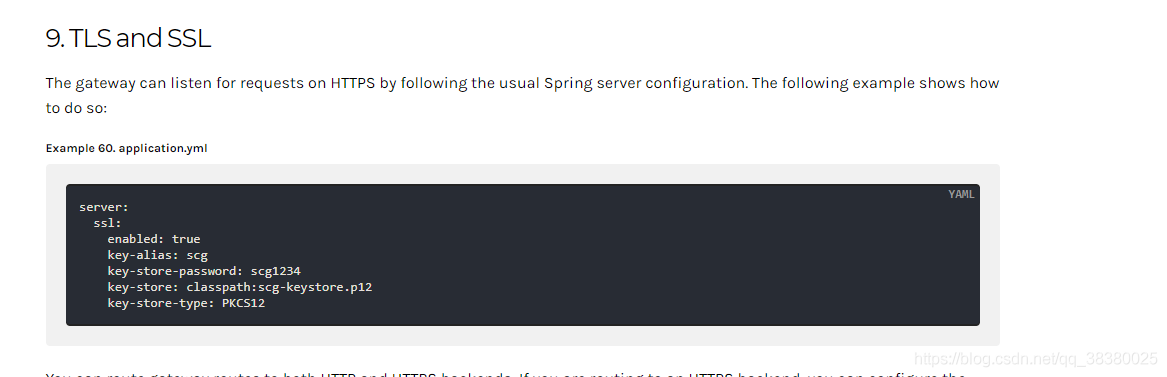
这是gateway 官网的配置截图



这个配置简单，但是我们需要同时支持http 和https

目前这个问题官方还没有正式解决

https://cloud.spring.io/spring-cloud-static/spring-cloud-gateway/2.2.2.RELEASE/reference/html/#tls-and-ssl

https://github.com/spring-cloud/spring-cloud-gateway/issues/1103

想到的办法是 在启动一个Netty 监听另外一个端口 如：8080，这个端口是不带 s 的也就是 http

当访问8080时 跳转到配置https 的如8443端口

查找

https://github.com/spring-projects/spring-boot/issues/12035

https://stackoverflow.com/questions/49045670/spring-webflux-redirect-http-to-https/53000573#53000573

具体实现：

package cn.com.test.gateway.config;

import org.springframework.beans.factory.annotation.Value;

import org.springframework.boot.web.embedded.netty.NettyReactiveWebServerFactory;

import org.springframework.context.annotation.Configuration;

import org.springframework.http.HttpStatus;

import reactor.core.publisher.Mono;

import javax.annotation.PostConstruct;

import java.net.URI;

import java.net.URISyntaxException;

@Configuration

public class HttpToHttpsRedirectConfig {

@Value("${server.http.port}")

private int httpPort;

@Value("${server.port}")

private int serverPort;

@PostConstruct

public void startRedirectServer() {

NettyReactiveWebServerFactory httpNettyReactiveWebServerFactory = new NettyReactiveWebServerFactory(httpPort);

httpNettyReactiveWebServerFactory.getWebServer((request, response) -> {

URI uri = request.getURI();

URI httpsUri;

try {

httpsUri = new URI("https", uri.getUserInfo(), uri.getHost(), serverPort, uri.getPath(), uri.getQuery(), uri.getFragment());

} catch (URISyntaxException e) {

return Mono.error(e);

}

response.setStatusCode(HttpStatus.MOVED\_PERMANENTLY);

response.getHeaders().setLocation(httpsUri);

return response.setComplete();

}).start();

}

}

yml 配置文件配置：

分为俩个端口号，server.port 是配置https 的端口， http.port 是监听的端口

server:

http:

port: 4321

port: 1234

ssl:

enabled: true

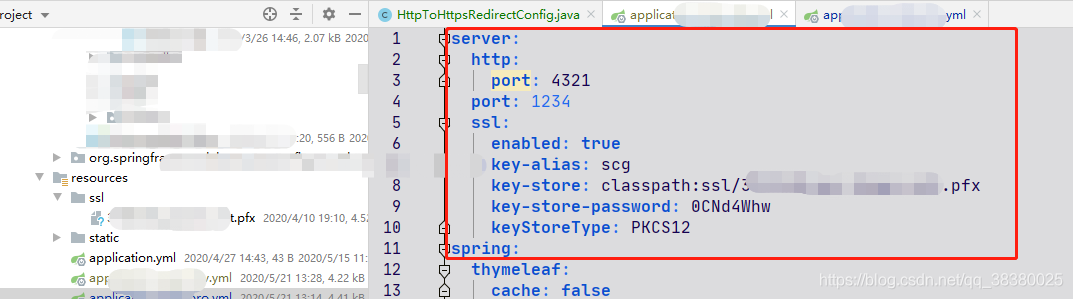
key-alias: scg

key-store: classpath:ssl/test.pfx

key-store-password: 6540CNd4Whw

keyStoreType: PKCS12

目录结构：



启动后访问，不带s 的4321端口会转发到1234端口。