

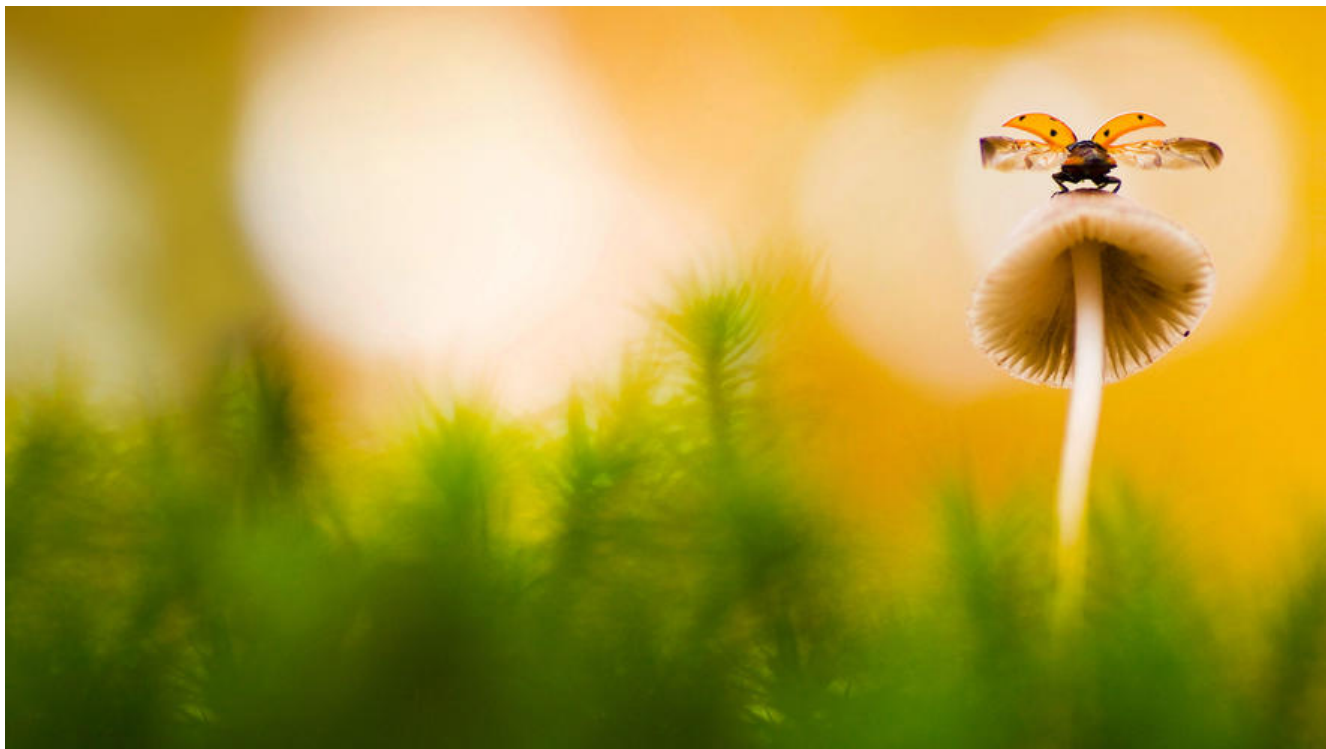
# Spring Cloud Gateway 2020.x 版本无法注入 Feign 服务和 RestTemplate 的问题

作者: [hjljy](#)

原文链接: <https://ld246.com/article/1630561196880>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



## 问题描述

Spring Cloud Gateway 2020.x版本无法注入Feign服务和RestTemplate，注入要么报错，要么IDE无法启动项目，一直转圈

## 问题相关代码

Feign服务代码如下：

```
@FeignClient(value = "oauth")
@Component
public interface TokenService {

    /**
     * 检查令牌
     *
     * @param token 令牌
     * @return {@link String}
     */
    @RequestMapping("/oauth/check_token")
    String checkToken(@RequestParam String token);
}
```

Gateway代码如下：

```
@Order(1)
@Component
public class Oauth2Filter implements GlobalFilter {

    @Autowired
```

```

GatewayConfig gatewayConfig; //这个是其他的bean 可以注入成功

@Autowired
TokenService tokenService; //这个是Feign服务的bean 无法注入，IDEA启动的时候也不报错，
是一只启动循环当中

@Autowired
RestTemplate restTemplate; //同上
@Override
public Mono<Void> filter(ServerWebExchange exchange, GatewayFilterChain chain) {

    ServerHttpRequest request = exchange.getRequest();
    String path = request.getPath().value();
//    1 判断请求路径是否需要放行
    if (whiteList(path)) {
        chain.filter(exchange);
    }
    // 2 不是白名单的需要鉴权
    String token = request.getHeaders().getFirst(HttpHeaders.AUTHORIZATION);
    if (StringUtils.hasText(token)) {
        // 存在token 校验token合法性
        String checkToken = tokenService.checkToken(token);
    } else {
        // 不存在 就报错返回啦
    }
    return chain.filter(exchange);
}

private boolean whiteList(String path) {
    AntPathMatcher antPathMatcher = new AntPathMatcher();
    List<String> ignoreUrl = gatewayConfig.getIgnoreUrl();
    if (ignoreUrl.size() > 0) {
        return ignoreUrl.stream().anyMatch(n -> antPathMatcher.match(n, path));
    }
    return false;
}
}

```

## 问题原因

网上有普遍说是filter是在bean创建前被创建的所以无法注入，但是我这里只是特定的bean无法注入，以我不认同这个说法。

强烈怀疑是Gateway 2020版的对Feign支持不友好导致的，然后就想到了，在2020版的Spring Cloud Gateway是移除了之前的robbion，并且在官方的issue里面找到了相关的问题：[Failed to invoke Feign and RestTemplate in Spring Cloud 2020's Gateway](#)

