NERC Data Grid Security

Computational Viewpoint

Version 0.1

Document Log

Version Number	Date	Comment
0.1	05/02/08	First Draft

Contents

1. REFERENCES

1. RM-ODP (The Reference Model for Open Distributed Processing) - http://www.rm-odp.net/

2. MyProxy Credential Management Service http://myproxy.ncsa.uiuc.edu/

3. Lawrence, B.N., P. Kershaw and J. Blower, 2007: Practical Access Control with NDG-Security. Submitted and accepted, UK e-Science AHM 2007.

4. NDG Security - Passing authentication details across Domains, P J Kershaw, 06/04/2006

5. Securing an Application with NDG Security B N Lawrence, Ag Stephens, P J Kershaw, Version 0.6 28/02/2006

6. NERC Data Grid Security - Cross Domain Cookie Interface, P J Kershaw, version 0.2, 02/08/2006

7. Bennett, Neil, Ray Cramer, Glen Drinkwater, Marta Gutierrez, Phil Kershaw, Kerstin Kleese van Dam, Siva Kondapalli, Susan Latham, Bryan Lawrence, Roy Lowry, Ananta Manandhar, Kevin O'Neill, Ag Stephens, Shoaib Sufi, Andrew Woolf, 2005: Nerc Datagrid Authorisation Architecture. Proceedings of the UK e-Science AHM 2005, Simon J Cox and David W Walker (ed), ISBN 1-904425-53-4.

8. NERC Data Grid Security Authentication and Authorisation Across Different Data Centres, P J Kershaw, version 0.2, 18/11/2005

9. NERC Data Grid Security - Design Options for Authentication with MyProxy, P J Kershaw, version 0.1, 02/08/2005

10. NERC DataGrid Architecture: Computational viewpoint Phase One, version 0.5.2, 25/09/2004

11. NERC DataGrid Architecture: Information Viewpoint Phase One, version 0.6, 30/11/2004

12. NERC DataGrid Architecture: Enterprise Viewpoint Phase One, version 0.6, 30/11/2004

13. Implementation of NERC Data Grid Security, Neil Bennett, 2004

14. Access Control Issues for the NERC DataGrid, The NDG Team 02/03/2004, Revised, B N Lawrence 25/03/2004

2.INTRODUCTION

The intention of this document is to bring together the previous NDG Security design documentation and technical notes into one place to enable newcomers to the project to obtain an overview of the NDG Security architecture and an insight into how the architecture has evolved to its current state. It is brought together within the existing RM-ODP based framework used within the NDG1 project: the Enterprise, Information and Computational Viewpoint documents. This document is intended to supersede the latter draft document for the specific case of NDG Security.

NDG Security can be divided into these key areas: authentication and management of user credentials, mechanism for allocation of user attributes and access control. Orthogonal to this NDG Security envisages separate profiles for browser based and application client usage.

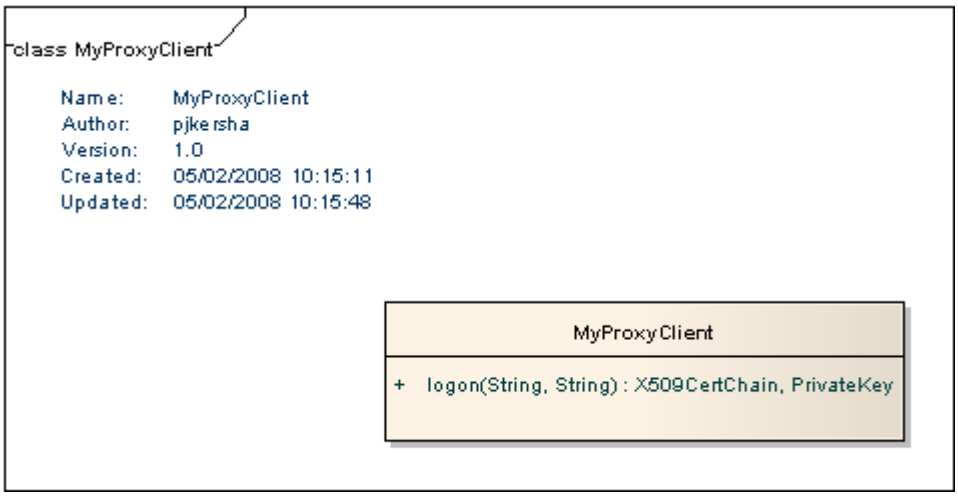
The browser based profile includes the mechanisms for secured access to resources via a web browser. The application client profile is a more generic case which could encompass client scripting for secured download of data perhaps for regular data retrievals; or the retrieval of data within a separate application environment.

NDG Security's web service based architecture enables flexibility and variation in how and which components are deployed.

3.AUTHENTICATION

3.1User Certificates and Username/password based authentication

The use of individual user certificates enable users to be authenticated in security transactions. However, NDG partners did not wish to change their existing authentication systems based on username/password. MyProxy from the Globus Toolkit enables authentication in this way whilst at the same time making individual user certificates available to security services to broker access to secured resources. The diagram below shows the NDG Security client interface to MyProxy. The logon method requests MyProxy to issue a user certificate valid for the duration of a session.



