HOWTO Install and Configure Grouper 2.2.1
on Ubuntu Linux 12.04

18 November 2014

Author: Marco Malavolti, Andrea Biancini
## Index

1) Introduction ................................................................................................................................. 3
2) Packages required ....................................................................................................................... 3
3) Phase 1 – Installation of Grouper ............................................................................................... 4
   3.1) Install the Operating System (Ubuntu 12.04 – Precise Pangolin) on target machines .......... 4
   3.2) Prepare the environment ...................................................................................................... 4
   3.3) Install Grouper ................................................................................................................... 5
4) Phase 2 – Shibbolize Grouper and add Subjects to DB ............................................................ 8
   4.1) Install a Shibboleth Service Provider on Grouper machine ............................................... 8
   4.2) Add Subjects to Grouper DB .............................................................................................. 9
5) Phase 3 – Install the Grouper VOOT Connector ..................................................................... 11
   5.1) Prepare the environment ................................................................................................... 11
6) Phase 4 – Configure an Attribute Authority on Grouper machine .......................................... 13
7) Phase 5 – Configure a Shibboleth SP to use the “isMemberOf” attribute ............................. 18
8) Phase 6 – Configure Grouper to release the “isMemberOf” attribute to a Service Provider .... 20
1 Introduction

This is a tutorial for users that want to know how to install Grouper on a Ubuntu Linux 12.04 machine and that want to know how to add the attribute “isMemberOf” and retrieve eduPersonEntitlement from Grouper.

2 Packages required

- ntp
- vim
3 Phase 1 – Installation of Grouper

3.1 Install the Operating System (Ubuntu 12.04 – Precise Pangolin) on target machines

Install Ubuntu Linux 12.04 on the target machines, from the installation parameters, choose ONLY “Standard system utilities” and “SSH server” to minimize the number of packages to be installed on the target machine.

Configure the network and the name resolution so that the machine reachable with its FQDN (as returned by `hostname -f` command). It can be obtained by editing the `/etc/hosts` file.

Install the following packages and their dependencies:

- `sudo apt-get install vim ntp`

3.2 Prepare the environment

1. Assume the role of root user for all the process steps:
   - `sudo su -`

2. Install the required packages and their dependencies:
   - `apt-get install python-software-properties`
   - `add-apt-repository ppa:webupd8team/java`
   - `apt-get update ; sudo apt-get dist-upgrade`
   - `apt-get install oracle-jdk7-installer ant tomcat7 dos2unix mysql-server git`

3. Configure JAVA_HOME for correctly execution of tomcat7:
   a) Select the ‘oracle’ version of Java after execute the command:
      - `update-alternatives --config java`
   b) Add the following line to “/etc/default/tomcat7” to set the JAVA_HOME:
      - `JAVA_HOME="/usr/lib/jvm/java-7-oracle"

4. Create the Grouper Log directories:
   - `mkdir /var/log/grouperUi ; chown tomcat7:tomcat7 /var/log/grouperUi`
   - `mkdir /var/log/grouperWs ; chown tomcat7:tomcat7 /var/log/grouperWs`

5. Create the Grouper DB:
   - `mysql -u root -p`
   - `mysql> create database grouper;`
   - `mysql> create user 'grouperdb'@'localhost' identified by '###DB-PASSWORD###';`
   - `mysql> grant all on grouper.* to 'grouperdb'@'localhost' identified by '###DB-PASSWORD###';`
   - `mysql> flush privileges;`
   - `mysql> exit;`
3.3 Install Grouper

1. Download the Grouper installer, extract it and move it in the /opt directory:
   - cd /usr/local/src
   - wget http://software.internet2.edu/grouper/release/2.2.1/grouper.installer-2.2.1.tar.gz
   - tar xzvf grouper.installer-2.2.1.tar.gz
   - mv grouper.installer-2.2.1 /opt/grouper

