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

Aim

Securing an enterprise application

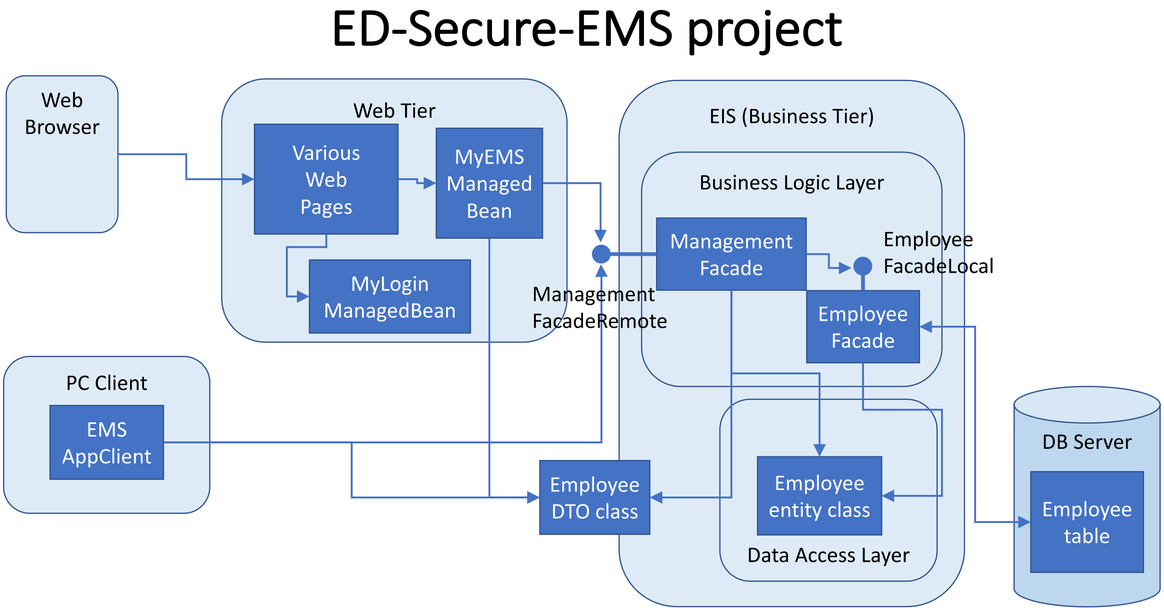

Figure 1 A rough architecture diagram of SECURE's EMS Application (in UML notation)

Case Study – A Secure Company's Employee Management System (Secure EMS)

SECURE is a company proud of developing secure enterprise applications. The company has a "secure" web-based system, called Employee Management System (EMS) to manage (that is the CRUD operations) its employees' records. Only the administrators of the EMS can login to the system and perform the normal CRUD operations. The current login credentials of the Administrators of EMS are stored in the GlassFish server as the "ED-APP-ADMIN" group (See Lab_07a_Secure_GlassFish for the credentials). Administrator can add, delete, edit, and review individual employee's record. All information of an employee's record (except the employee id, the primary key) can be changed including individual employee's password.

A sample copy of the application, "**Lab_07b_ED-Secure.zip**", can be downloaded from Canvas. At the moment, anyone can make changes to the employee information via EMS. In other words, it is not secure.

In this Lab, we will focus on making this application secure. That is, making changes so that only users with "ED-APP-ADMIN" credentials can perform the CRUD operations of EMS.

Overview of Lab Tasks

In this Lab, you will learn to perform the following tasks in programming a stateful session bean:

LT1. Download the sample app from Canvas
LT2. Set up the "SECURE EMS" database table on JavaDB and preload some data into it
LT3. Run the "ED-Secure" project

LT4. Run the “ED-Secure-AppClient” project
LT5. Secure the “ED-Secure-ejb” project
LT6. Secure the “ED-Secure-war” project

Assumption
Assume that you have done Lab_07a_Secure_GlassFish to set up the login credentials on GlassFish server