4. ATTRIBUTE ALLOCATION

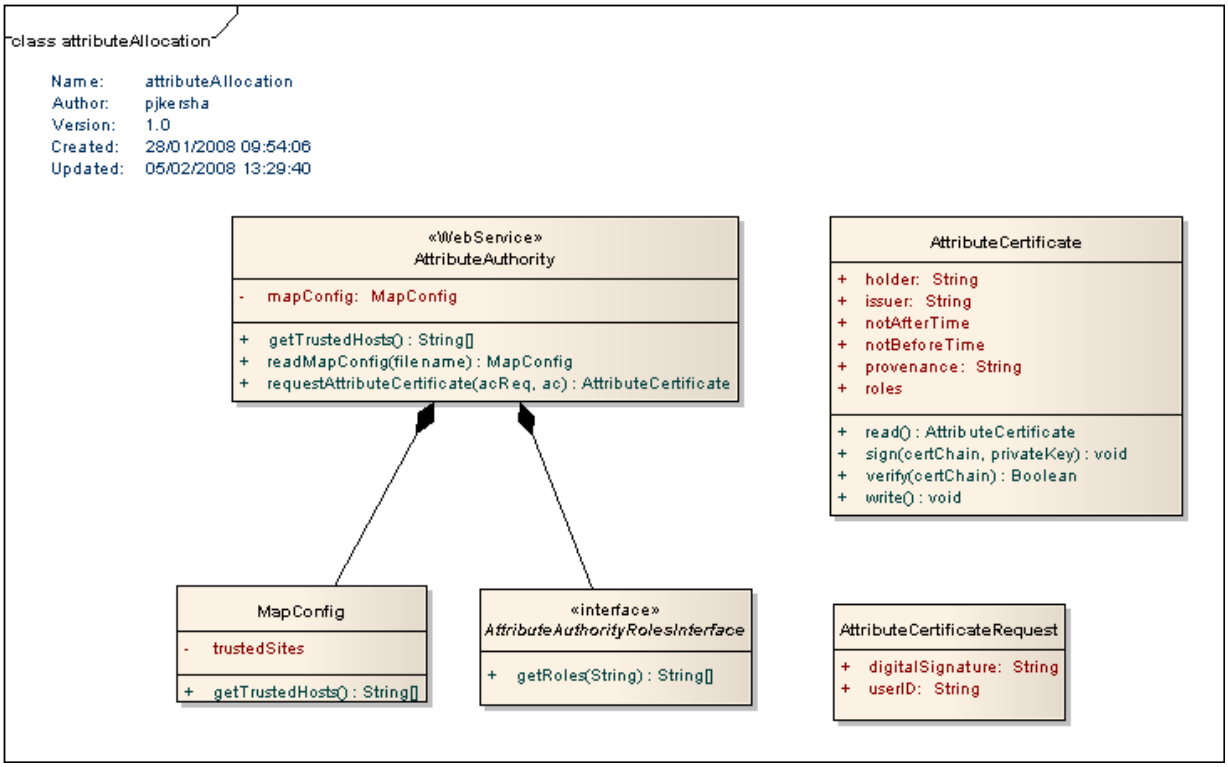
4.1 Roles, Attributes and Access Control

NDG Security uses RBAC (Role based Access Control). Secured resources have roles associated with them which govern access. Authenticated users are delegated roles. To gain access to a resource they must have the required role. Roles or *attributes* are used interchangeably within NDG Security although it is recognised that in some security systems they have distinct meanings.

4.2 Attribute Authority and Attribute Certificate

The roles a user is entitled to are issued by an Attribute Authority service. On receipt of a digitally signed request identifying a user, an Attribute Certificate is issued. The Attribute Certificate is digitally signed by the Attribute Authority that issued it. It has a time bound validity.

The diagram below shows the AttributeCertificate, AttributeCertificateRequest and AttributeAuthority class definitions:



4.3 Attribute Authority Roles Interface

A key requirement of NDG Security is the ability to easily integrate into partner organisation's existing security infrastructures. The Attribute Authority Roles Interface is an abstract interface class which enables the Attribute Authority to make a call out to a site's local security infrastructure matching the user ID of a request to the roles to which that user is entitled. For example, a site may have a user database containing tables for roles and user Ids. The specialisation of the Attribute Authority Roles Interface for this site could be a database query to extract this information.

4.4 Role Mapping Configuration

The Attribute Authority also contains a MapConfig instance. The MapConfig class describes the role mapping relationship between the owner institution's Attribute Authority and its trusted NDG partner organisations. Role mapping is negotiated bilaterally between parties. This negotiation is a human to human one outside the scope of the functionality expressed in the system architecture.

The exact role mappings are expressed in a role mapping configuration. This could be stored in a file or database tables. Each site which the Attribute Authority trusts has a series of its roles with agreed mappings to which the Attribute Authority's host organisation supports.

