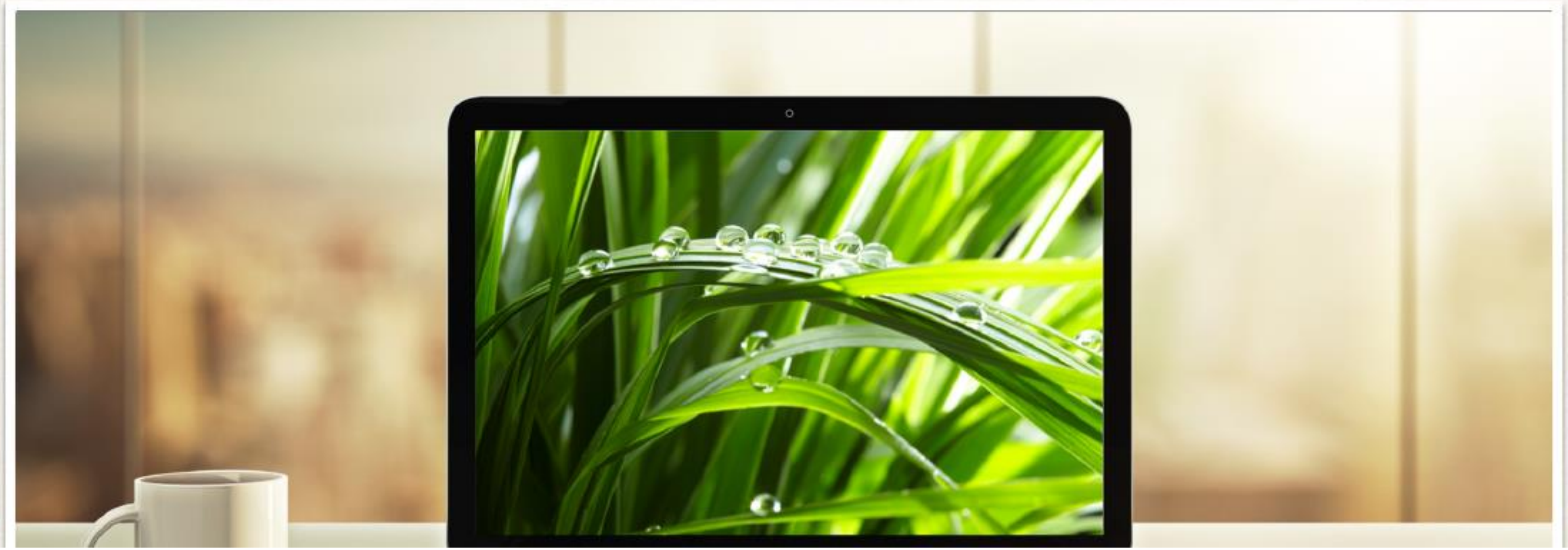
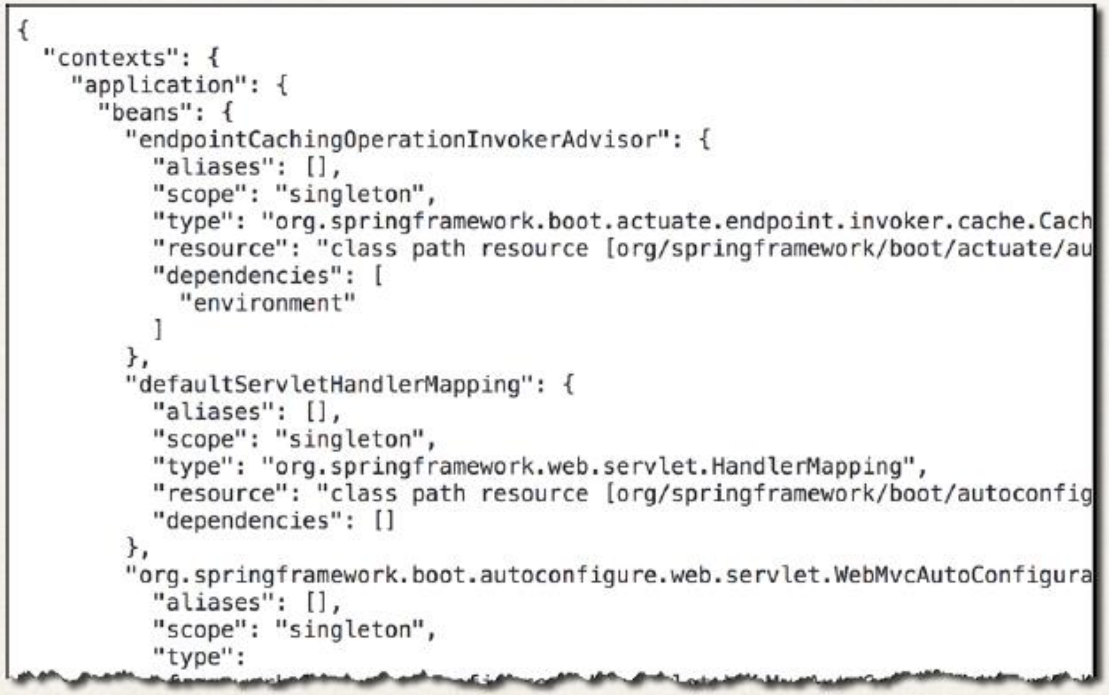
Spring Boot Actuator - Security



