



HOWTO Integrate Moodle with Grouper on Ubuntu Linux 12.04

20 November 2014

Author: Marco Malavolti, Andrea Biancini

Index

1) Introduction.....	3
2) Requirements.....	3
3) Installation.....	4
3.1) Install and Configure Shibboleth plugin for Moodle.....	6
3.2) Install and Configure the Moodle enrolment plugin for VOOT.....	8
4) Add the Moodle groups to Grouper.....	9
5) How to prevent the access to an user.....	13
6) Bibliography.....	14

1 Introduction

This is a tutorial for users that want integrate the authentication on a Moodle learning platform with Grouper on an Service Provider based on Ubuntu Linux 12.04.

2 Requirements

- A machine with Grouper installed and configured.
- A Shibboleth SP (sp.example.it) with Apache2 Web Server, PHP 5.3.3 or later, MySQL 5.1.33 or later and Ghostscript for pdf annotation.

3 Installation

Install and Configure Moodle on your Service Provider

1. Install the software requirements for Moodle on your SP:

- `sudo apt-get install apache2 mysql-server php5-mysql ghostscript php5-curl php5-xmlrpc php5-intl php-pear php5-dev php5-gd`

2. Install Moodle 2.6.6 on your SP:

- `sudo su -`
- `cd /usr/local/src`
- `wget https://download.moodle.org/download.php/direct/stable26/moodle-2.6.6.tgz`
- `tar xzf moodle-latest-26.tgz`
- `mv moodle/ /var/www/`
- `chown -R root:root /var/www/moodle ; chown www-data /var/www/moodle`
- `vim /etc/apache2/sites-available/moodle.conf :`

```
<IfModule mod_alias.c>
Alias /moodle /var/www/moodle/

<Directory /var/www/moodle/>
Options Indexes MultiViews FollowSymLinks
Order deny,allow
Allow from all
</Directory>

</IfModule>
```

- `a2ensite moodle.conf ; service apache2 reload`
- `mysql -u root -p` (type your mySQL root password to access)
 - `create database moodledb;`
 - `create user 'moodledbuser'@'localhost' identified by '###moodledbuserPassword_CHANGE_ME##';`
 - `grant all on moodledb.* to 'moodledbuser'@'localhost' identified by '###moodledbuserPassword_CHANGE_ME##';`
 - `flush privileges;`
 - `quit`

3. Create the “moodledata” directory with right permissions:

- `sudo mkdir /var/moodledata ; chown www-data:www-data /var/moodledata`

4. Configure Moodle on your SP:

- Open the page “<https://sp.example.it/moodle/install.php>” and follow the instructions until the end. (Remember that you have installed a MySQL Database and ignore the Environment Check message about OpCache)

For default, MySQL installs its socket to: `/var/run/mysqld/mysqld.sock`

Database host	<input type="text" value="localhost"/>
Database name	<input type="text" value="moodledb"/>
Database user	<input type="text" value="moodledbuser"/>
Database password	<input type="text" value="##moodledbuserPassword_CHANGE_ME##"/>
Tables prefix	<input type="text" value="mdl_"/>
Database port	<input type="text"/>
Unix socket	<input type="text" value="/var/run/mysqld/mysqld.sock"/>

Pay attention on the credential of Admin User that you will give to Moodle installation.

- At the end of the Moodle installation you will be redirect on the Moodle learning platform logged-in as Administrator.

5. Create the cron rule necessarily to Moodle:

- Add this line to the file opened with ``crontab -e``:
`*/* * * * * /usr/bin/php -c /etc/php5/apache/php.ini /var/www/moodle/admin/cli/cron.php`

3.1 Install and Configure Shibboleth plugin for Moodle

The following instructions are based on the **README.txt** you can find into the directory:
/var/www/moodle/auth/shibboleth

1. Configure Shibboleth SP to support the Moodle authentication plugin:

- vim /etc/shibboleth/shibboleth2.xml

```
<Host name="sp.example.it">
  <Path name="moodle" authType="shibboleth" requireSession="true"/>
  <Path name="auth/shibboleth/index.php" authType="shibboleth" requireSession="false"/>
</Host>

<SessionInitiator type="Chaining" Location="/DS" id="DS" relayState="cookie"
isDefault="true">
  <SessionInitiator type="SAML2" acsIndex="1" template="bindingTemplate.html"/>
  <SessionInitiator type="Shib1" acsIndex="5"/>
  <SessionInitiator type="WAYF" acsIndex="5" URL="https://dsc.example.it/DS"/>
</SessionInitiator>
```

