# 分布式服务认证(oauth2.0)

# 技术架构

springBoot+springSecurity+springCloud+oauth2.0+jwt

# 环境搭建

## 搭建父工程distributed-security

### 编写pom.xml

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <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 https://maven.apache.org/xsd/maven-4.0.0.xsd">  <modelVersion>4.0.0</modelVersion>  <parent>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-parent</artifactId>  <version>2.2.3.RELEASE</version>  <relativePath/> <!-- lookup parent from repository -->  </parent>  <groupId>com.liu.distributedSecurity</groupId>  <artifactId>distributed-security</artifactId>  <version>0.0.1-SNAPSHOT</version>  <name>distributed-security</name>  <description>分布式服务认证</description>   <properties>  <java.version>1.8</java.version>  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>  </properties>   <dependencyManagement>  <dependencies>  <!--springCloud的依赖-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-dependencies</artifactId>  <version>Greenwich.RELEASE</version>  <type>pom</type>  <scope>import</scope>  </dependency>  <!--servlet-api的依赖-->  <dependency>  <groupId>javax.servlet</groupId>  <artifactId>javax.servlet-api</artifactId>  <version>4.0.1</version>  <scope>provided</scope>  </dependency>  <!--jwt依赖-->  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-jwt</artifactId>  <version>1.0.10.RELEASE</version>  </dependency>  <!--oauth2的依赖-->  <dependency>  <groupId>org.springframework.security.oauth</groupId>  <artifactId>spring-security-oauth2</artifactId>  <version>2.3.6.RELEASE</version>  </dependency>  <dependency>  <groupId>org.springframework.security.oauth.boot</groupId>  <artifactId>spring-security-oauth2-autoconfigure</artifactId>  <version>2.1.3.RELEASE</version>  </dependency>   <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-web</artifactId>  </dependency>   <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-devtools</artifactId>  <scope>runtime</scope>  <optional>true</optional>  </dependency>   <dependency>  <groupId>org.projectlombok</groupId>  <artifactId>lombok</artifactId>  <optional>true</optional>  </dependency>   <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-test</artifactId>  <scope>test</scope>  <exclusions>  <exclusion>  <groupId>org.junit.vintage</groupId>  <artifactId>junit-vintage-engine</artifactId>  </exclusion>  </exclusions>  </dependency>  </dependencies>  </dependencyManagement>   <build>  <plugins>  <plugin>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-maven-plugin</artifactId>  </plugin>  </plugins>  </build>  </project> |

