

Grouper Provisioner Plugin: Prepare Apache

We recommend proxying the Tomcat container in which Grouper will run with Apache HTTP Server (Apache) to make authentication with SSO protocols like SAML/Shibboleth easier.

We assume that you have already deployed Apache HTTP Server and configured it to deliver all content securely using HTTPS. At a minimum you should configure Apache to use HTTPS for all content delivered for the locations /grouper and /grouper-ws, and for testing purposes only /examples.

1. Configure Apache with support for AJP proxying. For Debian 7 you may do

```
a2enmod proxy_ajp
```

On RHEL and its derivatives the proxy module is most likely already configured to be loaded.

2. Edit the Apache configuration for the virtual host that will support the Grouper UI and web services and add the lines below to enable AJP proxying to Tomcat:

```
ProxyPass /grouper/ ajp://localhost:8009/grouper/ retry=0
ProxyPass /grouper-ws/ ajp://localhost:8009/grouper-ws/ retry=0
ProxyPass /examples/ ajp://localhost:8009/examples/ retry=0
```

The /examples/ location will only be used for testing and will be deleted later.

3. Create a passwords file to allow simple basic auth for testing for the user 'GrouperSystem'. For example:

```
htpasswd -c /etc/apache2/passwords GrouperSystem
New password:
Re-type new password:
Adding password for user GrouperSystem
```

4. Edit the apache configuration for the virtual host that will support the Grouper UI and web services and add the lines below for simple basic auth:

```
<Location /grouper>
  AuthType Basic
  AuthName "Grouper"
  AuthUserFile /etc/apache2/passwords
  Require valid-user
</Location>

<Location /grouper-ws>
  AuthType Basic
  AuthName "Grouper"
  AuthUserFile /etc/apache2/passwords
  Require valid-user
</Location>

<Location /examples>
  AuthType Basic
  AuthName "Grouper"
  AuthUserFile /etc/apache2/passwords
  Require valid-user
</Location>
```

The /examples/ location is only used for testing and will be removed later. You may later integrate the Grouper UI with your SSO infrastructure, for example SAML/Shibboleth, but it is easier to first test using basic auth.

5. Reload the Apache configuration, eg. :

```
service apache2 reload
```