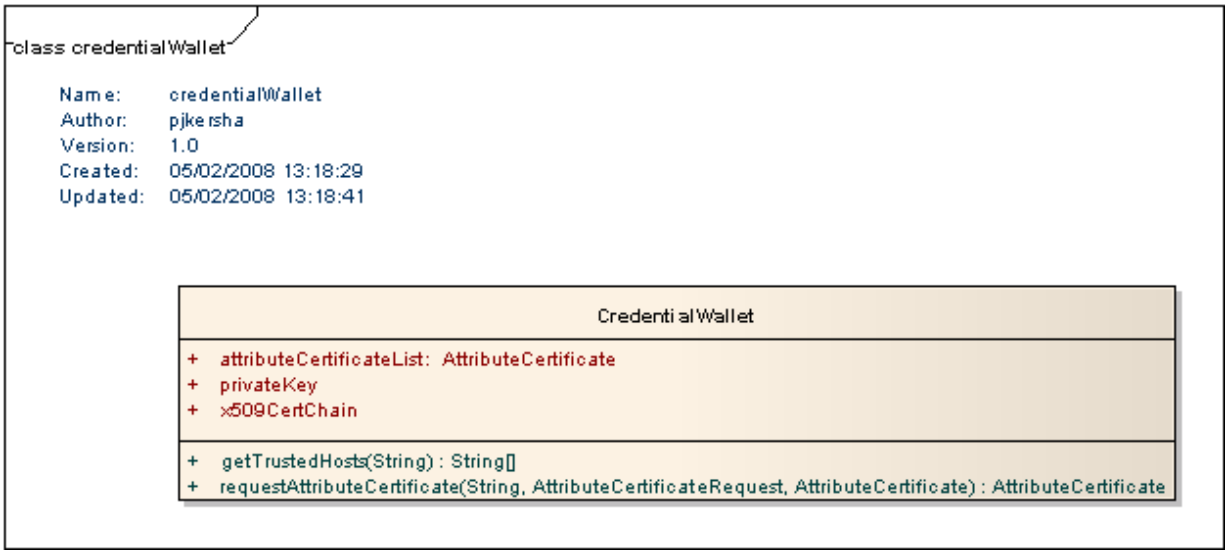
4.5Attribute Certificate Request from a Client from a Trusted Site

Role mapping is applied by the Attribute Authority on receipt of a valid Attribute Certificate issued by a trusted Attribute Authority. The roles contained in the certificate are mapped and a knew Attribute Certificate is issued containing these mapped roles. In this way the client can obtain access to resources at a trusted site by submitting the new *mapped* Attribute Certificate. Only one level of mapping is permitted. Attribute Certificates have a Provenance flag to indicate to a consumer whether the certificate is *original* or *mapped*.

5. USER CREDENTIAL MANAGEMENT

5.1 The CredentialWallet

The CredentialWallet class is a container for user session credentials. It contains their X.509 certificate and private key obtained at login and it also caches Attribute Certificates obtained from Attribute Authorities. This enables a previously obtained Attribute Certificate to be used for multiple access requests to a resource without repeated calls to an Attribute Authority.



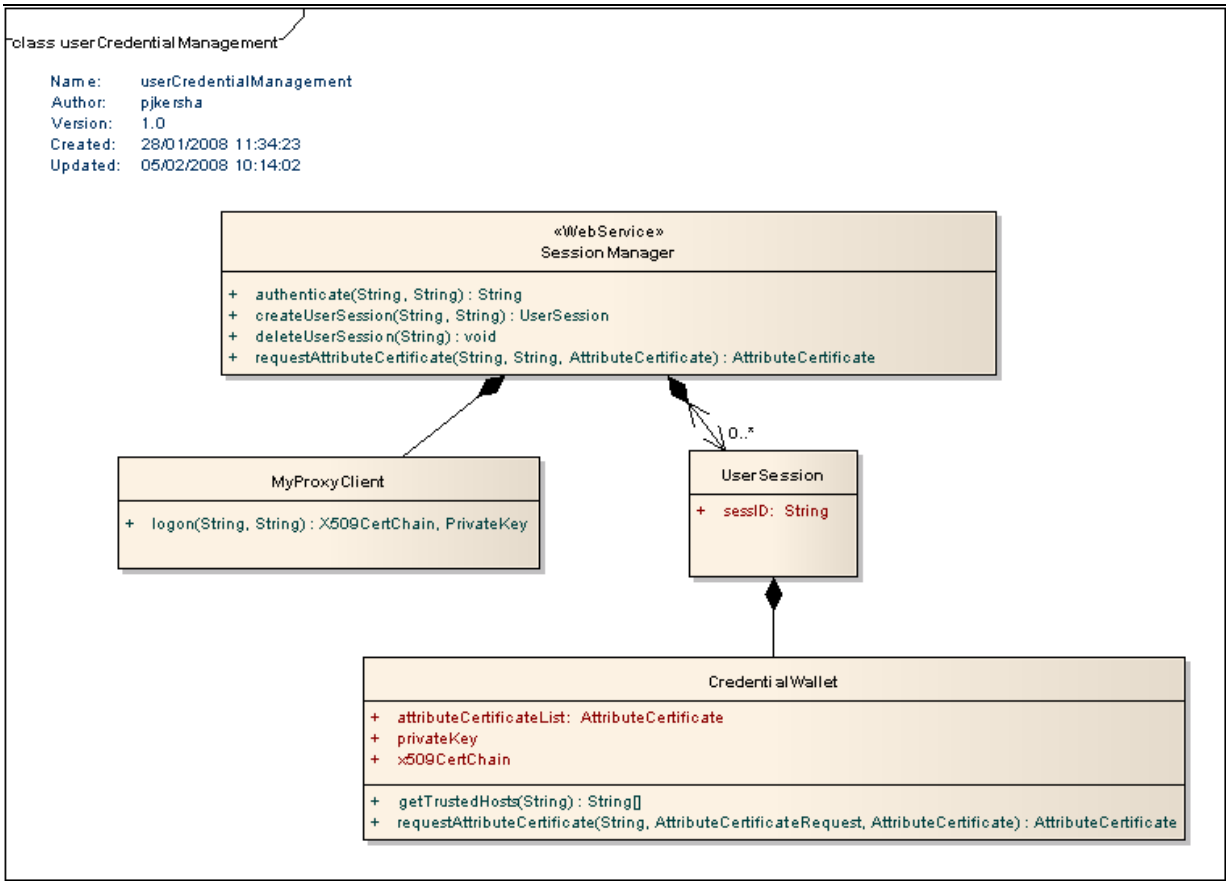
The requestAttributeCertificate method is used to make a request to an Attribute Authority for an Attribute Certificate. The getTrustedHosts method enables the CredentialWallet to query which organisations a given Attribute Authority has trust relationships with in its role mapping.

