轻松入门SpringSecurity

讲师：李新杰

https://github.com/endofprogram/eop-ssei

安全(Security)

Spring Security是一个基于Spring企业应用的安全框架

认证(Authentication)

授权(Authorization)

框架(Framework)

Spring是一个现在企业应用开发的框架

所谓框架就是定义了宏观架构和执行流程，但其本身并不能直接运行，需要按要求用代码和配置文件进行填充之后，方可运行

所以框架会定义很多规则和预留点，供用户进行简单配置或深度定制

规则(Rule)

规则定义一个框架的集成方式和使用方法

引入哪些依赖的jar包
需要提供什么配置文件
使用哪个注解启用功能
至少要定义哪些类
是否需要把它们注册到容器中

预留点(Reserve point)

预留点提供了用户按照自己的意愿来使用框架的能力，即个性化定制

设置属性
设置回调方法
重写指定方法

如何集成(Integration)

通过maven引入依赖

定义一个类来继承AbstractSecurityWebApplicationInitializer

定义一个类来继承WebSecurityConfigurerAdapter，并把该类放入容器

配置(Configuration)

登陆页面
点击登陆按钮提交到的url
登陆成功时跳转的页面
登陆失败时跳转的页面
登陆成功时返回的JSON
登陆失败时返回的JSON
如何与存在数据库里的用户名/密码关联
。。。。。。

找到规则和预留点

用户(User)

UserDetails ，接口，表示用户
User，用户实现类
UserDetailsService，接口，负责加载用户

PasswordEncoder，接口，负责密码加密

注意：这些内容只在登陆（认证）的时候使用

已认证对象(Authentication)

当用户成功登陆后，系统会生成一个Authentication对象，供后续权限校验（授权）时使用

从其中可以获取用户信息，权限信息等

RBAC(Role-Based Access Control)

用户：User
角色：Role
权限：Auth
用户-角色：User_Role
角色-权限：Role_Auth

登陆/登出(Login/Logout)

登陆页面
点击登陆按钮提交到的url
登陆成功时跳转的页面
登陆失败时跳转的页面
登陆成功时返回的JSON
登陆失败时返回的JSON
点击登出按钮提交到的url
登出成功时跳转的页面
登出成功时返回的JSON

权限/角色(Auth/Role)

Web应用通过url进行访问，所以权限其实和url有密切关系

所以需要配置url和角色（权限）的对应关系
无权访问时的提示页面
无权访问时返回的JSON

问题(Question)

Admin角色应该能访问所有的url

url和角色（权限）的对应关系应该动态加载