Use Case - MyLibrary and Reading List System

Scenario

A current project by Newcastle University's library is the MyLibrary project which is aimed at providing a customisable environment for students and staff to manage their library resources. A system which will work closely with this is the Reading List system. The purpose of this system is to allow module leaders and lecturers to manage the reading lists for their taught modules and to enable students to access reading lists applicable to their course from their MyLibrary area.

The focus of this use case is on the management of the module reading list information, rather than the access provided for students to view their reading lists. A structure is already in place to delegate access for students to their reading lists, this is achieved using Grouper. A data flow into Grouper allows for the representation of student to module enrolments, with a group being created for each module, with students being members of the appropriate module groups. These modules are then released as a Shibboleth attribute, so that it is possible for web based systems such as the reading list system to access this attribute and restrict access dependant on group memberships.

The reading list system will link directly into the ordering of books for the library. The reading lists submitted by the module leader, are used to determine which books are required within the Library's catalogue. It is therefore important that the system is provided with accurate module information and that module leaders are able to update reading lists which are appropriate to themselves.

Every module has a module leader associated with it and the module leader is therefore responsible for informing the library of the reading list for the module. Until now, this is a process that should be done through the University's module outline system (MOF'S), as on the creation of a module, in order for the module to be accepted a reading list should be provided for it. Though this is not always the case, with modules being accepted without a reading list being associated to it, and often is the case that module leaders provide the library a handwritten copy of the reading list.

The reading list system is aimed at providing a consistent tool for module leaders to be able to manage and update their own module reading lists. Module leaders will be provided with a web interface which will present them with the reading list information for their taught modules. Modules are not only associated with module leaders, they are also associated with lecturers and contributors. The system will allow for the lecturers and contributors to also be provided with access to the reading list so that they can make any appropriate updates.

Alongside the academic staff that have direct links to the module, it is necessary for library administrators to have overall access to all the reading lists. In an ideal world, all module leaders would look after their own reading lists, yet as previous experiences from the library staff have shown, they realise that this will not always be the case. In the scenario that module leaders do not keep up to date their reading list or insist on submitting details via other routes, library staff will need access to be able to update the reading list on their behalf.

Actors/Stakeholders

- Module leaders
- Lecturers
- Contributors
- Library staff

Forms of Delegation

The key areas of delegation are required;

- Academic staff who are able to manage reading lists
    - Module leaders should be able access and edit the modules that they are associated with
    - Lecturers should be able to access the modules that they teach on.
    - Contributors should be able to access the modules that they provide input to.
- Library administrators
    - Library administrators should be able to access and edit all reading lists for every module.
- Delegation of privileges
    - Library staff should be able to add members of staff to the access control group for individual reading lists.
    - Module leaders should be able to delegate administrator's privileges to other members of staff when required.

Approaches and Considerations

An advantage of using such tools as Grouper to manage access control to resources, is the ability to make use of data structures that are loaded into Grouper on a scheduled basis, for example the organisational structure as discussed in Use Case 1. These pre-loaded groups can then be used to facilitate the delegation of access control, as in the Org structure, the ability to allow all members of ISS to book a particular room.

As an output of the JISC funded IDMAPS project, Grouper and other applications throughout the University are able make use of numerous sources of institutional data. In the terms of this Use Case, it is possible to extract data regarding all the modules that are currently being taught or have been taught at the University. Module data has been used previously within Grouper to assist in the delegation of access control for students accessing lecture captures with the Recap system.

Delegation on a Module basis

The structuring of modules and student enrolments currently used in Grouper has proved successful with the provisioning of access control for the Recap system. This could provide the basis of a possible solution to this use case, though there is a difference between the two use cases. The student enrolments use case, does not require a member of staff to maintain the memberships of the groups, as memberships are managed through a scheduled provisioning job. For the reading list system there is a requirement for staff to be able to update the defined access groups.

A proposed solution is to replicate the module structure that is created for the Recap system, but instead of populating memberships with students, it will replace these with module staff members.

The proposed group structure is shown in figure 1. Each module access group would represent the reading list, and it would provide the structure to then be able to delegate access to the required users for this use case.

