1.How web-application works with security.

2.Procedure to place usernames,passwords,roles in tomcat server.

3.Configure the security information in web.xml

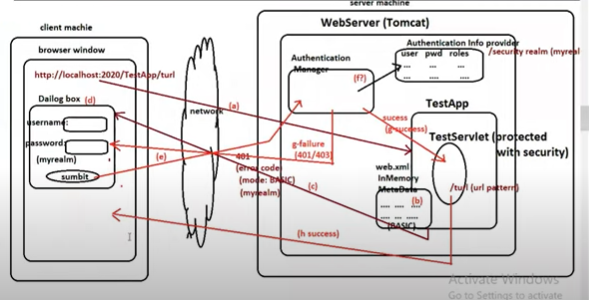
4.Basic or Digest Mode Authentication in servlet Security:

5.Form Mode Authentication in Servlet security.

6.Client-CERT Mode Authentication in servlet security.

1.How web-Application works with security.

================================



Step-a : From browser, a request is given to web-server for “TestServlet” .

Step-b: web server determines whether the TestServlet is protected or not based

On configuration details in web.xml file.

Step-c: The TestServlet is protected so web-server send the response to client.

The response have following info:

* Errorcode:401
* Authencation mode:BASIC
* Security realm name: Default name is myrealm.

Step-d: when client received response ,it display the dialog box for collecting the user credentials.

Step-e: After pressing submit button in dialog box, the given user credentials are submitted to “Authentication manager”.

Step-f: Authentication manager check user identity by using data of security realrm.

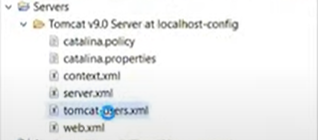
Step-g: If authentication success, Required servlet(Test Servlet) execution will happens and its output will be given to client as response.

Step-h: IF authentication or authorization fails, web-server will give 401 error code for indicating authentication failed and web-server will give 403 error code for indicating authorization failed to client.

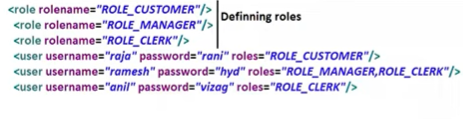
2.Procedure to place usernames,passwords,roles in tomcat server

=============================================

Step1: open tomcat-users.xml file which is in servers section of Eclipse IDE.

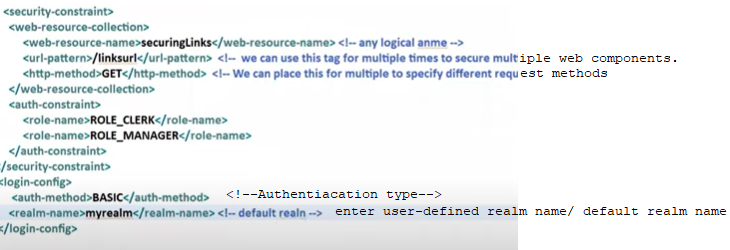


Step2: Define users,user roles as xml entries in that file.



3.Configure the security information in web.xml

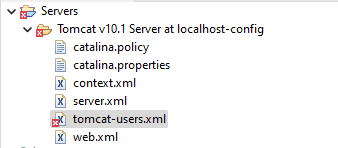
==================================

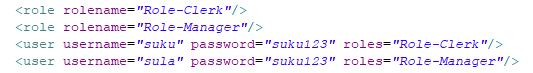


4.Basic or Digest Mode Authentication in servlet Security:

==================================

Step-1: Add users,password, roles to server.

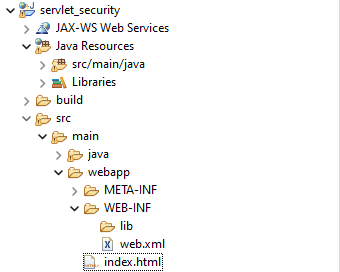




Step-2: Develop the web application.

