PP 单机游戏 SDK
开发参考说明书
_服务端接口

修订记录

归档日期	版本	说明	作者
2015-06-23	1.0.0	PP 单机游戏 SDK 开发参考说明书_服务端接口	PP 助手安卓版

目录

1. 服务器端接口说明 3
1.1. 概述 3
1.2. 数据协议 3
1.2.1 数据格式 3
1.2.2 字符编码 3
1.2.3 签名规则 3
1.2.4 签名示例 4
1.3. 充值结果通知接口说明 6
1.3.1 请求地址 6
1.3.2 调用方式 6
1.3.3 接口描述 6
1.3.4 请求方 6
1.3.5 响应方 6
1.3.6 请求内容（json 格式） 6
1.3.7 返回内容（文本） 9
1.3.8 通知机制 9

1. 服务器端接口说明

1.1. 概述

本部分主要提供 PP 助手安卓版“单机游戏支付服务器”和 PP 游戏合作商“游戏服务器”的交互的接口规范。

1.2. 数据协议

1.2.1 数据格式

请求消息和响应消息的内容都使用 json 格式表示数据，具体请参考下文的例子。

1.2.2 字符编码

请求与响应内容（json 格式数据）须采用 utf-8 字符编码。

1.2.3 签名规则

使用 md5 哈希对请求内容进行签名，算法如下：

```
md5(签名内容+apiKey)
```

说明：

MD5 使用 RFC 1321 标准，编码后需要转换回全小写(本文提到的所有 md5 结果，都需要转换成全小写)。

表达式中的“+”号表示字符串接，并不存在。

签名内容是指请求数据（data 字段）中，各字段名及其字段值的拼接，字段名与字段值之间使用等号（=）连接。拼接时需对字段名排序，排序方式是按字段名进行升序排列。

注意：签名内容不应包含“&”符号，拼接签名内容时需把“&”符号剔除。字符串中若有换行情况，也应把换行符（回车或换行）剔除。

计算 MD5 签名时，取签名内容的字节时，应以 UTF-8 编码取字符串的字节值。

apiKey 由 PP 分配，PP 负责游戏接入的人员在游戏接入前会把需要由 PP 分配的数据包括 apiKey 通知给游戏合作商。

1.2.4 签名示例

假设请求内容请求数据（data 字段）为 ：

```
"data":{
    "personid":value1,
    "code":"value2",
"name":"value3"
}
}
```

排序拼接后得出要签名的内容串为：

```
code=value2name=value3personid=value1
```

假定 apiKey=202cb962234w4ers2aa，

要进行 MD5 哈希的字符串为：

```
code=value2name=value3personid=value1202cb962234w4ers2aa
```

执行 MD5 哈希，下面为在 linux shell 中执行命令取得 MD5 值：

```
echo -n
code=value2name=value3personid=value1202cb962234w4ers2aa|md5sum
```

得出签名结果是：

```
8f468d03c2c42fbe294bdd49f038c031
```

最后发送的请求内容为：

```
{
    "id":1330395827,
    "service":"aa.aa.aa",
    "data":{
        "personid":value1,
        "code":"value2",
        "name":"value3"
    },
    "sign":"8f468d03c2c42fbe294bdd49f038c031"
}
```