2. Execute the installer of Grouper:
   - cd /opt/grouper
   - java -jar grouperInstaller.jar

3. Answer to the questions as follows:

   - Do you want to 'install' a new installation of grouper, or 'upgrade' an existing installation (enter: 'install' or 'upgrade' or blank for the default) [install]: install
   - Enter in the Grouper install directory (note: better if no spaces or special chars) [/opt/grouper]: /opt/grouper
   - Enter the default IP address for checking ports (just hit enter to accept the default unless on a machine with no network, might want to change to 127.0.0.1): [0.0.0.0]: 0.0.0.0
   - Do you want to set gsh script to executable (t|f)? [t]: t
   - Do you want to run dos2unix on gsh.sh (t|f)? [t]: t
   - Do you want to use the default and included hsqldb database (t|f)? [t]: f
   - Database user [sa]: grouperdb
   - Database password (note, you aren't setting the pass here, you are using an existing pass, this will be echoed back) [<blank>]: ###DB-PASSWORD###
   - Don’t care if this message appears:
   - Checking database with query: select 1
   - GRAVE: Problem rolling back
   - com.mysql.jdbc.exceptions.MySQLNonTransientConnectionException: Can’t call rollback when autocommit=true
     at com.mysql.jdbc.SQLWarning.createSQLException(SQLWarning.java:888)
     at com.mysql.jdbc.Connection.rollback(Connection.java:5257)
     at edu.internet2.middleware.grouperInstaller.util.GiDbUtils.rollbackQuietly(GiDbUtils.java:419)
     at edu.internet2.middleware.grouperInstaller.util.GiDbUtils.listSelect(GiDbUtils.java:403)
     at edu.internet2.middleware.grouperInstaller.util.GiDbUtils.listSelect(GiDbUtils.java:323)
     at edu.internet2.middleware.grouperInstaller.util.GiDbUtils.select(GiDbUtils.java:334)
     at edu.internet2.middleware.grouperInstaller.util.GiDbUtils.checkConnection(GiDbUtils.java:474)
     at edu.internet2.middleware.grouperInstaller.GrouperInstaller.checkDatabaseConnection(GrouperInstaller.java:4546)
     at edu.internet2.middleware.grouperInstaller.GrouperInstaller.mainInstallLogic(GrouperInstaller.java:3102)
     at edu.internet2.middleware.grouperInstaller.GrouperInstaller.mainLogic(GrouperInstaller.java:849)
     at edu.internet2.middleware.grouperInstaller.GrouperInstaller.main(GrouperInstaller.java:207)
   - It is enough that this message appears on the last line "Successfully tested database connection"
1. Do you want to init the database (delete all existing grouper tables, add new ones) (t|f)? t
2. Do you want to add quickstart subjects to DB (t|f)? [t]: t
3. Do you want to add quickstart data to registry (t|f)? [t]: t
4. Do you want to start the Grouper loader (daemons)?
   (note, if it is already running, you need to stop it now, check ps -ef | grep gsh | grep loader) (t|f)? [f]: t
5. Do you want to set the tomcat memory limit (t|f)? [t]: t
6. Do you want to run dos2unix on tomcat.sh files (t|f)? [t]: t
7. What ports do you want tomcat to run on (HTTP, JK, shutdown): [8080, 8009, 8005]: 8080, 8009, 8005
8. The tomcat HTTP port is in use or unavailable: 8080, do you want to pick different ports? (t|f): f
9. Do you want to set URIEncoding to UTF-8 in tomcat server.xml <Connector> elements (t|f)? [t]: t
10. Should we check ports to see if tomcat was able to stop (t|f)? [t]: t
11. Do you want to set the log dir of UI (t|f)? [t]: t
12. Enter the UI log dir: [/opt/grouper/apache-tomcat-6.0.35/logs/grouperUi]: /var/log/grouperUi
13. Enter the URL path for the UI [grouper]: grouper
14. Enter the GrouperSystem password: ###GROUPER-PASSWORD###
15. Do you want to set the GrouperSystem password in /opt/grouper/apache-tomcat-6.0.35/conf/tomcat-users.xml? [t]: t
16. Should we stop tomcat anyway? (t|f)? [f]: f
17. Should we check ports to see if tomcat was able to start (t|f)? [t]: t
18. The Grouper WS has been built in the past, do you want it rebuilt? (t|f) [t]: t
19. Should we check ports to see if tomcat was able to stop (t|f)? [t]: t
20. Do you want to set the log dir of WS (t|f)? [t]: t
21. Enter the WS log dir: [/opt/grouper/apache-tomcat-6.0.35/logs/grouperWs]: /var/log/grouperWs
22. Enter the URL path for the WS [grouper-ws]: grouper-ws
23. Should we stop tomcat anyway? (t|f)? [f]: f
24. Should we check ports to see if tomcat was able to start (t|f)? [t]: t
25. Do you want to install the provisioning service provider (t|f)? [t]: t