- shibd -t /etc/shibboleth/shibboleth2.xml
- service shibd restart

2. Edit the apache2 moodle configuration to support the Shibboleth authentication:

- vim /etc/apache2/sites-enabled/moodle.conf

```
<IfModule mod_alias.c>
  Alias /moodle /var/www/moodle/

  <Directory /var/www/moodle/>
    Options Indexes MultiViews FollowSymLinks
    Order deny,allow
    Allow from all
  </Directory>

  <Directory /var/www/moodle/auth/shibboleth/index.php>
    AuthType shibboleth
    ShibRequireSession On
    require entitlement urn:mace:garr.it:moodle
  </Directory>
</IfModule>
```

(This rules require apache 2.4 and Shibboleth SP software v2.4.3.

If you use a Shibboleth SP release >= 2.5.1, please, use this:

```
<Directory /var/www/moodle/auth/shibboleth/index.php>
  AuthType shibboleth
  ShibRequestSetting requireSession 1
  require shib-attr entitlement urn:mace:garr.it:moodle
</Directory>
```

)

- service apache2 restart

3. As Moodle admin, go to the **Site administration => Plugins => Authentication => Manage authentication** and click on the Shibboleth **“Settings”**.

4. Fill in the fields of the form:
 - a) 'Username': for example “eduPersonPrincipalName” o “**eppn**”
 - b) Shibboleth Service Provider logout handler URL: “/Shibboleth.sso/Logout”
 - c) 'First name': for example “**givenName**”
 - d) 'Surname': for example “**sn**”
 - e) 'Email address': for example “**mail**”

If you want that Moodle checks these value on each User Login, you must select the “**On every Login**” value from the listbox near the “**Update local**”.

The fields should contain the name of the Shibboleth attributes that you want to map onto the corresponding Moodle variable (refer the Shibboleth documentation or the documentation of your Shibboleth federation for information about the attributes that are available from IdPs that the Moodle learning platform must serve).

Shibboleth Attributes needed by Moodle:

For Moodle to work properly Shibboleth should at least provide the attribute that is used as username in Moodle. It has to be unique for all Shibboleth.

Be aware that Moodle converts the username to lowercase.

So, the overall behaviour of the username will be case-insensitive.

All attributes used for moodle must obey a certain length. You can find this value by searching the word “maxlength”. For example on the “*signup_form.php*” file:

```
$mform->addElement('text', 'username', get_string('username'), 'maxlength="100" size="12");
$mform->addElement('passwordunmask', 'password', get_string('password'), 'maxlength="32" size="12");
$mform->addElement('text', 'email', get_string('email'), 'maxlength="100" size="25");
$mform->addElement('text', 'email2', get_string('emailagain'), 'maxlength="100" size="25");
$mform->addElement('text', '$field', get_string($field), 'maxlength="100" size="30");
$mform->addElement('text', 'city', get_string('city'), 'maxlength="120" size="20");
```

Especially the 'Username' field is of great importance because this attribute is used for the Moodle authentication of Shibboleth users.

5. Save the changes with button “**Save changes**”.
6. When all is ready, enable the Shibboleth authentication by pressing on the “**eye**” into the “**Enable**” column for Shibboleth.
7. The last things to do is modify the **Site administration => Plugins => Authentication => Manage authentication => “Instruction”** blank space by putting some information like these:

If you do not have credentials to enter this Moodle site, you can use your federated identities by clicking the link below.



Where the Shibboleth image is linked to the page:

<https://sp.example.it/moodle/auth/shibboleth/index.php>

3.2 Install and Configure the Moodle enrolment plugin for VOOT

1. Download the enrolment plugin into the properly directory:

- `cd /var/www/moodle/enrol`
- `git clone https://github.com/biancini/moodle-enrol_voot.git`
- `mv moodle-enrol_voot/ voot/`

2. Open a new windows of your browser and go to the “**Site administration**”=>”**Notifications**” to see which plugin require to be installed and press on “**Upgrade Moodle database now**”:

This page displays plugins that may require your attention during the upgrade. Highlighted items include new plugins that are about to be installed, updated plugins that are about to be upgraded and any missing plugins. Add-ons are highlighted if there is an available update for them. It is recommended that you check whether there are more recent versions of add-ons available and update their source code before continuing with this Moodle upgrade.