Project Directory Structure:

===================



Index.html

=======

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h1 align="center "> Home page</h1>

<a href="./Family">Display Family Details</a>

</body>

</html>

Family.html

=========

**package** com.nit.suku;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**public** **class** Family **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse res) **throws** ServletException, IOException {

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

pw.write("<h2>"+"sukumar"+"</h2>");

pw.write("<h2>"+"samba"+"</h2>");

pw.write("<h2>"+"veena"+"</h2>");

pw.close();

}

}

**Step-3:** Configure security info in web.xml file.

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"https://jakarta.ee/xml/ns/jakartaee"* xsi:schemaLocation=*"https://jakarta.ee/xml/ns/jakartaee https://jakarta.ee/xml/ns/jakartaee/web-app\_6\_0.xsd"* version=*"6.0"*>

<security-constraint>

<web-resource-collection>

<web-resource-name>sukumar</web-resource-name>

<url-pattern>/Family</url-pattern>

<http-method>GET</http-method>

</web-resource-collection>

<auth-constraint>

<role-name>Role-Clerk</role-name>

<role-name>Role-Manager</role-name>

</auth-constraint>

</security-constraint>

<login-config>

<auth-method>BASIC</auth-method>

<realm-name>myrealm</realm-name>

</login-config>

<servlet>

<description></description>

<display-name>Family</display-name>

<servlet-name>Family</servlet-name>

<servlet-class>com.nit.suku.Family</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Family</servlet-name>

