



ShibboLEAP Project

Final Report:

Eprints Development

Simon McLeish

May 2006

<http://www.angel.ac.uk/ShibboLEAP/deliverables/finalreports/eprints.pdf>

1 Eprints and Shibboleth: Background

Each of the seven partners in this project maintains an institutional self-archive of academic publications, based on the Eprints.org software [eprints], as part of the SHERPA-LEAP consortium. Access to documents in these archives is public and unrestricted, but access management is required to authenticate the identities of academic staff depositing documents, and of library staff who must check or amend descriptive metadata, and approve documents to appear within the public collection. Currently authentication and authorisation is handled within the Eprints.org server, requiring the registration and use of (yet another) password by users, and administration of these registrations by staff supporting the archive.

Shibboleth [shibboleth] is an Internet2 project designed to make it possible to share resources securely and anonymously. Instead of the user presenting credentials to each resource, their home institution sends a set of attributes describing the user that need not identify them personally. Shibboleth has been identified as the main technology that will be used for authorisation in the future in UK higher education.

2 Approach to Shibbolizing Eprints

There are two sides to creating a Shibboleth version of any resource: authentication and authorisation. Authentication is usually simple for web-based resources, as Shibboleth will directly interact with the web server itself. However, the authentication for eprints is closely bound to the rest of the software, which means that the existing authentication code needs to be replaced.

Authorisation seemed to be potentially harder than authentication. There are technical and policy issues to sort out.

The technical issue is derived from the various possible methods for embedding the permissions in the code. Where permissions are embedded in a database of users, as they are in the standard eprints code, the issue is basically whether the permission listing is accessed fresh each time whenever it is required, or whether the permissions are set up once per session. In the first case, it would be necessary to change many different parts of the software, but eprints is organised the other way, so it is possible to replace the code which sets the permissions for the eprints session. This makes it much simpler to make eprints use eduPerson [eduPerson] or other attributes delivered by Shibboleth as the basis for its authorisation decisions.

The policy issue is to decide on how to use generic Shibboleth attributes to represent the different permissions required for the eprints software. Firstly, eprints needs to know the identity of users other than anonymous browsers, and it needs more than the anonymised eduPersonTargetedID offers. A digital repository must make it possible to find the user who claims the right to deposit a copy of an item in the repository, because of the obvious rights issues; and browsers who suspect a copyright violation need to have an idea of the identity of the potential offender. However, as the users of a repository are likely to be members of the institution which runs it, there is no privacy issue in passing the actual identity of the user via Shibboleth. We therefore decided that it would be requirement within the project to use eduPersonPrincipalName to identify the user (though the attribute used for the purpose is configurable in the eprints modifications we made). For the same reason, there is no problem in obtaining the user's email address from the Shibboleth Identity Provider, and the InetOrgPerson mail attribute is used to carry this information (again, the attribute

used is configurable).

Secondly, unless the eprints database continues to be used to contain data about a user's entitlements, which would be collected in the same way that it is in a standard eprints installation, it needs to be possible to collect the the information needed to distinguish staff permitted to deposit documents, and library staff permitted to amend metadata and approve deposits from the attributes passed by the Shibboleth IdP. This latter is problematic, because there is as yet unlikely to be any way to single out as small a group of individuals as those who are specifically entitled to act as editors in this fashion within an institutional directory: the information is simply not there. We decided to compromise, and assume that all staff in the library were entitled to act as editors. However, there remains the problem that departmental information is not necessarily consistent among the institutions which run repositories (and in particular is not consistent among the ShibboLEAP partners). This is for two reasons: the names used for the libraries are not the same for all partners (especially as some partners have several libraries on different campuses) and the directory entries reveal different levels of granularity being applied within the library entries (e.g. the LSE staff working on the ShibboLEAP project are listed as "Library: Electronic Projects" in the OU attribute of the directory). Example entries in partner directories include "Library Service" (Royal Holloway), "Library Services" (UCL), "Library" (SOAS) as well as the LSE entry already mentioned. This interoperability issue is likely to be problematic in any cross-institutional federated access scheme which requires levels of granularity for authorisation beyond that which can be calculated using those eduPerson attributes with with defined vocabularies.