[Check for available updates](#)

Last check done on 19 November 2014, 11:35 PM

Number of plugins requiring your attention: 1

[Display the full list of installed plugins](#)

Plugin name	Directory	Source	Current version	New version	Requires	Status
Enrolment methods						
VOOT Server	/enrol/voot	Add-on		2013110500	Moodle 2013110500	To be installed

[Reload](#)

[Upgrade Moodle database now](#)

3. Configure the settings for “**VOOT Server**” like this:

- VOOT host: **grouper.full.qualified.domain.name**
- VOOT user name: **GrouperSystem**
- VOOT password: **yourGrouperSystemPassword**
- Group prefix for courses: **sp.example.it:moodle:**
- Group name for teachers: **admin**

pay attention because the fields are case sensitive!

4. Save the changes with the proper button and go to the **Site administration** => **Plugins** => **Enrolments** => **Manage enrol plugins** and enable the VOOT Server plugin by pressing on the “**eye**” into the “**Enable**” column.
5. Add one test course with one test member for Moodle to Grouper by following the Chapter 4
6. Now press on “**Test settings**” to verify the correct working of VOOT Server plugin.
7. It is important to understand that Moodle need to know the users before their addition to the courses, so the users must login to Moodle Platform at least one time or the VOOT sync script will give, for example, the error: *error: skipping unknown user username 'example@eppn.it' in course 'philo'*
8. Create the cron rule necessarily to sync the course stored on Grouper:
 - Add this line to the file opened with ``crontab -e``:

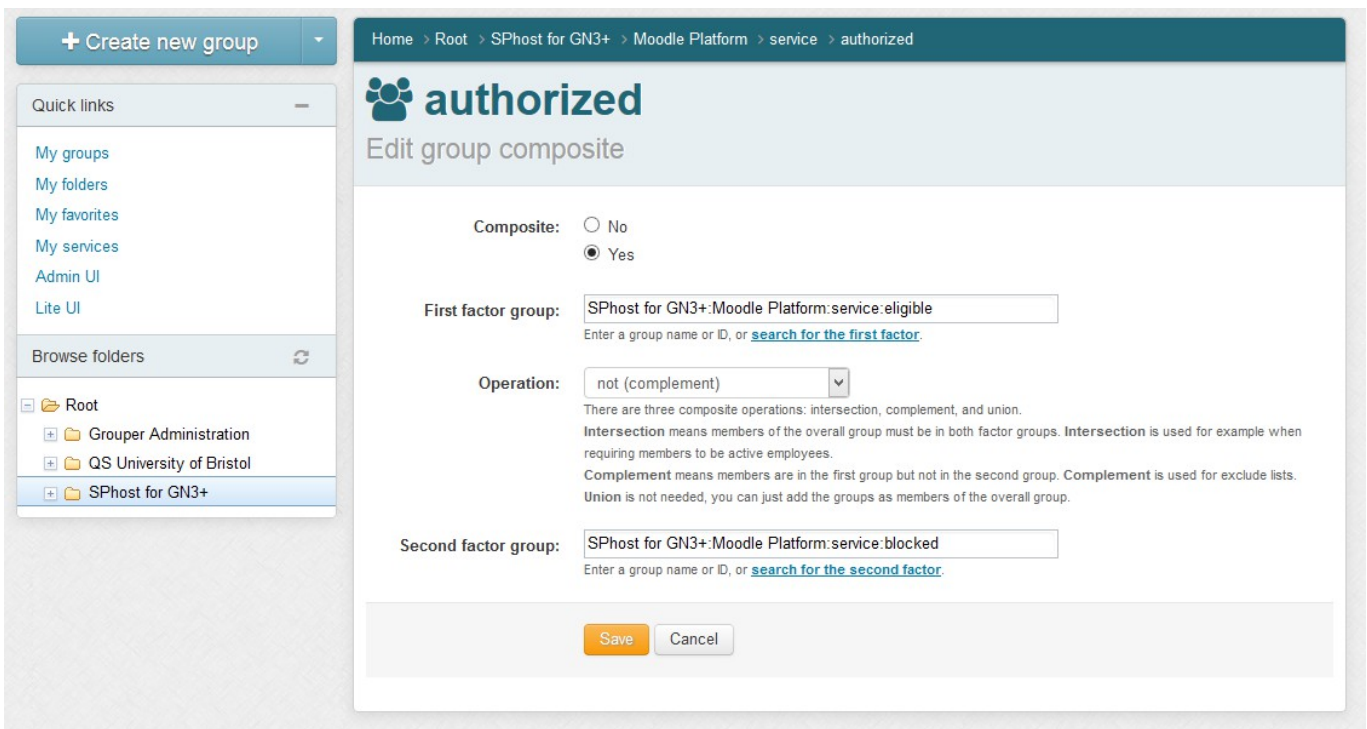
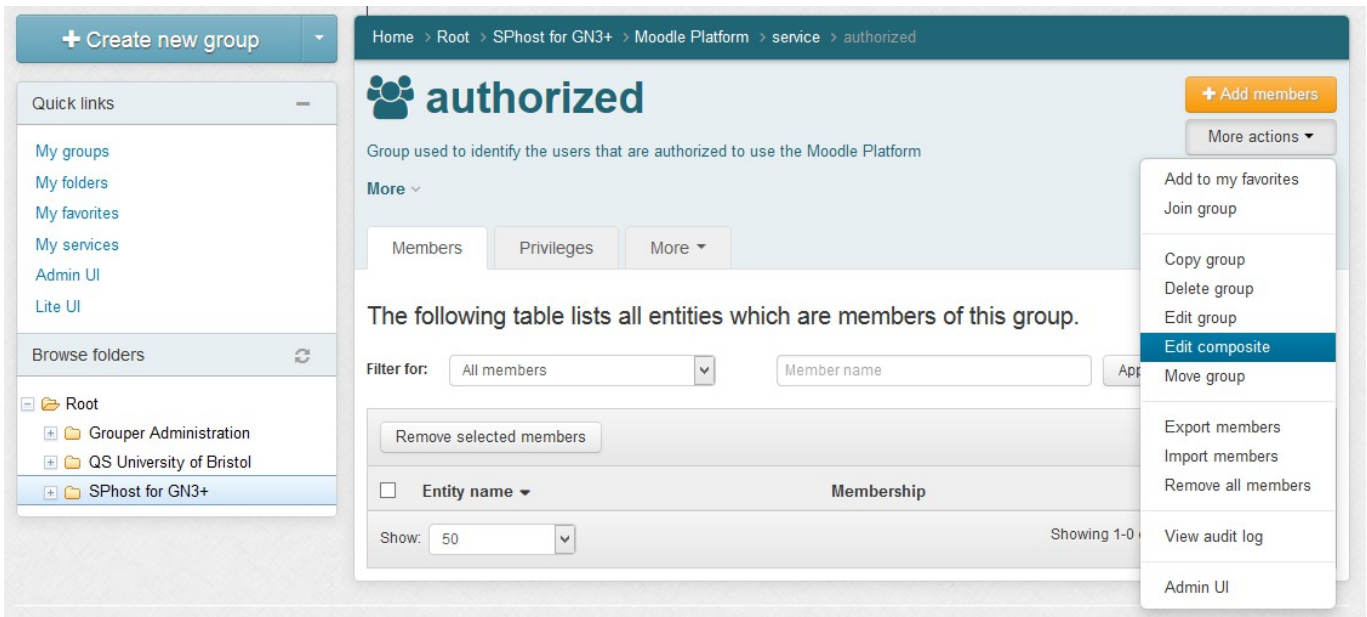

```
* /5 * * * * /usr/bin/php -c /etc/php5/apache/php.ini /var/www/moodle/enrol/voot/cli/sync.php
```


4 Add the Moodle groups to Grouper

1. Open Grouper application, log-in as Admin and create the folder for Moodle (“Moodle Platform” with Folder ID = “**moodle**”) under the relying party's folder:

2. Create the “**service**” folder into the new “**moodle**” folder and add to it these groups:
 - a) authorized group:
 - Name: **auhtorized**
 - ID: **authorized**
 - Description: **Group used to identify the users that are authorized to use the Moodle Platform**
 - b) eligible group:
 - Name: **eligible**
 - ID: **eligible**
 - Description: **Group used to identify the users that are eligible to use the Moodle Platform**
 - c) blocked group:
 - Name: **blocked**
 - ID: **blocked**
 - Description: **Group used to identify the users that are blocked when they try to use the Moodle Platform**