## 搭建common子工程

用于存放公共的工具类，实体类，service及mapper

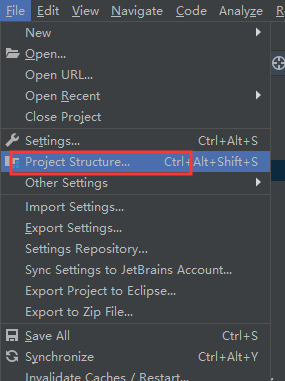
### 创建common子工程

### 编写公共的工具类，实体类，service及mapper

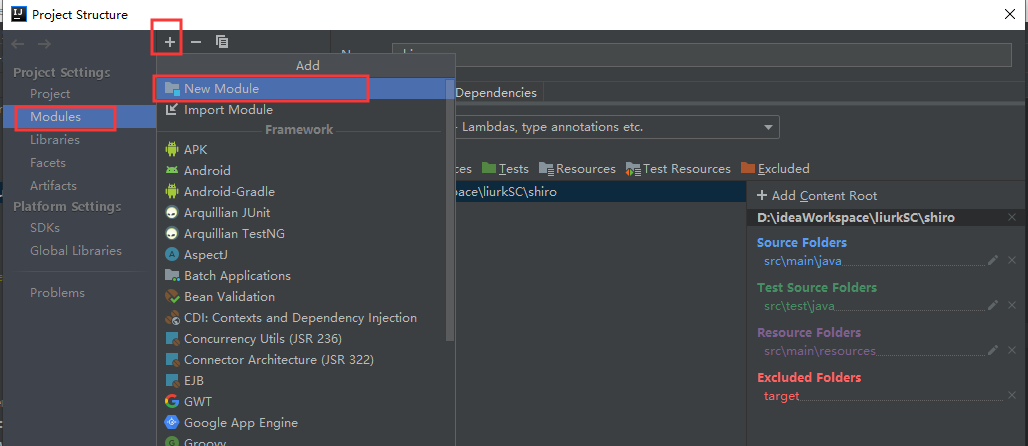
