

Information & Communication Technologies
Technology Operations

www.imperial.ac.uk/ict

ShibboLEAP Final Report

Mr Neil McLennan

19 May 2006

Copyright Notice

Imperial College of Science, Technology and Medicine
Information & Communication technologies

ShibboLEAP Final Report

© 2006 Neil McLennan
n.mclennan@imperial.ac.uk

This publication may be distributed freely in its entirety and in its original form without the consent of the copyright owner.

Use of this material in any other published works must be appropriately referenced, and, if necessary, permission sought from the copyright owner.

Information & Communication Technologies
Imperial College London
Mechanical Engineering Building
Exhibition Road
London SW3 6AZ

19 May 2006

www.imperial.ac.uk/ict

Background

Imperial College London

Consistently rated in the top three UK university institutions, Imperial College London is a world leading science-based university whose reputation for excellence in teaching and research attracts students (14,000) and staff (8,000) of the highest international quality. Innovative research at the College explores the interface between science, medicine, engineering and management and delivers practical solutions that enhance the quality of life and the environment - underpinned by a dynamic enterprise culture. The college is spread over 7 campuses based in central and west London and 2 other campuses in south-east England.

Shibboleth

Shibboleth is standards based protocol for exchanging information about users through the web written by MACE (Middleware Architecture Committee for Education) facilitating web Single Sign On (SSO) across or within organizational boundaries. It allows sites to make informed authorisation decisions for individual access of protected online resources in a privacy-preserving manner.

The Shibboleth software is open source middleware which has two parts. The service provider software (SP) at the site that controls access to the resources a user wishes to use (also known as the "target system"), and identity provider software (IdP) in a site which knows about the user. Communication between the two systems uses SAML, an XML standard framework for exchanging security information.

Shibboleth decouples the management of directories by the IdP from access control to online resources by service providers so that different kinds of directory can be used to provide information to the access control software. This document refers to the implementation of an Identity provider

When a user tries to access a Shibboleth protected resource they are redirected, to their home institution. After successful authentication at their institution, a one-time "handle" or session identifier is generated for this user session which is passed back to resource service provider allowing access to the resource.

ShibboLEAP and Imperial

Imperial is participating in ShibboLEAP (<http://www.angel.ac.uk/ShibboLEAP/>), the largest JISC funded Early Adopters project exploring Shibboleth, in which 7 SHERPA-LEAP (<http://www.sherpa-leap.ac.uk/>) partners will each implement Shibboleth-based access management for their institutional repositories based on Eprint.org software by April 2006. The partners are Birkbeck College, Imperial College London, Kings College London, Royal Holloway, School of Oriental & African Studies and University College London. The lasting benefit of the project will be that these institutions will be newly enabled as Shibboleth Identity Providers allowing staff and students' access to subscribed electronic resources.

The library also subscribes to the Athens service which required another set of login credentials to be maintained, until 2005 when we implemented the proprietary Athens Devolved Authentication which works in a similar way to Shibboleth.

By implementing a Shibboleth Identity Provider the college will be able to remove the need to maintain Athens DA, using the Athens-Shibboleth gateway instead. We will also be able to remove the administrative overhead caused by having to maintain user's credentials in other systems such as our Eprints system which are becoming Shibboleth enabled. Our staff and students will also benefit from having a trusted authentication gateway to these online resources using their standard college username and password reducing the need to remember other credentials.

User Information and Account Registration

The college has a custom built user management system (Son of Validate) where all user properties, permissions and account configuration information is stored and managed through a custom user interface which displays each user's setup. This system and its' predecessor (Validate) have been used to facilitate SSO (single sign on) internally in college for many years and therefore each college member has one username and password to access college systems.

