


# 微服务

bookservice为查询图书作者的微服务，bookstore为原来的工程。两者均被注册到Eureka上，使得原来的工程可以通过Feign调用bookservice微服务。

前端可以分别通过8082、11130两个端口，访问两个服务；可以通过8443端口访问gateway，从而调用到两个服务；也可以通过8082端口，访问到11130端口的微服务。

配置结果及各服务端口号如下图所示。

HOME    LAST 1000 SINCE STARTUP

System Status

Environment	test	Current time	2021-10-23T16:18:31 +0800
Data center	default	Uptime	00:19
		Lease expiration enabled	true
		Renews threshold	8
		Renews (last min)	12

DS Replicas

localhost

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
BOOKSERVICE	n/a (1)	(1)	UP (1) - 192.168.3.104:bookservice:11130
BOOKSTORE	n/a (1)	(1)	UP (1) - 192.168.3.104:bookstore:8082
GATEWAY	n/a (1)	(1)	UP (1) - 192.168.3.104:gateway:8443

## 搜索图书作者

货币的非国家化

×

🔍

搜索结果如下：

编号	名称	作者
3	货币的非国家化	[英] 弗里德里希·冯·哈耶克

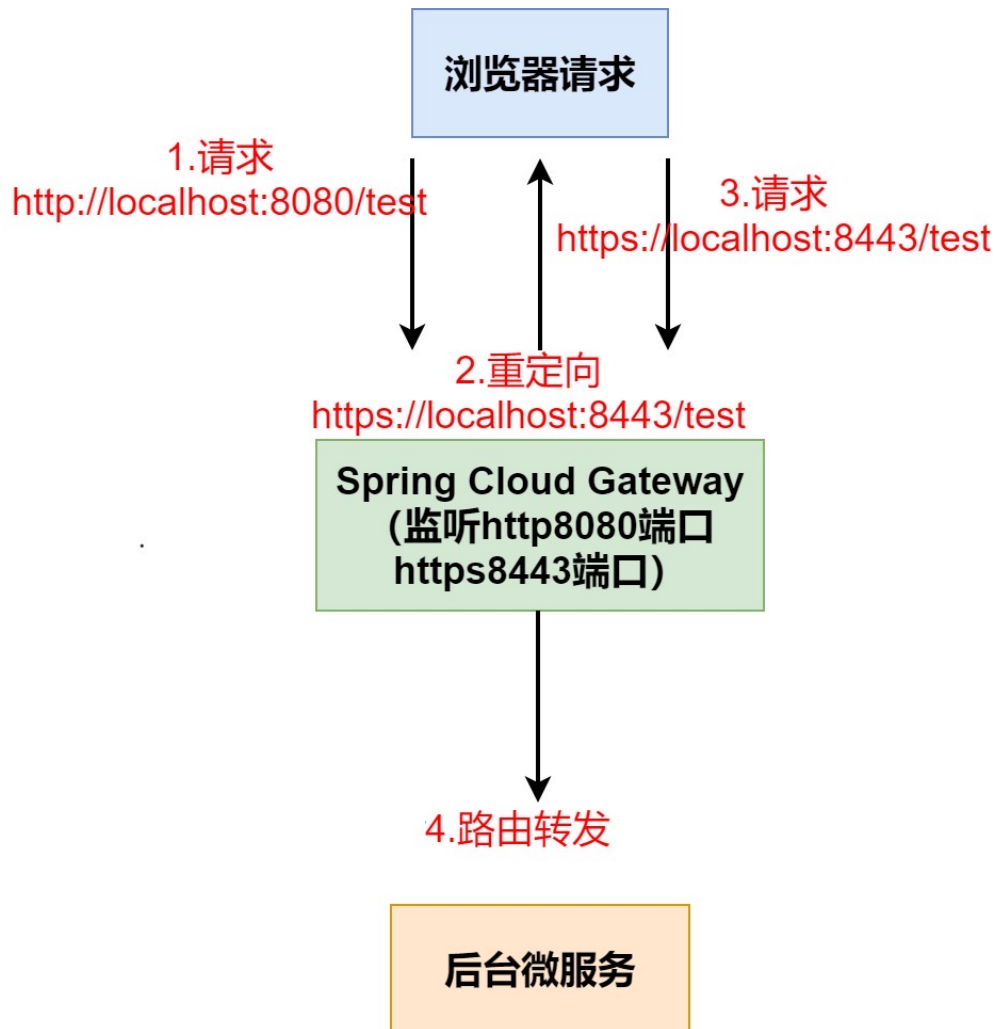
<

1

>

## HTTPS

我对gateway配置了HTTPS通信功能。如下图所示，gateway同时监听http 8080端口与https 8443端口。当访问8080端口时，会重定向至8443端口，8443端口再将请求转发给后台微服务。

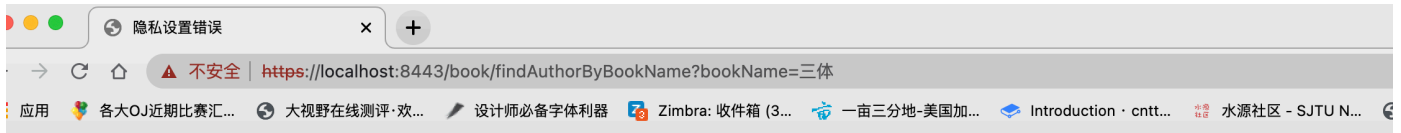


我们对gateway的运行结果进行观察。Spring Cloud Gateway是基于WebFlux实现的，而WebFlux框架底层使用了Netty作为服务容器，而不是Tomcat。Tomcat主要基于http协议，而Netty可以通过编程自定义各种协议。若程序没有采用https，则程序只会打开8080一个端口；若程序采用了https，则程序会打开8080和8443两个端口，同时8443端口注册到Eureka上。