## 搭建uaa授权服务子工程

### 2.1.1 创建uaa授权服务子工程

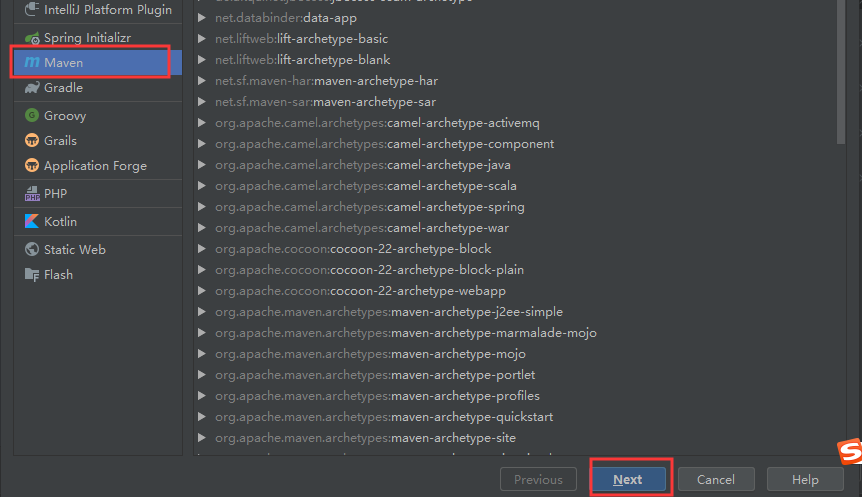
file --> Project Structure



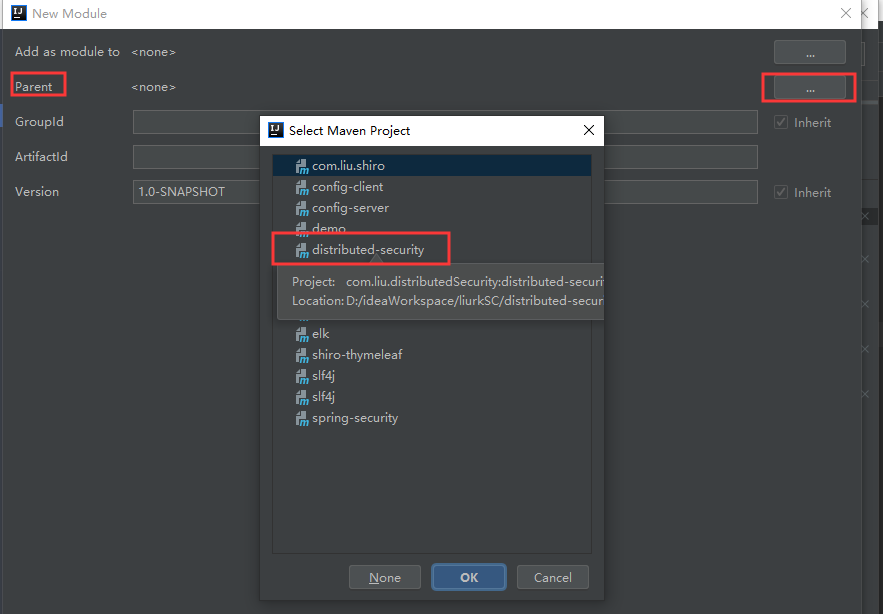
选择Modules，点击+号，选择New Module



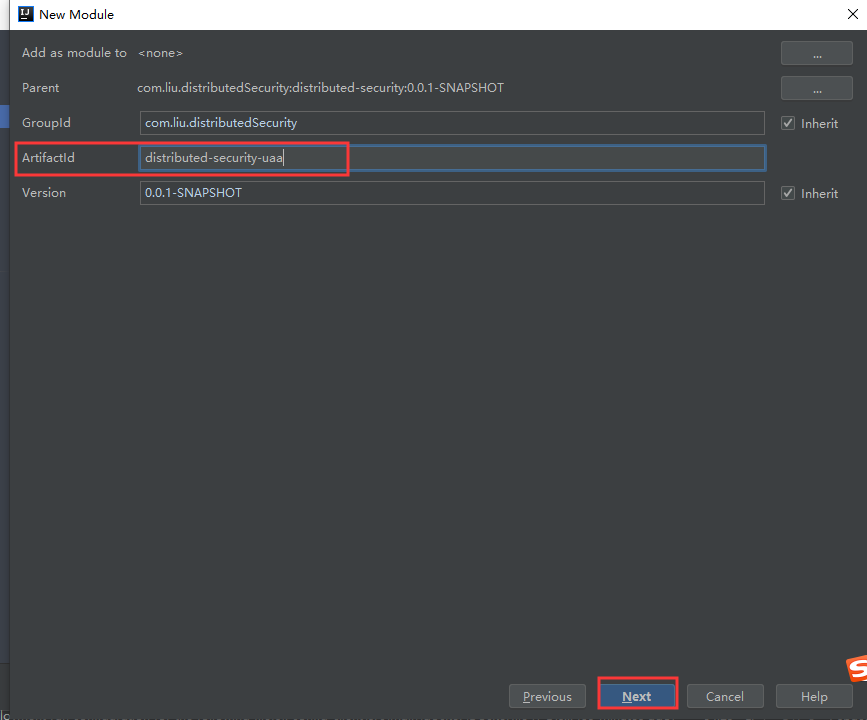