What about Security?

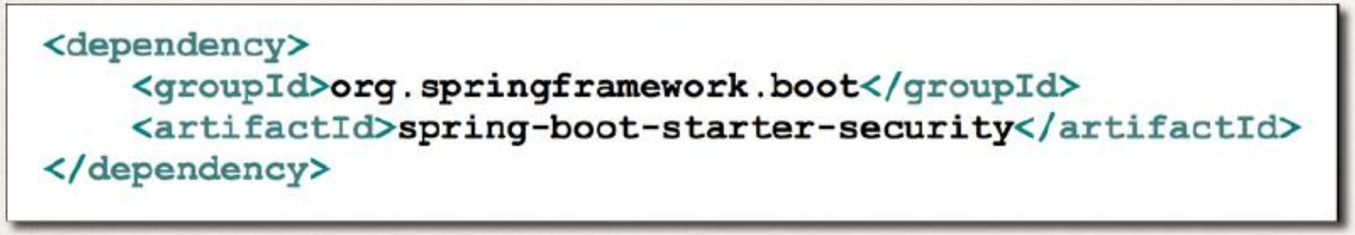
- You may **NOT** want to expose all of this information



```
{
  "contexts": {
    "application": {
      "beans": {
        "endpointCachingOperationInvokerAdvisor": {
          "aliases": [],
          "scope": "singleton",
          "type": "org.springframework.boot.actuate.endpoint.invoker.cache.Cach
          "resource": "class path resource [org/springframework/boot/actuate/au
          "dependencies": [
            "environment"
          ]
        },
        "defaultServletHandlerMapping": {
          "aliases": [],
          "scope": "singleton",
          "type": "org.springframework.web.servlet.HandlerMapping",
          "resource": "class path resource [org/springframework/boot/autoconfig
          "dependencies": []
        },
        "org.springframework.boot.autoconfigure.web.servlet.WebMvcAutoConfigura
          "aliases": [],
          "scope": "singleton",
          "type":
```

What about Security?

