

Tema 5: Spring Security

Índice







1. Creación de login con múltiples usuarios sin usar la dependencia Spring Security.
2. Creación de login básico de 1 solo usuario usando Spring Security.
3. Creación de login de 1 solo usuario usando Spring Security + clase Config (@Configuration)
4. Tarea de investigación: usa Spring Security para los siguientes casos:
 - Usuarios en memoria, rol único.
 - Usuarios en memoria, varios roles.
 - Usuarios en base de datos, varios roles.

1. Creación de login con múltiples usuarios sin usar la dependencia Spring Security.

En esta primera versión vamos a acceder directamente a una base de datos llamada usuarios. Dentro existe una tabla llamado usuario con 2 campos:

Tabla usuario	Field	Type	Null	Key	Default	Extra
	user	varchar(10)	NO	PRI	NULL	
	pass	varchar(20)	NO		NULL	

Contiene los siguientes usuarios

<div>← T →</div>					user	pass		
<input type="checkbox"/>		Editar		Copiar		Borrar	user1	pass1
<input type="checkbox"/>		Editar		Copiar		Borrar	user2	pass2

Intento de login incorrecto

← → ↻ ⓘ localhost:8080/login

*User:

*Pass:

Entrar

*Campo obligatorio

Acceso Denegado

No tiene permisos para acceder a esta página.

Volver al Login



2. Creación de login básico de 1 solo usuario usando Spring Security.

Spring Security es un módulo de Spring usado para incorporar seguridad de acceso a las aplicaciones hechas con Spring Boot. Permite controles de acceso por URL entre otras muchas opciones y es más que suficiente para proteger tu programa.

El control de acceso permite limitar las opciones que pueden ejecutar un determinado conjunto de usuarios o roles sobre la aplicación. Cuando el usuario intente acceder a cualquier URL de la aplicación, Spring Boot y el Security Filter de HTTP redirigirá al usuario al formulario de identificación, donde solicitará al usuario a insertar el nombre y password para proceder.

Usar Spring Security en un proyecto es tan fácil como incluir la dependencia al crear el proyecto o añadirla directamente a nuestro archivo pom.xml:

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
</dependency>
```

De esta forma ya tendremos una página de login completo. Pon la url que quieras de tu proyecto y redirigirá al login. Las credenciales por defecto son:

Username: user

Password: la que aparezca en tu consola. Ejemplo: Using generated security password: 217b800c-ed70-4f13-85aa-0c3afa45164k. This generated password is for development use only. Your security configuration must be updated before running your application in production. (Fijate en ese mensaje, parece que esa contraseña puede mejorarse).

Puedes crear usuarios y contraseñas propias desde el archivo application.properties usando las siguientes líneas de código:

```
spring.security.user.name = tuNombreDeUsuario
```

```
spring.security.user.password = tuContraseña
```

3. Creación de login de 1 solo usuario usando Spring Security + clase Config (@Configuration)

En este caso, creamos una clase Config, la cual interactúa con Spring Security. El código es:

```
@Configuration
```

```
public class SecurityConfig {
```

```
    @Bean
```

```
    public UserDetailsService userDetailsService() {
```

```
        var userDetailsService = new InMemoryUserDetailsManager();
```

```
        UserDetails user1 = User.builder().username("admin")
```

```
            .password(this.passwordEncoder().encode("admin"))
```

```
            .authorities("write")
```

```
            .build();
```

```
        userDetailsService.createUser(user1);
```

```
        return userDetailsService;
```

```
    }
```

```
    @Bean
```

```
    public PasswordEncoder passwordEncoder() {
```

```
        return new BCryptPasswordEncoder();
```

```
    }
```

```
}
```

4. Tarea de investigación: usa Spring Security para los siguientes casos:

- Usuarios en memoria, rol único.**
- Usuarios en memoria, varios roles.**
- Usuarios en base de datos, varios roles.**

Es tu momento para mejorar tus proyectos de manera avanzada gracias a Spring Security. Para ello, ayúdate del documento que está subido a la Moodle llamado “Proyectos seguros con SpringBoot + Security”. Está dividido en 3 niveles de dificultad, puedes realizar entre 1 y 3 entregas en la plataforma:

- Usuarios en memoria, rol único. (Puntuación: 5)
- Usuarios en memoria, rol único + Usuarios en memoria, varios roles. (Puntuación: 7,5)
- Usuarios en memoria, rol único + Usuarios en memoria, varios roles. + Usuarios en base de datos, varios roles. (Puntuación: 10)