**Spring Cloud与Docker微服务架构实战笔记**

**151150072 姚政**

GitHub：<https://github.com/yaozheng1998/MovieTicket>

微服务：敏捷响应平台；组件化，松耦合，自治，去中心化；每个服务自己的进程，API交互，可全自动部署；不同的服务可用 不同语言开发，使用不同的数据存储技术。sc轻轻松松几行代码就完成了熔断、均衡负责、服务中心的各种平台功能

微服务设计原则：单一指责原则，服务自治原则，轻量级通信机制，微服务粒度

Rest：<https://www.cnblogs.com/imyalost/p/7923230.html>

Eureka: 实现微服务的注册与发现；集群

Ribbon：控制HTTP和TCP客户端的行为;请求均匀分布到用户微服务节点上实现负载均衡

Feign: 便捷优雅调用HTTP API；仅需创建接口，添加注解;实际业务中会根据不同的角色，执行不同的业务操作，可以自定义完成复杂的服务鉴权。

项目Step:

1. WARNING: Your ApplicationContext **is** unlikely **to** start due **to** a @ComponentScan **of** the **default** **package**.

解决：[main启动类需要放在一个包里]

1. schema.sql文件的路径问题；

解决：classpath

1. Spring Boot Actuator
2. EurekaServer在pom.xml中的配置
3. <http://localhost:8019/log-instance> 页面空白问题
4. Controller调用Feign接口报错：Could not Autowire.

解决：此错误提示不会产生影响，1.ij本身问题；2.导入包的问题；只需设置降低Autowired检测的级别即可；

1. SpringBoot启动多个实例？

解决：<https://blog.csdn.net/forezp/article/details/76408139>

1. Feign报错 Hystrix熔断器会判断服务是否能通，超时则错误返回；

Nested exception is com.netflix.hystrix.exception.HystrixRuntimeException:

UserFeignClient#findById(Long) timed-out and no fallback available.] with root cause

解决：<https://www.cnblogs.com/EasonJim/p/7722372.html>

采用了F3: feign.hystrix.enabled: false

1. Collection<? Extends GrantedAuthority>

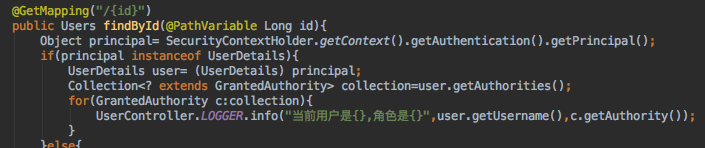
声明该Collection模版类的类型变量可以是所有GrantedAuthority的子类；

1. Logger VS System.out logger打印出详细信息，可以配置，还能保存到日志文件；sys在正式项目基本被禁用，不雅观，还影响效率；【指slj4的logger】

再看6.7 Feign的日志，Feign的logger只对DEBUG级别作出响应；

1. 自定义feign实现服务鉴权，登录无效问题

解决：在WebSecurityConfig配置文件中，isEnabled等参数默认为false，修改为true；而且配置文件需要放在com.yz文件夹下。



user.getAuthorties是Collection集合类；user一个，authority多个？

解决：这里我们只添加了一个auth！

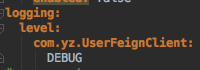


1. 6.7报错：

feign.FeignException: status 401 reading UserFeignClient#findById(Long); content:

{"timestamp":1522402291489,"status":401,"error":"Unauthorized","message":"Full authentication is required to access this resource","path":"/1"}

解决：因为我运行了with-auth，需要验证安全性



需要根据自己的项目结构配置。