3. Make the “authorized” group a composite group that involves “eligible” and “blocked” groups:



4. Add the group designed to contain the users that are blocked on all services hosted by the SP to the “**blocked**” group:

The screenshot shows the Moodle Grouper interface for the 'blocked' group. The breadcrumb trail is: Home > Root > SPhost for GN3+ > Moodle Platform > service > blocked. The main content area has a search bar for 'Member name or ID' with the value 'SPhost for GN3+:service:quickdisable' and a link to 'search for an entity'. Below this is a section for 'Assign these privileges' with radio buttons for 'Default privileges' (selected) and 'Custom privileges', and an 'Add' button. A description below reads: 'Group used to identify the users that are blocked when they try to use the Moodle Platform'. There are tabs for 'Members', 'Privileges', and 'More'. Below the tabs, a message states: 'The following table lists all entities which are members of this group.' There is a filter section with a dropdown set to 'All members' and a search input for 'Member name', with 'Apply filter' and 'Reset' buttons. Below the filter is a 'Remove selected members' button. The table has a header with 'Entity name' and 'Membership'. At the bottom, there is a 'Show: 50' dropdown and pagination text: 'Showing 1-0 of 0 · First | Prev | Next | Last'.

5. Into the main “**moodle**” folder create these groups:

a) Computer Science Course:

- Name: **Computer Science Course**
- ID: **computer**
- Description: **Brief description of Computer Science Course**

b) Mathematics Course

- Name: **Mathematics Course**
- ID: **mate**
- Description: **Brief description of Mathematics Course**

c) Philosophy and Modern Languages Course

- Name: **Philosophy and Modern Languages Course**
- ID: **philo**
- Description: **Brief description of Philosophy and Modern Languages Course**

6. Add all members that you want on the proper group and, in the same time, assign their privileges:

The screenshot shows the Moodle Grouper interface for a group named "Computer Science Course". On the left is a sidebar with "Quick links" (My groups, My folders, My favorites, My services, Admin UI, Lite UI) and "Browse folders" (Root, Grouper Administration, QS University of Bristol, SPhost for GN3+). The main content area has a breadcrumb trail: Home > Root > SPhost for GN3+ > Moodle Platform > Computer Science Course. Below the breadcrumb is the group name "Computer Science Course" with an "Add members" button and a "More actions" dropdown. A form for adding members is displayed, with "Member name or ID:" set to "Name Surname". Under "Assign these privileges:", "Default privileges" is unselected and "Custom privileges" is selected. The following checkboxes are checked: MEMBER, ADMIN, UPDATE, READ, VIEW, OPTIN, OPTOUT, ATTRIBUTE READ, and ATTRIBUTE UPDATE. An "Add" button is visible. Below the form is a "Brief description of Computer Science Course" and a "More" dropdown. A "Members" tab is active, showing a message: "The following table lists all entities which are members of this group." Below this is a "Filter for:" section with "All members" selected and a search box for "Member name". A "Remove selected members" button is present. A table header shows "Entity name" and "Membership". At the bottom, it says "Showing 1-0 of 0" with navigation links "First | Prev | Next | Last".

This screenshot is identical to the one above, showing the Moodle Grouper interface for the "Computer Science Course" group. The "Add members" form is still visible, but now only the "MEMBER" checkbox is checked under "Assign these privileges:". The "Add" button remains visible. The rest of the interface, including the sidebar, breadcrumb trail, and the empty members table, is the same as in the previous screenshot.

7. Wait at least 5 minutes, to permit to the cron scripts to be executed, and try to access into the Moodle site with one member and navigate through the courses pages to understand if your user is administrator or not.
If you are administrator of a Moodle course you can find, into its “**Administration**” => “**Course administration**” => “**Users**”, the “**Permissions**” voice.

5 How to prevent the access to an user

1. Add the user that you want block to the “**blocked**” group on the Grouper instance.
2. After that, when the user will try to access on the Wiki, he will meet the **401 Unauthorized** error page.

6 Bibliography

[1] Moodle: <https://moodle.org/>