SecurityConfig.java

package com.mkyong.config;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.context.annotation.Configuration;

import org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;

import org.springframework.security.config.annotation.web.builders.HttpSecurity;

import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;

import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

@Configuration

@EnableWebSecurity

public class SecurityConfig extends WebSecurityConfigurerAdapter {

@Autowired

public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception {

auth.inMemoryAuthentication()

.withUser("mkyong").password("123456").roles("USER");

}

//.csrf() is optional, enabled by default, if using WebSecurityConfigurerAdapter constructor

@Override

protected void configure(HttpSecurity http) throws Exception {

http.authorizeRequests()

.antMatchers("/admin/\*\*").access("hasRole('ROLE\_USER')")

.and()

.formLogin().loginPage("/login").failureUrl("/login?error")

.usernameParameter("username").passwordParameter("password")

.and()

.logout().logoutSuccessUrl("/login?logout")

.and()

.csrf();

}

}

**Spring Security XML file :**

<http auto-config="true">

<intercept-url pattern="/admin\*\*" access="ROLE\_USER" />

<form-login

login-page="/login"

default-target-url="/welcome"

authentication-failure-url="/login?error"

username-parameter="username"

password-parameter="password" />

<logout logout-success-url="/login?logout" />

<!-- enable csrf protection -->

<csrf/>

</http>

<authentication-manager>

<authentication-provider>

<user-service>

<user name="mkyong" password="123456" authorities="ROLE\_USER" />

</user-service>

</authentication-provider>

</authentication-manager>

login.jsp

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

<html>

<head>

<title>Login Page</title>

<style>

.error {

padding: 15px;

margin-bottom: 20px;

border: 1px solid transparent;

border-radius: 4px;

color: #a94442;

background-color: #f2dede;

border-color: #ebccd1;

}

.msg {

padding: 15px;

margin-bottom: 20px;

border: 1px solid transparent;

border-radius: 4px;

color: #31708f;

background-color: #d9edf7;

border-color: #bce8f1;

}

#login-box {

width: 300px;

padding: 20px;

margin: 100px auto;

background: #fff;

-webkit-border-radius: 2px;

-moz-border-radius: 2px;

border: 1px solid #000;

}

</style>

</head>

<body onload='document.loginForm.username.focus();'>

<h1>Spring Security Custom Login Form (Annotation)</h1>

<div id="login-box">

<h2>Login with Username and Password</h2>

<c:if test="${not empty error}">

<div class="error">${error}</div>

</c:if>

<c:if test="${not empty msg}">

<div class="msg">${msg}</div>

</c:if>

<form name='loginForm'

action="<c:url value='j\_spring\_security\_check' />" method='POST'>

<table>

<tr>

<td>User:</td>

<td><input type='text' name='user' value=''></td>

</tr>

<tr>

<td>Password:</td>

<td><input type='password' name='pass' /></td>

</tr>

<tr>

<td colspan='2'>

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

</td>

</tr>

</table>

<input type="hidden"

name="${\_csrf.parameterName}" value="${\_csrf.token}" />

</form>

</div>

</body>

</html>

**hello.jsp**

<%@page session="false"%>

<html>

<body>

<h1>Title : ${title}</h1>

<h1>Message : ${message}</h1>

</body>

</html>

admin.jsp + logout

<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

<%@page session="true"%>

<html>

<body>

<h1>Title : ${title}</h1>

<h1>Message : ${message}</h1>

<c:url value="/j\_spring\_security\_logout" var="logoutUrl" />

<!-- csrt support -->

<form action="${logoutUrl}" method="post" id="logoutForm">

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}" />

</form>

<script>

function formSubmit() {

document.getElementById("logoutForm").submit();

}

</script>

<c:if test="${pageContext.request.userPrincipal.name != null}">

<h2>

Welcome : ${pageContext.request.userPrincipal.name} | <a

href="javascript:formSubmit()"> Logout</a>

</h2>

</c:if>

</body>

</html>

HelloController.java

package com.mkyong.web.controller;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.servlet.ModelAndView;

@Controller

public class HelloController {

@RequestMapping(value = { "/", "/welcome\*\*" }, method = RequestMethod.GET)

public ModelAndView welcomePage() {

ModelAndView model = new ModelAndView();

model.addObject("title", "Spring Security Custom Login Form");

model.addObject("message", "This is welcome page!");

model.setViewName("hello");

return model;

}

@RequestMapping(value = "/admin\*\*", method = RequestMethod.GET)

public ModelAndView adminPage() {

ModelAndView model = new ModelAndView();

model.addObject("title", "Spring Security Custom Login Form");

model.addObject("message", "This is protected page!");

model.setViewName("admin");

return model;

}

//Spring Security see this :

@RequestMapping(value = "/login", method = RequestMethod.GET)

public ModelAndView login(

@RequestParam(value = "error", required = false) String error,

@RequestParam(value = "logout", required = false) String logout) {

ModelAndView model = new ModelAndView();

if (error != null) {

model.addObject("error", "Invalid username and password!");

}

if (logout != null) {

model.addObject("msg", "You've been logged out successfully.");

}

model.setViewName("login");

return model;

}

}

SpringSecurityInitializer.java

package com.mkyong.config.core;

import org.springframework.security.web.context.AbstractSecurityWebApplicationInitializer;

public class SpringSecurityInitializer extends AbstractSecurityWebApplicationInitializer {

}

SpringMvcInitializer.java

package com.mkyong.config.core;

import org.springframework.web.servlet.support.AbstractAnnotationConfigDispatcherServletInitializer;

import com.mkyong.config.AppConfig;

public class SpringMvcInitializer

extends AbstractAnnotationConfigDispatcherServletInitializer {

@Override

protected Class<?>[] getRootConfigClasses() {

return new Class[] { AppConfig.class };

}

@Override

protected Class<?>[] getServletConfigClasses() {

return null;

}

@Override

protected String[] getServletMappings() {

return new String[] { "/" };

}

}

AppConfig.java

package com.mkyong.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.ComponentScan;

import org.springframework.context.annotation.Configuration;

import org.springframework.context.annotation.Import;

import org.springframework.web.servlet.config.annotation.EnableWebMvc;

import org.springframework.web.servlet.view.InternalResourceViewResolver;

import org.springframework.web.servlet.view.JstlView;

@EnableWebMvc

@Configuration

@ComponentScan({ "com.mkyong.web.\*" })

@Import({ SecurityConfig.class })

public class AppConfig {

@Bean

public InternalResourceViewResolver viewResolver() {

InternalResourceViewResolver viewResolver

= new InternalResourceViewResolver();

viewResolver.setViewClass(JstlView.class);

viewResolver.setPrefix("/WEB-INF/pages/");

viewResolver.setSuffix(".jsp");

return viewResolver;

}

}