4. Test the correct execution of Grouper into the default environment by opening the web page:
   • http://###YOUR.GROUPER.FQDN###:8080/grouper/
   (As username use “GrouperSystem”, as password use “###GROUPER-PASSWORD###”)

5. Remove all unnecessary files:
   • cd /opt/grouper ; rm -rf *.tar ; rm -f *.tar.gz

6. Edit the /etc/default/tomcat7 file by adding this JAVA_OPTS line under the default ones:

   \n
   JAVA_OPTS="-server -Xmx512M -XX:MaxPermSize=256M"

7. Replace the default tomcat-users.xml of Tomcat7 with the grouper’s ones:
   • cp /opt/grouper/apache-tomcat-6.0.35/conf/tomcat-users.xml
     /etc/tomcat7/tomcat-users.xml

8. Edit the /etc/tomcat7/server.xml as follows:

   <Host name="localhost" appBase="webapps"
    unpackWARs="true" autoDeploy="true"
9. Replace “$” with “#” on the value "${uiException.class.simpleName}" into /opt/grouper/grouper.ui-2.2.1/dist/grouper/WEB-INF/jsp/dynamicTile.jsp file.

10. Remove the log's files from their directories to permit to Tomcat7 to write its logs:
    • rm -f /var/log/grouperUi/*
    • rm -f /var/log/grouperWs/*

11. Shutdown the Grouper's Tomcat Server to leave place to the Apache Tomcat7 installed:
    • sh /opt/grouper/apache-tomcat-6.0.35/bin/shutdown.sh

12. Start the Apache Tomcat7 server:
    • service tomcat7 start

13. Test the correct execution of Grouper into the default environment by opening the web page:
    http://###YOUR.GROUPER.FQDN###:8080/grouper/
    (As username use “GrouperSystem”, as password use “###GROUPER-PASSWORD###”)
**4 Phase 2 – Shibbolize Grouper and add Subjects to DB**

**4.1 Install a Shibboleth Service Provider on Grouper machine**

1. Install a Shibboleth SP for grouper application, protect it with SSL/HTTPS certificate and exchange its metadata with your federation. This SP will authenticate the users via EPPN attribute, then modify the “shibboleth2.xml” in this way:

   ```xml
   ...<ApplicationDefaults entityID="https://grouper.example.com/shibboleth"
   REMOTE_USER="eppn">
   ...```

2. Add AJP support to Tomcat7:
   a) Modify the `/etc/tomcat7/server.xml` file by adding this:

   ```xml
   <Connector port="8009" protocol="AJP/1.3" tomcatAuthentication="false"
   redirectPort="8443" />
   ```

   b) Ensure that the mod “proxy_ajp” is enabled:
   - `a2enmod proxy_ajp ; service apache2 restart`

3. Create the apache2 site “/etc/apache2/sites-available/grouper.conf” with this content:

   ```
   ProxyPass /grouper ajp://localhost:8009/grouper
   ProxyPassReverse /grouper ajp://localhost:8009/grouper
   ProxyPass /grouper-ws ajp://localhost:8009/grouper-ws
   ProxyPassReverse /grouper-ws ajp://localhost:8009/grouper-ws
   <Location /grouper>
   AuthType shibboleth
   ShibRequireSession On
   require valid-user
   </Location>
   ```

   And enable it:
   - `a2ensite grouper.conf ; service apache2 restart`
4.2 Add Subjects to Grouper DB

1. Comment out all the "<security-constraint>" , "<login-config>" and "<security-role>" from /opt/grouper/grouper.ui-2.2.1/dist/grouper/WEB-INF/web.xml.

2. Create the bash script "/root/addSubject.sh" that permits you to add a Subject to Grouper:

```bash
#!/bin/bash

function ask_param {
    local VALUE=$1
    local NAME=$2
    if [ -z "$VALUE" ]; then
        read -p "Insert the $NAME: " VALUE
    fi
    echo $VALUE
}

EPPN=$(ask_param "$1" "eppn")
NAME=$(ask_param "$2" "name")
SURNAME=$(ask_param "$3" "surname")
EMAIL=$(ask_param "$4" "email")

echo "The provided informations for the user to be inserted in Grouper, are as follows:"

echo ""
echo "eppn:    $EPPN"
echo "name:    $NAME"
echo "surname: $SURNAME"
echo "email:   $EMAIL"
echo ""
echo "Press ENTER to continue or CTRL+C to exit..."
read -p "" DEL

cd /opt/grouper/grouper.apiBinary-2.2.1
./bin/gsh <<EOF
addSubject("$EPPN", "person", "$NAME $SURNAME");
EOF
cd -

#GrouperSession.startRootSession();
#subj = findSubject("horberg@umu.se");
#attr = subj.getAttribute();
#attr.put("loginid", new HashSet(java.util.Arrays.asList(new String[] { "horberg@umu.se" })));

mysql --user=grouperdb --password=###DB-PASSWORD### --database=grouper <<EOF
insert into subjectattribute values('$EPPN','loginid','$EPPN','$EPPN');
insert into subjectattribute values('$EPPN','description','$NAME $SURNAME',lower('$NAME $SURNAME'));
insert into subjectattribute values('$EPPN','name','$NAME $SURNAME',lower('$NAME $SURNAME'));
insert into subjectattribute values('$EPPN','email','$EMAIL',lower('$EMAIL'));
EOF
```
3. Create the bash script “/root/addMemberToSysAdmin.sh” that permits you to add a Member to the Sysadmin group:

```bash
#!/bin/bash
function ask_param {
    local VALUE=$1
    local NAME=$2
    if [ -z "$VALUE" ]; then
        read -p "Insert the $NAME: " VALUE
    fi
    echo $VALUE
}
EPPN=$(ask_param "$1" "eppn")
echo "The provided information for the user to be inserted in Grouper, are as follows:"
echo ""
echo "eppn:    $EPPN"
echo ""
echo "Press ENTER to continue or CTRL+C to exit..."
read -p "" DEL
cd /opt/grouper/grouper.apiBinary-2.1.5
./bin/gsh <<EOF
addMember("etc:sysadmingroup","$EPPN");
EOF
```

4. Add the right privileges to addSubject.sh and to addMemberToSysAdmin.sh:
   - chmod +x /root/addSubject.sh /root/addMemberToSysAdmin.sh

5. Execute the addSubject.sh script to add the user stored on your IdP (as many as you want) into Grouper DB:
   - /bin/bash /root/addSubject.sh

6. Modify the callLogin path from “login.do” to “home.do” into “structs-config.xml” file:
   - vim /opt/grouper/grouper.ui-2.2.1/dist/grouper/WEB-INF/struts-config.xml

```xml
<action path="/callLogin" scope="request"
type="edu.internet2.middleware.grouper.ui.actions.CallLoginAction"
unknown="false" validate="false">
   <forward name="callLogin" path="/home.do" redirect="true"/>
</action>
```

7. Modify the grouper.properties to be able to edit the system groups by adding the following lines:
   - vim /opt/grouper/grouper.ui-2.2.1/dist/grouper/WEB-INF/classes/grouper.properties:
# HOWTO Install and Configure Grouper on Ubuntu Linux 12.04

