

CUCKOO Project Grouper/Signet technical findings to date

April 2008

1 Introduction

This report describes the findings we have found whilst implementing Grouper and Signet thus far in our test environment. Our goal is to test the viability of using Grouper and Signet to provision our Virtual Learning Environment (which consists of multiple Moodle's), a repository and a Wiki. In addition to this, we are also investigating how they may provide us with the ability to set groups and permissions for external users, giving us more functionality for setting up Virtual Organisations.

Our main application, Moodle, currently uses Shibboleth for both authentication and authorisation. Authentication uses Kerberos against our Active Directory, authorisation is done using the eduPersonEntitlement attribute, this states the users role on a course and the course they are enrolled on, a user can have any number of Moodle entitlements. Active Directory is provisioned by our MIS database every night to ensure entitlement data is accurate and up to date. This setup works great for automating student and staff access to the VLE, however it offers no solution for ad hoc role/course assignments within the VLE or other applications, unless of course we go to each application and set these roles manually. This means roles are very static, and often group and entitlement information has to be duplicated across different applications; we require a more dynamic way of assigning these roles to all applications, and if possible delegate the management of this to the relevant people to reduce the overhead for IT staff.

From the outset, we have been looking at the COmanage (Collaborative Organization Management Platform) [<http://middleware.internet2.edu/co/>] method of providing this functionality. COmanage is the name of a suite of programs that are designed to deliver this functionality, the main components are Grouper for administering groups, Signet to administer permissions, and LDAPPC for provisioning LDAP directories. The diagram below shows how the two tools would fit into a typical institution.

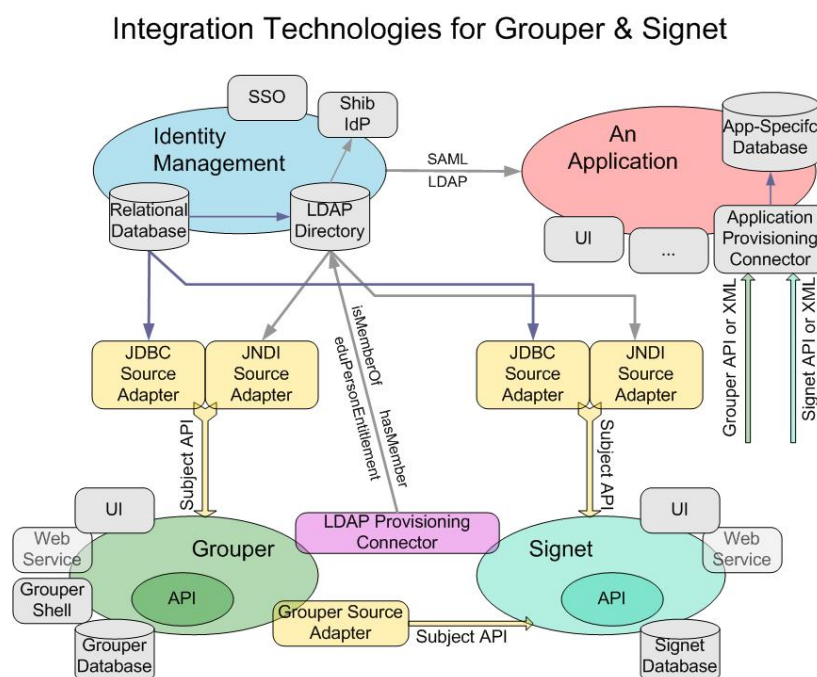


Diagram sourced from <https://wiki.internet2.edu/confluence/display/i2miCommon/Ldappc+v1.0>

2 Grouper Installation

Grouper will provide us with the ability to centrally administer user group assignments across all our web applications, whether it be for the courses they are on or ad hoc groups which are only valid for a short period of time, administration can be done not only by a central admin team, but also by tutors or anyone else with the given delegation privileges.

Our test platform has the following components:

CentOS 5

Java 1.5

Tomcat 5.5

Apache 2.2

MySQL 5

The Grouper Wiki is good for aiding administrators in the installation of Grouper, however at the time of our first install there was little documentation for installing the Grouper database on a MySQL 5 server; with help from the mailing list we were able to implement a working (all be it empty) instance of Grouper.

We pointed the sources.xml file to our Active Directory; this will allow us to pull out all our staff and students via LDAP, this was relatively easy to do by looking at existing examples on the Wiki and other sites. We manually added a small number of test accounts into test groups using the Grouper UI. For proof of concept we created a simple scenario with 10 groups, each group representing a real course, these groups were created through the Grouper UI, for a real world scenario where hundreds or thousands of groups are present this would either be done through the API or directly into the grouper database.

