

milestone 3 - Installing & Configuring Drupal Filter

About the Drupal Filter

 The Drupal Filter is a library package that gets installed in the Fedora Web-App on the Tomcat Application Server. It is the Fedora side of the communication between Islandora's Tuque library and the Fedora repository system.

1. Download the Latest Version of the Islandora Drupal Filter

Download the appropriate Drupal Filter file, and copy it to \$FEDORA_HOME/tomcat/webapps/fedora/WEB-INF/lib

Place the file in:

```
$ wget https://github.com/Islandora/islandora_drupal_filter/releases/download/v7.1.9/fcrepo-drupalauthfilter-3.8.1.jar  
$ cp -v fcrepo-drupalauthfilter-3.8.1.jar $FEDORA_HOME/tomcat/webapps/fedora/WEB-INF/lib
```

2. Make the Fedora Repository Aware of the New Filter

Navigate to \$FEDORA_HOME/server/config and open the file jaas.conf in a text editor.

To allow the Drupal Servlet Filter to authenticate against Drupal's database, replace the "fedora-auth" entry with the following lines that reference the DrupalServlet filters class files:

```
fedora-auth  
{  
org.fcrepo.server.security.jaas.auth.module.XmlUsersFileModule required  
debug=true;  
ca.upei.roblib.fedora.servletfilter.DrupalAuthModule required  
debug=true;  
};
```

3. Configure the new filter to connect to Drupal (filter-drupal.xml)

Create the file filter-drupal.xml in \$FEDORA_HOME/server/config using the following text as a template (or download a sample file from https://raw.githubusercontent.com/Islandora/islandora_drupal_filter/master/filter-drupal.xml), then modify the attributes of the <connection> tag to match the server, port, database name, username and password of your site's Drupal database.

Fedora requires a separate <connection> entry for each connecting Drupal site.

The Drupal Filter does not currently escape the database url before attempting to connect to the Mysql database, which can cause problems if the user name or password has '%' symbol within it.


```

<!--File to hold drupal connection info for the FilterDrupal servlet filter. For multisite drupal installs you can include multiple connection elements. We will query all the databases and assume any user in any drupal db with the same username and password are the same user. We will gather all roles for that user from all databases. This is a potential security risk if a user in one drupal db has the same username and password as another user in a separate drupaldb. We are also assuming all drupal dbs to be mysql. This file should be located in the same directory as the fedora.cfcg file-->

<FilterDrupal_Connection>
  <connection server="localhost" dbname="[drupal_database]" user="[drupal_db_user]" password="[drupal_db_password]" port="3306">
    <sql>
      <!--Different sql statement for each connection. This is for drupal multisites that are setup using one database with table prefixes. We don't do this but some people might.-->
      SELECT DISTINCT u.uid AS userid, u.name AS Name, u.pass AS Pass, r.name AS Role FROM (users u LEFT JOIN users_roles ON u.uid=users_roles.uid) LEFT JOIN role r ON r.rid=users_roles.rid WHERE u.name=? AND u.pass=?;
    </sql>
  </connection>
</FilterDrupal_Connection>

```

Now change the owner to the user that runs the web server, as Islandora needs to modify the file:

```
# chown www-data:www-data filter-drupal.xml
```

If you use the Drupal servlet filter to connect to multiple Drupal databases there is potential for users with the same username in each database to access each others private objects. To avoid this, use the Drupal LDAP module. A Drupal multi-site environment utilizing the LDAP module for all sites ensures a unique username/site configuration.

4. Stop and Restart tomcat to load the filter

This will enable the Drupal Servlet Filter.

```
# $FEDORA_HOME/tomcat/bin/shutdown.sh
# $FEDORA_HOME/tomcat/bin/startup.sh
```

5. Test the Drupal Servlet Filter

This Python script was developed to verify the database connection(s) defined in filter-drupal.xml files. The script is available at:

https://github.com/mjordan/drupal_filter_validator

To download and run the test, *on the server where Fedora is running*, issue the following command:

```
# git clone https://github.com/mjordan/drupal_filter_validator
# ./test_db_connection.py $FEDORA_HOME/server/config/filter-drupal.xml

You should see something like this:
Connection 1 - OK: connection to Drupal database successful, and user drupalAdmin has select,insert,update privileges on the users table.
Connection 2 - OK: connection to Drupal database successful, and user drupalAdmin has select,insert,update privileges on the users table.
Connection 3 - OK: connection to Drupal database successful, and user drupalAdmin has select,insert,update privileges on the users table.
```

If you don't get the a successful set of tests, please retrace your steps.

Create a milestone 3 snapshot backup:

```
# $FEDORA_HOME/tomcat/bin/shutdown.sh
# /etc/init.d/mysql stop
# /etc/init.d/apache2 stop

# cd /usr/local
# tar -cvzf fedora-milestone3.tar.gz fedora

# cd /var/lib
# tar -cvzf mysql-milestone3.tar.gz mysql

# cd /var/www
# tar -cvzf drupal-milestone3.tar.gz drupal

# /etc/init.d/mysql start
# $FEDORA_HOME/tomcat/bin/startup.sh
# /etc/init.d/apache2 start
```