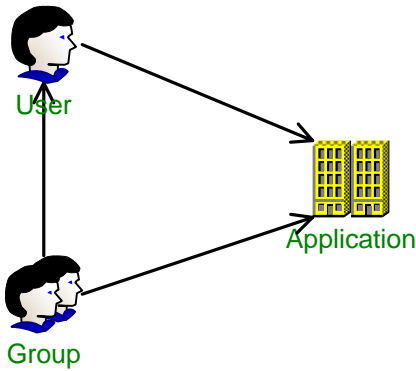


1. Current Mapbender User/Group Model



This information is taken from wiki documentation and testing using Mapbender 2.5 rc2.

Features

- + Ease to administer
- 0 Fits the needs of most users?
- Rights are given at the application level
- Groups cannot inherit rights of other groups

User Management

- A user is created by another user that has access to mod_editFilteredUser
- A user can have only one owner (by default the user that created them)
- A user can only be edited/deleted by their owner
- An owner of a user can assign the user to any application/gui they have access

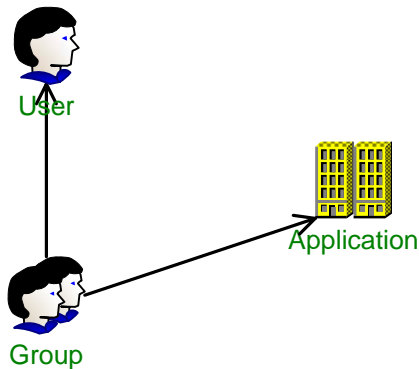
Group Management

- A group is created by a user that has access to mod_editFilteredGroup
- A group can have only one owner (by default the user that created them)
- A group can only be edited/deleted by their owner
- An owner of a group can assign the group to any application/gui they have access

Application Management

- Groups and users can be granted access to applications
- A user/group granted access to an application has full access to all modules and WMS & WFS services
- Only users can be owners of an application
- Only owners can modify GUI elements of an application

2. Mapbender User/Group/Group Model



This model is similar to the current design. A new table is added (mb_group_mb_group) which allows groups to be members of groups. Thus groups could inherit the rights of other groups. Only groups would have access to applications, this is suggested as a way to simplify the administration.

Features

- + Ease to administer
- + Minor change to current model
- 0 Fits the needs of most users?
- Only one owner of a group or application
- Rights are given at the application level

User Management

- A user is created by another user that has access to mod_editFilteredUser
- A user is owned by a group which is assigned at the time of creation
- A user can only be edited/deleted by member of the owning group
- A member of the owning group can assign the user to any application/gui they have access

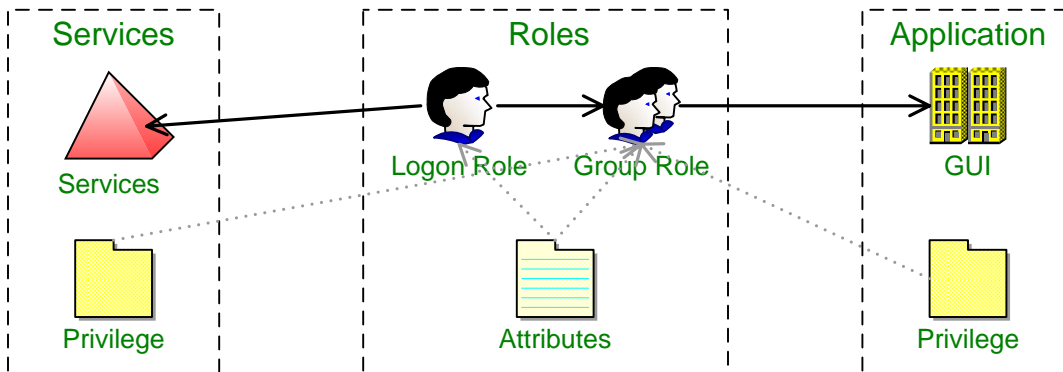
Group Management

- A group is created by a user that has access to mod_editFilteredGroup
- A group can have only one owner (by default the user that created them)
- A group can only be edited/deleted by their owner
- An owner of a group can assign the group to any application/gui they have access

Application Management

- Groups can be granted access to applications
- A group granted access to an application has full access to all modules and WMS & WFS services
- Only users can be owners of an application
- Only owners can modify GUI elements of an application

3. Alternative Mapbender Role Model



This model is similar to how Postgres handles DB management and would require significant changes to the Mapbender security model, but this would offer far more flexibility with Mapbender administration. This document includes examples of possible attributes and privileges and is not meant to be an exhaustive list.

How it works:

Roles

Roles replace users and groups. A logon role is a role which has the "logon" attribute – its only function is to provide access to Mapbender. Group roles can have Mapbender attributes such as "createLogon", "createApplication" and "createService"; in general these attributes would be given to Mapbender administrators. Default roles such as "Administrator" and "Guest" would be provided with Mapbender. When a role is created, it can inherit the attributes of the parent role.

Applications

Access to an application will be provided by privileges given to roles. Examples of privileges include "access", "editApplication" and "grantAccess". For example, application GUI1 grants ACCESS to role USERS.

Services

WMS and WFS services would be similar to applications in that they can grant privileges to roles for access and administration.

Features

- + Ease to administer
- + Applications give privileges to roles, much more flexibility
- + Roles can inherit attributes and privileges of other roles
- + Roles are granted privileges to services separately from applications
- 0 Do Mapbender installations need this much control?
- Major change to current model

An Example

The default roles überadmin and Mapbender Admin are used to create the new logon role User3 and group role Application Admin. User3 has the attributes needed to create logon roles and applications and creates the application GUI1 and roles User1, User 2 and Users. When GUI1 is ready for use, User3 gives the role Users the accessApplication privilege and people are ready to use the application. Note, that services have been left out, but would be configured similarly to an application.

