of the credentials. This eliminates a problem whereby non–audit aware certificates are hidden from the system resulting in incomplete data or, worse, a mechanism for malicious users to disable tracking of their actions. Instead events are published at the resource using information from the certificate the user cannot alter and interested parties can then acquire events as needed.

This design decision introduces a new problem however as it uses a pull instead of a push since it requires consumers to know which sites have data they are interested in. Should a proxy cert be used on an unexpected site the user has no way of discovering this unless the attempt raises an event which includes details of the new system. However, as we associate events with the certificate used to invoke them the user can locate them by searching. Thus given a proxy certificate such as the one shown in Figure 1 all proxy certificates generated share the same sequence of parent DNs from the root of the issuing CA to the end entity. More generally, each entity that requires access to the data discovers events using data known to them. Users, as mentioned above, can use details of their certificate to obtain associated event information.

Altering the model to collate events at the service also changes the security model of the system. In work such as [11], events are forwarded to a logging service whose URL was embedded in the proxy certificate. This provides two potential security problems. Firstly, it is possible for a malicious party to embed the URL of a third party in a proxy certificate and then use it to access services in order to generate traffic as part of a Denial of Service (DoS) attack. Secondly, it would be possible for an attacker to direct services to forward events

to a server which maintained the HTTP connection for as long as possible opening the possibility of submitting sufficient jobs to exhaust the server's supply of ports again causing a DoS.

In environments such as the NGS, it is highly desirable to refine individual level auditing information of resource usage. One mainstream way that this is achieved is through establishment and support of Virtual Organisations (VO). In this model, technologies such as VOMS [16] are used to establish the VO structure including the roles and privileges assigned to individuals in that VO. This information when included in a proxy credential (as an extension to the X.509 credential itself) is subsequently used by solutions such as LCMAPS and LCAS to transparently map VO-specific resource requests onto Grid resources targeted to the needs of the VO itself. Often this is to map VO user requests onto pooled accounts established on resources such as the NGS for the purposes of that VO. With this model, a VO administrator is typically tasked with establishing the software environment on the Grid resources, i.e. configuring the software and data resources associated with that VO. As with individual level usage tracking through the DN and the hierarchy of parent certificates outlined in Figure 1 VO specific usage can be logged and auditing through extracting the associated VOMS attribute certificates and the DN of the users embedded into the certificates. Key to this solution is to recognise that the same individuals can belong to multiple VOs and thus want/need access to multiple auditing services.

By separating the event processing between the service, user and applicable third parties the system can support many different groups with differing requirements. The type of questions the administrator of a site may wish to answer will likely be based around who and how their resources are being used. Clients will likely only wish information relating to when their credentials were used and if they accessed further sites. Third parties may similarly attach to a filtered feed appropriate to their needs and potentially emit events from their own interface for consumption by other entities.

A. Auditing Events

Based on the previous discussions, the PCA system has been designed to offer a wider range of event options instead of simply credential acceptance and logging / auditing information capture. A key requirement was how to associate multiple events, potentially taking place across multiple machines, with their causal predecessors. We currently solve this problem by associating events with a connection object which represents a specific TCP connection between a client and local service. We create the connection object after the TCP connection is established but before any security context is established in the SSL or GSI layer. Should the context fail to be established we emit a connection failure event if successful we emit a connection succeeded event. In both cases if a client credential is provided it is associated with the connection by a further event.

We assume that we can insert extra data into a request as it passes up the service's stack. This allows us to insert an identifier that event emitters can use to associate events with the correct connection. Our current use cases use both HTTP messages and / or shell scripts to propagate through the stack which are trivial to modify in this way.

The system uses an event logging service at each resource which is responsible for collecting event data from a specific site. It is then responsible for providing this data to external entities in a secure and useful manner.

