Aim

This lab aims to guide you through setting up GlassFish server to use JDBCRealm for its authentication purposes.

Software

To finish the lab, you may need the following software:

1. NetBeans IDE version 12.2 or version 12.5
2. JDK version 1.8.0 (jdk1.8.0_202, or later)
3. GlassFish Server Open Source Edition version 5.1.0
4. JavaDB – the database server that comes with GlassFish

Overview of the Tasks

In this Lab, you will learn how to set up GlassFish Server to use JDBCRealm for an enterprise app to access protected resources

LT1. Run the Admin Console of GlassFish server
LT2. Set up user credentials on GlassFish server using JDBCRealm
LT3. Select the appropriate JDBCRealm settings and Enable the "Default Principal to Role Mapping" option
LT4. Restart GlassFish

Pre-Lab Task

Get the JavaEE 7 Tutorial from Oracle (either pdf or html)

Lab Tasks

This Lab should be run on MS Windows Platform

LT1. Run the Admin Console of GlassFish server using NetBeans

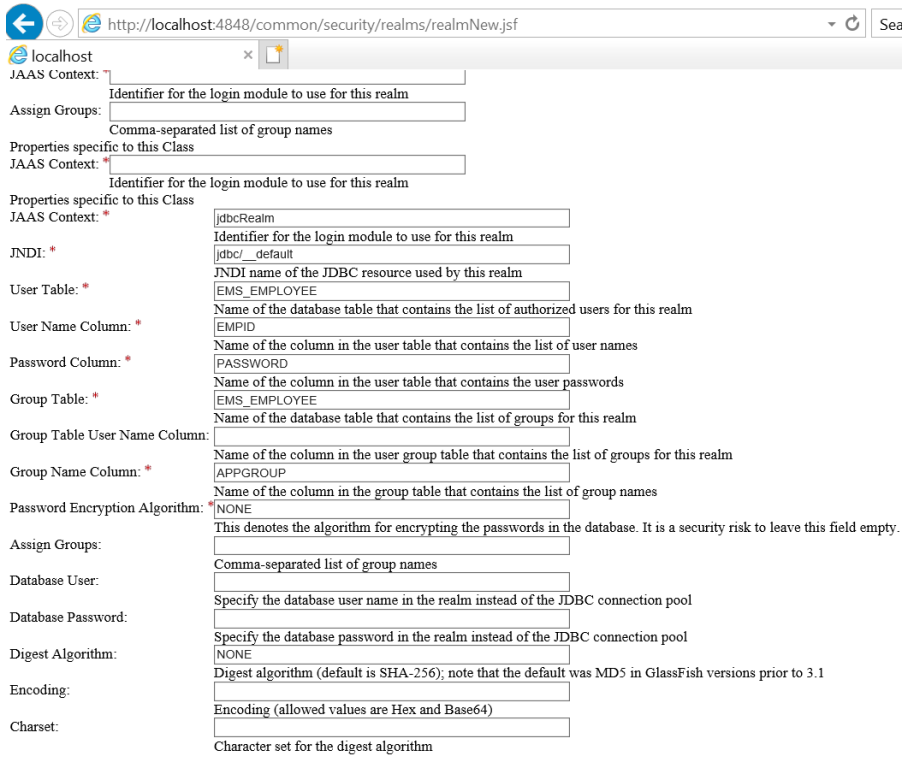
- LT1.1. Start GlassFish server in NetBeans, if needed
- LT1.2. Right click on "GlassFish Server 5.1.0" and select "View Domain Admin Console"
  - Note: A browser will start and load the "Admin Console" page of GlassFish server.
- LT1.3. Expand on the "Configurations", if needed
- LT1.4. Expand on the "server-config", if needed
- LT1.5. Expand on the "Security", if needed
- LT1.6. Select "Security"
  - Note: The browser will then show the "Security" page on the right pane. Remember this page. We will refer to this page as the "Security" page under "server-config", or simply "[server-config] > Security" page.
  - Note: Please be reminded that there is also another "Security" page under "default-config". This is not for us to use. Any changes on this "[default-config] > Security" page will have NO effect on your Glassfish server.

Our "EMS_EMPLOYEE" database table, used in previous labs, has all the user credential information stored there. In the following Lab Task, we will configure GlassFish server to point to this database table via JDBCRealm.

LT2. Set up user credentials in GlassFish Server using JDBCRealm

- LT2.1. Go to the "[server-config] > Security" page, if you have not done so
- LT2.2. Select "Realms" on the left pane
  - Note: The browser will then show the "Realms" page on the right pane.
  - Note: In the table below the "Realms" page, there should be something like "admin-realm" (a file realm actually – look under the "Class Name" heading), "certificate" (a certificate realm") and "file" (another file realm) if you have not made any changes to these realms.
- LT2.3. Click "New…" in the "Realms" page
  - Note: The browser will then show the "New Realm" page on the right pane.
- LT2.4. In the "New Realm" page,

a. Enter "jdbcRealm" in "Name:" (Assuming you do not have a "jdbcRealm" in the GlassFish server)
b. Choose "com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm" in "Class Name:" combo box
Note: The browser will then show more text fields for you to enter.
c. Click the **Add Property** button
d. Enter the following information in the corresponding field (without quotations, case sensitive)



LT2.5. Click "OK" (on the upper right corner)
Note: The browser will now bring you back to the "Realms" page.
Note: Please be reminded that there is a new entry in the "Realms" table below with "jdbcRealm" as the name. Click on this new "jdbcRealm" will bring to a page that allows you to make changes.

LT3. Select the appropriate JDBCRealm to use and Enable the "Default Principal To Role Mapping" option on GlassFish
LT3.1. Go back to the "[server-config] > Security" page, as mentioned in LT1.3 above
LT3.2. On the right pane of the "Security" page, do the following
a. Select "jdbcRealm" in the "Default Realm" related combo box
b. Check the "Enable" checkbox on the "Default Principal to Role Mapping" setting, if it has not been checked
LT3.3. Click "Save" (on the upper right corner)
Note: GlassFish server will response with "New values successfully saved."

LT4. Restart the "GlassFish" server for the settings to take effect.
Note: in this case, password is plain text as “NONE” is filled in for Password Encryption Algorithm. If you are going to do Credit Task 7.2C, if you use jdbcRealm, you should choose "SHA-256" for “Password Encryption Algorithm”. You should “hash” the password using SHA-256 algorithm, so the size of password field in the database table would be different from previous one.

- jdbcRealm set up with "SHA-256" for “Password Encryption Algorithm”:

  - "JAAS Context:" "jdbcRealm"

- "JNDI:" "jdbc/__default"
- "User Table:" "EMS_EMPLOYEE"
- "User Name Column:" "EMPID"
- "Password Column:" "PASSWORD"
- "Group Table:" "EMS_EMPLOYEE"
- "Group Table User Name Column:" "" (can be blank if using the same table)
- "Group Name Column:" "APPGROUP"
- "Password Encryption Algorithm:" "SHA-256"
- "Assign Groups:" "" (leave blank)
- "Database User:" "" (leave blank)
- "Database Password:" "" (leave blank)
- "Digest Algorithm:" "" (leave blank, GlassFish server will use the default SHA-256, which is the same as ours)
- "Encoding:" "" (leave blank)
- "Charset:" "" (leave blank, some suggest to use UTF-8 but I suggest not to do it for the time being)