

SOURCE CODE –Spring Authentication

DemoSecurityController.java

```
package com.example.demo;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller
public class DemoSecurityController
{

    @ResponseBody
    @RequestMapping("/")
    public String demoMsg() {
        return "general audeinece";
    }

    @ResponseBody
    @RequestMapping("/access1")
    public String request1() {
        return "admin";
    }

    @ResponseBody
    @RequestMapping("/access2")
    public String request2() {
        return "user";
    }

    @ResponseBody
    @RequestMapping("/access3")
```

```
        public String request3() {  
            return "user and admin";  
        }  
  
    }  
  
}
```

MphasisSecurityyApplication.java

```
package com.example.demo;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class MphasisSecurityyApplication {  
  
    public static void main(String[] args) {  
        SpringApplication.run(MphasisSecurityyApplication.class, args);  
    }  
  
}
```

SecurityConfig.java

```
package com.example.demo;  
  
import org.springframework.context.annotation.Bean;  
import  
org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;  
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
```

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

import org.springframework.security.crypto.password.NoOpPasswordEncoder;

import org.springframework.security.crypto.password.PasswordEncoder;

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

```
@EnableWebSecurity
```

```
public class SecurityConfig extends WebSecurityConfigurerAdapter{
```

```
    @Override
```

```
    protected void configure(AuthenticationManagerBuilder auth) throws Exception {
```

```
        auth.inMemoryAuthentication()
```

```
            .withUser("admin")
```

```
            .password("admin")
```

```
            .roles("ADMIN")
```

```
            .and()
```

```
            .withUser("user")
```

```
            .password("user")
```

```
            .roles("USER");
```

```
    }
```

```
    @Override
```

```
    protected void configure(HttpSecurity http) throws Exception {
```

```
        http.authorizeHttpRequests()
```

```
            .antMatchers("/access1").hasRole("ADMIN")
```

```
            .antMatchers("/access2").hasRole("USER")
```

```
            .antMatchers("/access3").hasAnyRole("ADMIN","USER")
```

```
            .antMatchers("/").permitAll().and().formLogin();
```

```
//form login is a view use .disable for postman
```

```
    }
```

```
    @Bean

    public PasswordEncoder getPassword() {

        return NoOpPasswordEncoder.getInstance();

    }

}
```

ServletInitializer.java

```
package com.example.demo;

import org.springframework.boot.builder.SpringApplicationBuilder;
import org.springframework.boot.web.servlet.support.SpringBootServletInitializer;

public class ServletInitializer extends SpringBootServletInitializer {

    @Override

    protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {

        return application.sources(MphasisSecurityApplication.class);

    }

}
```