Lab Tasks

This Lab should be run on MS Windows Platform

LT1. Get the Secure-EMS enterprise app
- LT1.1 Download the “Lab_07b_ED-Secure.zip” project from Canvas
- LT1.2 Expand it to a folder of your choice (Remember no space char in your folder path name!)
- Note: You will have the following folders
  - a. ED-Secure [The Secure-EMS Enterprise App project folder; in this folder there are two more sub-projects, the “ejb” and the “war” projects]
  - b. ED-Secure-AppClient [The Secure-EMS Application Client project folder]
  - c. ED-Secure-jdbc [The JDBC program to set up the “EMS_EMPLOYEE” database table]
  - d. ED-Secure-RI [The Secure-EMS remote interface project folder]
- LT1.3 Open all these projects in NetBeans, including the “ejb” and “war” projects

LT2. Set up the “EMS_EMPLOYEE” table in JavaDB and pre-load some data into it
- LT2.1 Start the JavaDB server in NetBeans
- LT2.2 Run the “ED-Secure-jdbc” project in NetBeans
- Note: After this, you will have the “EMS_EMPLOYEE” table in “sun-appserv-samples” with some data in it for you to use

LT3. Run the “ED-Secure” enterprise application project
- LT3.1 Add a new employee to the database with the following information via the web application

EmpId	00009
Name	Issac
Phone	9876543210
Address	9 Newton Street, Hawthorn
Email	issac@secure.com.au
Password	99999999
AppGroup	ED-APP-USERS
Bank Account Id	999-765432-1
Salary	20000.0
Active	true

LT4. Run the “ED-Secure-AppClient” project

LT4.1 Run the “ED-Secure-AppClient” project

a. Enter ‘N’ when prompted to remove the employee just added.

Note: It will create another user in the EMS_EMPLOYEE database table with the following information. You can check it yourself.

EmpId	00099
Name	Edmonds
Phone	9214436789
Address	99 John Street, Hawthorn
Email	edmonds@secure.com.au
Password	password
AppGroup	ED-APP-USERS
Bank Account Id	123-456789-0
Salary	12345.0
Active	true

LT4.2 Run the “ED-Secure-AppClient” project again.

a. Enter ‘Y’ when prompted to remove the employee just added.

Note: This time, the add employee operation will fail as the record is there.

Note: We need to remove this record to make sure things are done properly in later LTs.

Note: After this, the employee record added in LT4.1 above has been removed.

Now, we “secure” the enterprise application “ED-Secure”. We need to do this in the two sub-projects. First, we do this on the EJB project. Second, we do this on the WAR project.

LT5. Secure the EJB project

LT5.1 Expand the “ED-Secure-ejb” project and the “session” package

LT5.2 Open the “EmployeeManagement.java” file

LT5.3 Add the line “`@DeclareRoles({"ED-APP-ADMIN"})`” just before the line with “@Stateless” annotation of the class. So, it looks like the following

```
@DeclareRoles({"ED-APP-ADMIN"})
@Stateless
public class EmployeeManagement implements EmployeeManagementRemote {
```

Remember to fix the imports and save the file.

Note: This is to say that only users in the “ED-APP-ADMIN” group registered on the GlassFish server can access this EJB. See Lab_07a_Secure_GlassFish for details.

LT5.4 Add the line “`@RolesAllowed({"ED-APP-ADMIN"})`” in the following methods. Add it in between the “`@Override`” and the method signature.

a. “public boolean hasEmployee()”. So, it looks like the following

```
@Override
@RolesAllowed({"ED-APP-ADMIN"})
public boolean hasEmployee(String empId) {
```

Remember to fix the imports. We leave the “save” action till last.

b. “public boolean addEmployee()”

c. “public boolean updateEmployeeDetails()”

d. “public boolean updateEmployeePassword()”

e. “public EmployeeDTO getEmployeeDetails()”

f. “public boolean deleteEmployee()”

g. “public boolean removeEmployee()”

Note: These methods are those provided in “EmployeeManagementRemote.java”.

Note: This is to say that only users in the “ED-APP-ADMIN” group registered on the GlassFish server can access the required methods.

Note: Remember to save the file.

LT5.5 Redeploy “ED-Secure” project [If you want, you may rebuild the project before deploy.]

LT5.6 Run the “ED-Secure-AppClient” project again

a. Now, you will get a “Login for user:” prompt from GlassFish (this is standard login prompt provided by the GlassFish server)

b. Enter “ed-admin1” for username and password (these are the credentials you used to set up the GlassFish server in Lab_07a for “ED-APP-ADMIN”)

c. Click OK

d. When prompted to remove the record just added, enter ‘Y’ or ‘N’

Note: If you enter ‘N’, remember to rerun it to remove it for the future task.