- You may **NOT** want to expose all of this information
- Add Spring Security to project and endpoints are secured :-)

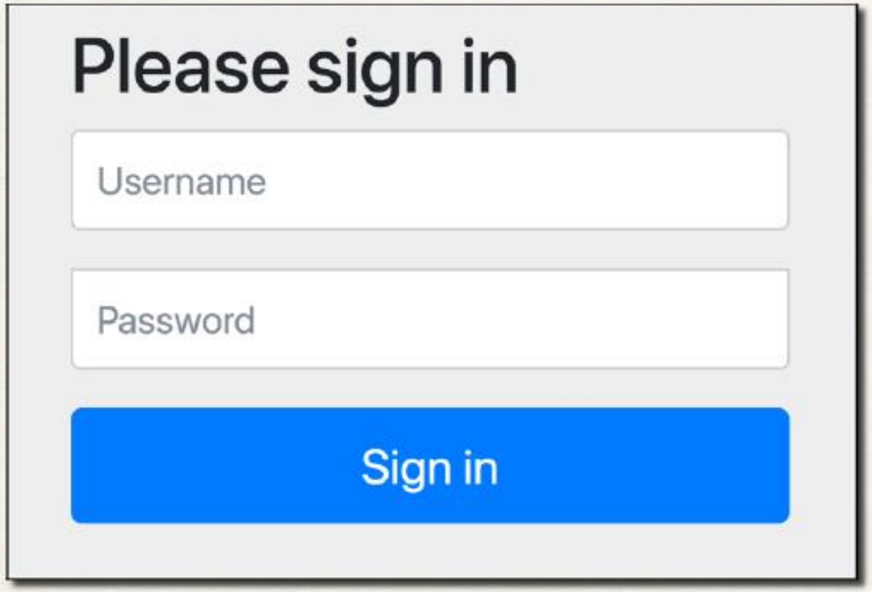


```xml
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
</dependency>
```

/health and /info
are still available
You can disable them
if you want

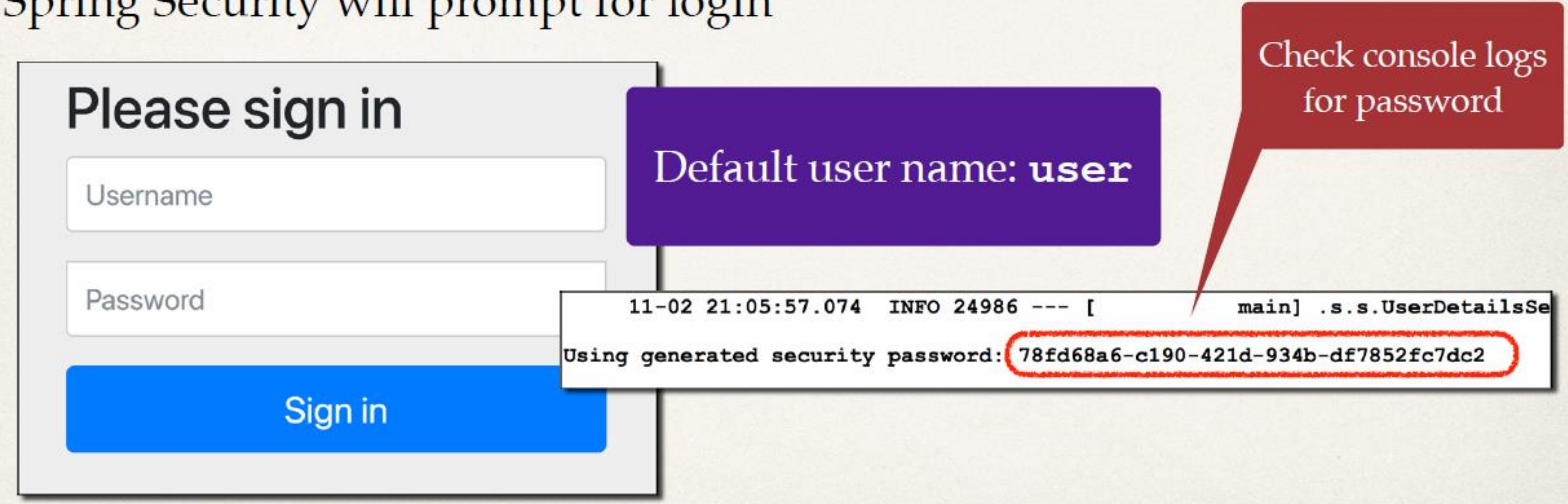
Secured Endpoints

- Now when you access: **`/actuator/beans`**
- Spring Security will prompt for login



