



链滴

# SpringCloud Alibaba 微服务番外篇二 - Feign 传递 Access\_Token

作者: [jianzh5](#)

原文链接: <https://ld246.com/article/1585036937217>

来源网站: [链滴](#)

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

<p></p>

<h2 id="概述">概述</h2>

<p>还记得我们之前创建订单的流程吗? </p>

<p>通过网关调用 order-service 的 <code>create</code> 方法, 然后 order-service 再通过 feign 调用 account-service 的 <code>reduce</code> 和 product-service 的 <code>deduct</code> 方法进行业务逻辑处理。 <br>

</p>

<p>当我们项目整合好 oauth2 后再来调用之前的方法发现调用异常, 查看后端日志发现如下的错误信息: </p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">org.springframework.security.authentication.InsufficientAuthenticationException: Full authentication is required to access this resource
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">Request processing failed; nested exception is feign.FeignException$Unauthorized: status 401 reading AccountFign#reduce(String,BigDecimal)
```

```
</span></span></code></pre>
```

<p>错误信息告诉我们是接口未授权导致访问异常, 实际上我们已经将 access\_token 放在了 header 上, 如下: <br>

</p>

<p>最后通过对自定义异常类进行跟踪发现是 access\_token 经过 order-service 访问 account-service 时丢失了, 获取的 accessToken 为 null。 <br>

</p>

<p>本节主要是来解决使用 feign 调用时 access\_token 丢失的问题。 </p>

<h2 id="代码示例">代码示例</h2>

<p>Feign 支持请求拦截器, 在发送请求前, 可以对发送的模板进行操作, 例如设置请求头等属性, 定请求拦截器只需要实现 <code>RequestInterceptor</code> 接口, 该接口的方法有 RequestTemplate 类型的参数, 我们可以根据实际情况对请求的信息进行调整。 </p>

```
<pre><code class="language-java highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-kd">public</span> <span class="highlight-kd">interface</span> <span class="highlight-nc">RequestInterceptor</span> <span class="highlight-nc">{</span>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="highlight-nc">void</span></span> <span class="highlight-nc">apply</span></span> <span class="highlight-nc">RequestTemplate</span> <span class="highlight-nc">var1</span></span> <span class="highlight-nc">};</span>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nc">}</span></span>
```

```
</span></span></code></pre>
```

<p>我们要实现的功能很简单, 就是取出原来的请求头, 再将其传递给下游服务, 在 order-service 建立 <code>FeignRequestInterceptor</code> 并给其添加 <code>@Configuration</code> 注解, 代码如下: </p>

```
<pre><code class="language-java highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-cm">/**
```

```
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-cm"> * Description:
```

```
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-cm"> * 微服务之间feign调用请求头丢失的问题
```

```
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-cm"> * @author javadaily
```

```

</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-cm"> * @date 2020/02/24 9:28
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-cm"> */</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="high
ight-nd">@Configuration</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="high
ight-nd">@Slf4j</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="high
ight-kd">public</span> <span class="highlight-kd">class</span> <span class="highlight-n
">FeignRequestInterceptor</span> <span class="highlight-kd">implements</span> <span
lass="highlight-n">RequestInterceptor</span> <span class="highlight-o">{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-nd">@Override</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-kd">public</span> <span class="highlight-kt">void</span> <span class="highli
ght-nf">apply</span><span class="highlight-o">(</span><span class="highlight-n">RequestT
mplate</span> <span class="highlight-n">template</span> <span class="highlight-o">)</s
an> <span class="highlight-o">{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class=
highlight-n">HttpServletRequest</span> <span class="highlight-n">httpServletRequest</s
an> <span class="highlight-o">=</span> <span class="highlight-n">getHttpServletRequest
</span><span class="highlight-o">();</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class=
highlight-k">if</span> <span class="highlight-o">(</span><span class="highlight-n">http
ervletRequest</span> <span class="highlight-o">!</span> <span class="highlight-kc">nul
</span> <span class="highlight-o">){</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span clas
="highlight-n">Map</span> <span class="highlight-o">&lt;</span> <span class="highlight-
">String</span> <span class="highlight-o">,</span> <span class="highlight-n">String</sp
n> <span class="highlight-o">&gt;</span> <span class="highlight-n">headers</span> <sp
n class="highlight-o">=</span> <span class="highlight-n">getHeaders</span> <span class
"highlight-o">(</span><span class="highlight-n">httpServletRequest</span> <span class="
ighlight-o">);</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span clas
="highlight-c1"> // 传递所有请求头,防止部分丢失
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span> <span class="highlight-c1"> //此处也可以只传递认证的heade
</span></span></span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span> <span class="highlight-c1"> //requestTemplate.header("Au
thorization", request.getHeader("Authorization"));
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span> <span class="highlight-k">for</span> <span class="highli
ght-o">(</span><span class="highlight-n">Map</span> <span class="highlight-o">.</spa
> <span class="highlight-na">Entry</span> <span class="highlight-o">&lt;</span> <span cl
ss="highlight-n">String</span> <span class="highlight-o">,</span> <span class="highlight
n">String</span> <span class="highlight-o">&gt;</span> <span class="highlight-n">entry
/</span> <span class="highlight-o">:</span> <span class="highlight-n">headers</span> <sp
n class="highlight-o">.</span> <span class="highlight-na">entrySet</span> <span class="h
ghlight-o">())</span> <span class="highlight-o">{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span c
ass="highlight-n">template</span> <span class="highlight-o">.</span> <span class="highli
ht-na">header</span> <span class="highlight-o">(</span><span class="highlight-n">entr

```