<url-pattern>/Family</url-pattern>

</servlet-mapping>

<display-name>servlet\_security</display-name>

<welcome-file-list>

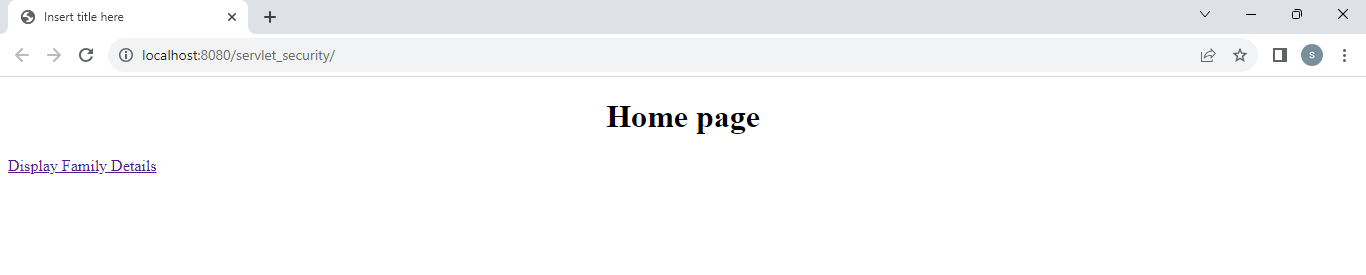
<welcome-file>index.html</welcome-file>

</welcome-file-list>

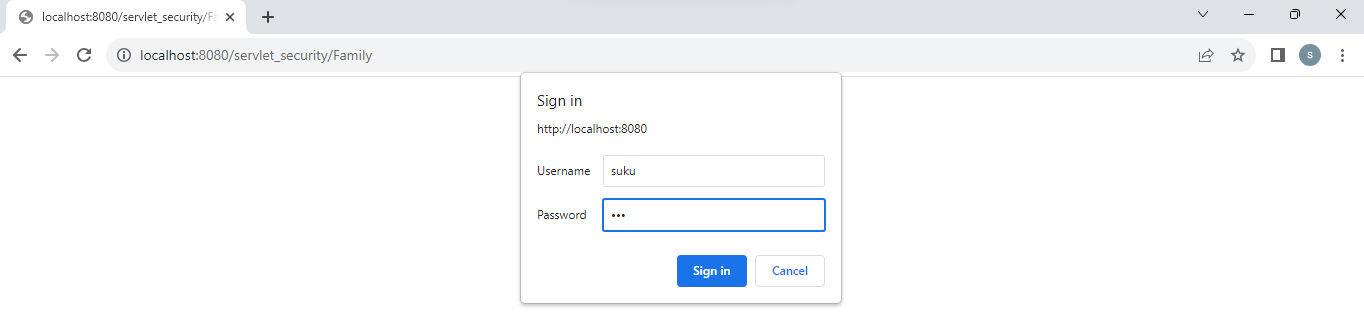
</web-app>

Output:-

=====

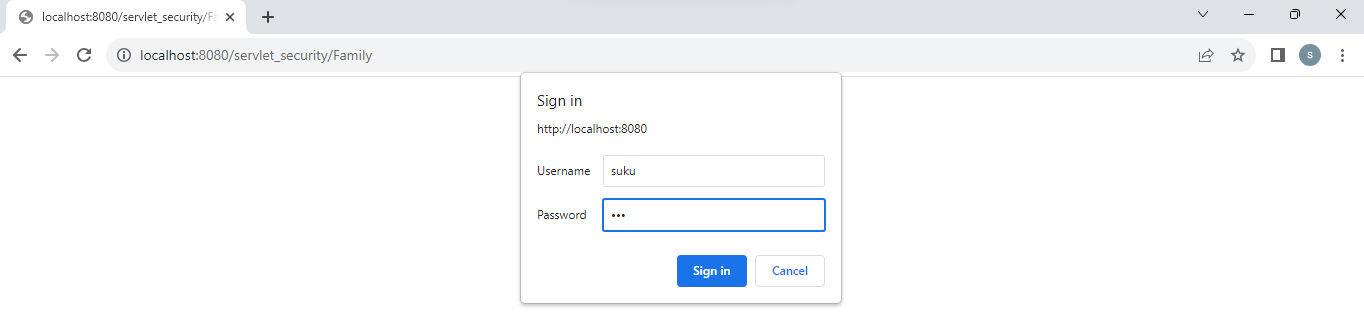


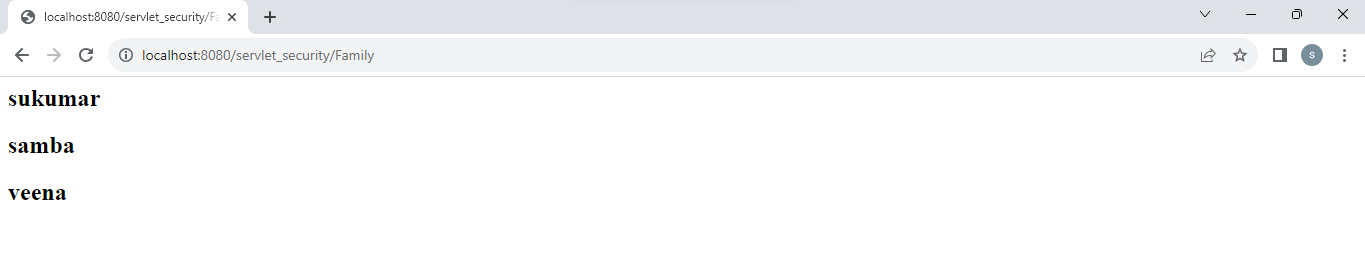
Press Display\_family\_Details Link.



I gave wrong password in diagbox. Till u will have entered right credentials, browser displays the dialogbox.

Now I entered correct user-credentials in above dialog-box.

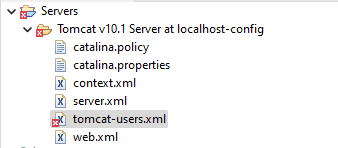


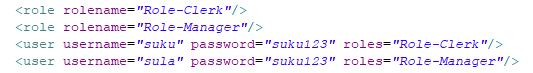


5.Form Authentication in servlet Security:

=============================

Step-1: Add users,password, roles to server.

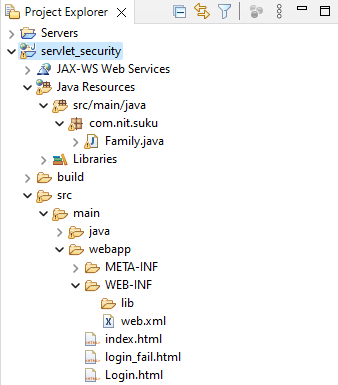




Step-2: Develop the web application.