B. Auditing Event Emitters

The original service described in [15] used a modified GSI library which performed a SOAP call to push the event to the logging service. In addition to the previous limitations, this design had a major drawbacks in that the SOAP call introduced latency into the stack i.e. the GSI handshake would not complete until the call had completed. As outlined above, the general design pattern for an audit/usage event emitter was based on the proxy pattern. This was selected as it allows emitters to be inserted between the layers of the existing stack without requiring any modifications to software forming important parts of the system. As noted previously, we felt that this minimum impact approach was most likely to be accepted by site administrators who would be hesitant to install modified replacements to vital components such as the GSI security library or the GRAM job manager.

C. Credentials

The original design used in [15] required an SSLv3 extensions mechanism to embed a field containing the URL of the event logging service. This requirement was dropped in the PCA solution as it allowed hostile users to simply omit the extension from their certificates to avoid their actions being monitored. Instead events are always captured at the service site with the option of events being sent to further sites if requested. This design allows events to be forwarded to potentially many third party sites allowing usage to be monitored.

D. External Interface

As the PCA design no longer pushed events directly to a location accessible by the user there is a need to provide an external interface for clients to acquire auditing and usage information. As we are no longer simply forwarding events as they occur it becomes possible to offer more sophisticated services. For example, a consumer may not simply wish to receive a stream of all events generated but to filter them into, potentially several, different streams matching specific patterns. Such patterns could capture undesired behaviour, such as proxy rejection, or potentially malicious behaviour, such as a the usage of a sub-proxy certificate not generated by the user.

E. Analysis

Simply collecting logs of events is in of itself not directly useful. There is a need for the events to be analysed to identify patterns of usage both, permitted and forbidden, so that useful

information can be extracted from the raw data. Again, the PCA design permits the different entities in the system to focus on processing data for their own needs.

III. Implementation

The entire project is currently written in the Ruby programming language. The event logging service is a Ruby on Rails application providing a RESTful API for event emitters to access.

A. Events

There was a need for some format with which to publish generated events. As we commonly produce a time series the Atom Syndication format [17] presents itself as an obvious candidate being an IETF standardised XML format for publishing data in reverse chronological order. Adopting an existing format allowed us to exploit the resources developed for it and reduce our development time. Events themselves are are stored in a MongoDB document store [18]. This was selected for the scalability, functionality and overall compatibility with the event information that is captured. That is, given that events are modelled as a bag of name value pairs associated with a connection a document store offers a close semantic match and allows for highly performant key-value stores look-ups.

B. Event Emitters

Development of the complete set of event emitters is currently on-going. Initially we focused on the development of an SSL server which emits connection and security events and allowed for security information to be captured and logged by an event emitter. Following this a primitive GRAM job manager event emitter was developed which emits events when it receives a Globus based job (either through globus-job-submit or globus-job-run). The event emitters wrap these services and are completely transparent to GRAM itself and thus not intrusive into the overall Globus software. This software has been developed and tested on an NGS-like cluster at the National e-Science Centre at the University of Glasgow, i.e. a cluster with the same job submission software stack as currently exists on the NGS.

C. Public Interface

As mention previously the public interface is implemented as a Ruby on Rails application that provides an to the secure web service interface to the event store. It provides both a HTML and programmatic atom based access to the data. Currently, the secure web service front end supports a simple query interface which permits querying the data store using URL encoded queries. Queriable attributes include the DN of the subject one of the members of the certificate chain of a credential and the name of an action as a string using regular expressions. It is also possible to specify maximum and minimum points in time within which an events occurred or when a connection was established. By encoding the query in the URL instead of the HTTP request body queries can be treated as first class entities by the system and provided with the same services as a connection and events group is allowing users to create a filtered view of events that suits their needs.