选择maven，点击next



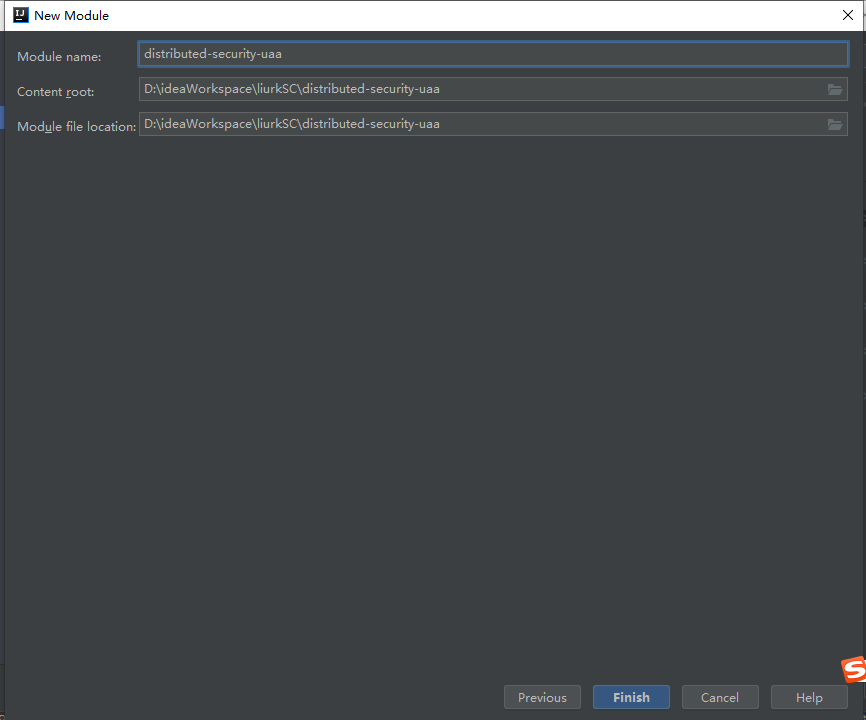
parent选择我们的父工程，点击ok



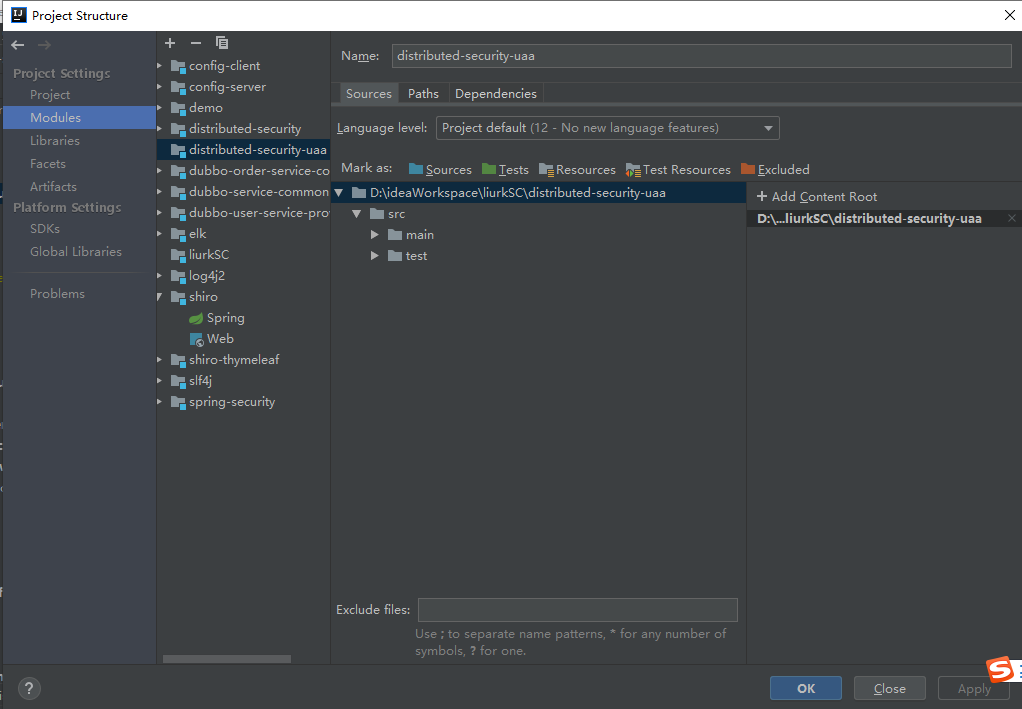
输入Artifactld(distributed-security-uaa)，点击Next



