An Authentication Strength Linked Access Control Middleware for the Grid

J. Chin[1], M. Parkin[2], N. Zhang[1], A. Nenadic[1], J.M. Brooke[2]
School of Computer Science
The University of Manchester

[1]{jchin, nzhang, nenadic}@cs.man.ac.uk
[2]{m.parkin, j.m.brooke}@manchester.ac.uk

Abstract

Current Grid software lacks support for authentication techniques other than PKI-based digital certificates. Support for multiple authentication technologies is important, particularly in information Grids, where classified resources/data should enjoy fine-grained protection. This paper describes our work in the design and implementation of a Grid authentication middleware. The solution, called Flexible Authentication Middleware Extension for Grids (GridFame), based on the Web Services Resource Framework (WSRF), will enable authentication strength linked access control to Grid resources.

Keywords : Authentication, Authorization, computational grid security, multi-factor authentication, secure communication

I. Introduction

A Grid is a collaboration of computational resources, applications, and data that spans multiple domains and institutions. This results in a Virtual Organisation (VO) [20] where users are able to gain access to resources across multiple domains from anywhere on the Internet. The existing middleware [11,12] developed by the Grid community mainly relies on digital certificates for user authentication. It does not support more advanced authentication techniques such as smart cards, smart tokens, and biometrics.

In the Grid environment, it is expected that scientists or researchers will gain access to Grid resources not only through wired devices such as desktop machines and home computers, but also through wireless devices such as laptops, PDAs and smart phones. This poses problems of supporting user mobility and of secure management of the private keys associated with the certificates. Current Grid Authentication requires that a user's digital certificate be present on each and every device that is used to access the Grid – ultimately, this expose the certificate's private key on multiple locations and is considered as a security weakness. Additionally, when a certificate expires, is revoked, or for any reason needs to be updated or removed, the changes would have to be replicated on every device that is used by the user, and this is often a costly and time-consuming process. The use of Remote Repositories [4] manages to solve a part of this problem by having an on-line credential repository for storing the certificates and private keys. Currently, the MyProxy [3] on-line credential repository is used to provide access to Grid resources via Grid portals. The biggest disadvantage of using Grid portals is that it only allows accesses through a Web browser, thus limiting the types of authentication methods that can be used. Furthermore, ‘Smart’ devices such as Personal Digital Assistants (PDAs) and smart phones may not have the computational resources to manage digital certificates. Therefore, alternative, preferably less expensive, authentication techniques should be supported.

Other proposed projects, such as GridLogon [13], often use the Pluggable Authentication Module (PAM) which allows a user to be authenticated with a variety of mechanisms.

Our aim is to not only support a variety of mechanisms for authentication, but also to derive an authentication strength, or the Level of Assurance (LoA), from the authentication method(s) used. The LoA can be fed into an authorisation engine to derive an authorisation decision based upon appropriate access control policies. Our work will be based upon an implementation of the recently standardised Web Service Resource Framework (WSRF) specification. We aim to make our solution sufficiently lightweight and portable for it to be executable on handheld devices .

II. Background

In this section we present a brief overview of current Grid authentication approaches, the WSRF-lite software, and the related Flexible Access Middleware Extension (FAME) project.

A. Grid Authentication

Mutual authentication in a Grid environment is established based on a Public Key Infrastructure (PKI) that requires the use of X.509 identity certificates [14]. In order to access a Grid resource, a user needs to have a valid X.509 certificate together with the corresponding private key. This credential (i.e. the certificate and the private key) is then used to generate a short-lived(usually valid for 8-24 hours) credential known as a X.509 proxy credential [16]. The proxy credential consists of a private key and a certificate that has been signed by the user's original credential. This in effect allows the user to delegate his/her privileges to a proxy certificate that can then be used for authentication within the Grid environment. Proxy certificates can be used to provide “Single Sign-On” (SSO) in a scenario where a user submits a job, and uses a proxy certificate to allow the submitted job to act on the behalf of the user to spawn other jobs or authenticate with other Grid resources.

B. MyProxy

The MyProxy [7] online credential repository is a service designed for on-line storage and management of user credentials in Grid environments. A user can store his/her credential in the repository, and retrieve it later from anywhere on the network whenever he/she is to access a Grid service. With MyProxy, Grid users no longer need to store credentials on every device he/she uses to access the Grid, thus significantly simplifying the management of credentials.