Although this system is authoritative for user account information, user details e.g. name, department, telephone number are non-authoritative so this information is updated from the relevant authoritative source e.g. Human Resource, Registry, Telephone systems etc. As this information is applied rules are fired to automatically update users account information adding and removing permissions where appropriate. New user accounts are automatically generated using these sources of information. The information required for each college system e.g. mail-relays, directories, home directory servers, WebCT etc is then either pulled or pushed out whereupon scripts automatically update the systems locally. By doing this we keep all systems in college in sync and if a new system is introduced we can push the required data to it and can quickly build scripts to process the information.

Shibboleth Implementation Decisions

Information Repository for Shibboleth

The college already runs three different resilient LDAP systems, Microsoft Active Directory, Oracle Internet Directory and OpenLDAP. Each of these systems contains every account in college, being fed from our user registration system, so it was obvious that one of these should be used to provide the attribute information for Shibboleth.

1) Microsoft Active Directory – This is college's core directory system and is normally the directory of choice because of the resilient implementation and the high level of expert support. However at the time of our decision we were concerned about the level of support from Microsoft if the schema was significantly altered given the system's importance to college it was decided that it could not be used.

2) Oracle Internet Directory – Support is limited within college.

3) OpenLDAP – High level of support within college and it allows a good degree of flexibility by allowing users credentials to be stored in under multiple organisational units. It is already setup to provide pass through authentication to Active Directory. Unlike Active Directory the organisation of the system is not highly tied in with other products in use e.g. Exchange.

We finally decided on OpenLDAP and a brief survey of the web showed that many there was a considerable knowledge base about Shibboleth working with OpenLDAP.

Unique User Key for Shibboleth

The college uses many unique id's, College Person ID, login, email address, UNIX uid, security card number etc and the majority are stored in the user registration system. Out of these the login name is the unique id which we have greatest control of and that changes least over time. We did have concerns about exposing the login name externally from college however our IT security team confirmed that it would be allowed.

Attributes

The attributes for ShibboLEAP were defined after consultation with the other participating institutions.

- eduPersonPrincipalName: <username>@ic.ac.uk
- eduPersonScopedAffiliation: (HTTP_SHIB_EP_AFFILIATION): User Status (STAFF, STUDENT)
- eduPersonTargetedID: Encrypted value used for holding persistent id information between sessions.
- mail (HTTP_SHIB_INETORGPERSO_MAIL): Email address
- ou (HTTP_SHIB_INETORGPERSO_OU): mapped to the department field

For the ShibboLEAP Eprints resource we need to use a combination of department user status (Staff, Student etc) to distinguish between people who could view the ShibboLEAP Eprints resources (everybody) and those who could administer them (Library staff). These were already available in the standard populated attributes received from our user registration system.

OU

Although our existing OpenLDAP server already has various organisational units to support our UNIX facilities, we realised early on that different attributes values would be required to authorise access to different resources and that the values of certain attributes may be required to be changed at a later date. To give us the flexibility to change the attribute values without effecting the support for our other UNIX facilities would need a new OU for Shibboleth which contained the all the college users.

Web Server Authentication

The OpenLDAP server used for the Information Repository allows pass through authentication to Active Directory where college passwords are stored so the college standard authentication module mod_auth_ldap was used.

Server Software

The selection of server software was based on the college Linux standard. This is Redhat Enterprise Linux operating system, Apache web server, Tomcat as a servlet container, OpenSSL and mod_auth_ldap authentication module. The latest production releases at the time were namely Redhat Enterprise 3.0, OpenSSL 0.9.7a, Apache 2.0.46 web server, java 1.5 (5.0), Tomcat 5.5.9 and mod_auth_ldap 3.0.7.

Server certificate

The college already uses a mixture of VeriSign and GlobalSign certificates on the college web servers. GlobalSign was chosen as a certificate supplier as it was supported by the ShibboLEAP service provider.

Shibboleth Implementation Experience

All implementation was done by the Information & Communication Technologies department (ICT) at Imperial College London.

