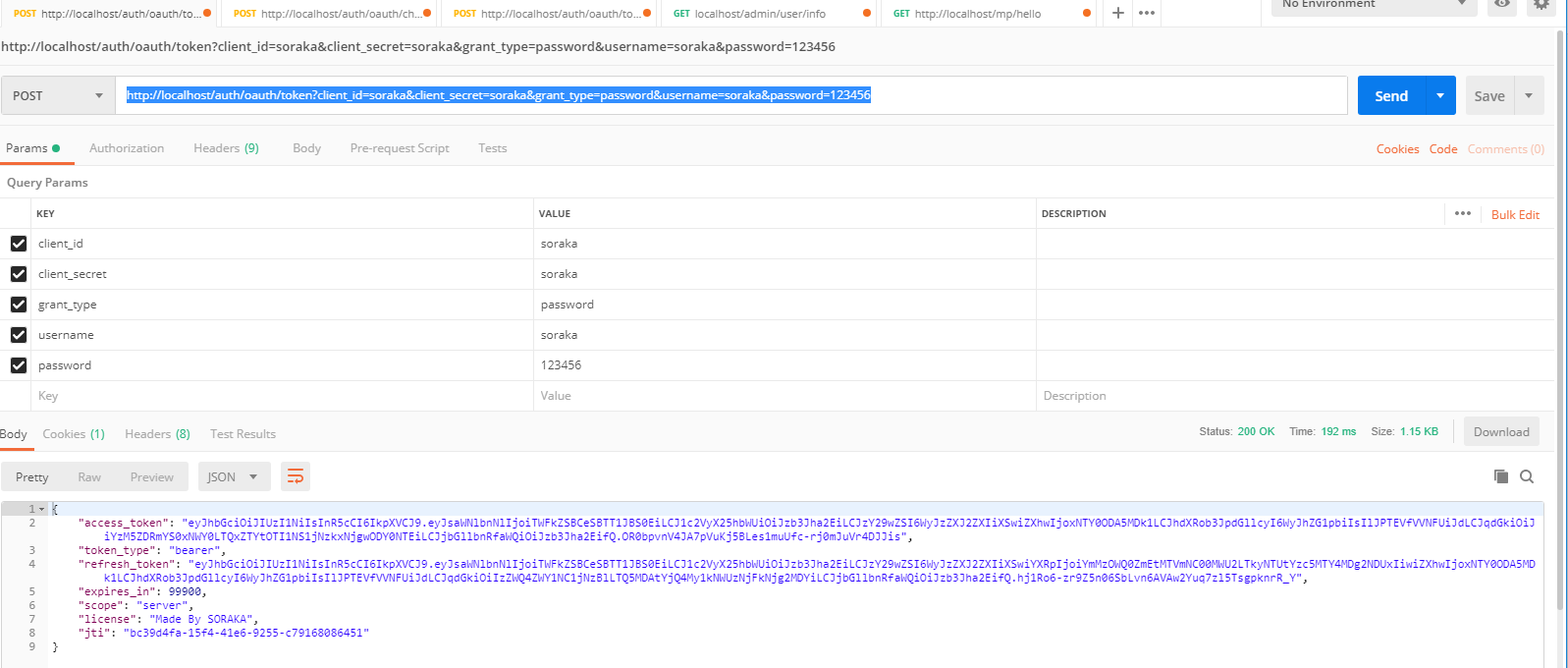
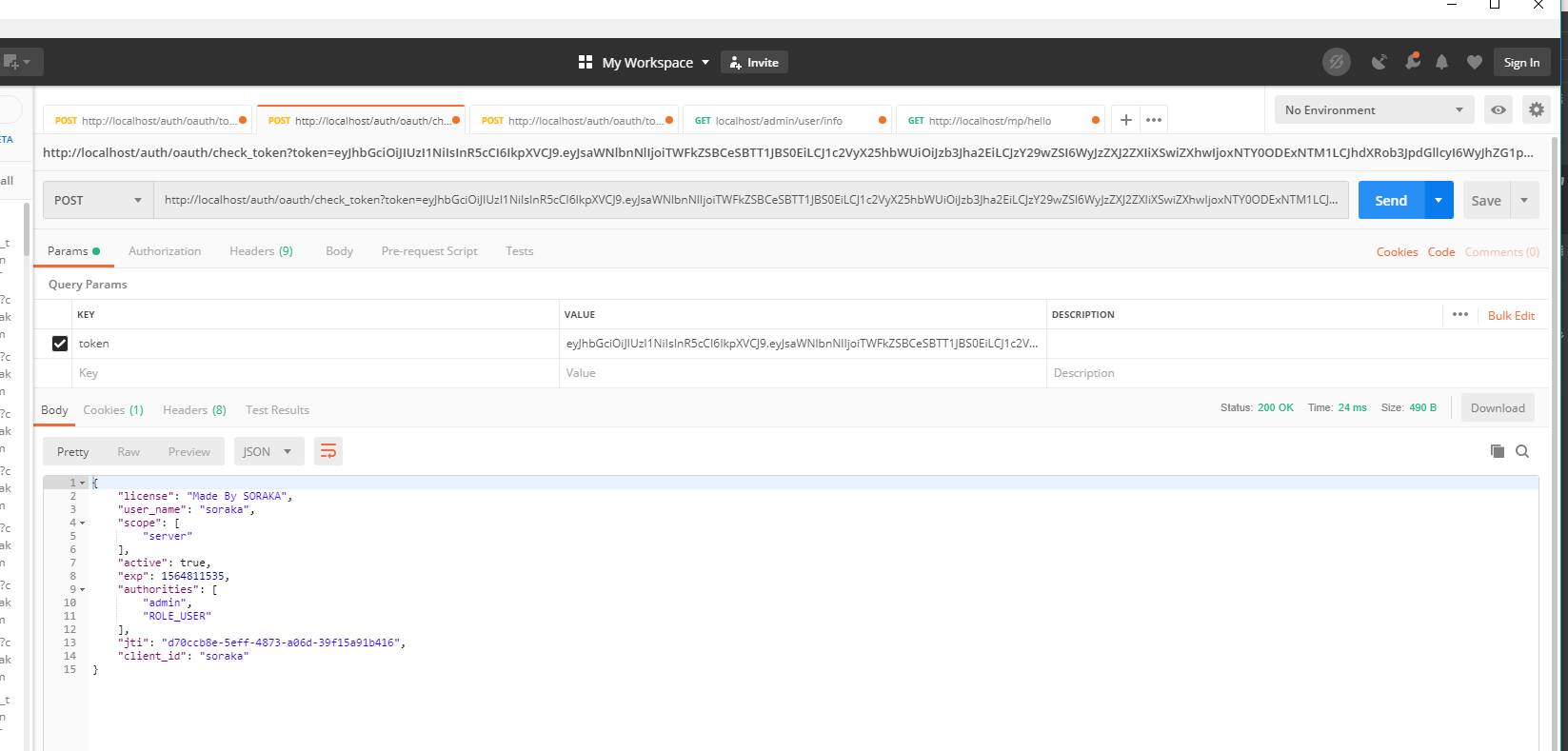
1. zuul,Hystrix
2. 登陆鉴权 Spring Security OAuth jwt token.
3. 鉴权原理: OAuth 2.0
4. 部署步骤，数据库数据导入admin下的SQL.依次启动discovery，auth，admin，gateway
5. 测试通过zuuul网关去测试.
6. 拿token