IV. DAMES Case Study

To demonstrate the application of the PCA proxy credential auditing infrastructure we have identified a portfolio of projects using the NGS and related resources. The ESRC funded DAMES project is a prime candidate for proxy credential usage tracking. The DAMES project is focused upon the challenges of data management facing the social sciences. It is the case, as with many other domains, that the social sciences are facing unprecedented challenges in the volume and heterogeneity of data sets from an increasingly diverse portfolio of data providers. When deal with issues around e-Health for example, it is often necessary to leverage data resources crossing the social, clinical and geospatial domain where individual and autonomous data providers are extremely aware of, and bound by criteria associated with information governance on data access and usage. The DAMES project has four primary areas of data management in the social science domain and is developing a family of specialist data environments to tackle the challenges that arise. These include:

- Grid-Enabled Educational Data Environment (GEEDE) – where researchers are able to access and analyse national and international data resources associated with education and associated qualifications;
- Grid-Enabled Occupational Data Environment (GEODE) – where researchers are able to access and analyse data resources associated with occupational classifications and associated coding systems;
- Grid-Enabled Minority Data Environment (GEMDE) – where researchers are able to access and analyse a variety of data resources associated with minority and ethnicity;
- Grid-Enabled Health Data Environment (GEHDE) – where researchers are able to access and analyse a variety of data resources associated with clinical and health related resources;

The GEODE and GEHDE related work is described in [19] and [20] respectively.

All of these data environments require access to and usage of statistical analysis tools. A variety of such solutions are widely used in this space including SAS, STATA and R. In the on-going work in DAMES we have supported exploitation of R on large scale HPC facilities. This was due in part to R being open source and already deployed on facilities such as the NGS and the expertise of the social science community. R itself can be used for a variety of statistical analysis. Of particular relevance to DAMES is the coding, recoding and subsequent statistical analysis of social science data sets. Many of these data resources are large and can require significant processing, especially when re-purposing is needed. One example from GEMDE is the classification of UK Census data from the UK Data Archive related to the ethnic classification of the UK population over the past 40 years. Researchers may want to analyse this data from a variety of perspectives. Considering white/British and Others; considering white British;

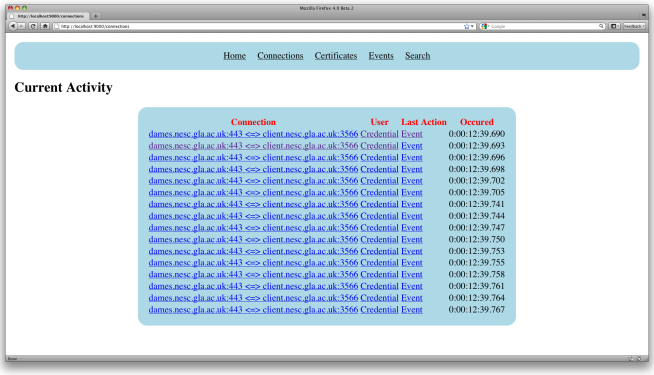

Fig. 5. The overview page of the PCA interface showing recent activity

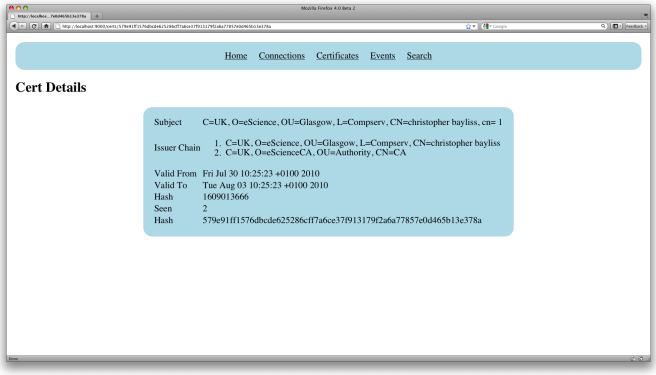

Fig. 6. The PCA interface showing details of a certificate.

