WAY2LEARN

Spring Security -User Signup and Signin

In this exercise, you will understand how to create a new Spring Security User and logging in the user automatically.

1) **Open /WEB-INF/jsp/application/signup.jsp and observe that we are asking E-mail, fullname and password for signing up.**

2) **Open SecurityController.java and observe that there is a method mapped for GET requests to /signup.**
3) **Deploy the application and click on Signup link or give request for /signup. you should Observe the Sign up Page. Once u submit the page, it is making a post request to same url**
4) **There is already one method in SecurityController.java to map POST requests for /signup. In this method, we want to write the code for saving user details in to default users table in database and roles into authorities.**
5) **Fortunately, we dont need to Write the code. There is a UserDetailsManager called JdbcUserDetailsManager which contains methods for performing CRUD operation on UserDetails Object which represents user**
   JdbcUserDetailsManager is already configured as a bean using <security:jdbc-user-service >

6) **AutoWire UserDetailsManager in to SecurityController.**
7) **Create UserDetails Object from the data coming in request as shown below** :

```
      List<GrantedAuthority> authorities=
AuthorityUtils.createAuthorityList("ROLE_USER");
            UserDetails springSecurityUser= new User(userDetails.getEmail(),
      userDetails.getPassword(),true, true, true, true, authorities);
```

8) **Create User Using following code**

```
            userDetailsManager.createUser(springSecurityUser);
```

9) **We want the User to be logged automatically once signup is successful. In Spring Security, if the User has to be logged in, SecurityContext has to be populated with authentication token as shown below** :

```
Authentication authentication=
                        new UsernamePasswordAuthenticationToken(userDetails.getEmail(),
   userDetails.getPassword(), authorities);

   SecurityContext securityContext=SecurityContextHolder.getContext();
   securityContext.setAuthentication(authentication);
```

10) **Now deploy the application and try to sign up. After successful sign up, you should see that user is sign in automtically. If login is successful, on the right corner of the page, you should see Welcome User message. Open topMenu.jsp page and see the code which checks if the user is already logged in . If user is logged in, it is showing the welcome message and also Logout link**

Following is the code   in topMenu.jsp

```
<security:authorize access="authenticated" var="isAuthenticated" />
      <c:if test="${isAuthenticated}">
            <c:url var="logouturl" value="/logout" />
            Welcome  <security:authentication property="name"/>
      <form action="${logouturl}" method="post">
        <input type="submit" value="Log out"  class="btn btn-green btn-login" />
        <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
      </form>

      </c:if>
```

Congratulations !! U have understood how to Create a new Spring Security User and Logging in user manually