MyProxy is typically used in a Grid Portal environment. Once a user logs on to a Grid Portal with a valid ID and password, the portal will obtain a proxy credential on the user's behalf from MyProxy. The proxy credential is then used by the Grid Portal to authenticate the user to other Grid resources. Access to MyProxy in order to retrieve credentials is commonly access-controlled by the “username/password” sign-on procedure. Other authentication methods such as PAM and Kerberos are still in the experimental stage at the moment [8].

C. Web Service Resource Framework (WSRF)

WSRF [1] is a new specification developed by OASIS (Organisation for Advancement of Structured Information Standards). It is designed to merge Grid and Web technologies based on the requirements of OGSA (Open Grid Services Architecture) [2]. WSRF was introduced in 2004, and is a product of the re-factoring and extension of the OGSI framework to make it more compliant with the Web Services Standards [3]. The Globus toolkit (version 4) will include a complete implementation of WSRF.

WSRF is a set of specifications that allows the modeling and creation of stateful resources using Web services [4]. This allows a Web service to create, address, discover, inspect, and manage stateful Grid resources.

The work presented in this paper uses the WSRF-Lite [4] software which is a WSRF compliant Grid container implemented using the Perl programming language. WSRF-Lite was developed at the University of Manchester and is being supported by the Open Middleware Infrastructure Institute (OMII) Managed Programme. WSRF-Lite [4] builds on the success of OGSI-Lite, which was the OGSI implementation used by the RealityGrid [5] and TeraGyroid [6] projects.

WSRF-Lite is lightweight and highly portable, allowing us to deploy Grid Services on a variety of platforms ranging from Personal Digital Assistants (PDAs) to High Performance Computing (HPC) Clusters - any platform where a Perl Interpreter is available. WSRF-Lite is also interoperable with other Java and .NET WSRF implementations.

D. Flexible Access Middleware Extension to Permis (FAME-Permis)

FAME-Permis [10] is a project sponsored by the Joint Information Systems Committee (JISC), and jointly undertaken by the University of Manchester and the University of Salford. This project is aimed at developing an authentication middleware extension that supports multifaceted authentication within the context of Web services and applications, derives an authentication strength based upon the authentication method used, and feeds this strength into the Permis authorisation decision engine. GridFAME is an extension of the FAME-PERMIS project to support light-weight Grid clients using WSRF-Lite.

The FAME part of the FAME-Permis project [9] focuses on supporting a wide variety of authentication methods through the use of a plug-in architecture. At the moment, there are plug-ins for supporting authentication methods based on IP addresses, username and password pairs, certificate-based soft tokens and other Java cards based authentication methods.