The ShibboLEAP project ran its own mini-federation (see [shibboleap]) for project partners, and this meant it could specify policies to overcome both these issues [attributes]. These requirements insisted that four attributes be available from IdPs (eduPersonScopedAffiliation, eduPersonPrincipalName, mail, and ou). One of the design requirements for the Shibbolized eprints software was that it should be possible to specify in the configuration files both which attributes would be assumed to carry the relevant information, and (in the case of departmental information) what to look for in the data to show that a person has the relevant permission (e.g. the word "Library" if the user is to be allowed to act as an editor).

3 Technical Implementation Challenges

The first issue that caused problems was the installation of a copy of eprints to use as a testbed for the Shibbolization. The initial idea was to install this on a server which was a clone of the main SHERPA LEAP machine running on Slackware, but this ran into difficulties due to project team inexperience with this variety of Linux. Installation on a project Fedora machine was far smoother, though there were still difficulties in running eprints on secured HTTP, as is required for (secure) use with Shibboleth. This issue is a well-known problem with eprints, and has been addressed by the eprints development team via documentation [eprintssecure] and improvements to the installation process. The machine used for the installation already had a working Shibboleth SP on it, which was the reason it was chosen for the purpose.

Once the software was installed, the process of integrating eprints with Shibboleth could begin. This necessitated replacing the whole of the existing eprints authentication mechanism with a simple cookie-based authentication tracking scheme, using a slightly modified version of the eprints database to do this. This makes it possible to use both the existing login scripts or Shibboleth, which would make it simple to cater for special case users to add to the repository without themselves being able to authenticate to a Shibboleth IdP. A part of this login script needs to be protected by the Shibboleth IdP in the usual way, through mod_shib configured in the web server. Additional

configuration is needed to the eprints installation, as mentioned above, to set up the different authorisation levels and link them to Shibboleth attributes.

4 Future

The major issues that need to be tackled in the future are integrating the ShibboLEAP code with the main eprints code. Discussions are under way with the core eprints development team about the best way to do this, and the ShibboLEAP code should be available from the main eprints site once the current migration to a new WIKI is complete.

We have discussed our experience with the core eprints development team, and it is likely that future versions of eprints will see improved installation procedures (particularly in secure HTTP environments) and simpler mechanisms for replacing the existing authentication/authorisation code with customised modules for integration with external identity management systems including Shibboleth.

5 Technical notes

The ShibboLEAP software was written to work with eprints 2.3.13.1 and the Shibboleth 1.3c Service Provider, and was tested on a Fedora Core 3 Linux server with Apache 2.0.53-3.3. The software can be downloaded from

<https://gabriel.lse.ac.uk/twiki/pub/Projects/ShibbolethIntegration/eprints-shibboleth.tgz> and installation and configuration instructions can be found at <https://gabriel.lse.ac.uk/twiki/pub/Projects/ShibbolethIntegration/install.html>.

6 References

[attributes] Attribute Requirements for ShibboLEAP federation -

<https://gabriel.lse.ac.uk/twiki/bin/view/Projects/AttributeRequirements>

[eduperson] EduPerson Object Class Specification (200604) - [http://www.nmi-](http://www.nmi-edit.org/eduPerson/internet2-mace-dir-eduperson-200604.html)

[edit.org/eduPerson/internet2-mace-dir-eduperson-200604.html](http://www.nmi-edit.org/eduPerson/internet2-mace-dir-eduperson-200604.html)

[eprints] Eprints Free Software Website - <http://www.eprints.org/software/>

[eprintssecure] Eprints WIKI: Securing with HTTPS -

<http://wiki2.eprints.org/w/EPrints2/SecuringWithHTTPS>

[shibboleap] ShibboLEAP Project WIKI -

<https://gabriel.lse.ac.uk/twiki/bin/view/Projects/ShibboLeap>

[shibboleth] Shibboleth Project Website - <http://shibboleth.internet2.edu/>