## 处理方式（并没有处理方式）

尽管找到了原因，但是官方并没有给出一个解决方式，很难受！！

发现ServerWebExchange可以获取到ApplicationContext，那么就用ApplicationContext 获取bean

试试!!! 发现可以获取到  
但是获取到之后又出现了新的问题,

```
@Order(1)
@Component
public class Oauth2Filter implements GlobalFilter {

    @Autowired
    GatewayConfig gatewayConfig;

    TokenService tokenService;

    @Override
    public Mono<Void> filter(ServerWebExchange exchange, GatewayFilterChain chain) {

        ServerHttpRequest request = exchange.getRequest();
        String path = request.getPath().value();
        // 1 判断请求路径是否需要放行
        if (whiteList(path)) {
            chain.filter(exchange);
        }
        // 2 不是白名单的需要鉴权
        String token = request.getHeaders().getFirst(HttpHeaders.AUTHORIZATION);
        try {
            if (null == tokenService) {
                ApplicationContext applicationContext = exchange.getApplicationContext();
                assert applicationContext != null;
                tokenService = applicationContext.getBean(TokenService.class);
            }
        } catch (BeansException e) {
            e.printStackTrace();
        }
        if (StringUtils.hasText(token)) {
            // 存在token 校验token合法性
            String checkToken = tokenService.checkToken(token);
        } else {
            // 不存在 就报错返回啦
        }
        return chain.filter(exchange);
    }

    private boolean whiteList(String path) {
        AntPathMatcher antPathMatcher = new AntPathMatcher();
        List<String> ignoreUrl = gatewayConfig.getIgnoreUrl();
        if (ignoreUrl.size() > 0) {
            return ignoreUrl.stream().anyMatch(n -> antPathMatcher.match(n, path));
        }
        return false;
    }
}
```

上述完成之后, 调用网关服务进行鉴权, 发现第一次可以成功, 后续同样的接口直接报错:

```
2021-06-01 16:30:28.770 WARN 26788 --- [ parallel-2] o.s.c.l.core.RoundRobinLoadBalance
: No servers available for service: oauth
2021-06-01 16:30:28.771 WARN 26788 --- [oundedElastic-8] .s.c.o.l.FeignBlockingLoadBalanc
rClient : Load balancer does not contain an instance for the service oauth
2021-06-01 16:30:28.774 ERROR 26788 --- [oundedElastic-8] a.w.r.e.AbstractErrorWebExcepti
nHandler : [78a0055a-1] 500 Server Error for HTTP GET "/api/user/getName"
```

```
feign.FeignException$ServiceUnavailable: [503] during [GET] to [http://oauth/oauth/check_to
en?token=eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJ7XCJpZFwiOlwiMTQwMjMyMTYzNDgxMzcxMT
0XClcXCl1c2VybWVwZVwiOlwiZ3JpZFwiLFwibGVzZ2VlSWRcljpcJlEyOTc1NDgyMzc1NjM5ODU
MlwiLFwidW5pdElkXCI6XCliXmJQ3NTUxNTAyOTk3NDIyMDNclxcIndlY2hhdFwiOnRydWUsXCJ
c2VyVHlwZVwiOjUsXCJhY2NvdW50Tm9uRXhwaXJlZFwiOnRydWUsXCJjcmVhZCJmVjZlZW50aWFsc05v
kV4cGlyZWVwZVwiOjUsXCJhY2NvdW50Tm9uRXhwaXJlZFwiOnRydWV9liwiZXhwIjoxNjE5MDkyNjc
LCJpYXQiOiJlY2MTkwNzgyNzV9.CgLRkP5dGfRrnWXqK2gbRDbUeXqq7hC9r7jI_Ge9GRI0Al1stX
Tvg5emPydha4i36fM5QXoGloJJYmUX1-wVA1] [TokenService#checkToken(String)]: [Load bal
ancer does not contain an instance for the service oauth]
    at feign.FeignException.serverResponseStatus(FeignException.java:237) ~[feign-core-10.10.1.jar:
a]
```

```
Suppressed: reactor.core.publisher.FluxOnAssembly$OnAssemblyException:
Error has been observed at the following site(s):
|_ checkpoint | org.springframework.cloud.gateway.filter.WeightCalculatorWebFilter [Defaul
WebFilterChain]
|_ checkpoint | HTTP GET "/api/user/getName" [ExceptionHandlerWebHandler]
```

哎，头大，放弃了，本来是打算在gateway统一调用oauth2鉴权服务进行鉴权的，还是让具体的服自己调用oauth2鉴权服务进行鉴权吧！！

## 2021-09-02 处理方式

抱着不死心的态度，发现在官方的issue里面已经给出了处理方式，当时没有仔细阅读，结果走了很弯路！！

最终处理方式: 使用webclient 替换feign

代码如下:

```
//第一步 注入webclident
@Bean
@LoadBalanced // 如果不添加, 无法通过服务名进行调用, 只能通过ip调用
public WebClient.Builder webBuilder(){
    return WebClient.builder();
}

//第二步 在gateway当中注入
@Autowired
WebClient.Builder webBuilder;

//第三步 具体调用方式 这里的 "lb" 也可以换成http
Mono<Object> toMono = webBuilder.baseUrl("lb://服务名/").build().get().uri(uriBuilder ->
    uriBuilder.path("/api/oauth/check_token").queryParam("参数名称", "参数值").build()
).header(HttpHeaders.AUTHORIZATION, token).exchangeToMono(clientResponse -> clie
tResponse.bodyToMono(Object.class));
// 不调用subscribe或者block是不会调用服务的
Object block = toMono.subscribe();
```