Project Directory Structure:

===================



Index.html

=======

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h1 align="center "> Home page</h1>

<a href="./Family">Display Family Details</a>

</body>

</html>

Family.html

=========

**package** com.nit.suku;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**public** **class** Family **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse res) **throws** ServletException, IOException {

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

pw.write("<h2>"+"sukumar"+"</h2>");

pw.write("<h2>"+"samba"+"</h2>");

pw.write("<h2>"+"veena"+"</h2>");

pw.close();

}

}

Login.html

========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<form action="**j\_security\_check**" method="GET" align="center">

UserName: <input type="text" name="**j\_username**"><br><br>

PassWord:<input type="password" name="**j\_password**"/><br><br>

<input type="submit" value="Login"/>

</form>

</body>

</html>

Login\_fail.html

===========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h2 align="center" style="color: red">Authentication Error:401</h2>

<a href="Login.html">TryAgain</a>

</body>

</html>

Web.xml

=======

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"https://jakarta.ee/xml/ns/jakartaee"* xsi:schemaLocation=*"https://jakarta.ee/xml/ns/jakartaee https://jakarta.ee/xml/ns/jakartaee/web-app\_6\_0.xsd"* version=*"6.0"*>

<security-constraint>

<web-resource-collection>

<web-resource-name>sukumar</web-resource-name>

<url-pattern>/Family</url-pattern>

<http-method>GET</http-method>

</web-resource-collection>

<auth-constraint>

<role-name>Role-Clerk</role-name>

<role-name>Role-Manager</role-name>

</auth-constraint>

</security-constraint>

<login-config>

<auth-method>FORM</auth-method>

<realm-name>myrealm</realm-name>

***<form-login-config>***

***<form-login-page>/Login.html</form-login-page>***

***<form-error-page>/login\_fail.html</form-error-page>***

***</form-login-config>***

</login-config>

<servlet>

<description></description>

<display-name>Family</display-name>

<servlet-name>Family</servlet-name>

<servlet-class>com.nit.suku.Family</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Family</servlet-name>