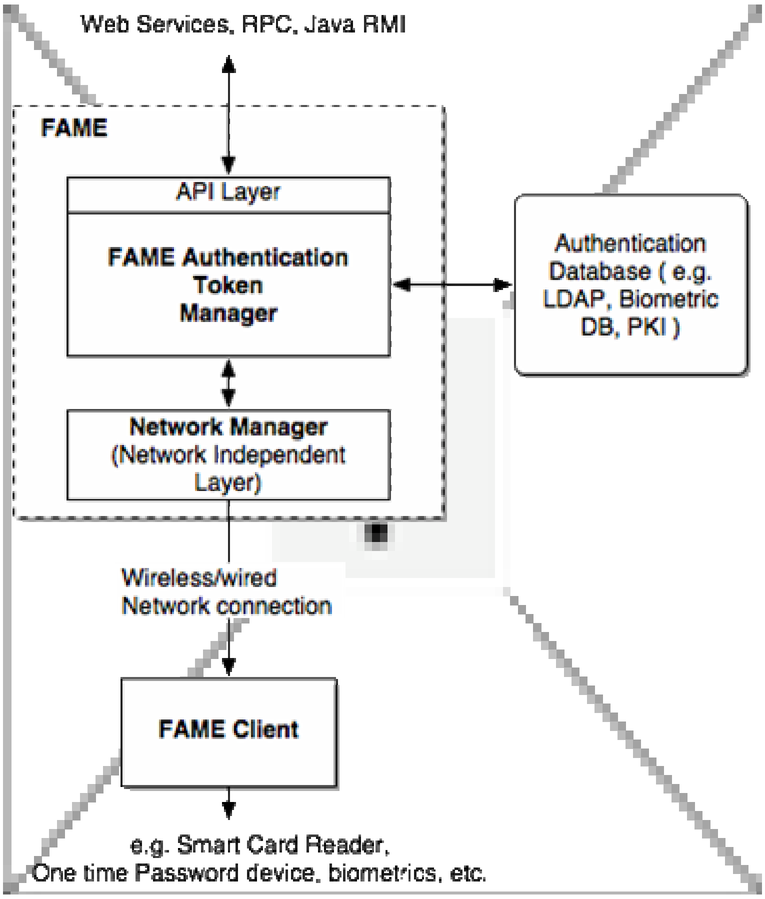

Web Services, RPC, Java RMI
FAME
API Layer
FAME Authentication Token Manager
Authentication Database (e.g. LDAP, Biometric DB, PKI)
Network Manager (Network Independent Layer)
Wireless/wired Network connection
FAME Client
e.g. Smart Card Reader, One time Password device, biometrics, etc.

Figure 1. FAME System

Figure 1 shows the main components of the FAME system. Requests for authentication tokens can be invoked through the FAME Authentication Token Manager (ATM) using either Web services, Remote Procedure Call (RPC) or Java Remote Method Invocation (RMI). A FAME request contains information about the target machine from which the authentication token could be retrieved. The ATM will then verify and process the request. If the request is valid, the ATM will connect to the FAME client on the target machine. The FAME client interfaces with any connected authentication device and communicates with the ATM in order to retrieve the authentication token before sending it back to the ATM for verification. The architecture of the FAME client includes a device independent layer that allows interfacing with future devices through the use of plug-in modules. Once the authentication token has been verified with the authentication database, the ATM may derive the LoA before sending it along with the authentication token to the requester. The LoA could then be used by a decision engine, such as PERMIS [17], to derive user privileges based on the authentication method used.

III. GridFAME

As indicated in the introduction, the fundamental problem that we address in this work is to allow the integration of multiple authentication services in a Grid environment. Our work also allows authentication strength to be linked with access control decision making in environments where different applications may require finer grained access controls with varying authentication requirements – in essence, a stronger authentication credential such as smart tokens or one time passwords should grant a user with a higher level of privileges. Another requirement is that our work should interoperate with WSRF implementations and be sufficiently lightweight to be deployed on PDAs or smart phones.

We address these problems by introducing a Grid Flexible Access Middleware Extension (GridFAME), which is an extension to the FAME project. This extension enables Grid Services to utilise the multiple authentication capability of FAME via a lightweight WSRF implementation. Furthermore, based on the authentication mechanism used, it is able to derive the authentication level (LoA) and feed it to a third party authorisation service for fine-grained access controls based on authentication strength.

