

支持Nacos 2.1.0! 这套Spring Cloud Gateway+Oauth2终极权限解决方案升级了!

原创 梦想de星空 macrozheng 2022-07-13 09:02 发表于江苏

收录于合集

#Spring Cloud学习教程

26个

最近经常有小伙伴问我关于在微服务中使用Oauth2的问题，其实之前已经写过一篇相关文章了。这次抽空把之前文章中的Demo给升级了，支持了最新版的Spring Cloud和Nacos。今天再来介绍下这套微服务权限终极解决方案，希望对大家有所帮助!

实现思路

首先还是来聊聊这套解决方案的实现思路，我们理想的解决方案应该是这样的，认证服务负责统一认证，网关服务负责校验认证和鉴权，其他API服务则负责处理自己的业务逻辑。安全相关的逻辑只存在于认证服务和网关服务中，其他服务只是单纯地提供服务而没有任何安全相关逻辑。

这套解决方案中相关服务的划分如下:

- **micro-oauth2-gateway** : 网关服务，负责请求转发和鉴权功能，整合Spring Security+Oauth2;
- **micro-oauth2-auth** : 认证服务，负责对登录用户进行认证，整合Spring Security+Oauth2;
- **micro-oauth2-api** : API服务，受网关服务的保护，用户鉴权通过后可以访问该服务，不整合Spring Security+Oauth2。

升级注意点

- 这里项目的依赖版本都升级了，支持 **SpringBoot 2.7.0** 和最新版的Spring Cloud;

<properties>

<spring-boot.version>2.7.0</spring-boot.version>

<spring-cloud.version>2021.0.3</spring-cloud.version>

```
<spring-cloud-alibaba.version>2021.0.1.0</spring-cloud-alibaba.version>
</properties>
```

- 这里不得不吐槽下Spring Cloud的版本号，之前名字采用了伦敦地铁站的名字，根据字母表的顺序来对应版本时间顺序，后来又改成了年份，现在又添加了之前废弃的地铁站名字作为别名，真是让人迷惑；



- SpringBoot 2.7.0版本中使用Maven插件需要添加版本号；

```
<plugin>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-maven-plugin</artifactId>
  <version>${spring-boot.version}</version>
</plugin>
```

- 最新版本的Spring Cloud已经放弃使用Ribbon来做负载均衡了，转而使用LoadBalancer，所以网关服务 `micro-oauth2-gateway` 中还需添加LoadBalancer依赖；

```
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-loadbalancer</artifactId>
</dependency>
```

- 如果不添加LoadBalancer依赖，从网关调用任意服务会返回 `Service Unavailable` 错误信息；

```
{
  "timestamp": "2022-06-28T02:36:31.680+00:00",
  "path": "/auth/oauth/token",
```

```
"status": 503,  
"error": "Service Unavailable",  
"requestId": "c480cefa-1"  
}
```

- **micro-oauth2-auth** 认证服务需要升级版本, 注意使用的JWT库也要同步升级;

```
<dependencies>  
  <dependency>  
    <groupId>org.springframework.cloud</groupId>  
    <artifactId>spring-cloud-starter-oauth2</artifactId>  
    <version>2.2.5.RELEASE</version>  
  </dependency>  
  <dependency>  
    <groupId>com.nimbusds</groupId>  
    <artifactId>nimbus-jose-jwt</artifactId>  
    <version>9.23</version>  
  </dependency>  
</dependencies>
```

- 下载Nacos 2.1.0版本, 下载地址: <https://github.com/alibaba/nacos/releases>



- 下载完成后解压到指定目录即可，使用如下命令启动Nacos;

```
startup.cmd -m standalone
```

- 访问Nacos控制台，使用账号密码 `nacos:nacos` 进行登录，访问地址：
`http://localhost:8848/nacos/`



使用

本文仅作为微服务权限终极解决方案，Spring Cloud Gateway + Oauth2 实现统一认证和鉴权！ 升级版本的补充，具体代码实现可以参考该文，下面演示下该解决方案中的统一认证和鉴权功能。

- 首先需要启动Nacos和Redis服务，然后依次启动 `micro-oauth2-auth` 、 `micro-oauth2-gateway` 及 `micro-oauth2-api` 服务，启动完成后Nacos服务列表显示如下；



- 使用密码模式获取JWT令牌，访问地址：<http://localhost:9201/auth/oauth/token>



- 不带JWT令牌访问受保护的API接口，访问地址：<http://localhost:9201/api/hello>



- 带JWT令牌访问受保护的API接口，注意请求头 **Authorization** 添加 **Bearer** 前缀，可以正常访问；



- 使用获取到的JWT令牌访问获取当前登录用户信息的接口，访问地址：
<http://localhost:9201/api/user/currentUser>



- 当JWT令牌过期时，使用接口返回的 **refreshToken** 获取新的JWT令牌，访问地址：
<http://localhost:9201/auth/oauth/token>



- 使用没有访问权限的 **andy:123456** 账号登录，访问接口时会返回如下信息，访问地址：
<http://localhost:9201/api/hello>



总结

在微服务系统中实现权限功能时，我们不应该把重复的权限校验功能集成到每个独立的API服务中去，而应该在网关做统一处理，然后通过认证中心去统一认证，这样才是优雅微服务权限解决方案！

项目源码地址

<https://github.com/macrozheng/springcloud-learning/tree/master/micro-oauth2>

微信**8.0**将好友放开到了一万，小伙伴可以加我大号了，先到先得，再满就真没了

扫描下方二维码即可加我微信啦， **2022**，抱团取暖，一起牛逼。



推荐阅读

- [JetBrains正式宣布：产品涨价！](#)
- [当开发同事辞职，接手到垃圾代码怎么办？](#)
- [全网显示 IP 归属地，用上这个开源库，实现也太简单了！](#)
- [新来个技术总监，把DDD落地的那叫一个高级，服气！](#)
- [还在用命令行看日志？快用Kibana吧，可视化日志分析YYDS！](#)
- [Mall电商实战项目全面升级！支持最新版SpringBoot，干掉循环依赖...](#)
- [重磅更新！Mall实战教程全面升级，瞬间高大上了！](#)
- [40K+Star！Mall电商实战项目开源回忆录！](#)



macrozheng

专注Java技术分享，解析优质开源项目。涵盖SpringBoot、SpringCloud、Docker...
240篇原创内容

公众号



阅读原文

喜欢此内容的人还喜欢

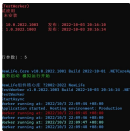
是什么让一段20行代码的性能提升了10倍

阿里开发者



支持多种操作系统的新一代服务主机

智能大石头



技术分享 | Jenkins中，如何管理用户及其相对应权限？

霍格沃兹测试学院