```
2021-10-23 19:39:39.520 INFO 20265 --- [main] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.cloud.client.loadbalancer.reactive.LoadBalancerB
2021-10-23 19:39:39.523 INFO 20265 --- [main] trationDelegate$BeanPostProcessorChecker : Bean 'deferringLoadBalancerExchangeFilterFunction' of type [org.springfram
2021-10-23 19:39:40.245 INFO 20265 --- [main] o.s.b.web.embedded.netty.NettyWebServer : Netty started on port(s): 8080

2021-10-23 19:56:02.854 INFO 24234 --- [InfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient_GATEWAY/192.168.3.104:gateway:8443: registering service..
2021-10-23 19:56:02.937 INFO 24234 --- [InfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient_GATEWAY/192.168.3.104:gateway:8443 - registration status:
2021-10-23 19:56:02.944 INFO 24234 --- [main] o.s.b.web.embedded.netty.NettyWebServer : Netty started on port(s): 8443
2021-10-23 19:56:02.945 INFO 24234 --- [main] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8443
```

可以看到，当使用Chrome浏览器进行HTTPS访问时，由于使用的是级别较低的自签名证书，因此Chrome会认为证书不受信任，阻止访问。



## 您的连接不是私密连接

攻击者可能会试图从 **localhost** 窃取您的信息（例如：密码、通讯内容或信用卡信息）。

[了解详情](#)