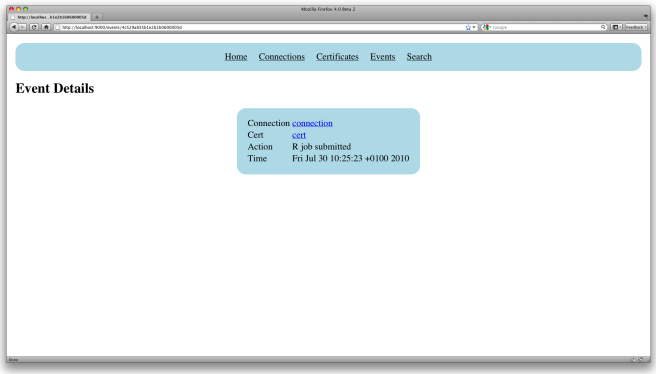

Fig. 7. The PCA interface showing details of an event.

black African; Indian; Pakistani/Bangladeshi; as a unit group, and Chinese and Others as another group etc and comparing this data at a regional levels versus a national level and indeed comparing across regions. Different coding and classification schemes are used for this purpose, and re-purposing of the data is often needed.

To support this, the DAMES project intends to exploit the UK e-Science NGS however our work here is based on a representative cluster in Glasgow, i.e. using the same software stack. The social scientists involved in the DAMES project exploit a portal developed using the LifeRay framework itself and accessible through the UK Access Management Federation (http://www.ukfederation.org.uk). Details of how the SAML assertions provided by the UK Access Management Identity Providers are used to configure the portal contents, and subsequently used to create user X.509 proxy credentials are described in [21]. At present users are able to create their own X.509 proxy credentials through a targeted portlet that exploits a dedicated MyProxy service. Other approaches also exist for this purpose as outlined previously, e.g. SARoNGS.

The user interface for social simulation exploiting R is shown in Figure 4 along with the output showing the average age and sex distribution of the UK ethnic population from the period 1971-2001 based on the Socio-Economic Position and Political Support of Black and Ethnic Minority Groups in the United Kingdom [22].

The auditing information that was obtained in the execution of this R script, i.e. to illustrate where these R jobs were, when and by whom these jobs were run is shown in Figure IV. The result shown is the browsable HTML interface displaying a summary of discrete connections. Each connection shows the end point addresses and links to detailed information on the credential used and events associated with this credential.

V. Future Work

The work described here has demonstrated the proof of concept in auditing of proxy credential usage and its application in the DAMES project. The work is far from complete however. There are numerous avenues and case studies that remain to be explored as part of the PCA project. However it is the case that the software has reached a stage where we can begin to deploy it at test NGS sites. A workshop is scheduled with the NGS technical support staff to demonstrate the solutions put forward in August 2010. Case studies also exist as part of the PCA project to demonstrate this software when used in an international context, e.g. in supporting access to and use of the NGS nodes, ScotGrid and international HPC facilities including TeraGrid in the US and D-Grid in Germany. Such international auditing efforts represent a key requirement in establishing global Grid infrastructures.

As the auditing work in PCA continues we expect to explore a variety of other research areas in auditing and usage of Grid facilities. In particular, once basic auditing capabilities exist, it will be possible to explore research avenues in the area of identifying irregular patterns of usage. Our focus here is on training algorithms to predict potentially suspicious proxy credential usage. We intend to apply Bayesian Neural Networks in this regard. However given the often sporadic access to and usage of Grid facilities by user communities, we expect that this in turn will be a challenge in itself.

Acknowledgement

The PCA work described here is funded by the Joint Information Systems Committee (JISC) in the UK. The DAMES project is funded by the Economic and Social Sciences Research Council (ESRC) in the UK. The authors gratefully acknowledge this support.