OpenLDAP Server Implementation and Population

The college already uses OpenLDAP so there was no installation necessary however the eduPerson classes required by Shibboleth were not already present in the current college OpenLDAP schema so these were installed by our Network Service team. The Shibboleth OU was created and our custom scripts which process the data from our user registration system were altered to populate this OU with all members of college. The separate OU was created so that we could control how the attributes were populated without having to worry about how it may affect other systems using the same LDAP service. The attributes were populated with the relevant data with the eduPersonPrincipalName populated with <username>@ic.ac.uk. Although it took 20 minutes to install and change the scripts, the LDAP took several hours processing to populate the attributes of the 40,000 entries. An example of the attributes can be seen in appendix A.

Shibboleth Server Implementation

The installation of the Shibboleth server (shibbo.cc.ic.ac.uk) by our UNIX support team went smoothly with no problems. The current college standard Redhat Enterprise 3.0 Linux operating system was installed with OpenSSL 0.9.7a, Apache 2.0.4.6 web server , java 1.5 (5.0), Tomcat 5.5.9, local firewall software and lastly mod_auth_ldap 3.0.7 for connection to our LDAP server. This took 3 hours including the build of Tomcat from source.

A certificate request was generated for the web site and sent to GlobalSign. As we are already GlobalSign customers there was no need to produce the corporate accreditation required for new customers which significant sped up the process. We had hoped to receive the certificate within a week, however three weeks later the certificate had still not had arrived and on investigation it was discovered that the certificate had been issued the day after they received our request however our anti-spam software had blocked it. Once the certificate was finally received it was installed without any problems.

After the system had been locked down, the security services team opened up ports 443 for the Single sign on traffic and port 8443 for the AA (Attribute Authority) on the college firewall for this server. These two different ports were used only because of a bug in Apache. Ideally we would have only used port 443.

The documentation at <http://shibboleth.internet2.edu/> on both installation and configuration was found to be unclear and poorly-structured, however we did find the implementation guide at <https://authdev.it.ohio-state.edu/twiki/bin/view/Shibboleth/InQueueIdPInstall> better laid out and more useful.

The installation of the Shibboleth Identity server software proceeded using this guide however the "Villain Verification Service (VVS)" which is mentioned was also installed removing our configured mod_auth_ldap setup. It took four hours before this mistake was eventually spotted and it was decided to do a clean install.

We configured the four setup files (See Appendix C) and started testing.

- idl.xml – Main configuration file which specify's the location of the other configuration files and holds the certificate.
- metadata.xml – This details which service providers are allowed.

- resolver.xml – The attribute resolver configuration file which specifies the required attributes and how to query them.
- arp.site.xml – Details which attributes specified in resolver.xml are released to the service provider.

Most of the problems we encountered were related to typographical mistakes made in the configuration files which took a lot of effort to discover and correct, and this was not helped by the server firewall accidentally becoming enabled during testing which prevented the AA details form being received. We did have one major problem related to using choice of mod_auth_ldap as an access control mechanism.

Mod_auth_ldap does an LDAP search against the user id (username) to retrieve the distinguished name (DN), it then attempts to bind to LDAP using the DN and the supplied password. The REMOTE_USER variable is then set to this user id and passed back to the IDP. The IDP requires the REMOTE_USER value to correspond to the eduPersonPrincipleName which is <username>@ic.ac.uk.

All college systems ask for username without @ic.ac.uk so to get around this problem a custom Perl script was created to filter the single sign on requests. The script (Appendix B) runs after the basic apache authentication and checks if REMOTE_USER contains the suffix @ic.ac.uk. If it is missing the script appends adds the @ic.ac.uk suffix and adjusts the HTTP headers appropriately before passing it on to the SSO module. Therefore if I login as “test” the REMOTE_USER variable becomes “[test@ic.ac.uk](#)” and then the system works.

In total it took us 20 hours to get our Shibboleth Identity Provider installed and working correctly.

Future of the Shibboleth Identity Provider

We will continue to use this Identity Provider and proceed to join various other Shibboleth federations however we are also investigating Microsoft’s Identity Integration server and Oracle’s Identity Management Suite both of which contain Shibboleth Identity Providers.