<url-pattern>/Family</url-pattern>

</servlet-mapping>

<display-name>servlet\_security</display-name>

<welcome-file-list>

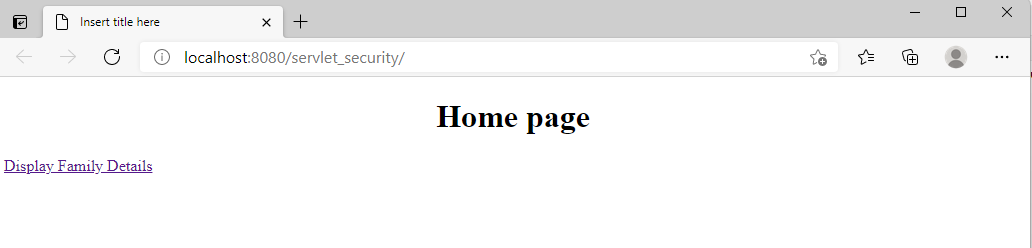
<welcome-file>index.html</welcome-file>

</welcome-file-list>

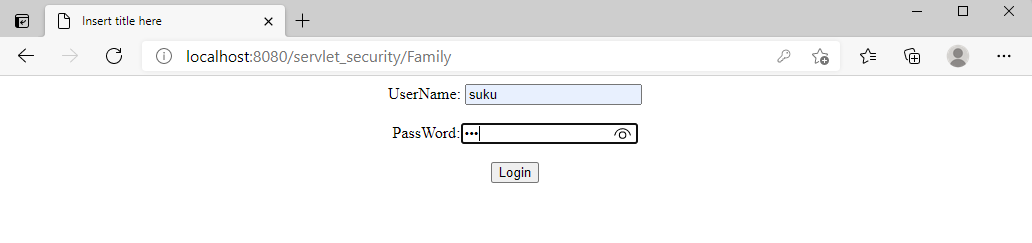
</web-app>

Output:1

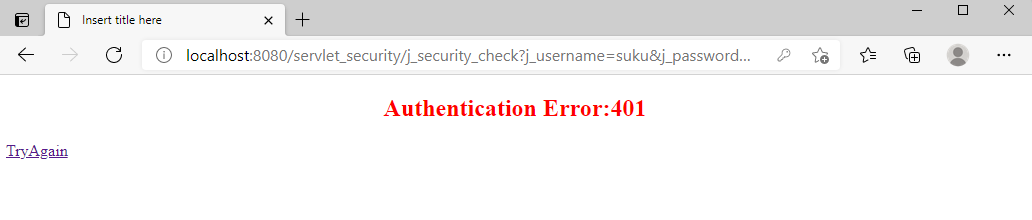
=======



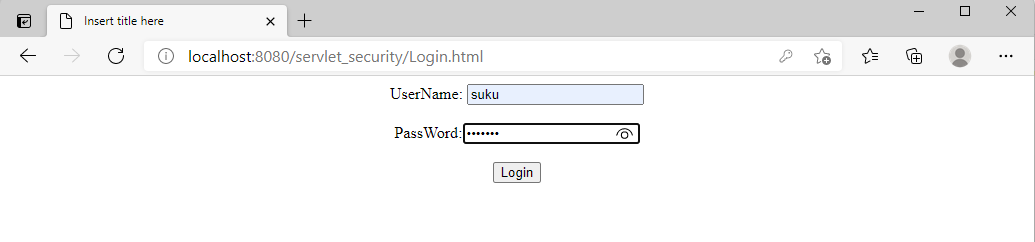
After clicking the hyperlink on page.



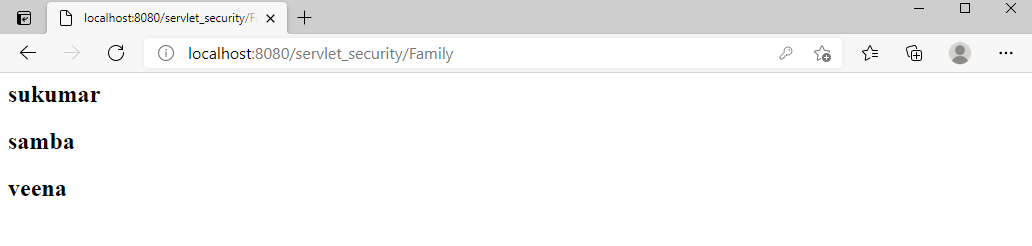
I entered the wrong user credentials.



I click on the “Try Again hyperlink”.



I entered the correct user credentials.



6. Client-CERT Mode Authentication in servlet security:

========================================

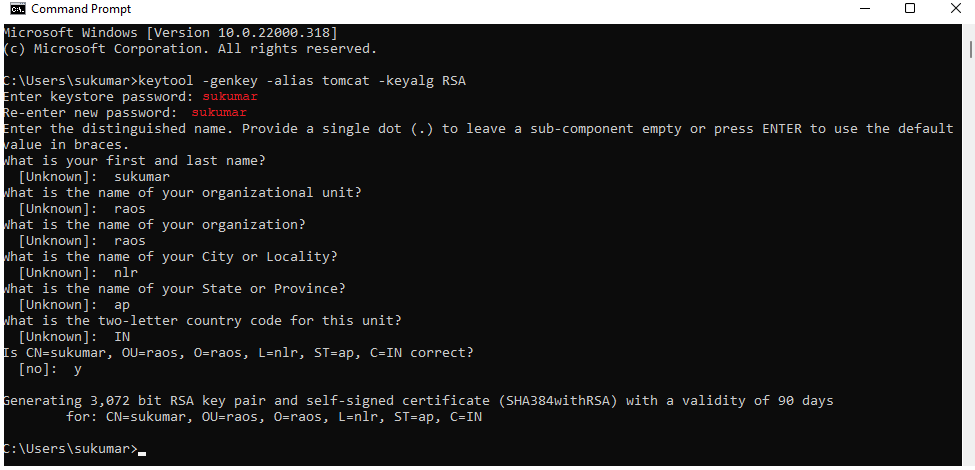
We prepare digital certificate using RSA or some other algorithm and we cfg that certificate in server by eanbling “https”. So if the web applications are requested with protocol “https” then the digital certificate will be downloaded browser side(client machine). Now onwards the browser and web application will encode/decode data according to digital certificate.

Since digital certificate contains personalized information.. the encryption and decryption that happens is very much stronger and hacker can not decode it.