Secured Endpoints

- Now when you access: **`/actuator/beans`**
- Spring Security will prompt for login



Default user name: **`user`**

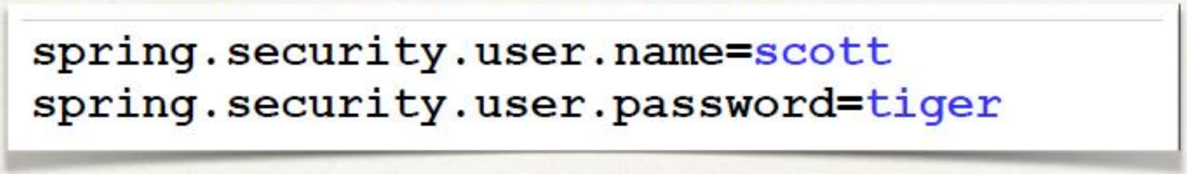
Check console logs for password

```
    11-02 21:05:57.074  INFO 24986 --- [           main] .s.s.UserDetailsSe
Using generated security password: 78fd68a6-c190-421d-934b-df7852fc7dc2
```

Spring Security configuration

- You can override default user name and generated password

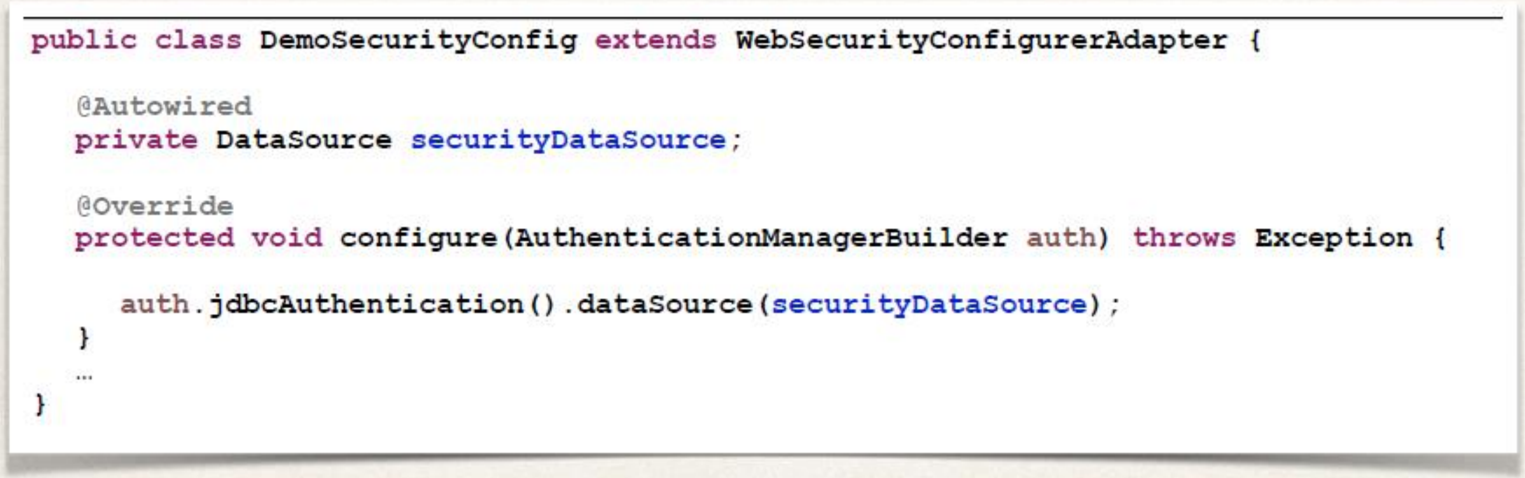
File: src/main/resources/application.properties



```
spring.security.user.name=scott
spring.security.user.password=tiger
```