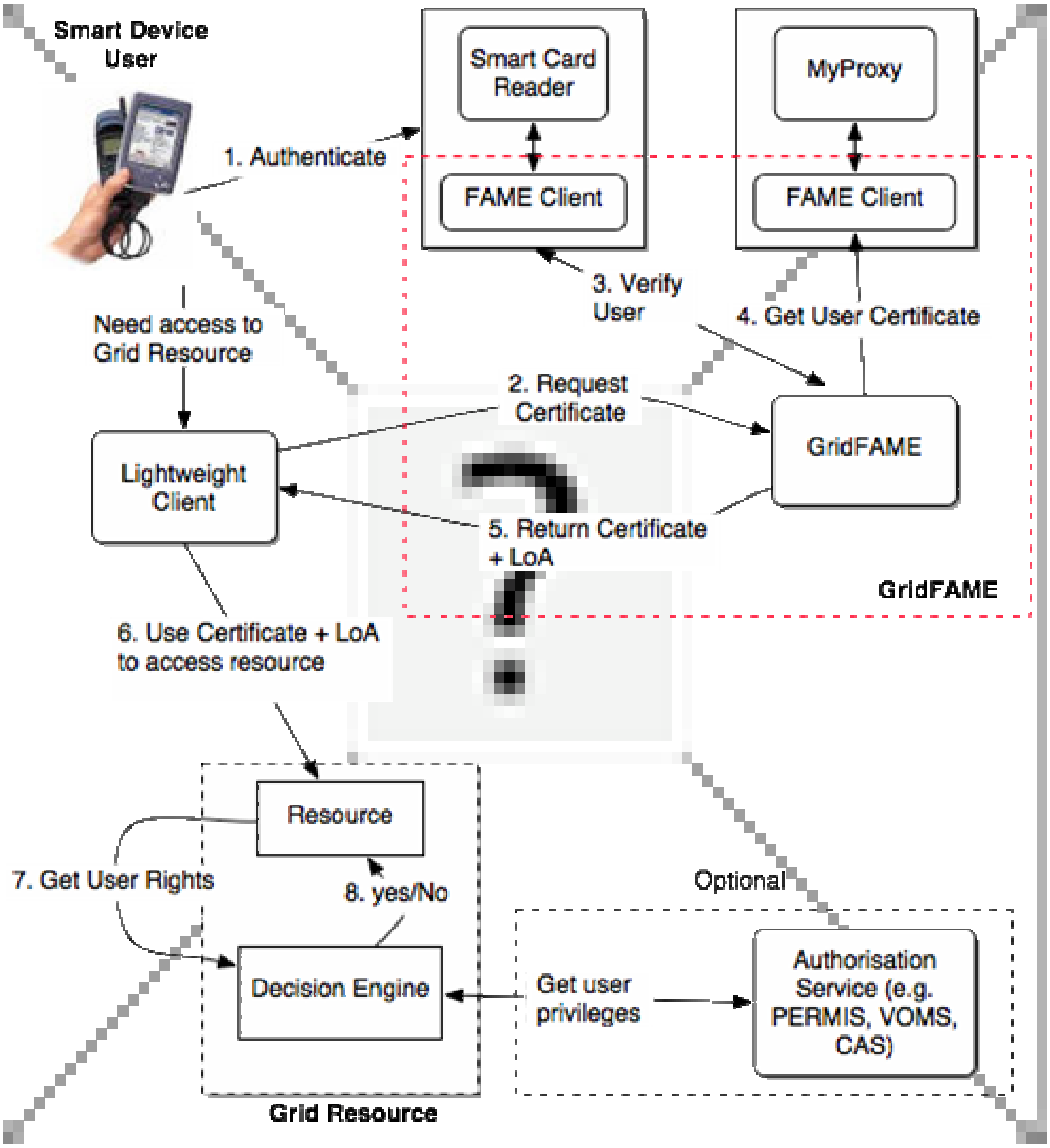


Figure 2. GridFAME

The GridFAME extension maps various authentication tokens to a username/password pair that can then be used to retrieve a proxy certificate from the MyProxy server. WSRF-Lite is used to create a lightweight wrapper for the original FAME middleware - presenting FAME as a Grid service through a well-defined Web Service Definition Language (WSDL).

Figure 2 shows an example operation of GridFAME where a user utilises the lightweight client installed on a PDA to access a Grid Resource using a smart card based authentication token.

The steps shown in Figure 2 are explained as follows:

User inserts his smart card into a card reader that is connected to a device that runs the FAME client.

User uses the lightweight client on the PDA to request a certificate from GridFAME. The request also contains a description and location of the target machine which authenticates the user's smart card. Figure 3 shows a snippet of the WSDL document for GridFAME.

GridFAME authenticates the user's smart card by communicating with the FAME client on the machine where the smart card is located.

