1、引入jar的依赖

|  |
| --- |
| <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  <modelVersion>4.0.0</modelVersion>  <groupId>cn.itcast.demo</groupId>  <artifactId>casclient\_demo1</artifactId>  <version>0.0.1-SNAPSHOT</version>  <packaging>war</packaging>   <properties>  <spring.version>4.2.4.RELEASE</spring.version>  </properties>  <!-- spring和spring security的依赖 -->  <dependencies>   <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-core</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-web</artifactId>  <version>${spring.version}</version>  </dependency>   <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-webmvc</artifactId>  <version>${spring.version}</version>  </dependency>   <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context-support</artifactId>  <version>${spring.version}</version>  </dependency>   <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-test</artifactId>  <version>${spring.version}</version>  </dependency>   <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jdbc</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>javax.servlet</groupId>  <artifactId>servlet-api</artifactId>  <version>2.5</version>  <scope>provided</scope>  </dependency>   <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-web</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>   <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-config</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>    <!-- cas -->  <!-- spring security和cas的集成 -->  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-cas</artifactId>  <version>4.1.0.RELEASE</version>  </dependency>  <!-- 排除cas中的slf4j包，防止和spring的slf4j冲突-->  <dependency>  <groupId>org.jasig.cas.client</groupId>  <artifactId>cas-client-core</artifactId>  <version>3.3.3</version>  <exclusions>  <exclusion>  <groupId>org.slf4j</groupId>  <artifactId>log4j-over-slf4j</artifactId>  </exclusion>  </exclusions>  </dependency>   </dependencies>  <build>  <plugins>  <!-- java编译插件 -->  <plugin>  <groupId>org.apache.maven.plugins</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <version>3.2</version>  <configuration>  <source>1.7</source>  <target>1.7</target>  <encoding>UTF-8</encoding>  </configuration>  </plugin>  <plugin>  <groupId>org.apache.tomcat.maven</groupId>  <artifactId>tomcat7-maven-plugin</artifactId>  <configuration>  <!-- 指定端口 -->  <port>9002</port>  <!-- 请求路径 -->  <path>/</path>  </configuration>  </plugin>  </plugins>  </build>  </project> |

2、修改spring-security.xml：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <beans:beans xmlns="http://www.springframework.org/schema/security"  xmlns:beans="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd  http://www.springframework.org/schema/security http://www.springframework.org/schema/security/spring-security.xsd">    <!-- entry-point-ref 设置cas入口点引用 -->  <http use-expressions="false" entry-point-ref="casProcessingFilterEntryPoint">   <intercept-url pattern="/\*\*" access="ROLE\_USER"/>   <csrf disabled="true"/>   <!-- custom-filter为过滤器， position 表示将过滤器放在指定的位置上，before表示放在指定位置之前 ，after表示放在指定的位置之后 -->   <custom-filter ref="casAuthenticationFilter" position="CAS\_FILTER" />   <custom-filter ref="requestSingleLogoutFilter" before="LOGOUT\_FILTER"/>   <custom-filter ref="singleLogoutFilter" before="CAS\_FILTER"/>   </http>    <!-- CAS入口点 开始 -->  <beans:bean id="casProcessingFilterEntryPoint" class="org.springframework.security.cas.web.CasAuthenticationEntryPoint">   <!-- 单点登录服务器登录URL -->   <beans:property name="loginUrl" value="http://localhost:8080/cas/login"/>  <beans:property name="serviceProperties" ref="serviceProperties"/>   </beans:bean>   <beans:bean id="serviceProperties" class="org.springframework.security.cas.ServiceProperties">   <!--com.pinyougou.user.service 配置自身工程的根地址+/login/cas -->  <beans:property name="service" value="http://localhost:9002/login/cas"/>  </beans:bean>   <!-- CAS入口点 结束 -->     <!-- 认证过滤器 开始 -->  <beans:bean id="casAuthenticationFilter" class="org.springframework.security.cas.web.CasAuthenticationFilter">   <beans:property name="authenticationManager" ref="authenticationManager"/>   </beans:bean>   <!-- 认证管理器 -->  <authentication-manager alias="authenticationManager">  <authentication-provider ref="casAuthenticationProvider">  </authentication-provider>  </authentication-manager>  <!-- 认证提供者 -->  <beans:bean id="casAuthenticationProvider" class="org.springframework.security.cas.authentication.CasAuthenticationProvider">   <beans:property name="authenticationUserDetailsService">   <beans:bean class="org.springframework.security.core.userdetails.UserDetailsByNameServiceWrapper">   <beans:constructor-arg ref="userDetailsService" />   </beans:bean>   </beans:property>   <beans:property name="serviceProperties" ref="serviceProperties"/>   <!-- ticketValidator 为票据验证器 -->  <beans:property name="ticketValidator">   <beans:bean class="org.jasig.cas.client.validation.Cas20ServiceTicketValidator">   <beans:constructor-arg index="0" value="http://localhost:8080/cas"/>  </beans:bean>   </beans:property>   <beans:property name="key" value="an\_id\_for\_this\_auth\_provider\_only"/>   </beans:bean>   <!-- 认证类 -->  <beans:bean id="userDetailsService" class="cn.itcast.demo.service.UserDetailServiceImpl"/>    <!-- 认证过滤器 结束 -->      <!-- 单点登出 开始 -->   <beans:bean id="singleLogoutFilter" class="org.jasig.cas.client.session.SingleSignOutFilter"/>  <!--经过此配置，当用户在地址栏输入本地工程/logout/cas-->  <beans:bean id="requestSingleLogoutFilter" class="org.springframework.security.web.authentication.logout.LogoutFilter">   <beans:constructor-arg value="http://localhost:8080/cas/logout?com.pinyougou.user.service=http://www.baidu.com"/>  <beans:constructor-arg>   <beans:bean class="org.springframework.security.web.authentication.logout.SecurityContextLogoutHandler"/>   </beans:constructor-arg>   <beans:property name="filterProcessesUrl" value="/logout/cas"/>   </beans:bean>   <!-- 单点登出 结束 -->    </beans:beans> |