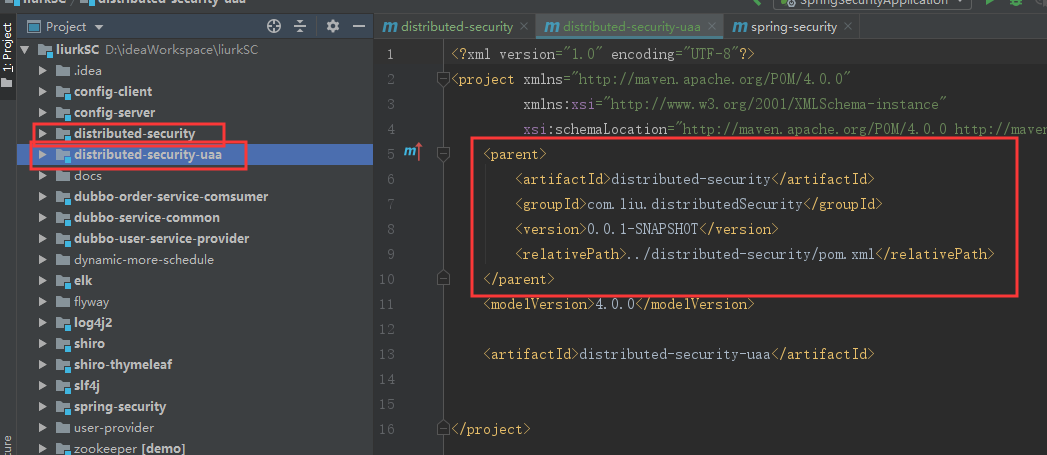
修改module name，点击finish



点击apply，再点击ok



至此，父子工程创建完毕



### 2.2.2 编写pom.xml

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <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">  <parent>  <artifactId>distributed-security</artifactId>  <groupId>com.liu.distributedSecurity</groupId>  <version>0.0.1-SNAPSHOT</version>  <relativePath>../distributed-security/pom.xml</relativePath>  </parent>  <modelVersion>4.0.0</modelVersion>   <artifactId>distributed-security-uaa</artifactId>   <!--子工程里面的依赖不用写版本，因为都依赖父工程里面的版本-->  <dependencies>  <!--依赖common模块-->  <dependency>  <groupId>com.liu.distributedSecurity</groupId>  <artifactId>distributed-security-common</artifactId>  <version>0.0.1-SNAPSHOT</version>  </dependency>  <!--eureka-client服务注册中心客户端-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-netflix-eureka-client</artifactId>  </dependency>  <!--hystrix熔断-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-netflix-hystrix</artifactId>  </dependency>  <!--ribbon负载均衡-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-netflix-ribbon</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-openfeign</artifactId>  </dependency>  <!---->  <dependency>  <groupId>com.netflix.hystrix</groupId>  <artifactId>hystrix-javanica</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.retry</groupId>  <artifactId>spring-retry</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-actuator</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-freemarker</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.data</groupId>  <artifactId>spring-data-commons</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-security</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-oauth2</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-jwt</artifactId>  </dependency>  <!---->  <!--<dependency>  <groupId>javax.interceptor</groupId>  <artifactId>javax.interceptor.api</artifactId>  </dependency>-->  <!--mysql数据库-->  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  <scope>runtime</scope>  </dependency>  <!--mybatis-plus插件-->  <dependency>  <groupId>com.baomidou</groupId>  <artifactId>mybatis-plus-boot-starter</artifactId>  <version>3.1.0</version>  </dependency>  <!--数据库连接池-->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>druid</artifactId>  <version>1.1.6</version>  </dependency>  </dependencies>  </project> |

### 2.2.3 编写UaaApplication启动类和application.yml配置文件

## 2.4 搭建order服务资源子工程

### 2.4.1 创建order服务资源子工程

### 2.4.2 编写pom.xml

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?> <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">  <parent>  <artifactId>distributed-security</artifactId>  <groupId>com.liu.distributedSecurity</groupId>  <version>0.0.1-SNAPSHOT</version>  <relativePath>../distributed-security/pom.xml</relativePath>  </parent>  <modelVersion>4.0.0</modelVersion>   <artifactId>distributed-security-order</artifactId>   <!--子工程里面的依赖不用写版本，因为都依赖父工程里面的版本-->  <dependencies>  <!--依赖common模块-->  <dependency>  <groupId>com.liu.distributedSecurity</groupId>  <artifactId>distributed-security-common</artifactId>  <version>0.0.1-SNAPSHOT</version>  </dependency>  <!--eureka-client服务注册中心客户端-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-netflix-eureka-client</artifactId>  </dependency>  <!--hystrix熔断-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-netflix-hystrix</artifactId>  </dependency>  <!--ribbon负载均衡-->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-netflix-ribbon</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-openfeign</artifactId>  </dependency>  <!---->  <dependency>  <groupId>com.netflix.hystrix</groupId>  <artifactId>hystrix-javanica</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.retry</groupId>  <artifactId>spring-retry</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-actuator</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-freemarker</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.data</groupId>  <artifactId>spring-data-commons</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-security</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.cloud</groupId>  <artifactId>spring-cloud-starter-oauth2</artifactId>  </dependency>  <!---->  <dependency>  <groupId>org.springframework.security</groupId>  <artifactId>spring-security-jwt</artifactId>  </dependency>  <!---->  <!--<dependency>  <groupId>javax.interceptor</groupId>  <artifactId>javax.interceptor.api</artifactId>  </dependency>-->  <!--mysql数据库-->  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  <scope>runtime</scope>  </dependency>  <!--mybatis-plus插件-->  <dependency>  <groupId>com.baomidou</groupId>  <artifactId>mybatis-plus-boot-starter</artifactId>  <version>3.1.0</version>  </dependency>  <!--数据库连接池-->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>druid</artifactId>  <version>1.1.6</version>  </dependency>  </dependencies>  </project> |

### 2.4.3 编写UaaApplication启动类和application.yml配置文件