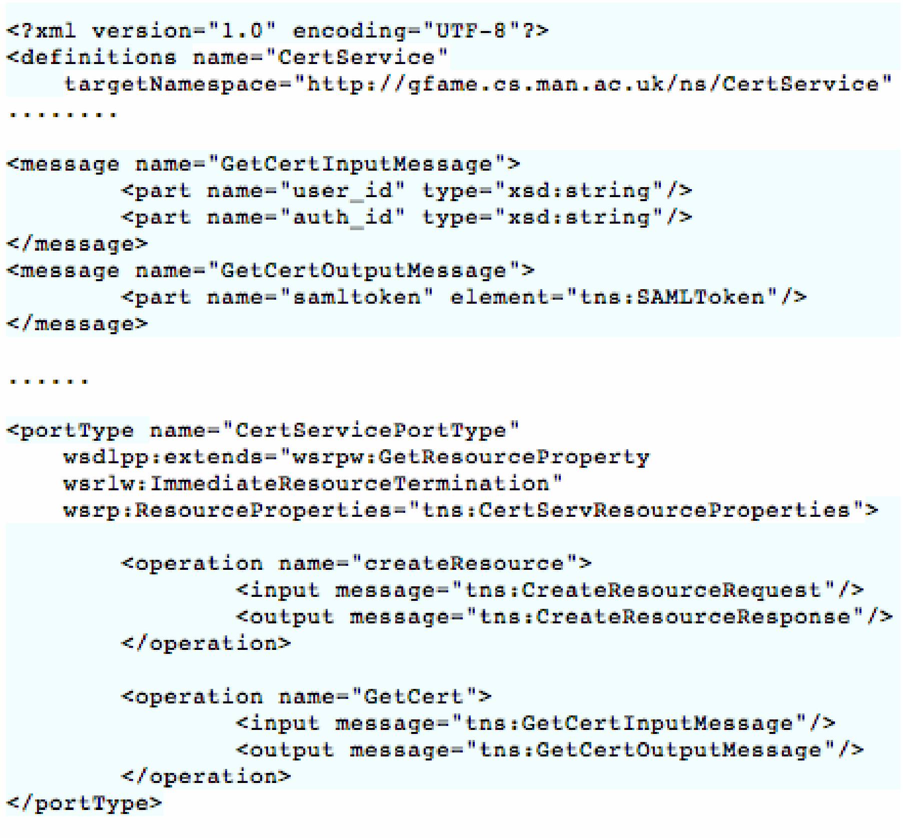
The user's verified identity is then mapped to a username/password pair stored on the GridFAME machine. The username/password pair is then used to request a proxy certificate from the MyProxy service.

The LoA can then be derived and returned to the user in the form of Security Assertion Markup Language (SAML) assertion. The SAML token will also contain a timestamp and details of the authentication (e.g. the location of the authentication token used). GridFAME then digitally signs the SAML token and the proxy certificate before sending them to the user.

User will then use the certificate and LoA to access a Grid resource.

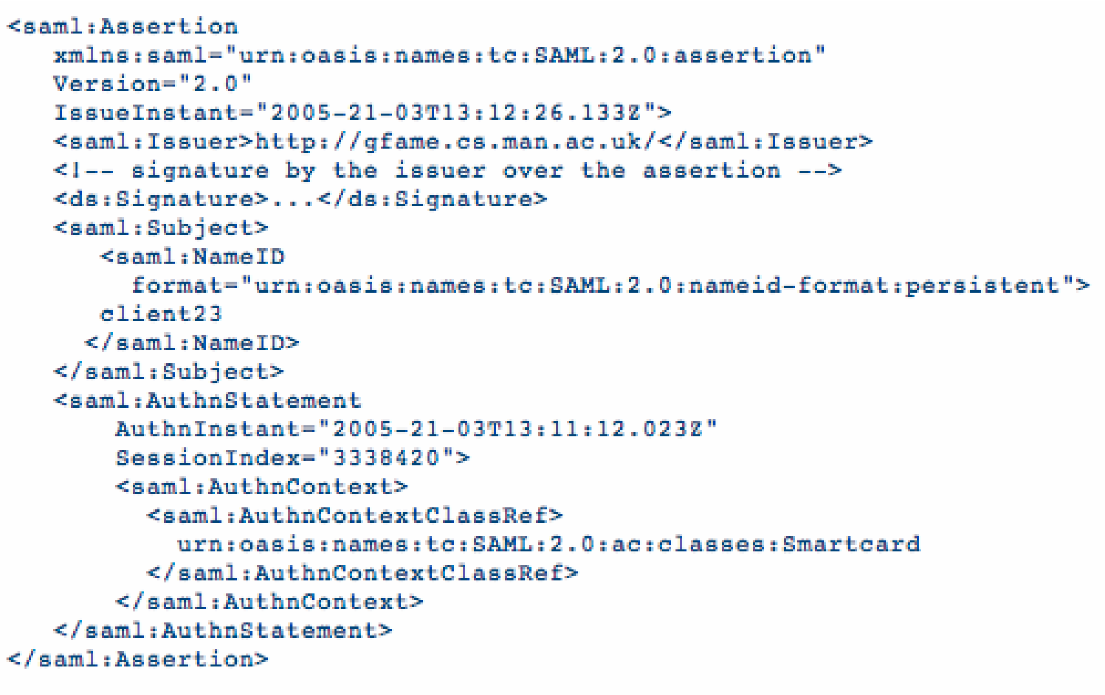
Grid Resource allows or denies the user access based on the proxy certificate and LoA presented by the user.

