# Implement Spring Security with Authentication- Source Code

**SellerController.java:**

**package** com.example.demo;

**import** org.springframework.stereotype.Controller; **import** org.springframework.web.bind.annotation.GetMapping;

@Controller

**public** **class** SellerController {

@GetMapping("/seller/display") **public** String sellerDisplay(){

**return** "seller/seller";

}

@GetMapping("/home/display") **public** String homeDisplay(){

**return** "home/home";

}

}

**SpringBootSecurityDemoApplication:**

**package** com.example.demo;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** SpringBootSecurityDemoApplication

{

**public** **static** **void** main(String[] args) {

SpringApplication.run(SpringBootSecurityDemoApplication.**class**,

arg

s

)

;

}

}

**WebSecurityConf:**

**package** com.example.demo;

**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.HttpSecuri

t

y

;

## import

org.springframework.security.config.annotation.web.configuration.Enab leWebSecurity; **import**

org.springframework.security.crypto.factory.PasswordEncoderFactories; **import** org.springframework.security.crypto.password.PasswordEncoder;

@EnableWebSecurity @Configuration **public** **class** WebSecurityConf {

@Autowired

**public** **void** configureGlobal(AuthenticationManagerBuilder auth) **throws**

Exception {

PasswordEncoder encoder =

PasswordEncoderFactories.createDelegatingPasswordEncoder();

auth.inMemoryAuthentication()

.withUser("user").password(encoder.encode("password")).roles(" USER")

.and()

.withUser("user1").password(encoder.encode("password1")).roles ("USER")

.and()

.withUser("user2").password(encoder.encode("password2")).roles ("USER")

.and()

.withUser("admin1").password(encoder.encode("adminpassword

1")).roles("ADMIN");

} }

**Application Properties :**

#JSP files configuation

spring.mvc.view.prefix=/WEB-INF/views/ spring.mvc.view.suffix=.jsp

**Index.jsp :**

Welcome to index page.

**Home.jsp :**

Home Page.

**Seller.jsp :**

Seller Page.

**pom.xml:**

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

<project

xmlns=["http://maven.apache.org/POM/4.0.0"](http://maven.apache.org/POM/4.0.0) xmlns:xsi=["http://www.w3.org/2001/XMLSchemainstance"](http://www.w3.org/2001/XMLSchema-instance)

xsi:schemaLocation=["http://maven.apache.org/POM/4](http://maven.apache.org/POM/4.0.0)

[.0.0 h](http://maven.apache.org/POM/4.0.0)ttps://maven.apache.org/xsd/maven-

4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>3.1.0</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.example</groupId>

<artifactId>spring-boot-security-demo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-boot-security-demo</name>

<description>Demo project for Spring Boot Security</description>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<!--

https://mvnrepository.com/artifact/jakarta.servlet.jsp.jstl/jakarta.se rvlet. jsp.jstl-api -->

<dependency>

<groupId>jakarta.servlet.jsp.jstl</groupId>

<artifactId>jakarta.servlet.jsp.jstl-api</artifactId>

<version>3.0.0</version>

</dependency>

<!-- JSP support in Spring -->

<dependency>

<groupId>org.apache.tomcat.embed</groupId>

<artifactId>tomcat-embed-jasper</artifactId>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>org.apache.tomcat.embed</groupId>

<artifactId>tomcat-embed-jasper</artifactId>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build> <plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>