LT5.7 Try re-run the “ED-Secure-AppClient” with different credentials in Lab_07a_Secure_GlassFish and see what happens.

LT6. Secure the WAR project

LT6.1 Expand the “ED-Secure-war” project and the “Configuration Files” folder

LT6.2 Open the “web.xml”

Note: NetBeans opens the “web.xml” file on the editor pane

LT6.3 Select the “Security” option on the top of the “web.xml” tab

a. Set Login Configuration

1. Expand the “Login Configuration”
2. Select the “Form” option
3. Enter “/faces/login.xhtml” in the “Form Login Page” text field
4. Enter “/faces/retryLogin.xhtml” for the “Form Error Page” text field
5. Enter “fileRealm” in the “Realm Name:” text field

b. Add a Security Role

1. Click “Add” under “Security Roles”
2. In the “Add Security Role” pane, enter the following information
   a. “Role Name:” – “ED-APP-ADMIN”
   b. “Description:” – “EMS Administrators”
3. Click OK

c. Add a Security Constraint

1. Click “Add Security Constraint” on the right side of the page
2. Enter “EMS-AdminOnly” in the “Display Name:” text field
3. Click “Add” under “Web Resource Collection:”
4. In the “Add Web Resource” pane,
   a. Enter “AdminOnly” in the “Resource Name:” text field
   b. Resource Name: AdminOnly
   c. Enter “AdminOnly Access” in the “Description:” text field (Optional)
   d. Enter “/faces/admin/*” in the “URL Pattern(s):” text field
   e. Select “All HTTP Method(s)”
   f. Click OK
5. Check “Enable Authentication Constraint” checkbox
6. Click “Edit” next to the “Role Name(s):” text field
7. In the “Edit Role Names” pane
   a. Select “ED-APP-ADMIN” on the left pane
   b. Click “Add>” to move it to the right pane
   c. Click “OK”

Note: This is to say that only users in the “ED-APP-ADMIN” group registered on the GlassFish server can access the pages in “/faces/admin/*” (actually those in “admin” on the web pages). For unknown reasons, NetBeans is “putting” “/faces” in front of some “JSF pages” (not all).

LT6.4 Select the “Pages” option on the top of the “web.xml” tab

a. Set Welcome Files

1. Expand on “Welcome Files”

2. Check & make sure "faces/index.xhtml" in the "Welcome Files:" text field

Note: This set the first page to visit.
Since "Employee Management System" (i.e. mainmenu) is a protected resources, the web container component in the GlassFish server (most probably the Apache Tomcat) will then dispatch the "/faces/login.xhtml" page to prompt the user to login

b. Set Error Pages

1. Click "Add"
2. In the "Add Error Page" pane,
   a. Click "Browse…" next to "Error Page Location:" text field
   b. Select "authFailure.xhtml" in the "Browse Files" pane
   c. Click "Select File"
      Note: NetBeans shows "/authFailure.xhtml"
   d. Enter "403" in the "Error Code:" text field
   e. Click "OK"
3. Click the "Source" option on the top of the "web.xml" tab
   a. Search "/authFailure.xhtml" (or "403" for error-code)
   b. Replace it with "/faces/authFailure.xhtml"

Note: This tells the GlassFish server to dispatch the "/faces/authFailure.xhtml" page to the user instead of "throwing an HTTP 403 Forbidden status" to the user.

LT6.5 Save the "web.xml" file

LT6.6 Redeploy "ED-SECURE" project [If you want, you may rebuild the project before deploy.]

LT6.7 Run the "ED-SECURE" project again

a. Now, you will get a "Login Page"

b. Enter "ed-admin1" for username and password (these are the credentials you used to set up the GlassFish server in Lab_07a for "ED-APP-ADMIN")

c. Click OK

d. You will then get to the "mainmenu.xhtml" for the administrators

Note: You can try different credentials in Lab_07a to see what happens. For other credentials, you will see a page called "Authentication Failure" (this is the page for an HTTP 403 Forbidden status).

Finally, Take several deep breaths and relax.