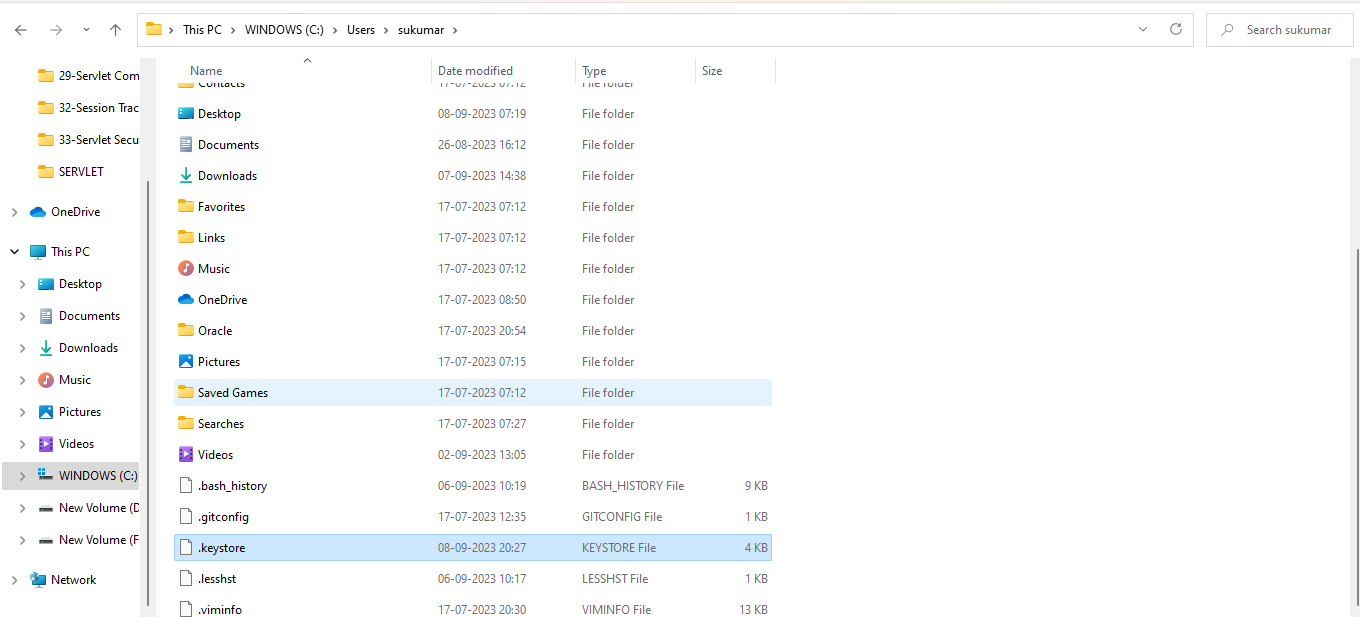
Usecase::

1. While sending login details over the network
2. While sending credit/debit card details over the network.

6.1. Create Digital certificate using tool “keytool” and RSA algorithm.



Creates digital certificates as” .keystore” file in C:\users\<windowsusername>folder.

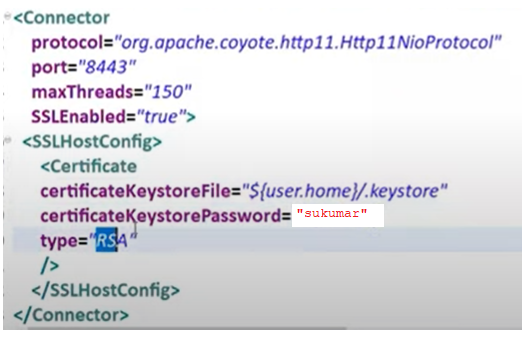


6.2. configure digital certificate with tomcat server in server.xml file.



Server.xml

========



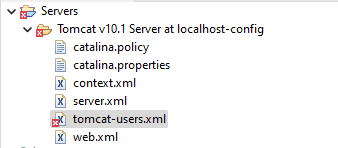
6.3. Steps to provide the security to servlet using Client-CERT.

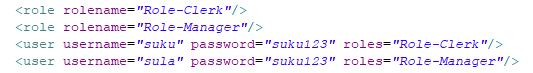
Step1: Create Digital certificate using tool “keytool” and RSA algorithm.

Step2: configure digital certificate with tomcat server in server.xml file.

Step3: Develop the web application .

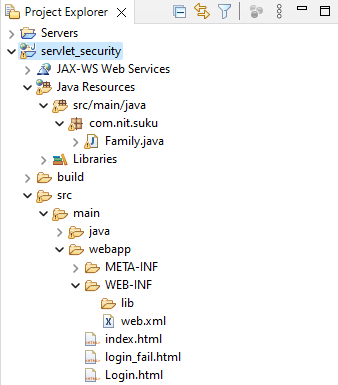
Add users,password, roles to server.