NET::ERR\_CERT\_INVALID

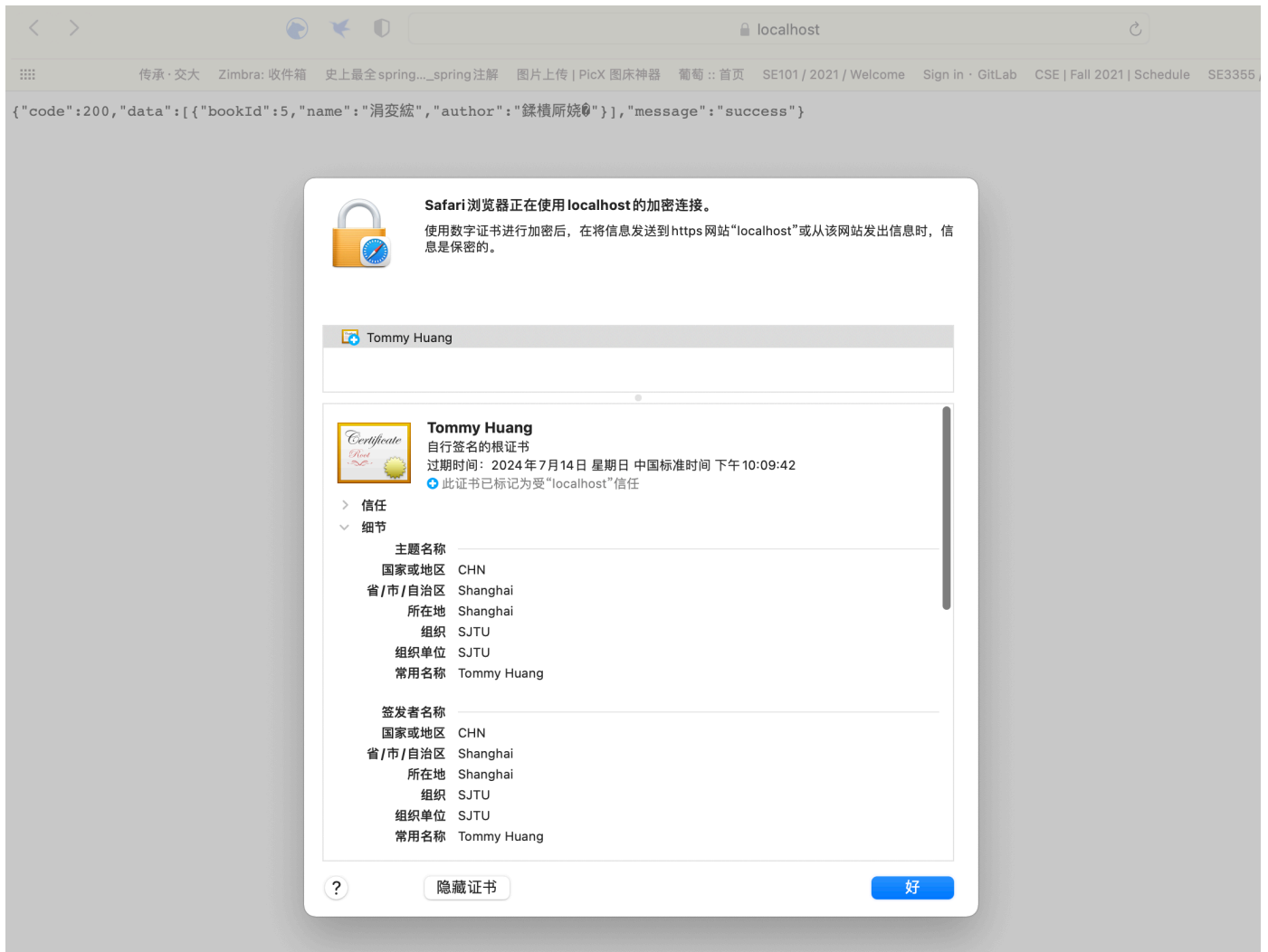


如果您想获得 Chrome 最高级别的安全保护，请[开启增强型保护](#)