The CredentialWallet acts as a client interface to broker the required attributes needed for a given resource access request.

5.2 The Session Manager Service

The Session Manager web service manages the security credential of user sessions. Rather a client managing it's own CredentialWallet, it can delegate this task to a remote Session Manager Service.

The Session Manager managers user authentication against a MyProxy service and the subsequent creation and management of a user session. A UserSession is itself a container for a CredentialWallet. The Session Manager makes use of the CredentialWallet interface to manage and broker the attributes required for access to secured resources.


class userCredentialManagement
Name: userCredentialManagement
Author: pjkersha
Version: 1.0
Created: 28/01/2008 11:34:23
Updated: 05/02/2008 10:14:02
«WebService»
Session Manager
+ authenticate(String, String) : String
+ createUserSession(String, String) : UserSession
+ deleteUserSession(String) : void
+ requestAttributeCertificate(String, String, AttributeCertificate) : AttributeCertificate
0..*
MyProxyClient
+ logon(String, String) : X509CertChain, PrivateKey
UserSession
+ sessID: String
CredentialWallet
+ attributeCertificateList: AttributeCertificate
+ privateKey
+ x509CertChain
+ getTrustedHosts(String) : String[]
+ requestAttributeCertificate(String, AttributeCertificateRequest, AttributeCertificate) : AttributeCertificate

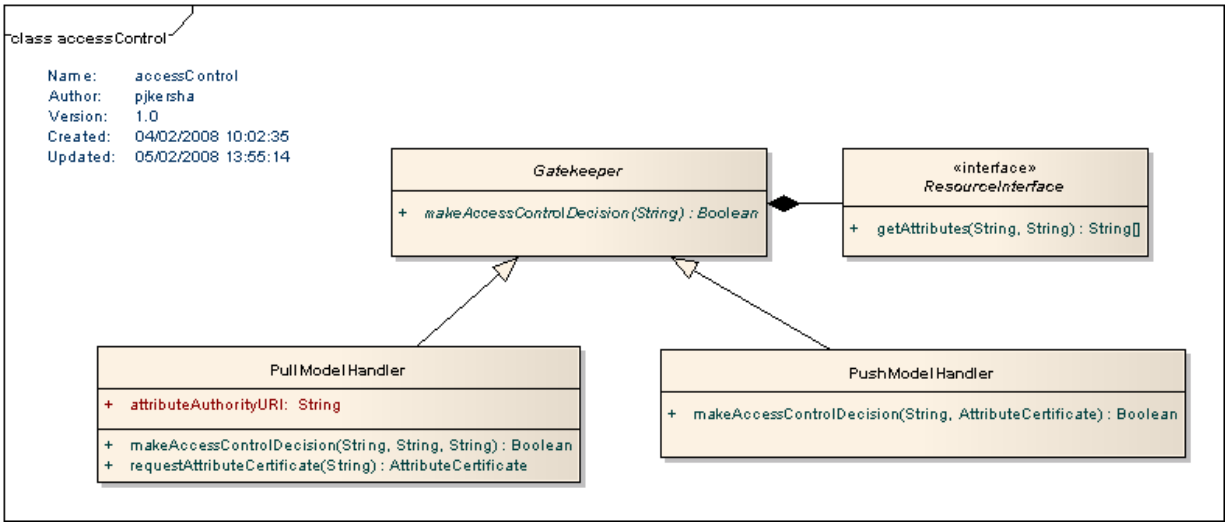
6. ACCESS CONTROL

Access to a secured resource is managed by a Gatekeeper. An access control decision is made based on the attributes which govern access to a resource and the attributes to which a user is entitled contained in an Attribute Certificate.

Pull and push models for Attribute Certificate access are envisaged.