Project Directory Structure:

===================



Index.html

=======

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h1 align="center "> Home page</h1>

<a href="./Family">Display Family Details</a>

</body>

</html>

Family.html

=========

**package** com.nit.suku;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** jakarta.servlet.ServletException;

**import** jakarta.servlet.http.HttpServlet;

**import** jakarta.servlet.http.HttpServletRequest;

**import** jakarta.servlet.http.HttpServletResponse;

**public** **class** Family **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse res) **throws** ServletException, IOException {

PrintWriter pw=res.getWriter();

res.setContentType("text/html");

pw.write("<h2>"+"sukumar"+"</h2>");

pw.write("<h2>"+"samba"+"</h2>");

pw.write("<h2>"+"veena"+"</h2>");

pw.close();

}

}

Login.html

========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<form action="**j\_security\_check**" method="GET" align="center">

UserName: <input type="text" name="**j\_username**"><br><br>

PassWord:<input type="password" name="**j\_password**"/><br><br>

<input type="submit" value="Login"/>

</form>

</body>

</html>

Login\_fail.html

===========

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<h2 align="center" style="color: red">Authentication Error:401</h2>

<a href="Login.html">TryAgain</a>

</body>

</html>

Web.xml

=======

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"https://jakarta.ee/xml/ns/jakartaee"* xsi:schemaLocation=*"https://jakarta.ee/xml/ns/jakartaee https://jakarta.ee/xml/ns/jakartaee/web-app\_6\_0.xsd"* version=*"6.0"*>

<security-constraint>

<web-resource-collection>

<web-resource-name>sukumar</web-resource-name>

<url-pattern>/Family</url-pattern>

<http-method>GET</http-method>

</web-resource-collection>

<auth-constraint>

<role-name>Role-Clerk</role-name>

<role-name>Role-Manager</role-name>

</auth-constraint>

</security-constraint>

<login-config>

<auth-method>FORM</auth-method>

<realm-name>myrealm</realm-name>

***<form-login-config>***

***<form-login-page>/Login.html</form-login-page>***

***<form-error-page>/login\_fail.html</form-error-page>***

***</form-login-config>***

</login-config>

<servlet>

<description></description>

<display-name>Family</display-name>

<servlet-name>Family</servlet-name>

<servlet-class>com.nit.suku.Family</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Family</servlet-name>

