一个Shiro的简单例子

|  |
| --- |
| https://timgsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1514915023242&di=80a818e34d2652f6526c56f9e3c5befb&imgtype=jpg&src=http%3A%2F%2Fimg4.imgtn.bdimg.com%2Fit%2Fu%3D4182909289%2C1217339124%26fm%3D214%26gp%3D0.jpg |

Pom依赖

|  |
| --- |
| <dependency>  <groupId>org.apache.shiro</groupId>  <artifactId>shiro-all</artifactId>  <version>1.2.2</version>  </dependency> |

Web.xml配置shiro框架(过滤器)

|  |
| --- |
| <filter>  <filter-name>shiroFilter</filter-name>  <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>  <init-param>  <param-name>targetFilterLifecycle</param-name>  <param-value>true</param-value>  </init-param>  </filter>  <filter-mapping>  <filter-name>shiroFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |

Spring容器中创建shiro的安全管理器

|  |
| --- |
| <!--  1. 配置 SecurityManager!  -->  <bean id=*"securityManager"* class=*"org.apache.shiro.web.mgt.DefaultWebSecurityManager"*>  <property name=*"realm"* ref=*"testRealm"*></property>  </bean>    <!--  2. 配置 Realm  2.1 直接配置实现了 org.apache.shiro.realm.Realm 接口的 bean  -->  <bean id=*"testRealm"* class=*"com.atguigu.realms.TestRealm"*></bean>    <!--  3. 配置 ShiroFilter.  -->  <bean id=*"shiroFilter"* class=*"org.apache.shiro.spring.web.ShiroFilterFactoryBean"*>  <property name=*"securityManager"* ref=*"securityManager"*/>    <!--  配置哪些页面需要受保护.  以及访问这些页面需要的权限.  1). anon 可以被匿名访问  2). authc 必须认证(即登录)后才可能访问的页面.  3). logout 登出.  4). roles 角色过滤器  -->    <property name=*"filterChainDefinitions"*>  <value>  /login.jsp = anon  /index.jsp = anon  /login.do = anon  /logout.do = logout    /user.jsp = roles[user]  /admin.jsp = roles[admin]    # everything else requires authentication:  /\*\* = authc  </value>  </property>  </bean> |