GridFAME: Flexible Authentication Middleware Extension for Grids

J.S. Chin\textsuperscript{1}, M. Parkin\textsuperscript{2}, N. Zhang\textsuperscript{1}, A. Nenadic\textsuperscript{1}, J.M. Brooke\textsuperscript{2}

School of Computer Science
The University of Manchester
\textsuperscript{1}\{jchin, nzhang, nenadic\}@cs.man.ac.uk
\textsuperscript{2}\{m.parkin, j.m.brooke\}@manchester.ac.uk

Abstract

Current Grid software lacks support for alternative authentication techniques other than using 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 an authentication strength linked access control solution to Grid resources. The solution, called Flexible Authentication Middleware Extension for Grids (GridFAME), is based on the Web Services Resource Framework (WSRF).

1 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)\cite{20} where users are able to gain access to resources across multiple domains from anywhere on the Internet. The existing middleware \cite{11}\cite{13} developed by the Grid community mainly relies on digital certificates for authentication purposes and does not support more advanced authentication techniques such as smart cards, biometrics, smart tokens, etc.

In this Grid environment, it is expected that scientists or researchers gain access to the Grid resources not only through wired devices such as desktop machines and home computers, but also wireless devices such as laptops, PDAs and smart phones. This scenario poses the problems of supporting user mobility and secure management of private keys associated to the certificates,
as requiring a digital certificate on each and every device that is used to access
the Grid and exposing the private key on multiple locations 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 [14] 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 [7] on-line credential repository is used to provide
access to Grid resources via Grid portals. However, one of the disadvantages
of using Grid portals is that it only allows access 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.

Another related project, GridLogon [12], uses Pluggable Authentication
Module (PAM) library, which allows a user to authenticate with a variety
of mechanisms that is supported by PAM. Our work presented in this paper
aims to not only support a variety of mechanisms for authentication, but also
to derive authentication strength, or the Level of Assurance (LoA), from the
authentication method(s) used. Then, any authorisation engine can make use
of the LoA to derive an appropriate access policy. 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 devices that lack a Java
runtime environment.

2 Background

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

2.1 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
[15]. 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 a 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.

2.2 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].

2.3 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-factor ing and extension done on the OGSI framework to make it more compliant with the Web Services Standards [3]. It is also important to note that the next generation of the Globus toolkit (version 4) will include a complete implementation of WSRF.

WSRF is a set of specifications that allows the modelling and creation of stateful resources using Web services [3]. 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 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 is also interoperable with other Java and .NET WSRF implementations.

2.4 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, deriving an authentication strength based upon the authentication method used, and feeding this strength into the Privilege and Role Management Infrastructure Standards Validation (PERMIS) authorisation decision engine.

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.

Fig. 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.

3 GridFAME

As we 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 fine 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 enough to be deployed on PDAs or smart phones.

We address these problems by introducing a Grid Flexible Access Mid-
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.

The GridFAME extension maps various authentication tokens to a user-name/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).

Fig. 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 Fig. 2 are explained as follows:

1. User inserts his smart card into a card reader that is connected to a device that runs the FAME client.
1. The GridFAME ATM issues a random challenge to the smart card via the FAME client that has directly access to the Smart Card Reader.
2. 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.

3. The GridFAME ATM decrypts the random challenge and user ID using the public key of the Java Card. The decrypted user ID is then mapped to a database to obtain the user’s corresponding username and password pair.

4. The retrieved username and password is then used by GridFAME to request a proxy certificate from MyProxy on behalf of the user.

5 Conclusion and Future Work

This paper describes our work in the integration of our FAME idea 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. This middleware can then be used by Grid users for authentication instead of MyProxy.

Our future work will be mainly focused on the linkage of authentication strength with third party authorisation services. The draft on authentication
strengths defined by The National Institute of Standards and Technology (NIST) [19] will be used as a guideline for our work.

References


[23] Qt: The cross platform C++ GUI/API. http://www.trolltech.com/