1.3. 充值结果通知接口说明

1.3.1 请求地址

即充值结果通知地址，由游戏合作商提供。游戏接入时，由游戏合作商提供给 PP 游戏运营人员，录入到 PP 的游戏平台中。

1.3.2 调用方式

HTTP POST

1.3.3 接口描述

用户在游戏中提交充值请求后，PP 游戏平台会异步执行充值，在充值操作完成后，PP 游戏平台通过该接口将充值结果发送给“游戏服务器”。

此处定义本接口的规范，游戏合作商需根据此规范在“游戏服务器”实现本接口。

此接口用于接收通过游戏 SDK 充值的结果通知。

1.3.4 请求方

单机游戏支付服务器

1.3.5 响应方

游戏服务器

1.3.6 请求内容（json 格式）

字段名称	字段说明	类型	必填	备注

ver	版本号	string	Y	接口版本号，1.0
data	支付结果数据	json	Y	
sign	签名参数	string	Y	MD5(签名内容+apiKey); 签名内容为 data 所有子字段按字段名升序拼接（剔除&符号及回车和换行符）
支付结果数据(对应 data,采用 json 格式)				
tradeId	交易流水号	string	Y	此订单号由 PP 游戏 SDK 生成，游戏客户端在进行充值时从 SDK 获得。
tradeTime	交易时间	String	Y	订单交易时间，格式：yyyyMMddHHmmss，如 20150527130000
orderId	CP 充值订单号	String	N	由游戏传递的订单号
gameId	游戏编号	string	Y	由 PP 分配
amount	支付金额	string	Y	单位：元。
attachInfo	游戏合作商自定义参数	string	N	游戏客户端在充值时传入，SDK 服务器不做任何处理，在进行充值结果回调时发送给游戏服务器，该字段建议使用字母或数字形式传入，建议不要传入特殊字符。
orderStatus	订单状态	string	Y	S-成功支付 F-支付失败
failedDesc	订单失败原因详细描述	string	Y	如果是成功支付，则为空串。

请求示例如下：

HTTP 请求的 body 内容：


```
{
    "ver": "1.0",
    "data":{
        "tradeId":"abcf1330",
        "tradeTime":"20150527130000",
        "orderId":"1234567",
        "gameId":123,
        "amount":"100.00",
        "attachInfo":"custominfo=xxxxx#user=xxxx",
        "orderStatus":"S",
         "failedDesc":""
    },
    "sign":"c041a6c1d70716ebf1396179e9e6b42f"
}
```


假定 apiKey=202cb962234w4ers2aaa

sign 的签名规则：

MD5(amount=...+attachInfo=...+failedDesc=...+gameId=...

+orderId=...+orderStatus=...+tradeId=...tradeTime=...+apiKey)（去掉+；替换...为实际值）

注意：若参数是非必填，如 orderId,attachInfo 等参数，仅当回调时具备该参数时，才需加入签名，如无，则不需要加入签名，下同

签名原文：

amount=100.00attachInfo=custominfo=xxxxx#user=xxxxfailedDesc=gameId=123order

Id=abcf1330orderStatus=StradeId=abcf1330tradeTime=20150527130000202cb962234

w4ers2aaa

1.3.7 返回内容（文本）

响应内容	响应内容描述
SUCCESS	成功，表示游戏服务器成功接收了该次充值结果通知，对于充值结果为失败的，只要能成功接收，也应返回 SUCCESS。
FAILURE	失败，表示游戏服务器无法接收或识别该次充值结果通知，如：签名检验不正确、游戏服务器接收失败

结果示例如下：

```
SUCCESS
```

1.3.8 通知机制

充值操作完成后，不论是否充值成功，“单机游戏支付服务器”都会将充值结果通过“充值结果回调接口”发送到“游戏服务器”。“游戏服务器”收到“单机游戏支付服务器”的充值通知后，根据处理结果返回字符串 SUCCESS 或 FAILURE。如果返回 SUCCESS，则“单机游戏支付服务器”结束通知任务；如果返回 FAILURE 或由于网络延时导致“单机游戏支付服务器”没有收到任何返回，单机游戏支付服务器将会在周期内进行重复通知。

“游戏服务器”在接收“单机游戏支付服务器”的充值结果通知时，不管订单是否成功，只要业务逻辑正常，都应该返回 SUCCESS，表示不需要“单机游戏支付服务器”再次发起通知。当业务逻辑异常（如：收到的单机游戏支付服务器的充值结果通知内容的签名不正确、充值结果与提交的充值请求不符等），认为需要再次通知，才返回 FAILURE。

【**注意**】

由于存在多次发送通知的情况，因此“游戏服务器”必须能够正确处理重复的通知。当接收到通知时，需要检查系统内对应业务数据的状态，以判断该通知是否已经处理过。在对业务数据进行状态检查或处理之前，需要采取数据加锁或时间戳判断等方式进行并发控制。

由于支付网关的通知机制原因，偶尔会发生通知支付失败后又通知支付成功的现象。基于这个情况，“游戏服务器”在处理充值结果通知时，对同一个订单，如果先接收到支付失败，再接收到支付成功的通知，应以成功的支付结果为准，替换原接收到的失败的支付结果。一旦通知支付成功，不会再发生通知支付失败的情形。