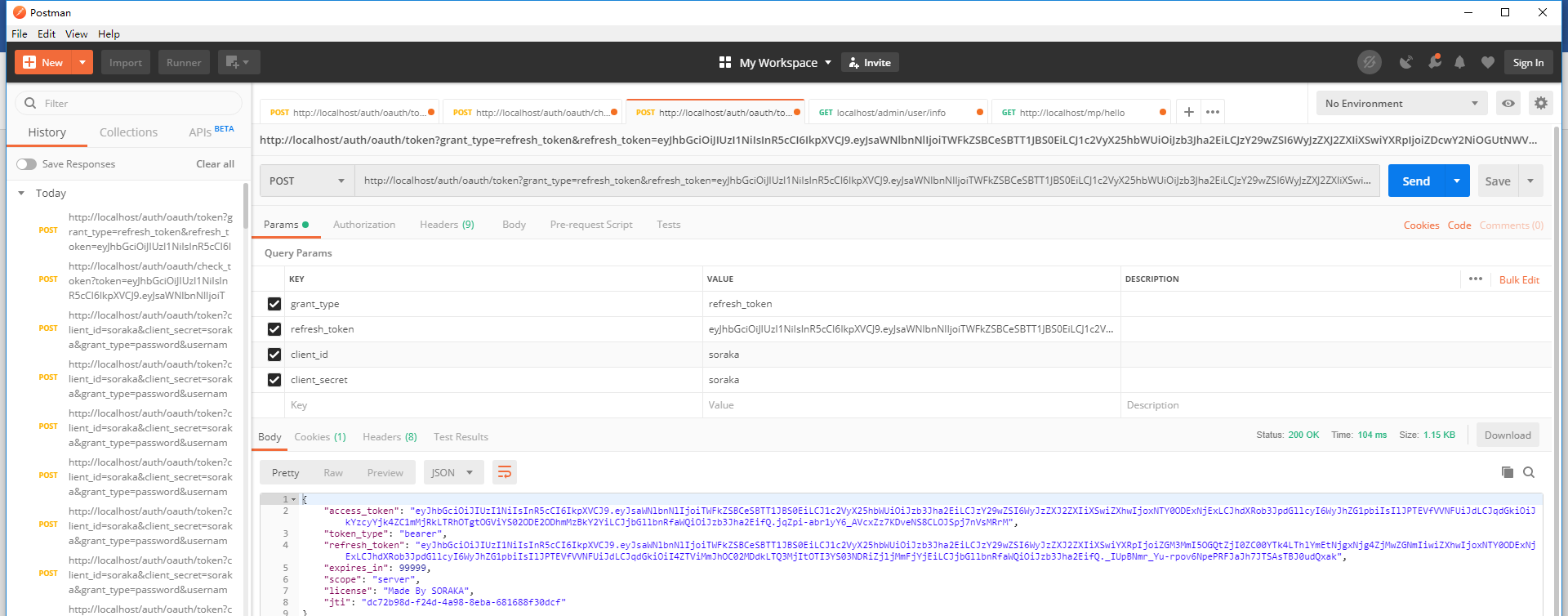
<http://localhost/auth/oauth/token?client_id=soraka&client_secret=soraka&grant_type=password&username=soraka&password=123456>