Customizing Spring Security

- You can customize Spring Security for Spring Boot Actuator
  - Use a database for roles, encrypted passwords etc …
- Follow the same techniques as regular Spring Security

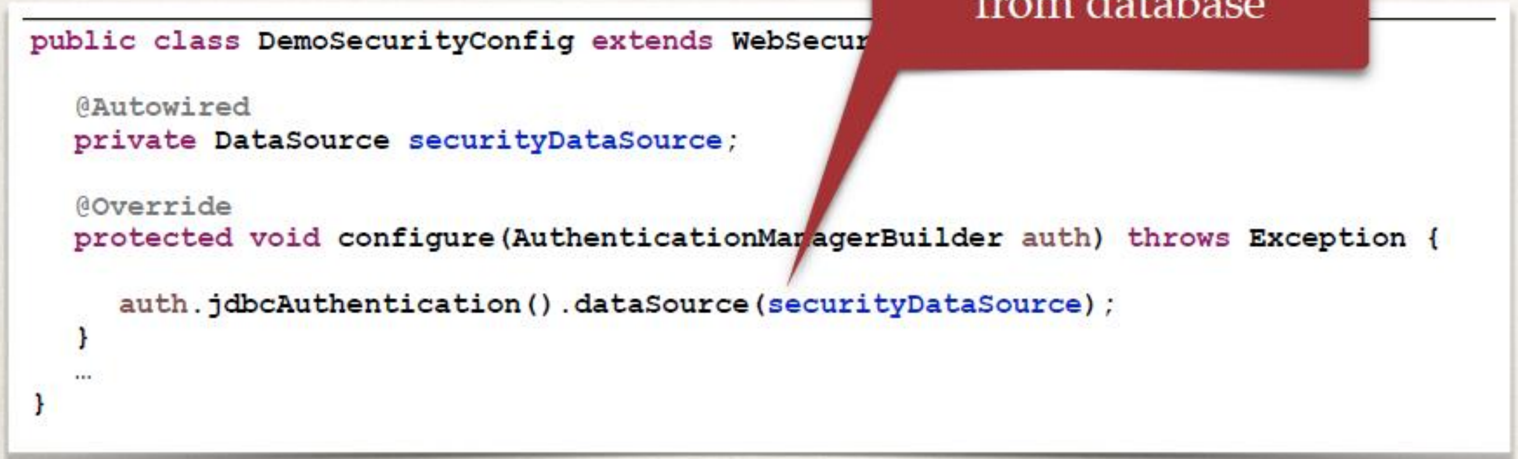


```
public class DemoSecurityConfig extends WebSecurityConfigurerAdapter {

  @Autowired
  private DataSource securityDataSource;

  @Override
  protected void configure(AuthenticationManagerBuilder auth) throws Exception {

    auth.jdbcAuthentication().dataSource(securityDataSource);
  }
  …
}
```

Customizing Spring Security

- You can customize Spring Security for Spring Boot Actuator
  - Use a database for roles, encrypted passwords etc …
- Follow the same techniques Security