APPENDICES

Appendix A

Example of LDAP directory entry

```
dn: uid=test,ou=People,ou=shibboleth,dc=ic,dc=ac,dc=uk
objectClass: top
objectClass: inetOrgPerson
objectClass: eduPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: test
cn: test
uidNumber: 51648
gidNumber: 6990
gecos: College Test Account
description: uid=test,nthomedir=\\icfs8.cc.ic.ac.uk\test
homeDirectory: /home/test
loginShell: /bin/bash
sn: \\icfs8.cc.ic.ac.uk\test
mail: test.mail@imperial.ac.uk
eduPersonPrincipalName: test@ic.ac.uk
eduPersonEntitlement: http://shibbo.cc.ic.ac.uk
eduPersonNickname: College Test Account
eduPersonOrgDN: dc=ic,dc=ac,dc=uk
eduPersonOrgUnitDN: OU=student,OU=people,dc=ic,dc=ac,dc=uk
eduPersonPrincipalName: test@ic.ac.uk
eduPersonPrimaryAffiliation: Staff
eduPersonAffiliation: Staff
```

Appendix B

Modifications to the httpd.conf file for our Shibboleth implementation used by Apache , including the PERL source code used by the SSO.

```
#####
## SHIB Config
#####

#
# Load the SHIBBOLETH module
#
LoadModule mod_shib /usr/libexec/mod_shib_20.so

#
# Global Configuration
# This is the XML file that contains all the global, non-apache-specific
# configuration. Look at this file for most of your configuration parameters.
#
ShibSchemaDir /usr/share/xml/shibboleth
```

ShibConfig /etc/shibboleth/shibboleth.xml

```
#
# Used for example logo and style sheet in error templates.
#
<IfModule mod_alias.c>
  Alias /shibboleth-sp/main.css /usr/doc/shibboleth/main.css
  Alias /shibboleth-sp/logo.jpg /usr/doc/shibboleth/logo.jpg
</IfModule>

#
# Configure the module for content
#
# You can now do most of this in shibboleth.xml using the RequestMap
# but you MUST enable AuthType shibboleth for the module to process
# any requests, and there MUST be a require command as well. To
# enable Shibboleth but not specify any session/access requirements
# use "require shibboleth".
#
<IfModule !mod_jk.c>
  LoadModule jk_module modules/mod_jk.so
</IfModule>

JkWorkersFile "/usr/tomcat5/conf/workers.properties"
JkLogFile "/var/log/httpd/mod_jk.log"

JkLogLevel emerg

JkMount /shibboleth-idp/* ajp13
JkMount /jsp-examples/* ajp13

<Location /secure>
  AuthType Basic
  AuthName "Enter your Imperial Login/Password to continue"
  LDAP_Server unixldap1.cc.ic.ac.uk
  Base_DN "ou=shibboleth,dc=ic,dc=ac,dc=uk"
  LDAP_StartTLS On
  LDAP_Debug On
  UID_Attr uid
  require valid-user
  PerlFixupHandler shibbo::remoteuser
</Location>
<Location /shibboleth-idp/SSO>
  AuthType Basic
  AuthName "Please enter your Imperial Login/Password to continue"
  LDAP_Server unixldap1.cc.ic.ac.uk
  Base_DN "ou=shibboleth,dc=ic,dc=ac,dc=uk"
  LDAP_StartTLS On
  LDAP_Debug On
  UID_Attr uid
  require valid-user
  PerlRequire /var/www/perl/startup.pl
  PerlOptions +GlobalRequest
  PerlFixupHandler shibbo::remoteuser
</Location>
```