```bash
# if groups like the wheel group should be auto-created for convenience (note: check config needs to be on)
configuration.autocreate.system.groups = true

# A wheel group allows you to enable non-GrouperSystem subjects to act
groups.wheel.use = true
```

8. Restart Tomcat7 service to apply the changes:
   - service tomcat7 restart

9. Add a created Subject to SysAdmin group:
   - /bin/bash /root/addMemberToSysAdmin.sh

10. Test the correct execution of Grouper on HTTPS by opening the web page:
    ```
    https://###YOUR.GROUPER.FQDN###/grouper/
    ```
    (And log-in into the grouper application with an IdP that releases the eppn of the user inserted with addSubject.sh script)

## 5 Phase 3 – Install the Grouper VOOT Connector

### 5.1 Prepare the environment

1. Download the code of Grouper VOOT Connector into `/usr/local/src`:
   - cd /usr/local/src
   - wget http://software.internet2.edu/grouper/release/2.2.1/grouper.vootBinary-2.2.1.tar.gz
   - tar zxf grouper.vootBinary-2.2.1.tar.gz

2. Extract and copy the `grouperVoot.jar` into the right position:

3. Modify the `sources.xml` by removing every "^M" character:
   - dos2unix /opt/grouper/grouper.ws-2.2.1/grouper-ws/build/dist/grouper-ws/WEB-INF/classes/sources.xml

   and ensure to see this:

   ```
   <!-- If using emails and need email addresses in sources, set which attribute has the email address in this source -->
   <init-param>
     <param-name>emailAttributeName</param-name>
     <param-value>email</param-value>
   </init-param>
   ```
4. Setup the Grouper **web.xml**:
   - `vim /opt/grouper/grouper.ws-2.2.1/grouper-ws/build/dist/grouper-ws/WEB-INF/web.xml`

```xml
<!-- Add this to filter-mapping -->
<filter-mapping>
  <filter-name>Grouper service filter</filter-name>
  <url-pattern>/voot/*</url-pattern>
</filter-mapping>

<!-- Add this to servlet -->
<servlet>
  <servlet-name>VootServlet</servlet-name>
  <display-name>Voot Servlet</display-name>
  <servlet-class>edu.internet2.middleware.grouperVoot.VootServlet</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>

<!-- Add this to servlet-mapping -->
<servlet-mapping>
  <servlet-name>VootServlet</servlet-name>
  <url-pattern>/voot/*</url-pattern>
</servlet-mapping>

<!-- Add this to security-constraint -->
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Voot services</web-resource-name>
    <url-pattern>/voot/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>grouper_user</role-name>
  </auth-constraint>
</security-constraint>
```

5. Restart Tomcat server:
   - `service tomcat7 restart`
6 Phase 4 – Configure an Attribute Authority on Grouper machine

1. Download the Shibboleth IdP package from Internet2 and store it into /usr/local/src directory:
   - cd /usr/local/src
   - wget https://shibboleth.net/downloads/identity-provider/latest/shibboleth-identityprovider-2.4.3-bin.tar.gz

2. Install the IdP on the Grouper VM into /opt/shibboleth-idp directory:
   - tar zxf shibboleth-identityprovider-2.4.3-bin.tar.gz
   - export JAVA_HOME="/usr/lib/jvm/java-7-oracle"
   - cd /usr/local/src/shibboleth-identityprovider-2.4.0 ; sh install.sh

3. Assign the right privileges by executing this:
   - chown tomcat7 /opt/shibboleth-idp/logs/
   - chown tomcat7 /opt/shibboleth-idp/metadata/
   - chown tomcat7 /opt/shibboleth-idp/credentials/
   - chmod 400 /opt/shibboleth-idp/credentials/idp.key
   - chmod 644 /opt/shibboleth-idp/credentials/idp.crt
   - chown tomcat7 /opt/shibboleth-idp/credentials/idp.key
   - chown tomcat7 /opt/shibboleth-idp/credentials/idp.crt

4. Deploy the idp.war application:
   - vim /etc/tomcat7/Catalina/localhost/idp.xml:
     ```xml
     <Context docBase="/opt/shibboleth-idp/war/idp.war"
             privileged="true"
             antiResourceLocking="false"
             antiJARLocking="false"
             unpackWAR="false"
             swallowOutput="true" />
     ```

5. Modify the “/etc/apache2/sites-enabled/default-ssl” by adding the bold text under the Virtual-Host:

   ```xml
   <VirtualHost _default_:443>
       ServerName grouper.example.com:443
       ...
       ProxyPass /idp ajp://localhost:8009/idp
       ProxyPassReverse /idp ajp://localhost:8009/idp
   </VirtualHost>
   ```
6. Copy the “/etc/apache2/sites-enable/default-ssl” to “/etc/apache2/sites-enable/default-ssl-8443” and change all “443” port to “8443”, and add this:

```
<VirtualHost _default_:8443>
    ServerName grouper.fqdn.example.com:8443
    ...
    SSLCertificateFile /opt/shibboleth-idp/credentials/idp.crt
    SSLCertificateKeyFile /opt/shibboleth-idp/credentials/idp.key
    ...
    SSLVerifyClient optional_no_ca
</VirtualHost>
```

7. Add the port 8443 to those that Apache2 listen by editing the “/etc/apache2/ports.conf” file:

```
Listen 8443
NameVirtualHost *:8443
```

8. Install mysql-java-connector:
   - sudo apt-get install libmysql-java
   - cp /usr/share/java/mysql-connector-java-5.1.16.jar /opt/shibboleth-idp/lib
   - cp /usr/share/java/mysql-connector-java-5.1.16.jar /var/lib/tomcat7/common

9. Restart the Tomcat7 and Apache2 service:
   - service tomcat7 restart
   - service apache2 restart

10. Configure the IdP to retrieve the federation’s metadata that contain the Grouper SP Metadata

11. Modify the “attribute-resolver.xml” on grouper machine by adding:
   - A new DataConnector:

```
<!-- Grouper Database connector -->
<resolver:DataConnector xsi:type="RelationalDatabase" xmlns="urn:mace:shibboleth:2.0:resolver:dc" id="grouper">
    <ApplicationManagedConnection jdbcDriver="com.mysql.jdbc.Driver"
        jdbcURL="jdbc:mysql://localhost:3306/grouper"
        jdbcUserName="grouperdb"
        jdbcPassword="geantdbpassword"/>

    <QueryTemplate>
        <![CDATA[
            SELECT DISTINCT REPLACE(GROUP_NAME, CONCAT(SUBSTRING_INDEX(SUBSTRING_INDEX($requestContext.getPeerEntityId(), '//', -1), '/', 1), ':'), '') AS GROUP_NAME
            FROM grouper_memberships_lw_v
            WHERE subject_id = '$requestContext.principalName'
            AND GROUP_NAME LIKE CONCAT(SUBSTRING_INDEX(SUBSTRING_INDEX($requestContext.getPeerEntityId(), '//', -1), '/', 1), '%')
            AND list_name = 'members'
            AND GROUP_NAME NOT LIKE '%:service:%'
        ]]]>
    </QueryTemplate>
```
• A new AttributeDefinition:

```xml
<Column columnName="GROUP_NAME" attributeID="isMemberOf" type="String" />
</resolver:DataConnector>

<resolver:DataConnector xsi:type="RelationalDatabase" xmlns="urn:mace:shibboleth:2.0:resolver:dc" id="grouperServices">
  <ApplicationManagedConnection jdbcDriver="com.mysql.jdbc.Driver"
    jdbcURL="jdbc:mysql://localhost:3306/grouper"
    jdbcUserName="grouperdb"
    jdbcPassword="geantdbpassword" />
  
