**CST-235**

**Securing Access**

**Project Report**

**12/8/2018**

**Roman Parkhomenko**

The purpose of this assignment was to develop knowledge and practice in securing a JavaEE application. We accomplished this by implementing FORM Based Authentication to the application we’ve been building this course.

To start implementing security, we had to make a configuration adjustment to our JBoss EAP. I went into the bin directory of JBoss EAP, ran a chmod 777 to grant myself write permissions, and ran the add-user script. I added a user with a role of GcuUser to the ApplicationRealm, with username “gcu” and password “Lopes2018!” (My configuration had specific security requirements). After finishing this, I double checked my configuration on the JBoss Server and saw that it had user properties in the standalone.xml file. I then went into the WEB-INF directory and added additional configuration parameters to the jboss-web.xml and web.xml files. The jboss file was just adding a security domain but the web.xml file added the details of a security constraint. This included protecting all pages in the project root from access and HTTP requests. I added allowances for the rest URL so the API could still be accessed, and set the default pages of /faces/LoginForm.xhtml and /faces/LoginError.xhtml to be accessible only by users with the GcuUser role.

After finishing the configuration, some refactoring was required to implement FORM Based authentication. In the LoginForm, I created a new form to collect the username and password. FORM-Auth requires standard HTML tags instead of <h:> tags, so I rebuilt the form and created a new LoginError.xhtml page that contains an error message when user credentials are not correct. The LoginError page also contains a button to redirect the user back to the login page.

Finally, in the FormController, we created a new onLogoff() method that invalidates the session, and redirects the user back to the login page. In the User bean we added a new init method, with the @PostConstruct annotation, that implements the Principal interface to create a user based on whether admin credentials exist.

The FORM Based Authentication is a great way to manage administrator credentials and filter application access. However, it seems really limited in flexibility since adding users requires command line access. It’s a great approach to security for smaller projects, but I don’t think this is the best approach to securing a user database for a web application.