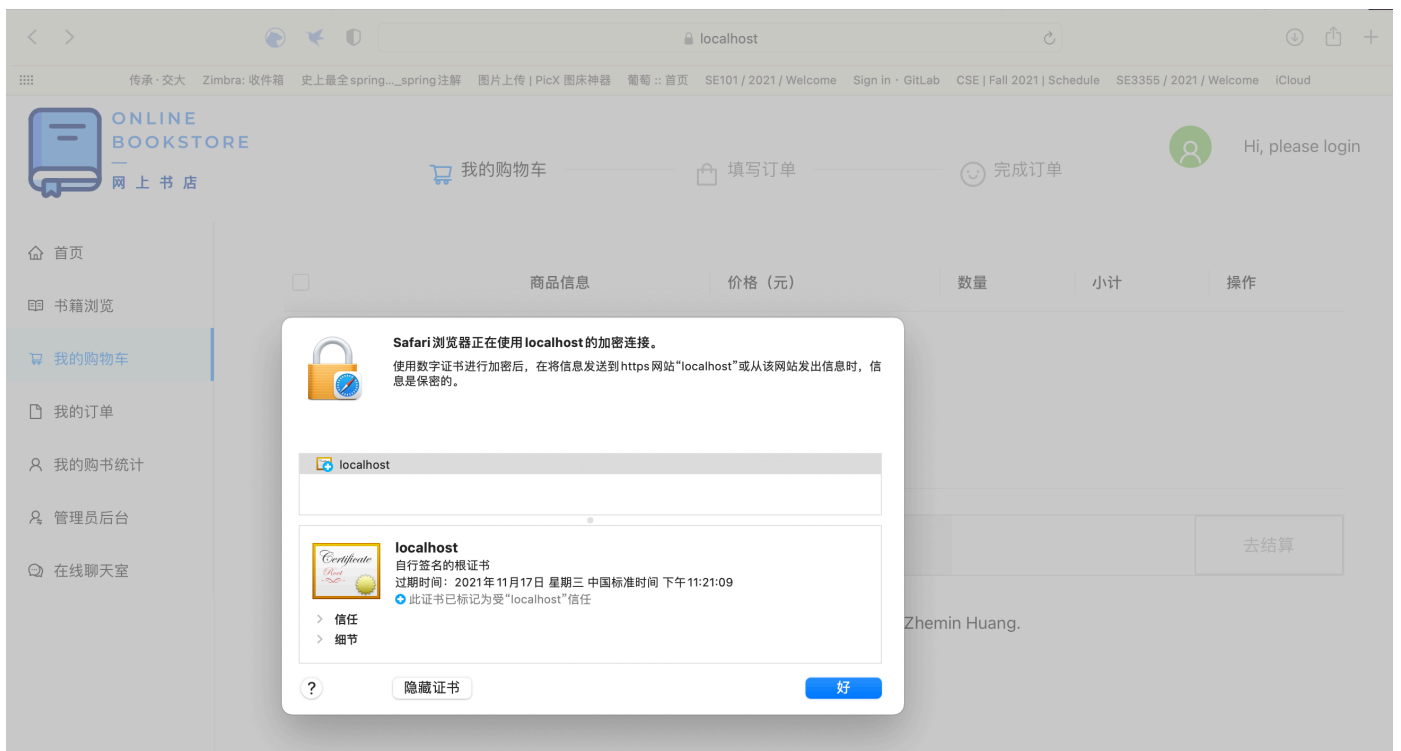
高级

重新加载

当使用Safari浏览器时，可以看到能够正常进行HTTPS访问，且浏览器的地址栏前会有一个锁的符号，代表使用了HTTPS协议，信息传输时进行了加密。HTTP的连接很简单，是无状态的；HTTPS协议是由SSL+HTTP协议构建的可进行加密传输、身份认证的网络协议，比HTTP协议安全。



根据教程，我生成了不同的证书，并对前端工程配置了HTTPS，结果如下图所示。



最后需要注意，由于Spring Cloud Gateway的依赖项中有**spring-webflux**，与spring-web冲突，因此不能引入spring-web依赖，也不能采用servlet。按照平时的跨域配置，则会出现一些问题。前端报错：**不允许有多个'Access-Control-Allow-Origin' CORS头**。因此我们需要写Header Filter，对跨域策略进行去重，前端才可以正常访问https服务。

## Reference:

Feign微服务相互调用：<https://blog.csdn.net/rookiediary/article/details/116785476>

React配置https：<https://segmentfault.com/a/1190000037611649>

Spring Cloud Gateway http转发https：[https://blog.csdn.net/qq\\_26932225/article/details/103019231](https://blog.csdn.net/qq_26932225/article/details/103019231)

解决Spring Cloud Gateway 跨域问题：<http://edisonxu.com/2020/10/14/spring-cloud-gateway-cors.html>