```

</span> <span class="highlight-o">.</span> <span class="highlight-na">getKey</span> <s
an class="highlight-o">(),</span> <span class="highlight-n">entry</span> <span class="hi
hlight-o">.</span> <span class="highlight-na">getValue</span> <span class="highlight-o">
));</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span clas
="highlight-o">}</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span clas
="highlight-n">log</span> <span class="highlight-o">.</span> <span class="highlight-na"
debug</span> <span class="highlight-o">(</span> <span class="highlight-s">"FeignRequest
nterceptor:{}"</span> <span class="highlight-o">,</span> <span class="highlight-n">templ
te</span> <span class="highlight-o">.</span> <span class="highlight-na">toString</span>
span class="highlight-o">());</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class=
highlight-o">}</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="h
ghlight-o">}</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="h
ghlight-kd">private</span> <span class="highlight-n">HttpServletRequest</span> <span c
ass="highlight-nf">getHttpServletRequest</span> <span class="highlight-o">()</span> <sp
n class="highlight-o">{</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class=
highlight-k">try</span> <span class="highlight-o">{</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span clas
="highlight-k">return</span> <span class="highlight-o">((</span> <span class="highlight-
">ServletRequestAttributes</span> <span class="highlight-o">)</span> <span class="highli
ht-n">RequestContextHolder</span> <span class="highlight-o">.</span> <span class="highl
ght-na">getRequestAttributes</span> <span class="highlight-o">()).</span> <span class="h
ghlight-na">getRequest</span> <span class="highlight-o">());</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class=
highlight-o">}</span> <span class="highlight-k">catch</span> <span class="highlight-o">
</span> <span class="highlight-n">Exception</span> <span class="highlight-n">e</span>
span class="highlight-o">)</span> <span class="highlight-o">{</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span clas
="highlight-k">return</span> <span class="highlight-kc">null</span> <span class="highlig
t-o">;</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class=
highlight-o">}</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="h
ghlight-o">}</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="h
ghlight-cm">/**
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span cla
s="highlight-cm"> * 获取原请求头
</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span cla
s="highlight-cm"> */</span>
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="h
ghlight-kd">private</span> <span class="highlight-n">Map</span> <span class="highlight
o">&lt;</span> <span class="highlight-n">String</span> <span class="highlight-o">,</spa
> <span class="highlight-n">String</span> <span class="highlight-o">&gt;</span> <span c
ass="highlight-nf">getHeaders</span> <span class="highlight-o">(</span> <span class="hi
hlight-n">HttpServletRequest</span> <span class="highlight-n">request</span> <span clas

```

```

="highlight-o">)/span> <span class="highlight-o">{/span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class=
highlight-n">Map</span><span class="highlight-o">&lt;</span><span class="highlight-n"
String</span><span class="highlight-o">,</span> <span class="highlight-n">String</span>
<span class="highlight-o">&gt;</span> <span class="highlight-n">map</span> <span clas
="highlight-o">=</span> <span class="highlight-k">new</span> <span class="highlight-n
">LinkedHashMap</span><span class="highlight-o">&lt;&gt;();</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class=
highlight-n">Enumeration</span><span class="highlight-o">&lt;</span><span class="high
ight-n">String</span><span class="highlight-o">&gt;</span> <span class="highlight-n">
numeration</span> <span class="highlight-o">=</span> <span class="highlight-n">reques
</span><span class="highlight-o">.</span><span class="highlight-na">getHeaderNames<
span><span class="highlight-o">);</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class=
highlight-k">if</span><span class="highlight-o">(</span><span class="highlight-n">enu
eration</span><span class="highlight-o">!</span><span class="highlight-kc">null</spa
"><span class="highlight-o">){</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span clas
="highlight-k">while</span> <span class="highlight-o">(</span><span class="highlight-n
">enumeration</span><span class="highlight-o">.</span><span class="highlight-na">has
oreElements</span><span class="highlight-o">())</span> <span class="highlight-o">{/sp
n>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span c
ass="highlight-n">String</span> <span class="highlight-n">key</span> <span class="high
ght-o">=</span> <span class="highlight-n">enumeration</span><span class="highlight-o
">.</span><span class="highlight-na">nextElement</span><span class="highlight-o">);</
span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span c
ass="highlight-n">String</span> <span class="highlight-n">value</span> <span class="hi
hlight-o">=</span> <span class="highlight-n">request</span><span class="highlight-o">
</span><span class="highlight-na">getHeader</span><span class="highlight-o">(</span>
<span class="highlight-n">key</span><span class="highlight-o">);</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span c
ass="highlight-n">map</span><span class="highlight-o">.</span><span class="highlight-
a">put</span><span class="highlight-o">(</span><span class="highlight-n">key</span>
span class="highlight-o">,</span> <span class="highlight-n">value</span><span class="h
ghlight-o">);</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span clas
="highlight-o">}</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class=
highlight-o">}</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class=
highlight-k">return</span> <span class="highlight-n">map</span><span class="highlight
o">;</span>
</span></span><span class="highlight-line"><span class="highlight-cl">    <span class="h
ghlight-o">}</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="high
ight-o">}</span>
</span></span></code></pre>

```

<p>完成配置后重启服务器，再次调用 create 方法，正常返回响应结果。 <br>
 </p>