- ➔ In a Pull model, the Gatekeeper is passed the SessionID and URI of the Session Manager service where that session is held. The Gatekeeper trusted by the Session Manager, can ask to the Session Manager to broker an Attribute Certificate on behalf of the user and return it to the Gatekeeper.

- ➔ In a Push model, the client to the Gatekeeper acting on behalf of the user, has obtained an Attribute Certificate. This is passed to the Gatekeeper so that it can make an access control decision.



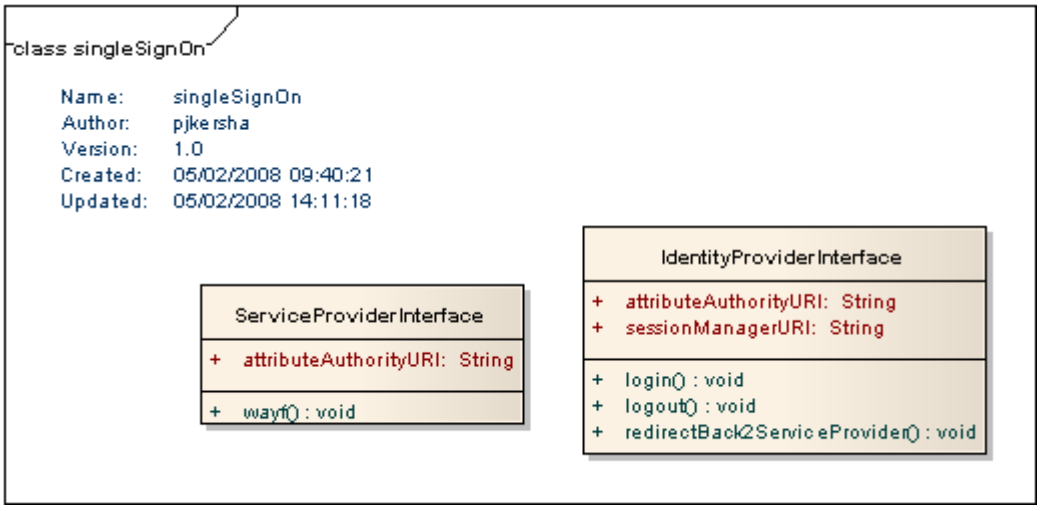
In the class model above, Pull and Push model handler are shown as specialisations of an abstract Gatekeeper super class.

6.1 Resource Interface

The ResourceInterface class encapsulates the association of resource to attributes which control access to that resource. An organisation deploying a Gatekeeper(s) can tailor this interface according to their existing site security infrastructure.

7.SINGLE SIGN ON

This is an extension of the functionality for authentication to enable cross-organisational authentication for a browser based profile. It builds upon the services previously described.



It consists of Service Provider and Identity Provider interfaces:

- ➔ The Service Provider Interface enables a user to select their home site login selecting from a list of Identity Provider sites which the Service Provider trusts. It does this by calling its Attribute Authority service with getTrustedHostInfo to find out the list of trusted sites. This analogous to the Shibboleth WAYF (Where Are You From) service.

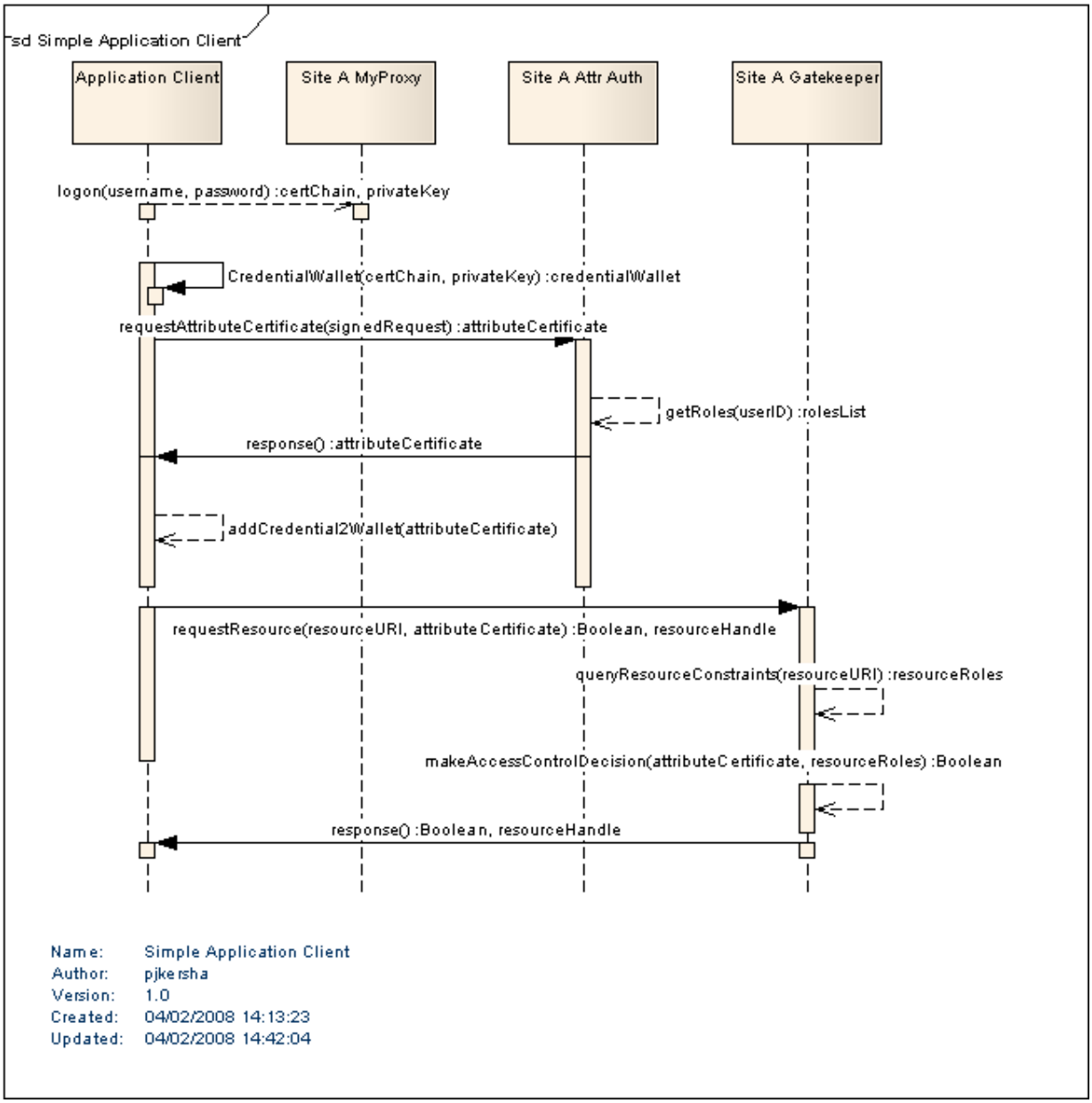
- ➔ The Identity Provider enables users to login on receipt of a request from a Service Provider WAYF. The login method authenticates the user using its Session Manager service. The redirectBack2ServiceProvider redirects the users browser back to the Service Provider page from which the WAYF call was initiated. The Identity Provider must be able to authenticate the Service Provider making the redirect request.