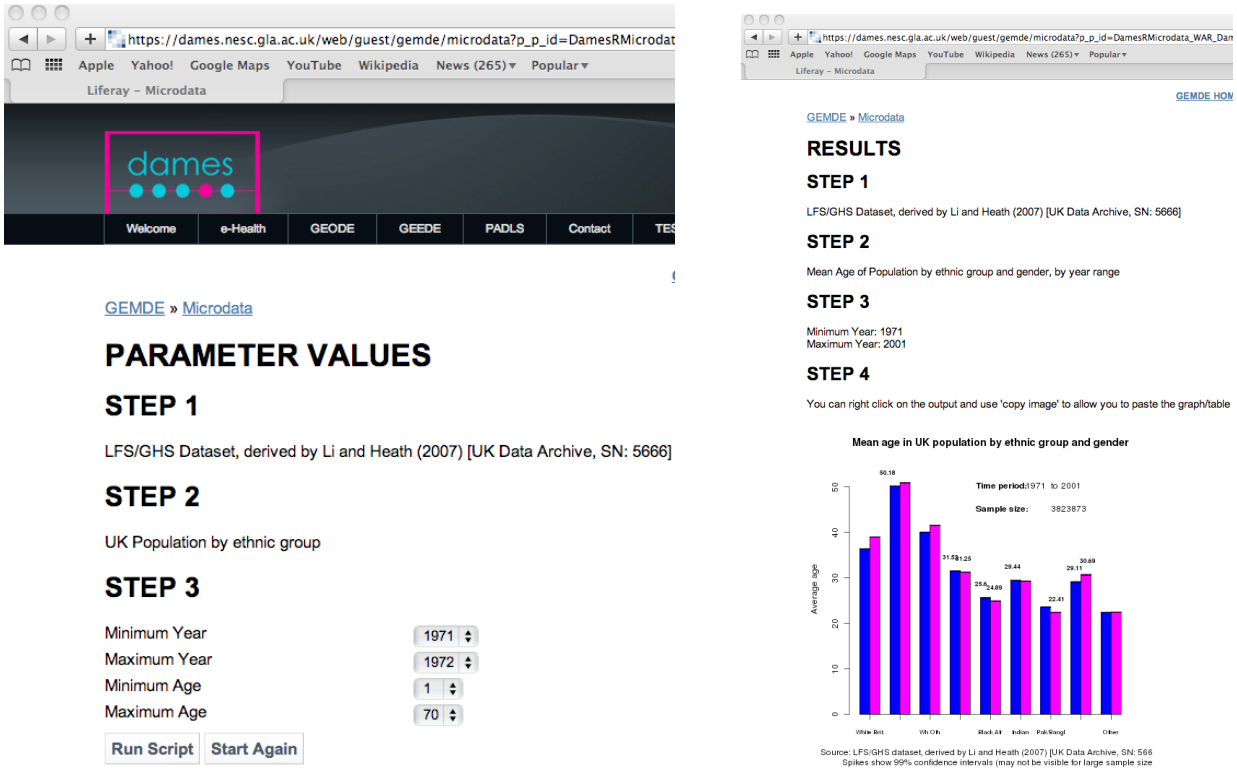

Fig. 4. R Job Submission via the DAMES portal to an Audit-enabled Grid Resource and Output Results

REFERENCES

[1] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, "Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile," RFC 5280 (Proposed Standard), Internet Engineering Task Force, May 2008.

[2] "Shibboleth." [Online]. Available: http://shibboleth.internet2.edu

[3] R. O. Sinnott, J. Jiang, D. J. Watt, and O. Ajayi, "Shibboleth-based access to and usage of grid resources," in *In Proceedings of the 7 th IEEE/ACM International Conference on Grid Computing, IEEE Computer*. Society Press, 2006, pp. 136–143.

[4] D. Spence, N. Geddes, J. Jensen, A. Richards, M. Viljoen, A. Martin, M. Dovey, M. Norman, K. Tang, A. Trefethen, and D. Wallom, "Shibgrid: Shibboleth access for the UK national grid service, e-science 0: 75," 2006.

[5] X. D. Wang, M. Jones, J. Jensen, A. Richards, D. Wallom, T. Ma, R. Frank, D. Spence, S. Young, C. Devereux, and N. Geddes, "Shibboleth access for resources on the national grid service (sarongs)," *Information Assurance and Security, International Symposium on*, vol. 2, pp. 338–341, 2009.