B．验证token



C．刷新token



1. 实现步骤

A:加密，验证

String p = new BCryptPasswordEncoder().encode("123456");

boolean f = new BCryptPasswordEncoder().matches("soraka",p);

可以修改sys\_oauth\_client\_details表中的密码

B: OAuth 2.0要有client\_id，client\_secret

clientDetailsService.setSelectClientDetailsSql(Constants.DEFAULT\_SELECT\_STATEMENT);

clientDetailsService.setFindClientDetailsSql(Constants.DEFAULT\_FIND\_STATEMENT);

密码可以通过以上的方法去做密码，默认用户密码是: soraka/ soraka

C:ZUUL

*#认证服务器地址，通过client-id，client-secret访问认证服务器***security**:  
 **auth**:  
 **server**: http://localhost:8005  
 **oauth2**:  
 **client**:  
 **client-id**: soraka  
 **client-secret**: soraka  
 **resource**:  
 **jwt**:  
 **key-uri**: ${**security.auth.server**}/oauth/token\_key *#解析jwt令牌所需要密钥的地址*

SorakaResourceConfigurerAdapter中配置antMatchers .permitAll()代表以下路径不去hasPermission中进行鉴权，//以下配置不会去hasPermission方法，但是都会需要登陆token验证，

//如果不需要登陆token验证则再SorakaSecurityConfigurerAdapter里配置。

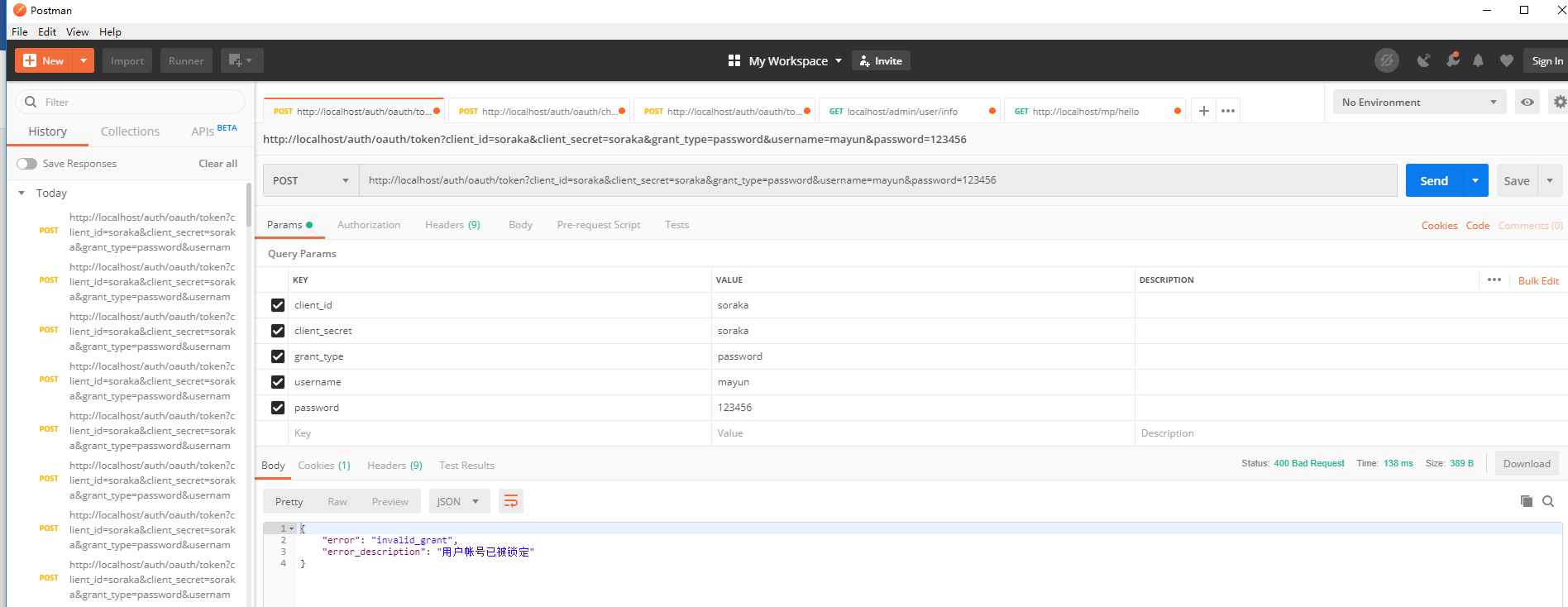
SorakaAdminFallbackProvider 里可以配置FallbackProvider内容