8. INTERFACE INTERACTIONS

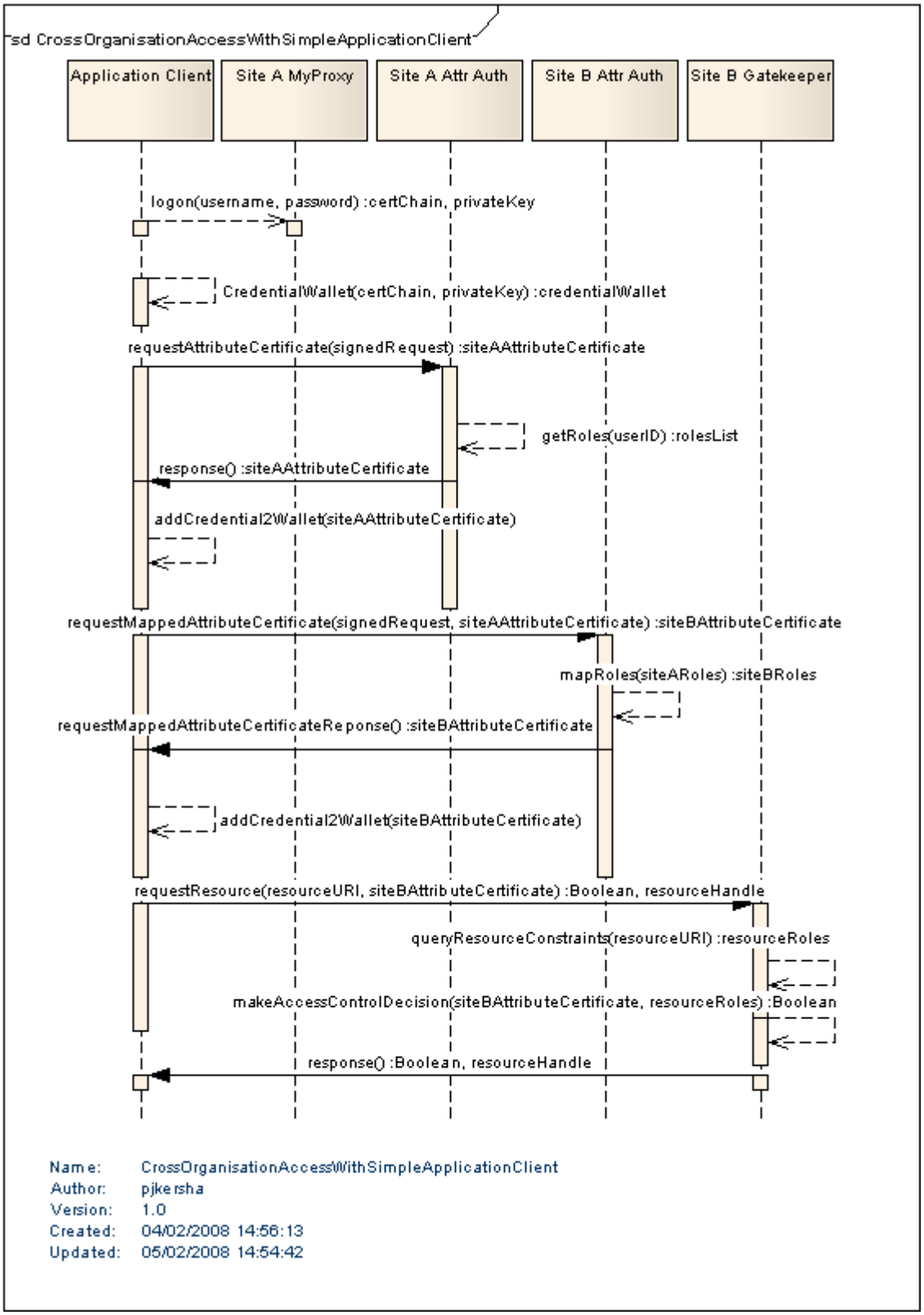
This section describes the interactions between services required for secured access to a resource.