Optionally, the Grid Resource may present the user's proxy certificate and LoA to an Authorisation service to derive a user's access policy.



```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="CertService"
    targetNamespace="http://gfame.cs.man.ac.uk/ns/CertService"
........

<message name="GetCertInputMessage">
        <part name="user_id" type="xsd:string"/>
        <part name="auth_id" type="xsd:string"/>
</message>
<message name="GetCertOutputMessage">
        <part name="samltoken" element="tns:SAMLToken"/>
</message>

......

<portType name="CertServicePortType"
    wsdlpp:extends="wsrpw:GetResourceProperty
    wsrlw:ImmediateResourceTermination"
    wsrp:ResourceProperties="tns:CertServResourceProperties">

        <operation name="createResource">
                <input message="tns:CreateResourceRequest"/>
                <output message="tns:CreateResourceResponse"/>
        </operation>

        <operation name="GetCert">
                <input message="tns:GetCertInputMessage"/>
                <output message="tns:GetCertOutputMessage"/>
        </operation>
</portType>
```

Figure 3. Sample WSDL Document for Grid FAME



```
<saml:Assertion
   xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
   Version="2.0"
   IssueInstant="2005-21-03T13:12:26.133Z">
   <saml:Issuer>http://gfame.cs.man.ac.uk/</saml:Issuer>
   <!-- signature by the issuer over the assertion -->
   <ds:Signature>...</ds:Signature>
   <saml:Subject>
      <saml:NameID
        format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">
      client23
     </saml:NameID>
   </saml:Subject>
   <saml:AuthnStatement
       AuthnInstant="2005-21-03T13:11:12.023Z"
       SessionIndex="3338420">
       <saml:AuthnContext>
         <saml:AuthnContextClassRef>
           urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard
         </saml:AuthnContextClassRef>
       </saml:AuthnContext>
   </saml:AuthnStatement>
</saml:Assertion>
```

Figure 4. Sample SAML 2.0 Assertion for Smart Card Authentication

A Implementation

We have developed a portable, lightweight client implemented in C++, based on the gSOAP [21], OpenSSL [22] and QT [23] libraries and deployed here on a Linux based PDA (a SHARP/ Zaurus SL-5500). This lightweight client was then used to request a proxy certificate from GridFAME while the user authenticates with his or her smart card on an adjacent machine running the FAME client. Our hardware token implementation is based on the Cyberflex e-gate 32K Java-enabled smart card provided by Axalto [9].

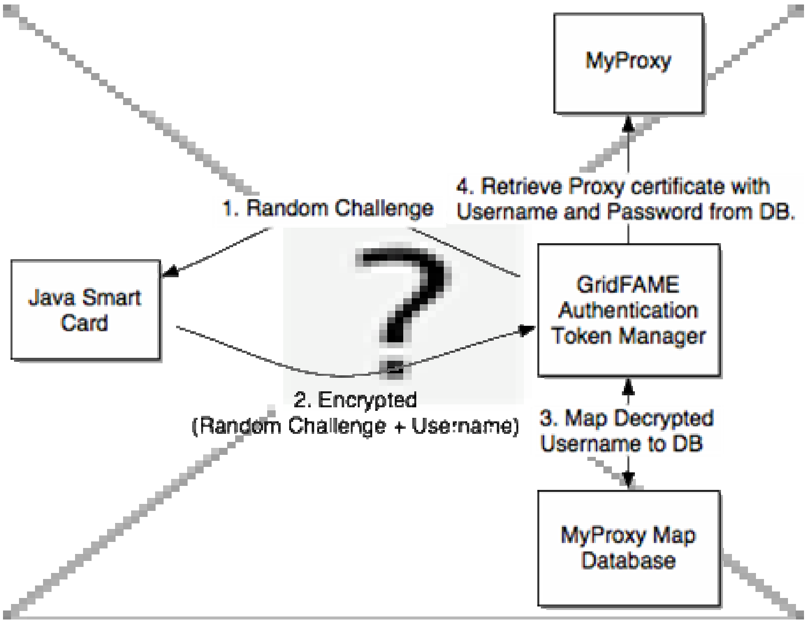

MyProxy
4. Retrieve Proxy certificate with Username and Password from DB.
1. Random Challenge
Java Smart Card
GridFAME Authentication Token Manager
2. Encrypted (Random Challenge + Username)
3. Map Decrypted Username to DB
MyProxy Map Database