[6] I. Foster, "Globus toolkit version 4: Software for service-oriented systems," in *IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779*, 2005, pp. 2–13.

[7] R. Butler, V. Welch, D. Engert, I. Foster, S. Tuecke, J. Volmer, and C. Kesselman, "A national-scale authentication infrastructure," *Computer*, vol. 33, no. 12, pp. 60–66, 2000.

[8] S. Staniford, V. Paxson, and N. Weaver, "How to 0wn the internet in your spare time," in *In Proceedings of the USENIX Security Symposium*, 2002.

[9] S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson, "Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile," RFC 3820 (Proposed Standard), Internet Engineering Task Force, June 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3820.txt

[10] C. Kunz, C. Szongott, J. Wiebelitz, and C. Grimm, "Design and implementation of a grid proxy auditing infrastructure," *E-Science Workshops, 2009 5th IEEE International Conference on*, pp. 11–18, December 2009.

[11] C. Szongott, "Webser vice-basier tes Auditing für Grid Proxy Credentials," Master's thesis, Gottfried Wilhelm Leibniz Universität Hannover, 2009.

[12] B. M. Cantrill, M. W. Shapiro, A. H. Leventhal, and S. Microsystems, "Dynamic instrumentation of production systems," 2004, pp. 15–28.

[13] "Systemtap."

[14] R. Fonseca, G. Porter, R. H. Katz, S. Shenker, and I. Stoica, "X-trace: A pervasive network tracing framework," in *In NSDI*, 2007.

[15] R. O. Sinnott, J. Jiang, D. J. Watt, and O. Ajayi, "Shibboleth-based access to and usage of grid resources," in *In Proceedings of the 7 th IEEE/ACM International Conference on Grid Computing, IEEE Computer*, 2009.

[16] R. Alfieri, R. Cecchini, V. Ciaschini, . Frohner, A. Gianoli, K. Lrentey, F. Spataro, and I. Firenze, "an authorization system for virtual organizations," in *In Proceedings of the 1st European Across Grids Conference, Santiago de Compostela*, 2003, pp. 13–14.

[17] M. Nottingham and R. Sayre, "The atom syndication format," 2005.

[18] "Mongodb." [Online]. Available: http://www.mongodb.org/

[19] P. S. Lambert, K. L. L. Tan, K. J. Turner, V. Gayle, K. Prandy, and R. O. Sinnott, "Utilising a grid enabled occupational data environment," in *16th World Congress of the International Sociological Association*, July 2006.

[20] S. McCafferty, T. Doherty, R. O. Sinnott, and J. Watt, "Supporting research into depression, self-harm and suicide across scotland," *Journal of the Philosophical Transactions of the Royal Society A*, July 2010.

[21] J. Watt, R. O. Sinnott, T. Doherty, and J. Jiang, "Portal-based access to advanced security infrastructures," in *UK e-Science All Hands Meeting*, September 2008.

[22] Y. Li and A. Heath, "Employment status of 1st and 2nd generation minority ethnic groups in britain: a tale of 35 years," *SRC Britain Today*, p. 95, March 2007.

Author/s:
BAYLISS, CHRISTOPHER; Sinnott, Richard O.; Jie, Wei; Arshad, Junaid

Title:
The design, development and application of a proxy credential auditing infrastructure for collaborative research

Date:
2011

Citation:
Bayliss, C., Sinnott, R. O., Jie, W. & Arshad, J. (2011). The design, development and application of a proxy credential auditing infrastructure for collaborative research. In 5th International MCETECH Conference on eTechnologies, Les Diablerets, Switzerland.

Publication Status:
Published

Persistent Link:
http://hdl.handle.net/11343/32651

File Description:
The design, development and application of a proxy credential auditing infrastructure for collaborative research