8.1 Simple Application Client

This is the most simple example of access. Fundamental to access are a means of authentication (MyProxy), attribute retrieval (Attribute Authority) and access control (Gatekeeper).

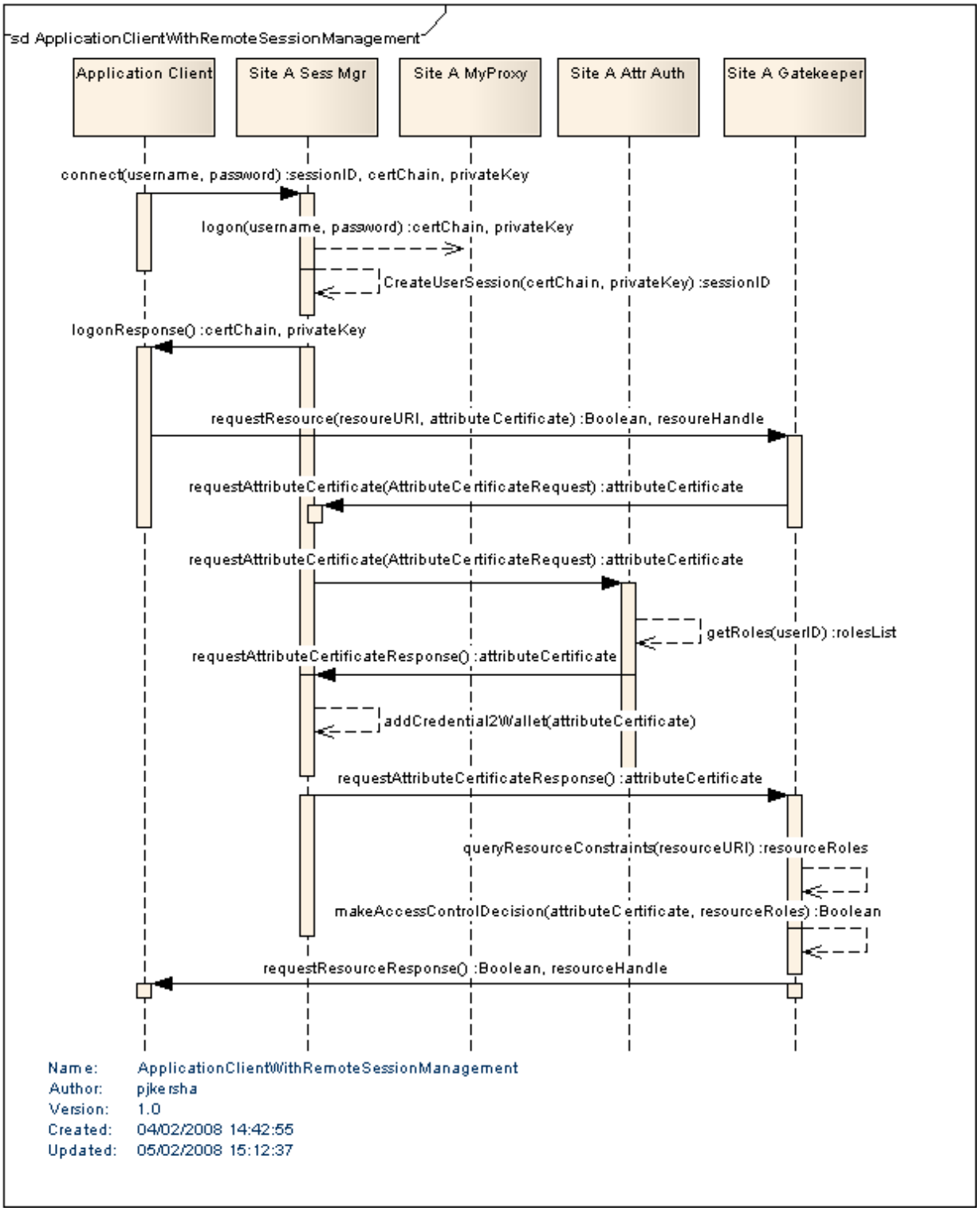


8.2 Cross-Organisation Access with a Simple Application Client



The client authenticates, obtains an Attribute Certificate from its home Attribute Authority and then uses this to obtain a mapped Attribute Certificate from the Service Provider's (Site B's) Attribute Authority. This mapped Attribute Certificate may be accepted by the Service Provider's Gatekeeper. Note that the Gatekeeper in this case conforms to a push model.