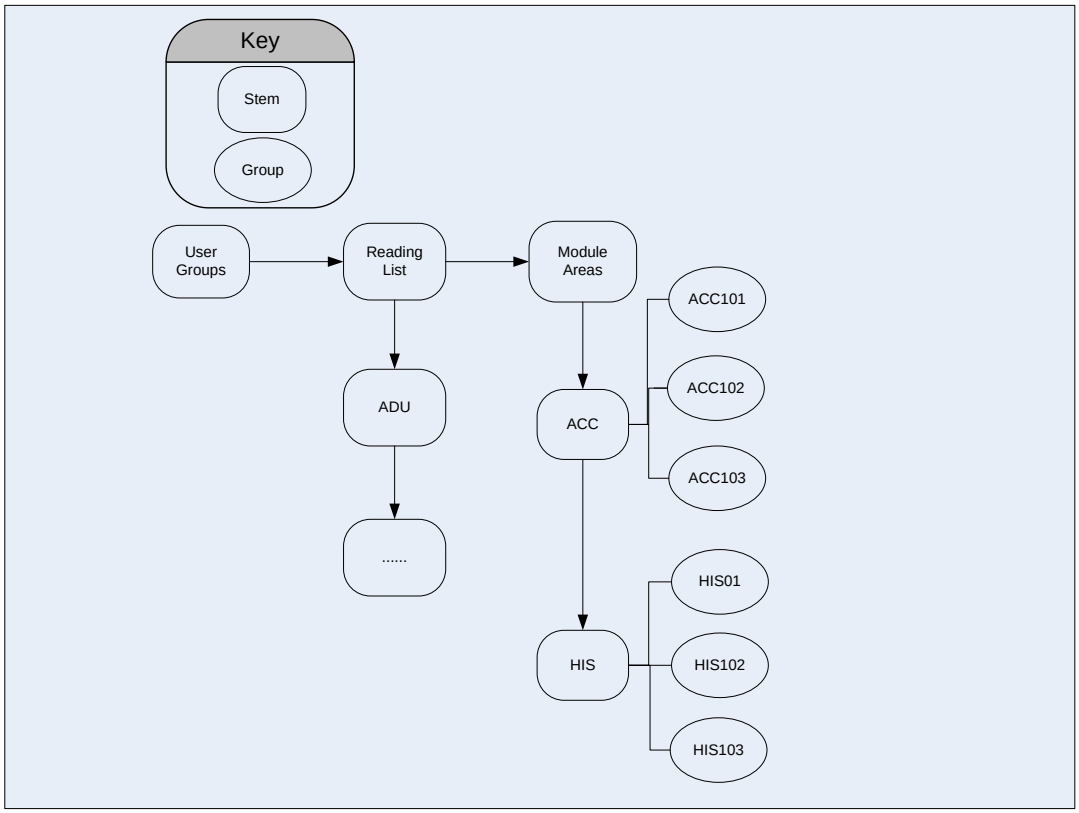

Figure 1: Proposed structure for delegation by module

The module leader is the main user who should have the responsibility for maintaining their respective reading lists, therefore on the creation of the structure they will be automatically assigned to the necessary module access group. This can be achieved by making use of the data that is loaded into the data warehouse that has been developed as part of the IDMAPS project. They will be added as a member to the group, and it is this membership that the reading list system will use to determine access.

This deals with the delegation of access within the system, but it is also necessary for the module leaders to assign other users with access to the reading list. This can be done by assigning the module leader with administrator privileges on the module access group. This will allow them to delegate access to other staff members to edit the reading list when necessary using the Grouper UI. It may also be the case that the module leader needs to pass the responsibility for the management of the reading list to a colleague. If this is the case, they can delegate appropriate privileges on the module access group to this colleague. All contributors and lecturers for the modules will also be assigned directly to their respective modules as members, allowing them to access the appropriate lists in the reading lists system.

As stated in the scenario the reluctance of some module leaders to maintain their reading lists means that this solution would require a further group of users who could have access to all

modules. This group would be made up of the library staff, if the module leader does not maintain the reading list, the responsibility then passes over to the library staff.