Figure 5. Java Smart Card Authentication with GridFAME

Figure 3 illustrates the authentication process using a Java Smart Card. Each step is explained in detail as follows:

The GridFAME ATM issues a random challenge to the smart card via the FAME client that has direct access to the Smart Card Reader.
The private key stored inside the smart card is used to encrypt the random challenge together with a user ID that uniquely identifies the smart card user. The purpose of the random challenge is to prevent replay attacks.
The GridFAME ATM decrypts the random challenge and user ID using the public key of the Java Card. The decrypted user ID is than mapped to a database to obtain the user's corresponding username and password pair.
The retrieved username and password is then used by GridFAME to request a proxy certificate from MyProxy on behalf of the user.

IV. Conclusion and Further Work

This paper describes our work in the integration of FAME and the Grid authentication infrastructure. It is aimed at solving two main issues in Grid authentication context: the ability to support different authentication methods, and authentication strength linked access controls. We have tackled these two issues by building a middleware (FAME) Grid Service interface based on WSRF, which supports multiple authentication technologies and the derivation of authentication strength. Our project would prove to be useful in security sensitive Grid applications that rely on multi-factor authentication and fine grained access controls.

The draft on authentication strengths as defined by The National Institute of Standards and Technology (NIST) [19] will be used as a guideline for our current work to integrate authentication strength with third party authorisation services. We are also working on integrating FAME with existing Grid systems such as the MyGrid [24] project.

References

[1] Czajkowski, K., Ferguson, D., Foster, I., Frey, J., Graham, S., Sedukhin, I., Snelling, D., Tuecke, S., Vambenepe, W. 2004. The WS-Resource Framework. http://www-106.ibm.com/ developerworks/library/ws-resource/ws-wsrf.pdf

[2] Foster, I., Kesselman, C., Nick, J. and Tuecke, S. 2002. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. Draft of 6/22/02. http://www.gridforum.org/ogsiwg/drafts/ogsa_draft2.9_2002-06-22.pdf

[3] Karl Czajkowski, Don Ferguson, Ian Foster , Jeff Frey, Steve Graham, Tom Maguire, David Snelling Steve Tuecke, From Open Grid Services Infrastructure to WSResource Framework: Refactoring & Evolution . Version 1.1 03/05/2004. http://www.globus.org/wsrf/specs/ogsi_to_wsrf_1.0.pdf

[4] OGSI::Lite and WSRF::Lite - Perl Grid Services, http://www.globus.org/wsrf/specs/ogsi_to_wsrf_1.0.pdf .

[5] Reality Grid, http://www.realitygrid.org/ .

[6] TeraGyroid: Grid-based Lattice-Boltzmann simulations of Defect Dynamics in Amphiphilic Liquid Crystals, http://www.realitygrid.org/TeraGyroid.html

[7] Novotny, J.; Tuecke, J.; Welch, V., "An Online Credential Repository for the Grid: MyProxy", To appear in Proc. 10 th IEEE Symp. On High Performance Distributed Computing, 2001.

[8] MyProxy SASL Support, http://grid.ncsa.uiuc.edu/myproxy/sasl.html .

[9] N. Zhang, J. Chin, A. Rector, C. Goble, Y. Li, "Towards an Authentication Middleware to Support Ubiquitous Web Access", In proceedings of the 28th Annual International Computer Software & Applications Conference, September 2004.

[10] FAME - PERMIS - Flexible Access Middleware Extensions to PERMIS, http://www.cs.man.ac.uk/fame-permis/ .

[11] V. Welch, et al "Security for Grid Services", available at http://www.globus.org/Security/GSI3/GT3Security-HPDC.pdf, 10 Feb 2004.