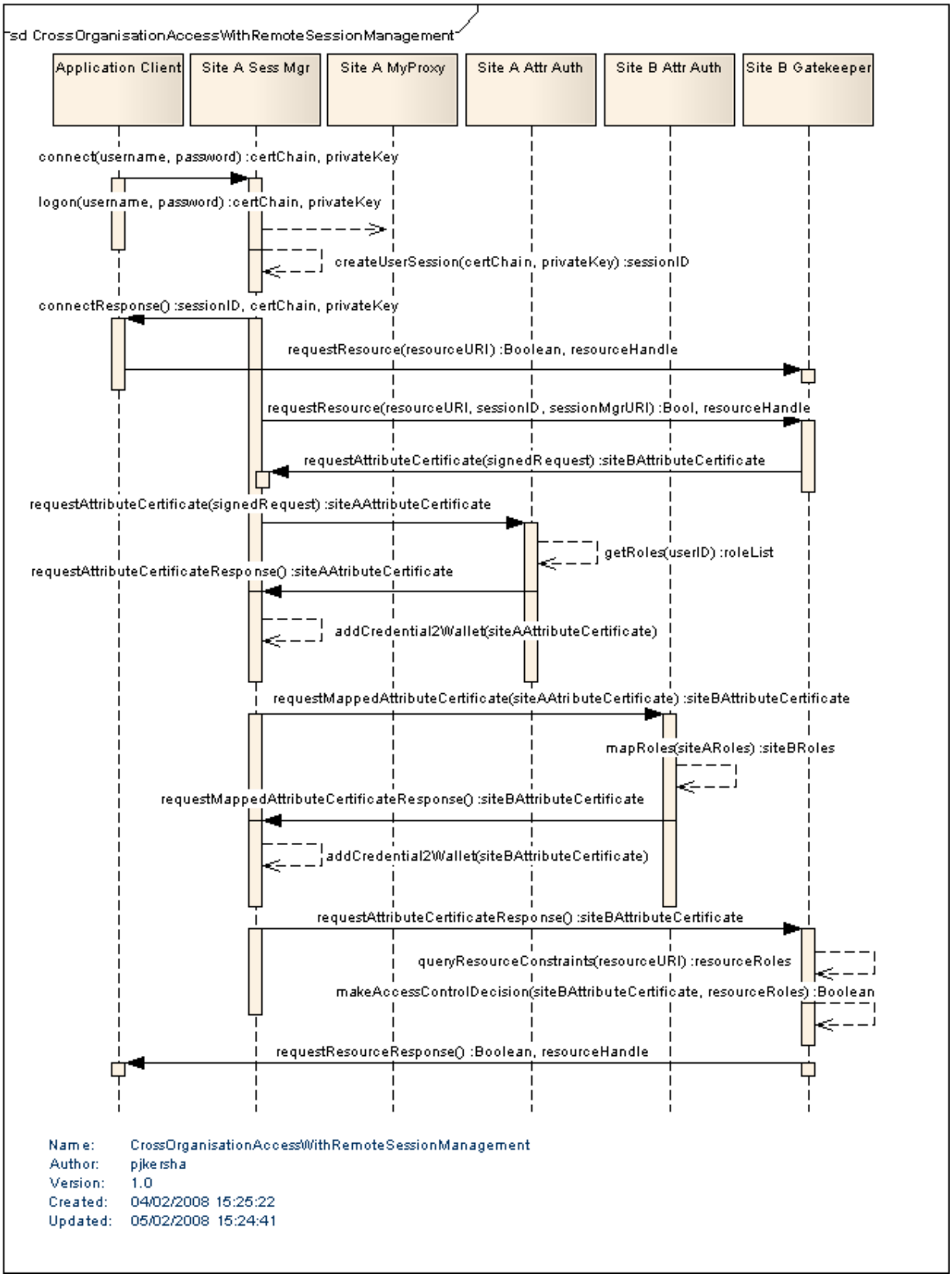
8.3 Application Client using Remote Session Management



This illustrates the use of a Session Manager service to manage user credentials on behalf of a client. Using a Session Manager service enables the Gatekeeper to adopt a pull model, pulling the required Attribute Certificate from the user's CredentialWallet held by the Session Manager. In this way the steps needed by the client are reduced since the Session Manager and Gatekeeper manage the brokering of attributes on its behalf. This is applicable to a browser based profile where the limited

capabilities of a browser and security considerations place restriction on how and where credentials are managed.

8.4 Cross Organisation Access with Remote Session Management



In this case, the user is not registered with the Service Provider (Site B) so the Session Manager must in addition broker a mapped Attribute Certificate from the Service Provider's Attribute Authority.

8.5 Single Sign On

This diagram shows the steps for login using the Single Sign On interface for use with a browser based profile. The user is browsing Site B but is registered with Site A. The trigger could be user request for a secured link or user selection of a login link. The sequence results in the user authenticated at Site B with credentials from Site A.