  <QueryTemplate>
          <![CDATA[
            SELECT DISTINCT CONCAT('urn:mace:garr.it:',
            REPLACE(REPLACE(GROUP_NAME, ':service:authorized', ''), CONCAT(SUBSTRING_INDEX(SUBSTRING_INDEX($requestContext.getPeerEntityId(), '//', -1), '/', 1), ':'), '')) AS GROUP_NAME
            FROM grouper_memberships_lw_v
            WHERE subject_id = '$requestContext.principalName'
            AND GROUP_NAME LIKE CONCAT(SUBSTRING_INDEX(SUBSTRING_INDEX($requestContext.getPeerEntityId(), '//', -1), '/', 1), '%')
            AND list_name = 'members'
            AND GROUP_NAME LIKE '%:service:authorized'
          ]]>}
  </QueryTemplate>
</resolver:DataConnector>

<!-- AttributeDefinition for "isMemberOf" attribute -->
<resolver:AttributeDefinition id="isMemberOf" xsi:type="ad:Simple" sourceAttributeID="isMemberOf">
  <resolver:Dependency ref="grouper" />
  <resolver:DisplayDescription xml:lang="en">List of groups retrieved from Grouper</resolver:DisplayDescription>
  <resolver:DisplayDescription xml:lang="it">Elenco dei gruppi ottenuti da Grouper</resolver:DisplayDescription>
  <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:isMemberOf" />
  <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:1.2.840.113556.1.666.1" friendlyName="isMemberOf" />
</resolver:AttributeDefinition>

<!-- AttributeDefinition for "eduPersonEntitlement" attribute -->
<resolver:AttributeDefinition id="eduPersonEntitlement" xsi:type="ad:Simple" sourceAttributeID="eduPersonEntitlement">
  <resolver:Dependency ref="grouperServices" />
  <resolver:DisplayDescription xml:lang="en">List of authorized services retrieved from Grouper</resolver:DisplayDescription>
  <resolver:DisplayDescription xml:lang="it">Elenco dei servizi autorizzati da Grouper</resolver:DisplayDescription>
</resolver:AttributeDefinition>
```
A change to the Principal Connector:

```xml
<!-- ------------------------ -->
<!--      Principal Connectors      -->
<!-- ------------------------ -->
<!--
<resolver:PrincipalConnector xsi:type="pc:Transient" id="shibTransient" nameIDFormat="urn:mace:shibboleth:1.0:nameIdentifier"/>
<resolver:PrincipalConnector xsi:type="pc:Transient" id="saml1Unspec" nameIDFormat="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified"/>
<resolver:PrincipalConnector xsi:type="pc:Transient" id="saml2Transient" nameIDFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"/>
-->
<resolver:PrincipalConnector xsi:type="pc:Direct" id="saml1Direct" nameIDFormat="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified"/>
<resolver:PrincipalConnector xsi:type="pc:Direct" id="saml2Direct" nameIDFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified"/>
```

12. Modify the “attribute-filter.xml” of Grouper IdP by adding this:

```xml
<!-- Release the transient ID to anyone -->
<afp:AttributeFilterPolicy id="releaseTransientIdToAnyone">
    <afp:Policy RequirementRule xsi:type="basic:ANY"/>
    <afp:AttributeRule attributeID="transientId">
        <afp:PermitValueRule xsi:type="basic:ANY"/>
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="isMemberOf">
        <afp:PermitValueRule xsi:type="basic:ANY"/>
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="eduPersonEntitlement">
        <afp:PermitValueRule xsi:type="basic:ANY"/>
    </afp:AttributeRule>
</afp:AttributeFilterPolicy>
```

13. Don't restart Tomcat7 before the end of Phase 5 !!!!
7 Phase 5 – Configure a Shibboleth SP to use the “isMemberOf” attribute and eduPersonEntitlement from Grouper AA

1. Install and Configure a Shibboleth SP (sp-test-grouper.example.com) and exchange its metadata with Grouper AA and Federation.

2. Modify the “shibboleth2.xml” by adding this AttributeResolver:

   ```xml
   <AttributeResolver type="Query" subjectMatch="true">
     <AttributeResolver type="SimpleAggregation" attributeId="eppn" format="urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified">
       <Entity>https://grouper.example.it/idp/shibboleth</Entity>
     </AttributeResolver>
   </AttributeResolver>
   ```

3. Edit the “attribute-map.xml” to resolve the new attribute “isMemberOf” and “eduPersonEntitlement”:

   ```xml
   <Attribute name="urn:mace:dir:attribute-def:eduPersonEntitlement" id="entitlement" />
   <Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7" id="entitlement" />
   ...
   <Attribute name="urn:oid:1.2.840.113556.1.666.1" id="isMemberOf"/>
   ```

4. Configure the policy of the SP to retrieve “isMemberOf” and “eduPersonEntitlement” attributes:

   ```xml
   ...
   <!-- Require isMemberOf and eduPersonEntitlement to be released only by Grouper AA -->
   <afp:AttributeRule attributeID="isMemberOf">
     <afp:PermitValueRule xsi:type="basic:AttributeIssuerString" value="https://grouper.example.it/idp/shibboleth" />
   </afp:AttributeRule>
   <afp:AttributeRule attributeID="eduPersonEntitlement">
     <afp:PermitValueRule xsi:type="basic:AttributeIssuerString" value="https://grouper.example.it/idp/shibboleth" />
   </afp:AttributeRule>
   ...
   <!-- Catch-all that passes everything else through unmolested. -->
   <afp:AttributeRule attributeID="*">
     ...
   </afp:AttributeRule>
   ```

5. Restart the “shibd” service:
   - service shibd restart

6. Restart Tomcat7 on Grouper machine

7. Configure the Federation’s IdPs to release the “eduPersonPrincipalName” of their users to the Grouper Application and other SPs.
NOTES:
The Federation's IDPs must know, by metadata exchange, the Grouper SP and the other SPs.

The Federation's SPs must know, by metadata exchange, the Grouper AA and the other IdPs.
Phase 6 – Configure Grouper to release the “isMemberOf” and “eduPersonEntitlement” attributes to a Service Provider

1. Open https://#YOUR.GROUPER.FQDN#/grouper and, working as Admin:
   - Create a new folder that has as FolderID the FQDN of the SP to which Grouper will provide the isMemberOf attribute and the eduPersonEntitlement attribute:
- Refresh the web page to update the visible folders.
• Move on the new folder and create another two folder
  ○ One, called “service”, that will be useful to quickly disable the users:
One, called “secure”, for Secure Application:

- Create another new folder called “service” into the “secure” folder:
• Create 3 new groups into the “service” folder:
  ○ authorized
  ○ blocked
  ○ eligible
HOWTO Install and Configure Grouper on Ubuntu Linux 12.04

New group

Create in this folder: SPHost for GNE3+ Secure Application service

Group name: blocked

Group ID: blocked

Description: Group used to identify the users that are blocked when they try to use the resource
• Create a new group called “quickdisable”, in the folder “secure” of Sphost, that will contain all the users blocked on all services hosted by the SPhost relying party:

• Add the “quickdisable” group to the “blocked” group of the Secure Application:
• Make the “authorized” group a composite group that involves “eligible” and “blocked” groups:
• Create the following new groups into the **secure** folder and **add** members by searching their surname:
  - administrator
  - member
• Add the “administrator” and “member” group as member of the “eligible” group created into “service” folder of Secure Application:
Finally add members to the “administrator” or “member” group created. This permits to Grouper to release the isMemberOf attribute and the eduPersonEntitlement attribute of the subjects added for your “secure” service on Grouper application.

2. Try to log-in on a simple application protected by the SP for which you have created a folder into Grouper and see if the attributes “isMemberOf” and “eduPersonEntitlement” are released by checking the /Shibboleth.sso/Session page of your SP.

```
entitlement = urn:mace:garr:it:secure
...
...isMemberOf = secure:administrator

Attribute = Value
```