[12] Butler, R.; Welch, V.; Engert, D.; Foster, I.; Tuecke, S.; Volmer, J.; Kesselman, C., "A national-scale authentication infrastructure", Computer, Volume: 33 Issue: 12, Dec. 2000, Page(s): 60 -66.

[13] A Roadmap for Integration of Grid Security with One-Time Passwords, http://www.ncsa.uiuc.edu/~jbasney/grid-otp.pdf, April 18, 2004.

[14] Samar, V., and Lai, C., Making Login Services Independent of Authentication Technologies. Third ACM Conference on Computer Communications and Security, 1996.

[15] X.509, http://www.ietf.org/html.charters/pkix-charter.html

[16] Welch, V., Kesselman, C., Mulmo, O., Pearlman, L., Tuecke, S., Gawor, J., Meder, S., and Siebenlist, F., X.509 Proxy Certificates for Dynamic Delegation, PKI R&D Workshop, 2004.

[17] PERMIS, http://sec.isi.salford.ac.uk/permis/.

[18] http://www.axalto.com/company/index.asp.

[19] W. E. Burr, et al, DRAFT Recommendation for Electronic

Authentication, NIST Special Publication 800-63, January, 2004.

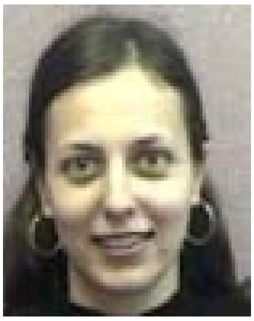
[20] Foster, I. Kesselman, C., and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International J. Supercomputer Applications, 15(3), 200-222, 2001.
[21] gSOAP: SOAP C++ Web Services. http://www.cs.fsu.edu/~engelen/soap.html
[22] OpenSSL: The Open Source Tookit for SSL/TLS. www.openssl.org
[23] Qt: The cross platform C++ GUI/API. http://www.trolltech.com/
[24] C.A. Goble, S. Pettifer, R. Stevens and C. Greenhalgh, Knowledge Integration: In silico Experiments in Bioinformatics in The Grid: Blueprint for a New Computing Infrastructure Second Edition eds. Ian Foster and Carl Kesselman, 2003, Morgan Kaufman, November 2003.

Jay Chin is a PhD student in the School of Computer Science at the University of Manchester. He received his MSc. in Advanced Computer Science from the Victoria University of Manchester in 2003. His current research interests are Security in Grid environments, distributed computing, and context-aware Grid Services.

Michael Parkin is a second-year PhD student in the School of Computer Science at the University of Manchester. He graduated with a BSc. in Physics from the University of Liecester in 1993 and spent 8 years in industry before returning to academia to gain a MSc. in Computer Science from the Victoria University of Manchester in 2003. His current research interests are the lightweight collaborative Grid environments, their formation, use, security models and scalability, and the development of extremely lightweight Grid clients.

Ning Zhang works as a lecturer at the Department of Computer Science,University of Manchester, UK. She received her PhD degree in Electronic Engineering from the University of Kent at Canterbury, UK. Her research interests include computer networks, mobile computing, and information and e-commerce security. She is supervising a number of research projects on these subjects, which are funded by various funding sources, including the UK Engineering and Physical Sciences Research Council (EPSRC), theDepartment of Trade and Industry (DTI), and the Joint Information Systems

Committee (JISC).



Aleksandra Nenadic is Research Associate at the Information Management Group at the School of Computer Science, University of Manchester. She received BSc in Computer Science from the Faculty of Mathematics, University of Belgrade in 1999, and is currently a PhD candidate at the School of Computer Science, University of Manchester, expected to graduate in September 2005. Her main research interest include design and formal verification of protocols for securing e-commerce transactions. She is also investigating solutions for secure authentication of parties over the Internet.

John Brooke is the co-Director of the North-West eScience Centre (ESNW) and an Honorary Lecturer in the School of Computer Science. His main research interests are in grid computing, visualization and nonlinear problems in astrophysics. He is a co-Investigator on the EPSRC RealityGrid project which pioneered the use of computational steering in very large simulations. This work won major awards at international supercomputing conferences in 2003 and 2004. ESNW acts as a focus to promote interdisciplinary collaboration between computational scientists, computer scientists and visualization experts.