Listen 8443 for AA traffic

```
<VirtualHost _default_:8443>
  SSLEngine on
  SSLCipherSuite
  ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP
  SSLVerifyClient optional_no_ca
  SSLVerifyDepth 10
  SSLOptions +StdEnvVars +ExportCertData
  SSLCertificateFile /etc/httpd/ssl/shibbo.crt
  SSLCertificateKeyFile /etc/httpd/ssl/shibbo.pem
  ErrorLog logs/ssl_error_log
  TransferLog logs/ssl_access_log
</VirtualHost>
```

mod_idap_auth fix

Perl code for [PerIFixupHandler](#)

```
package shibbo::remoteuser;
use strict;
use warnings;
use Apache::Const qw/:common/;
use Apache::RequestUtil;
use Apache::RequestRec;
use APR::Table;
use MIME::Base64;
use strict;
use Carp;

sub handler
{
  my $r = shift;
  return OK unless $r->user() !~ /^.+@ic.ac.uk$/ ;
  my $uname = $r->user() . "@ic.ac.uk" || return OK;
  _auth_ok ($uname) and do {
    $r->user($uname);
  };
  return OK;
}
sub _auth_ok
{
  return 1;
}
1;
```

Appendix C

Idp.xml – The amended idp.xml file used in our implementation.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!-- Shibboleth Identity Provider configuration -->

    <IdPConfig
        xmlns="urn:mace:shibboleth:idp:config:1.0"
        xmlns:cred="urn:mace:shibboleth:credentials:1.0"
        xmlns:name="urn:mace:shibboleth:namemapper:1.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="urn:mace:shibboleth:idp:config:1.0 ../schemas/shibboleth-
idpconfig-1.0.xsd"
        AAUrl="https://shibbo.cc.ic.ac.uk:8443/shibboleth-idp/AA"
        resolverConfig="file:/d01/shibboleth-idp/etc/resolver.xml"
        defaultRelyingParty="urn:mace:eduserv.org.uk:athens:federation:uk"
        providerId="https://ic.ac.uk/shibboleth/origin/1">
<!-- This section contains configuration options that apply only to a site or group of sites
The signingCredential attribute value here needs to match the name of some credential
defined
below -->
    <RelyingParty name="urn:mace:ac.uk:sdss.ac.uk:federation:sdss"
        providerId="urn:mace:ac.uk:sdss.ac.uk:provider:identity:ic.ac.uk"
        signingCredential="inqueue_cred">
        <NameID nameMapping="shm"/>
    </RelyingParty>
    <RelyingParty name="urn:mace:eduserv.org.uk:athens:federation:uk"
        providerId="urn:mace:eduserv.org.uk:athens:provider:ic.ac.uk"
        signingCredential="inqueue_cred">
    <NameID nameMapping="shm"/>
    </RelyingParty>

    <RelyingParty name="https://lse.ac.uk/shibboleth/federation/1"
        providerId="https://ic.ac.uk/shibboleth/origin/1"
        signingCredential="inqueue_cred">
    <NameID nameMapping="shm"/>
    </RelyingParty>

    <!-- Change only path here if necessary -->
    <ReleasePolicyEngine>
        <ArpRepository
implementation="edu.internet2.middleware.shibboleth.aa.arp.provider.FileSystemArpReposit
ory">
            <Path>file:/d01/shibboleth-idp/etc/arps</Path>
        </ArpRepository>
    </ReleasePolicyEngine>

<!-- Logging Configuration
The defaults work fine in this section, but it is sometimes helpful to use
"DEBUG" as the level for
the <ErrorLog/> when trying to diagnose problems -->
    <Logging>
        <ErrorLog level="WARN" location="file:/var/log/shibboleth/shib-error.log" />
```

```

        <TransactionLog level="INFO" location="file:/var/log/shibboleth/shib-
access.log" />
    </Logging>

    <!-- This configuration section determines how Shibboleth maps between SAML
Subjects and local principals.
        Do not change! -->
    <NameMapping
        xmlns="urn:mace:shibboleth:namemapper:1.0"
        id="shm"
        format="urn:mace:shibboleth:1.0:nameIdentifier"
        type="SharedMemoryShibHandle"
        handleTTL="28800"/>