PermissionServiceImpl，利用Feign获取menuservice,然后

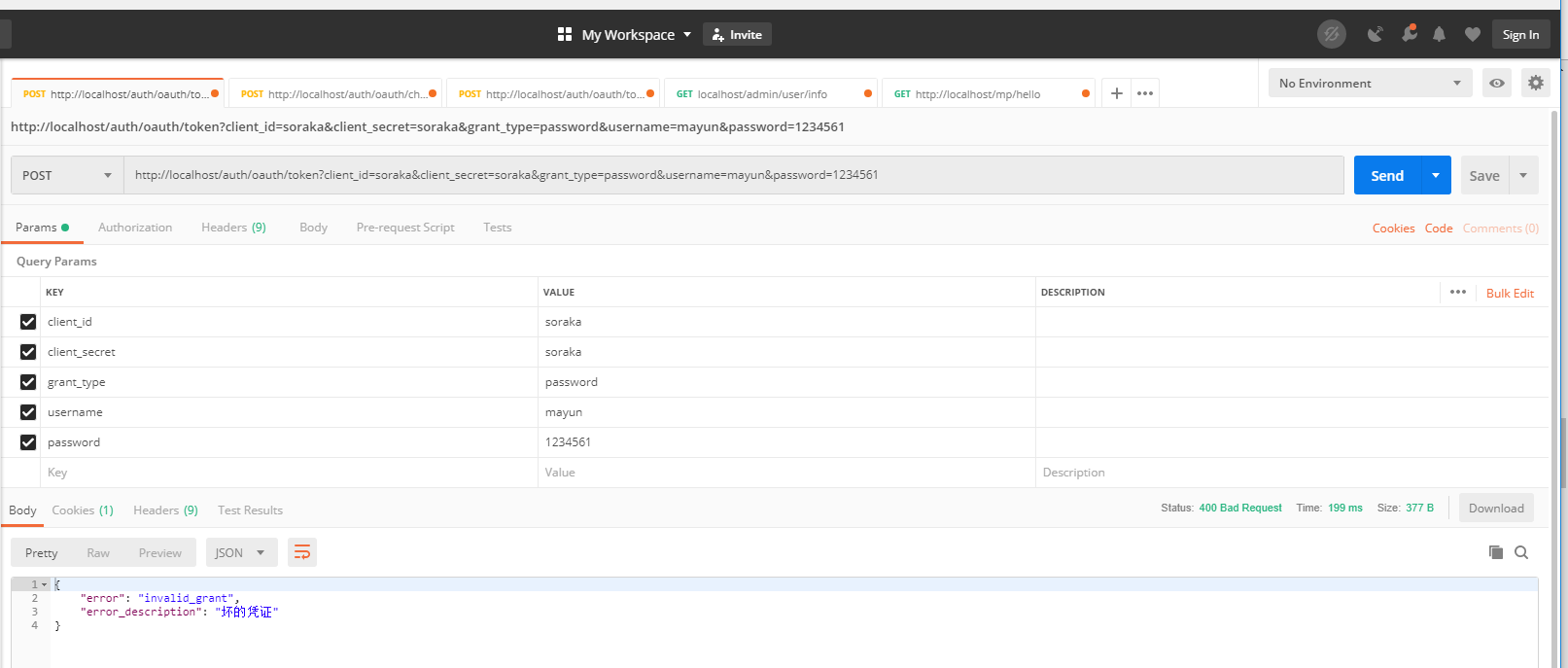
1. soraka-auth

生成token服务器，把user信息注册到UserDetailsService，实现里面注册的方法几个@Override方法。

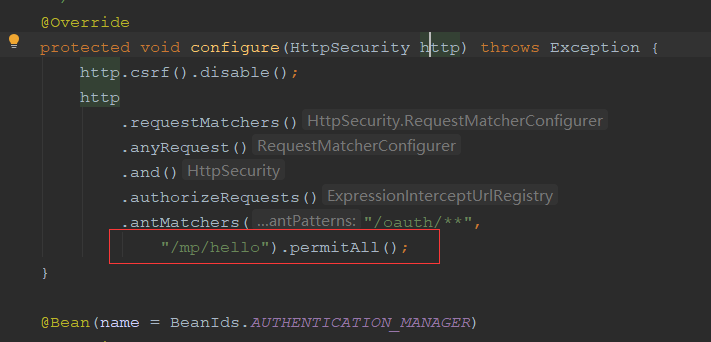
特殊情况mayun账号是status改成0，去生成token的时候会有错误提示



密码填错.



不需要token的话:



antMatchers 不需要用token鉴权的再此列出，SorakaResourceConfigurerAdapter也要放开鉴权，例如/mp/hello。

1. soraka-admin:获取token信息

private Claims getJwtClaims() {  
 String authorization = request.getHeader(Constants.*TOKEN\_HEADER*);  
 String token = StringUtils.*substringAfter*(authorization, Constants.*TOKEN\_BEARER*);  
 String key = Base64.*getEncoder*().encodeToString(Constants.*JWT\_SIGN\_KEY*.getBytes());  
 Claims claims = Jwts.*parser*().setSigningKey(key).parseClaimsJws(token).getBody();  
 return claims;  
}