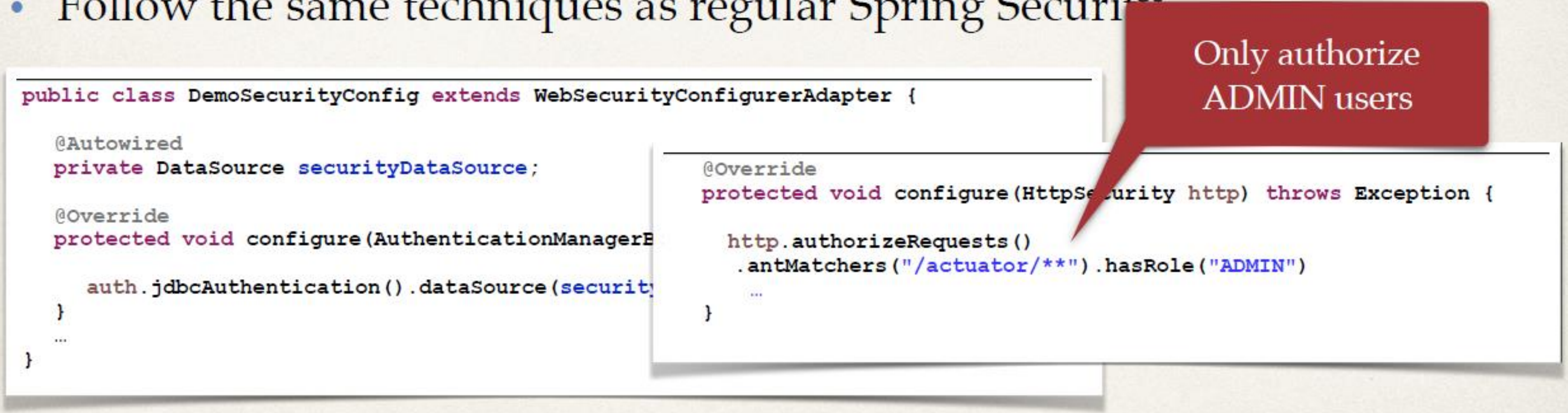
Read security info from database



```
public class DemoSecurityConfig extends WebSecur

   @Autowired
   private DataSource securityDataSource;

   @Override
   protected void configure(AuthenticationManagerBuilder auth) throws Exception {

      auth.jdbcAuthentication().dataSource(securityDataSource);
   }
   …
}
```

Customizing Spring Security

- You can customize Spring Security for Spring Boot Actuator
    - Use a database for roles, encrypted passwords etc …
- Follow the same techniques as regular Spring Security



```
public class DemoSecurityConfig extends WebSecurityConfigurerAdapter {

   @Autowired
   private DataSource securityDataSource;

   @Override
   protected void configure(AuthenticationManagerB

      auth.jdbcAuthentication().dataSource(securit
   }
   …
}
```

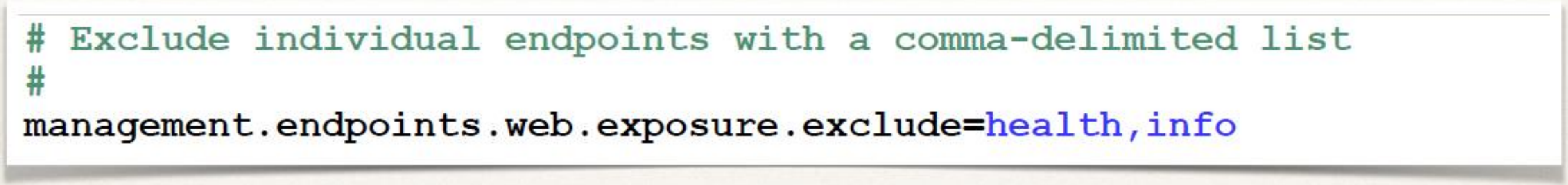
```
@Override
protected void configure(HttpSecurity http) throws Exception {

  http.authorizeRequests()
   .antMatchers("/actuator/**").hasRole("ADMIN")
    …
}
```