    <!-- Determines how SAML artifacts are stored and retrieved
        Do not change! -->
    <ArtifactMapper
implementation="edu.internet2.middleware.shibboleth.artifact.provider.MemoryArtifactMappe
r" />

    <!-- This configuration section determines the keys/certs to be used when signing
SAML assertions -->
    <!-- The credentials listed here are used when referenced within <RelyingParty/>
elements above -->
    <!-- Change only if key/certificate locations are changed -->
    <Credentials xmlns="urn:mace:shibboleth:credentials:1.0">
        <FileResolver Id="inqueue_cred">
            <Key format="PEM">
                <Path>file:/etc/httpd/ssl/shibbo.pem</Path>
            </Key>
            <Certificate format="PEM">
                <Path>file:/etc/httpd/ssl/shibbo.crt</Path>
                <CAPath>file:/etc/httpd/ssl/gs-root.crt</CAPath>
            </Certificate>
        </FileResolver>
    </Credentials>

    <!-- Protocol handlers specify what type of requests the IdP can respond to. The
default set listed here should work
        for most configurations. Do not change. -->
    <ProtocolHandler
implementation="edu.internet2.middleware.shibboleth.idp.provider.ShibbolethV1SSOHandler
">
        <Location>https?://[^\:]+(:(443|80))?.*/shibboleth-idp/SSO</Location> <!-- regex
works when using default protocol ports -->
    </ProtocolHandler>
    <ProtocolHandler
implementation="edu.internet2.middleware.shibboleth.idp.provider.SAMLv1_AttributeQueryH
andler">
        <Location>.+:8443/shibboleth-idp/AA</Location>
    </ProtocolHandler>
    <ProtocolHandler
implementation="edu.internet2.middleware.shibboleth.idp.provider.SAMLv1_1ArtifactQueryH
andler">

```

```
        <Location>.+:8443/shibboleth-idp/Artifact</Location>
    </ProtocolHandler>
    <ProtocolHandler
implementation="edu.internet2.middleware.shibboleth.idp.provider.Shibboleth_StatusHandler
">
        <Location>https://[^\:]+(:443)?/shibboleth-idp/Status</Location>
    </ProtocolHandler>

    <!-- This section configures the loading of SAML2 metadata, which contains
information about system entities and
        how to authenticate them. The metadatatool utility can be used to keep
federation metadata files in synch.
        Metadata can also be placed directly within this these elements. -->
    <MetadataProvider
type="edu.internet2.middleware.shibboleth.metadata.provider.XMLMetadata"
    uri="file:/d01/shibboleth-idp/shibboleap.xml"/>
</IdPConfig>
```

Resolver.xml – The following changes made to the resolver.ldap.xml file

```
<SimpleAttributeDefinition id="urn:mace:dir:attribute-def:eduPersonScopedAffiliation"
smartScope="ic.ac.uk">
  <AttributeDependency requires="urn:mace:dir:attribute-
def:eduPersonAffiliation"/>
</SimpleAttributeDefinition>
-->
<SimpleAttributeDefinition id="urn:mace:dir:attribute-def:eduPersonPrincipalName"
smartScope="ic.ac.uk">
  <DataConnectorDependency requires="directory"/>
</SimpleAttributeDefinition>

<JNDIDirectoryDataConnector id="directory">
  <Search filter="cn=%PRINCIPAL%">
    <Controls searchScope="SUBTREE_SCOPE"
returningObjects="false" />
  </Search>
  <Property name="java.naming.factory.initial"
value="com.sun.jndi.LdapCtxFactory" />
  <Property name="java.naming.provider.url"
value="ldap://unixldap1.cc.ic.ac.uk/ou=People,ou=shibboleth,dc=ic,dc=ac,dc=uk" />
  <Property name="java.naming.security.principal"
value="cn=root,dc=ic,dc=ac,dc=uk" />
  <Property name="java.naming.security.credentials" value="<secret pass>" />
</JNDIDirectoryDataConnector>
```