

retrofit 服务自定义签名认证

2017/11/29 23:23 - Posted By 起衣 - Tags: java,retrofit - Category: Code

0 条评论

因为用在一个不是特别主流的方案里，所以还是记录下。服务所有的请求通过 form data 格式传输，然后每一次的请求都需要使用服务端给客户端分配的密钥对请求参数进行签名，

直接基于 OkHttpClient 扩展 Interceptor 实现，如下：

```
01. public class SignInterceptor implements Interceptor {
02.     private final Mac mac;
03.     private final String clientId;
04.
05.     public SignInterceptor(String clientId, String clientKey) throws NoSuchAlgorithmException, InvalidKeyException {
06.         this.clientId = clientId;
07.         byte[] key = clientKey.getBytes(StandardCharsets.UTF_8);
08.         SecretKeySpec keySpec = new SecretKeySpec(key, "HmacSHA256");
09.         this.mac = Mac.getInstance("HmacSHA256");
10.         this.mac.init(keySpec);
11.     }
12.
13.     @Override
14.     public Response intercept(Chain chain) throws IOException {
15.         Request request = chain.request();
16.         RequestBody requestBody = request.body();
17.         if (requestBody instanceof FormBody) {
18.             FormBody originFormBody = (FormBody) requestBody;
19.             FormBody.Builder formBody = new FormBody.Builder();
20.             List fields = new ArrayList<>(originFormBody.size());
21.             int size = originFormBody.size();
22.             for (int i = 0; i < size; i++) {
```

```

23.         String encodedName = originFormBody.encodedName(i);
24.         String encodedValue = originFormBody.encodedValue(i);
25.         fields.add(encodedName + "=" + encodedValue);
26.         formBody.addEncoded(encodedName, encodedValue);
27.     }
28.     Collections.sort(fields);
29.     String baseString = String.join("&", fields);
30.     Base64.Encoder encoder = Base64.getEncoder();
31.     String signature = encoder.encodeToString(
32.         mac.doFinal(baseString.getBytes(StandardCharsets.UTF_8)));
33.     formBody.add("clientId", this.clientId);
34.     formBody.add("signature", signature);
35.     request = request.newBuilder()
36.         .method(request.method(),
37.             formBody.build())
38.         .build();
39.     }
40.     return chain.proceed(request);
41. }
42. }

```

然后在构建服务的时候添加进去:

```

01. Retrofit retrofit = new Retrofit.Builder()
02.     .baseUrl("http://127.0.0.1:8080/")
03.     .addConverterFactory(new FormBodyConverterFactory())
04.     .addConverterFactory(GsonConverterFactory.create())
05.     .client(new OkHttpClient.Builder()
06.         .addInterceptor(new SignInterceptor("dev", "key"))
07.         .build())
08.     .build();

```

相关日志 Relate Posts

- [retrofit 发起form data请求时使用自定义对象参数](#)
- [SpringBoot应用配置项加密](#)
- [SpringBoot 集成 Flyway 自动创建数据库表](#)
- [Java 里 long 可以计数多久](#)
- [并发编程 Promise, Future 和 Callback](#)

收藏与分享 : [Twitter](#) | [Facebook](#) | [微博](#) | [人人](#) | [Google+](#) | [PDF](#)

发表留言 (Ctrl+Enter提交)

昵称 (必填)

EMail (不公开, 必填)

网站 (选填)



发表