Only authorize ADMIN users

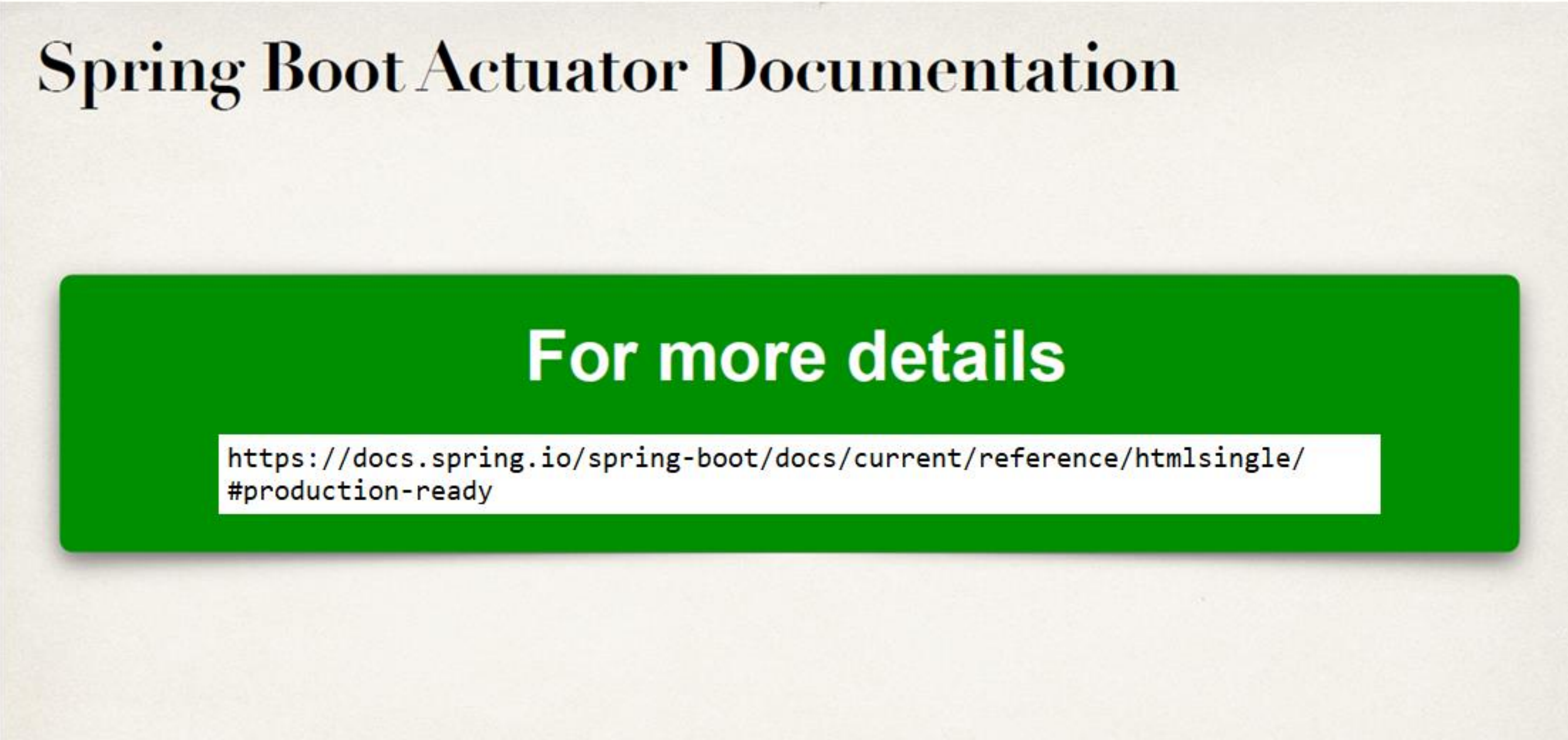
Excluding Endpoints

- To exclude **`/health`** and **`/info`**

File: src/main/resources/application.properties



```
# Exclude individual endpoints with a comma-delimited list
#
management.endpoints.web.exposure.exclude=health,info
```



Spring Boot Actuator Documentation

For more details

https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#production-ready

Development Process

Step-By-Step

1. Edit **`pom.xml`** and add **`spring-boot-starter-security`**

2. Verify security on actuator endpoints for: **`/beans`** etc

3. Disable endpoints for **`/health`** and **`/info`**