
接口简易描述

1、 接口类型

URL	说明
/buss/alive	心跳
/buss/op	计算

2、 基本流程

3、 接入规范与安全

3.1、接入规则

本文档中的所有接口均为 http 类型接口，支持 get 和 post 两种方式，参数均采用 UTF-8 编码。

所有接口的响应消息均为 json 格式。

➤ 基本请求参数

所有的接口除了业务参数之外还必须附带基本请求参数

字段名	名称	说明
msgId	消息 ID	32 位 uuid，每次请求必须保证唯一
timestamp	时间戳	yyyyMMddHHmmssSSS
sign	签名	

➤ 基本响应参数

returnCode、memo、errorCode 为基本 json 参数，content 为嵌套的业务字段。

```
{"returnCode":"返回结果",
```

```
"memo":"提示说明",  
"errorCode":"错误代码",  
"content":{"buss1":"value1","buss2":"value2"}}&sign
```

3.2、安全规范

- 双方的交互信息必须进行动态签名处理，即每次业务签名信息都不一样，防止信息重发导致的安全问题。
- 签名采用 RSA 非对称密钥体系。
- 双方在处理业务中一定要核对签名信息，只有在签名信息通过验证的情况下才能进行业务处理。

3.2.1、签名机制

- 待签名参数的确认

在请求和返回参数列表中，除去 sign 参数外，其他参数均是需要签名的参数。

- 规则

对数组里的每一个值从 a 到 z、0 到 9 的顺序排序,若遇到相同首字母，则看第二个字母,以此类推。 排序完成之后，再把所有数组值以“&”字符连接起来。

- 示例

对于如下的参数数组：

```
string[] parameters={
    "zname=zvalue",
    "bname=bvalue",
    "aname=avalue"
};
```

待签名字符串为：

```
aname=avalue&bname=bvalue&zname=zvalue
```

- 注意事项
 - 1) 签名时将字符转化成字节流时指定的字符集为 UTF-8;
 - 2) 根据 HTTP 协议要求,传递参数的值中如果存在特殊字符(如:&、@等),那么该值需要做 URL Encoding,这样请求接收方才能接收到正确的参数值。这种情况下,待签名数据应该是原生值而不是 encoding 之后的值。例如:调用某接口需要对请求参数 email 进行数字签名,那么待签名数据应该是 email=test@msn.com,而不是 email=test%40msn.com。
- 客户端签名 rsa 私钥以及验签公钥

见附件

4、 接口业务参数

4.1、心跳

请求业务参数		
字段名	名称	说明

响应业务参数		
字段名	名称	说明
systemTime	服务器时间	yyyyMMddHHmmss

4.2、计算

请求业务参数		
字段名	名称	说明
opType	操作类型	计算类型 1 表示加 2 表示减 3 表示乘 4 表示除
opl	操作数 1	
opr	操作数 2	
响应业务参数		
字段名	名称	说明
opRt	结果	计算结果

5、 示例报文

请求：

<http://localhost:8080/buss/op?msgId=xxxxxxxxxxxxxxxxxxxxxxxxxxxx×tamp=20200606121212123&sign=D5f40Wrgwomakz2jEQGZZoryyhBWx5XPD%2bedf8kEfoyvrdXlna7NZQhUyW%2fMLYX7s%2b6JmwxfU3rBEeDu2agykA%3d%3d&opType=1&opl=1&opr=2>

注意 sign 中的值是经过 urlencoding 之后的编码

响应：

```
{
  "errorCode": "",
  "memo": "成功",
  "content": {
    "opRt": 3,
    "returnCode": "00"
  }
}&f68LXpvlF+TCDzN+qKiiKtDBzvox0fGC37ZaRlyc8wUndBim+pY7piu2xnXGQaO3jhg6dgysQHCTj/YdlN98kA==
```