This is a finely grained approach to delegation of access control, as it restricting access on an individual module basis ,meaning that only staff members that have some kind of association to the module will be able to access the reading list in the system. As a result of this the system administrator can be assured that only the module staff and any staff the administrator has delegated access to will be able to edit the module reading list. As a result, this can lead to a real administrative burden, there are approximately 4000 modules taught at Newcastle University, this means 4000 module access groups being created within Grouper.

High Level Delegation

The above approach to the delegation of access control does provide the system administrator with a controlled and restrictive structure to ensure that reading lists are not edited mistakenly or maliciously. Yet as previously discussed with this fine grained approach it creates the administrative burden which can soon lead to the module access role groups becoming outdated if they are not regularly updated.

An approach to attempt to reduce the amount of administration required would be to create a higher level of delegation, with only a few access groups being used. As proposed in the first solution, it would be necessary to have a group which is made up of all the library staff, as they require edit access over all the modules.

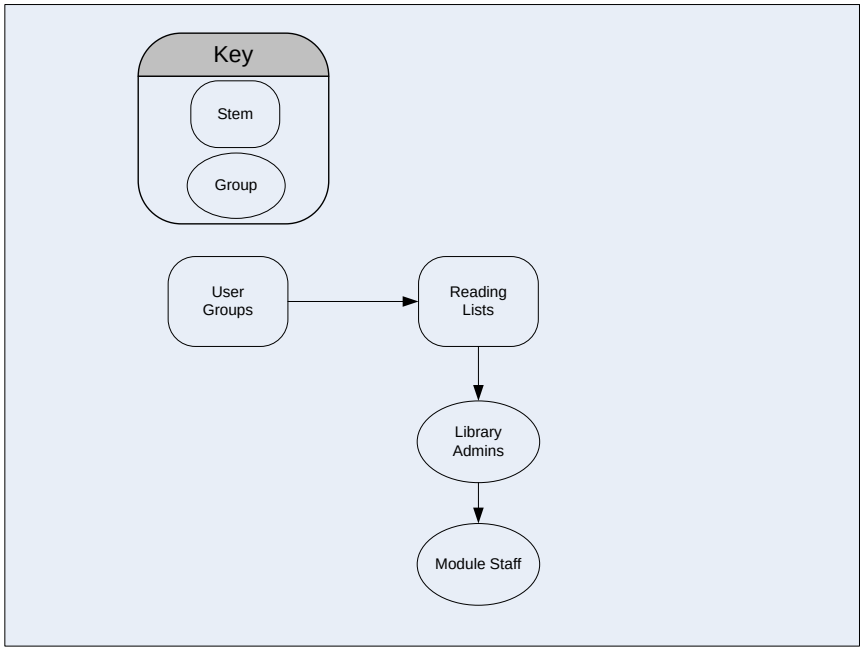

Figure 2: High level delegation

The next step would be, instead of loading all the modules into grouper and assigning module leaders to their respective module access role groups, is to create an overall module leaders access control group. This group would be automatically updated with all module leaders, adding and removing memberships on a scheduled basis. This approach takes an assumption that module leaders can access all reading lists not just the ones that they lead. This can be seen as being as

taking a lax approach to the access control, yet on careful consideration of the overall scenario, a question could be asked whether module leaders need to be restricted to their own modules or can there be an element of trust.

Access control does not need to be restricted to an environment such as Grouper, an artificial level of access control can also be provided at the application side. Although module leaders technically have access to all reading lists as they are part of the access group, at an application level they could be presented with only the modules that are associated with them. This approach would provide the same solution as in the first approach, yet reduces the level of administration that would be required. Further thought would be required on how module contributors and lecturers could be provided access to the reading list, though this could also take a similar approach of creating an access group for contributors and lecturers.

Initial Conclusions

From the initial review of the scenario it is clear that a solution to this use case can be achieved in a number of different ways, not just the two described above. Careful considerations by the system administrators will need to be fulfilled to decide upon the most suitable solution to enable a secure yet manageable approach for the delegation of access control. Proofs of concept will be put in place to determine a structure that can be used to deliver this delegation and help in the decision of which approach to adopt.