<url-pattern>/Family</url-pattern>

</servlet-mapping>

<display-name>servlet\_security</display-name>

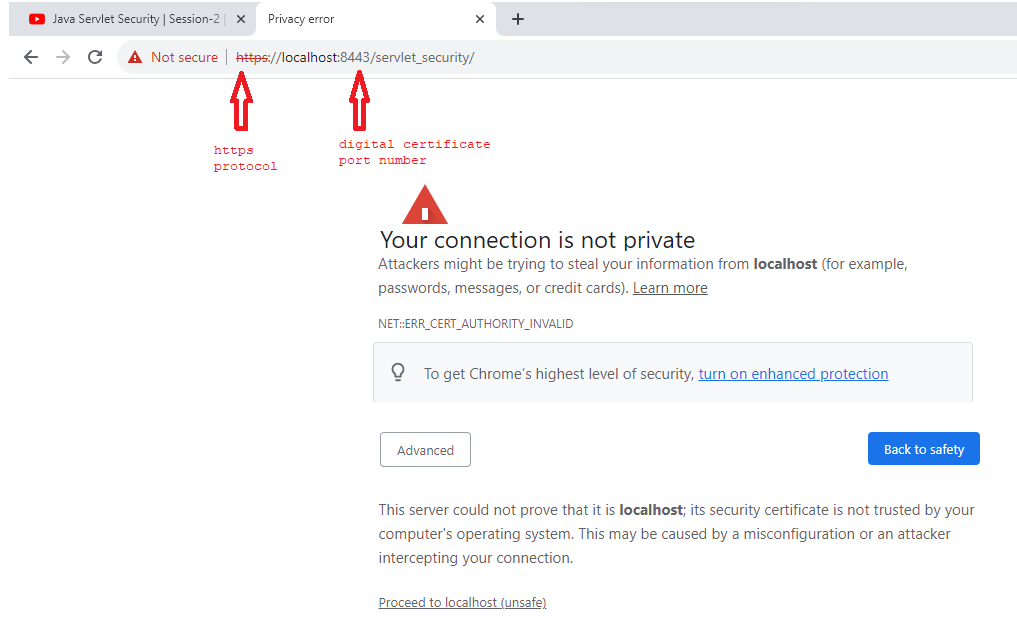
<welcome-file-list>

<welcome-file>index.html</welcome-file>

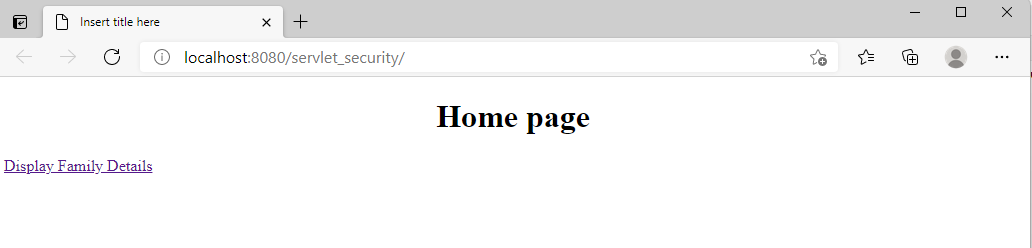
</welcome-file-list>

</web-app>

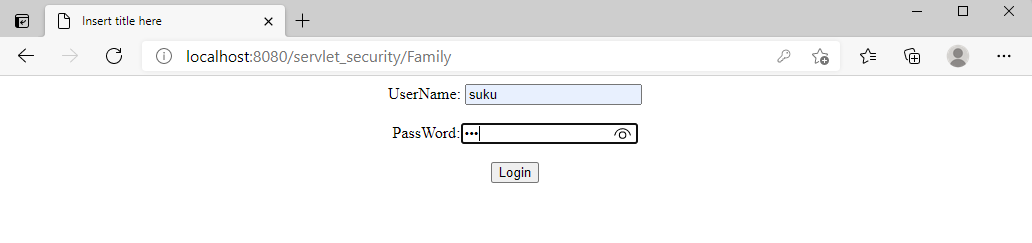
Step4: Test the web application using “https” protocol.



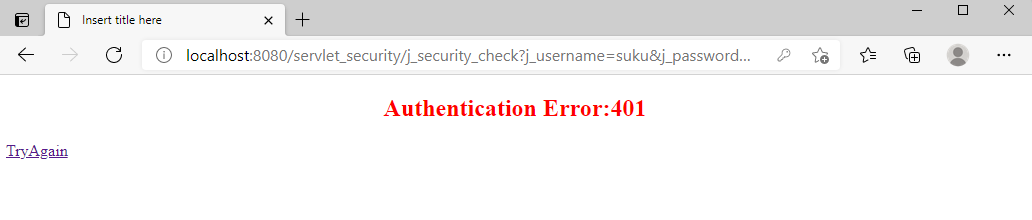
Click on proceed to localhost(unsafe).



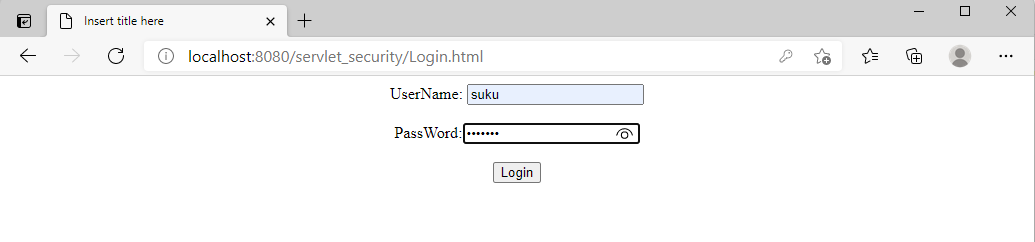
After clicking the hyperlink on page.



I entered the wrong user credentials.



I click on the “Try Again hyperlink”.



I entered the correct user credentials.