As we use Active Directory as our central store for users and their attributes we chose to use this as our store for group information. Our first initial tests have only been concerned with the 10 test groups, of which each one has a number of users assigned to it. Using LDAPPC we have implemented the provisioning of this information to Active Directory, so our 10 groups now have the correct users in them.

Perhaps the hardest part of implementing Grouper effectively is ensuring the applications can use the data correctly. In our environment, MediaWiki and our repository will be relatively easy to modify to use the group information, we have tested this initially with .htaccess rules, but believe it will not scale well, which has been confirmed by the GFIVO project team. More complex applications like Moodle may be harder to implement, a decision must be made very early on of how you want the group information to be used, for example, we already have course membership for Moodle using the eduPersonEntitlement attribute, so assigning users to courses with this information is pointless, a better way to use group information within Moodle maybe to assign users to the groups within a Moodle course, this way lecturers can assign groups within courses, something that cannot be done (currently) with the Shibboleth enrolment module, and also an area that doesn't always match the strict rules from our student enrolment data.

3 Signet Installation

Signet will provide a central way to administer user privileges across all of our applications, in most cases this will be provisioning the users eduPersonEntitlement attribute with the necessary privileges. As with Grouper, Signet provides us with the ability to delegate these tasks.

Signet work is still ongoing, however we found this harder to install as generally it appears uptake for this software is a lot less than that of Grouper. However the setup time of Signet was reduced due to the fact that some pre-requisites were installed as part of the Grouper installation.

As with Grouper, Signet will pull users from Active Directory, and provision it back into AD once permission information has been entered for users. Due to the model of Moodle that we have running, we see these granular permissions are more suitable than those of group information, as it allows more control over what the user can do. Of course most eduPersonEntitlement is already in place for our users, however with Signet setup these permissions are no longer bound to the static permissions obtained from our MIS database, we now have the ability to allow tutors or project leaders to delegate these permissions to users. This reduces the burden on IT staff that are best placed doing administration at a lower level, who have no idea what groups users should be in.

4 Conclusions

Through our initial tests it is clear to see that these two tools can provide great benefits, both from the end user point of view and the administrators. The benefits of having Grouper setup in an organisation are increased by the number of applications it protects, if you have only one application then having Grouper will not be of any benefit, as you could more easily use the applications internal user management. The power of Grouper will therefore become evident where a multitude of applications are made aware of the groups set within LDAP, or made aware with other provisioning techniques available through Grouper, such as htaccess rules or the soon to be released database provisioning connector. The RDMS connector we feel will open up the potential of Grouper, allowing nearly all web applications to be provisioned, all be it with an initial overhead of implementing the connector for the applications databases. Ongoing work with web services will also make application provisioning an easier task in the future.

There are limitations to the way Grouper and Signet are provisioned with data, whether it be databases or LDAP stores, most implementations currently poll the synchronisation at regular intervals, so those requiring real time updates will need to find another solution. We are looking into solutions for this, Cardiff University are intending to extend Grouper to sync in near real time with their eDirectory. Another concern that requires further research is instances where the Grouper database has been provisioned with groups and user information, and those groups need modifying to match the real world memberships within Grouper , however, the next time Grouper is provisioned this will be updated with the old group assignment values, Grouper needs to be updated so membership data set within it does not get overwritten when synched.

Institutions with a small number of resources, or resources which are already centralised (for example they all use LDAP, and LDAP is provisioned via an IDM solution already) may find Grouper and Signet unnecessary, and actually increase administration overhead.

Due to the way our identity management and VLE are setup, the Signet tool will be more useful for us. Groups are already taken care of in terms of courses students are on, when a student joins the college they are put in the correct group, this gets updated every night to ensure this is correct; so Grouper would only be useful for the ad hoc groups.

Signet however, with its ability to provision the entitlements of users is immediately useful to us, our VLE uses this information already, to provide an easy to use interface to delegate the provisioning of these entitlements would reduce admin overhead and mistakes made, as well as speeding up the provisioning process.

It is fair to say that the user interfaces are not very user friendly in their current states, being well above the heads of the staff we showed it to. The community has already begun working on improving this, we particularly like the simple user php front end developed by the GFIVO project, not only does the interface hide a lot of the unnecessary jargon and complexity for the end user, it would also fit very well into our environment of mainly php applications.

In environments lucky enough to have all their web applications reading from the same store, e.g. all applications point to LDAP, then simpler applications could be developed (for example a simple php interface to allow users to connect straight to the LDAP server itself), as we feel the learning curve for Grouper and Signet is quite high. This high learning curve can be justified by the fact that it has been written to operate in a heterogeneous environment where applications and user information is stored in many different ways.