

乐视网游 SDK 服务端接入指南 (Android_TV)

Ver 2.0.1

2016-8-15

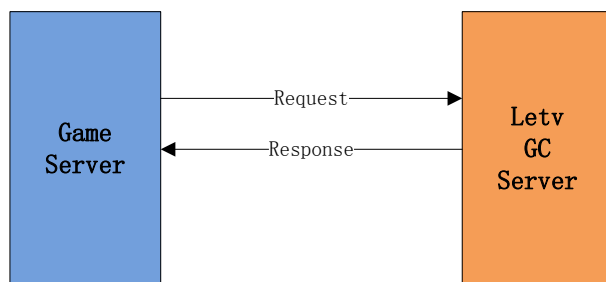
版本	更新时间	更新内容	操作人
1.0.0	2016/4/8	初始版本。	
1.0.1_v1	2016/5/9	服务端无变动	
1.0.1_v3	2016/5/24	服务端无变动	
2.0.1	2016/8/15	服务端无变动	

1. 登录校验接口（选择接入）

说明：

因为乐视服务端逻辑问题，应用需要在线上后，参数才能进行登录服务端校验，所以如果想做服务端校验测试，请使用demo中的参数。

交互过程：HTTP协议



参数介绍：

Request

请求方式：GET

Header参数

Header参数名	Header值
Accept	application/json
Content-Type	application/json

请求参数：

参数名称	参数描述	参数值举例
uid	游戏中心用户ID，即userId	1095(位数不定)
token	用户加密后的token，即access_token	
appkey	游戏唯一识别码，即AppId	a7a0983c71050a97a53c76d4ef217b701
sign	签名 算法：Hmac-SHA1 加密实体：uid+token+appkey 密钥：secretKey	算法见附件

Eg:

http://gc.letvstore.com/api/auth/login4Server?uid=247&token=10301a49f5JGgrM2fJDfKLicRCu8NT891NVVZ7XkTjm26F8N0eo46KqDA&appkey=d28b739c539330bf836c232706372d013b&sign=24710301a49f5JGgrM2fJDfKLicRCu8NT891N33VVZ7XkTjm26F8N0eo46KqDAd28b739c5390bf836c232706372d013b

以下为Demo,在游戏没上线之前可用DemoApp信息进行接口联调

DemoAPP

appkey: 61eb9bd756f84075ac5a3fff8b6fa008

secretkey: 17457b2af9564224179e5a6983792d1f

Response

返回类型JSON

参数名称	参数描述	数值类型	参数值举例
errCode	错误编码	String	如下表
errorMsgForClient	给请求者返回的错误信息	String	如下表
errMsg	错误信息	String	暂时不用
succMsg	成功信息	String	暂时不用
success	是否成功	Boolean	True/false
token	用户token	String	
uid	用户ID	String	
username	用户名称	String	

errCode	描述	ErrMsg
9999	认证成功	恭喜您，认证成功！
10000	系统错误	对不起，系统错误！
10001	Uid不能为空	对不起，UID不能为空！
10002	Token不能为空	对不起，Token不能为空！
10003	Appkey不能为空	对不起，APPKEY不能为空！
10004	Sign不能为空	对不起，SIGN不能为空！
10005	Appkey错误，无法查找到APP	对不起，APPKEY错误或者不存在！
10006	Sign出错，无法登录	对不起，SIGN错误或者不存在！
10007	用户认证信息失败	对不起，用户认证信息无效！

Eg:

```
{
  "status": 1,
  "code": null,
```

```
"msg": "success",
"serverTime": 1421983223364,
"entity": {
    "errCode": "9999",
    "errMsg": "认证成功",
    "succMsg": "",
    "errorMsgForClient": "恭喜您，认证成功！",
    "success": false,
    "token":
"7842900b6bbf9edc26638303c532e181d28def6059aa9a009014ce51b21911f80765ada5cd36c2cb6afc59c
07b3277e45684b6a0d0f4490cd5ec9746724bdb97dc4267a91d902a3d69cc4e68607b4b2563caa503bb0b
8c95cad90e879221f2afc81e6899a199b0fb2d55b3c83051f61aee828796a58159f02b60b72bdf179e",
    "uid": "2437",
    "username": "letv_50c84d34eb2875822"
}
}
```

加密算法java实现：



HmacSha1.java

2. 发货通知接口（必须接入）

接口地址：在 SDK 端发起支付时将 callback_url 在 AndroidManifest 中配置，具体请参考 demo 工程；

接口说明：CP 配置的用于接收支付成功信息和发货通知的回调接口

http 请求方式：GET

注：商户需要提供一个http协议的接口，包含在支付请求参数里，即callback_url。乐视服务器在支付成功后，会以GET方式调用callback_url，以key，value数据格式传输支付结果(示例：a=1&b=2)，**具体收到的参数和结果以实际收到的为准。**

CP服务器在成功处理后必须返回“SUCCESS”字符串，并且应该在一分钟内处理完毕。

若CP超过一分钟未处理，或者返回了除“SUCCESS”以外的字段，乐视服务器会认为第三方服务器处理失败，从而自动重试。

当CP服务器处理回调失败时，乐视服务器会进行重试，重试规则为间隔为5s 10s 20s 40s... 最大重试20次。

在回调CP服务器时，乐视服务器会使用URL的查询参数（Query Parameters）回填下面的数据。

参数名称	参数描述	是否必需	说明		
appKey	应用所属 AppId	是	乐视平台分配的应用参数		
currencyCode	货币种类	是	参 Currency.getInstance(Localelocale).getCurrencyCode() 默认 “CNY” ，		
params	CP 自定义参数	否	此值为客户端开发者传入, 在回调开发者服务器时, encode 之后返回给开发者服务器		
price	实际付费金额	是	订单总价格		
products	产品信息	是	购买产品 json 数组字符串方式传递,内容包括:		
			参数名称	参数描述	是否必须
			externalProductId	商品订单 id	否
			quantity	购买商品数量	是
			sku	支付系统产品标识	是
			total	商品单价，为 0	是
pxNumber	支付流水号	是	支付请求在乐视支付系统的唯一标识.		
userName	用户名	是	开发者请求时传入的 UserId 参数(ssoUid).		
sign	签名	是	做签名对比，具体可参考下方签名规则		

响应参数说明：

返回结果	含义
SUCCESS	处理成功，支付平台系统收到此结果后不再进行后续通知
FAIL 或其它字符	处理不成功，支付平台会不断重发通知，一直达到上限

支付回调回告参数，请严格按照上述要求返回，不要更改！！

附录

1. 商户签名规则:

合作密钥：

第三方和支付系统接入协议时，由乐视支付系统提供 secret_key，用于在第三方和支付系统计算签名信息。合作密钥可在必要的时候由双方重新商定。

报文签名：

在乐视回调开发者服务器时，会对参数进行摘要算法，用于保护参数在传递过程中不会被篡改，只有校验签名校验通过的情况下，才可以认为信息是可靠的。

签名算法：

目前使用 MD5。

将回调 URL 中除 sign 以外的查询参数解析为 key=value 的数组，注意这步需要进行 URLDecode 操作，然后将数组按照升序排列，最后将数组拼接为一个字符串(array[0]array[1] ... array[length - 1])。

例如下面的回调 url：

```
http://www.stv.com/?sign=5f5a8044dc03c02a4658fb3ce0c4b0c0&price=0.01&pxNumber=f052123c14d141c29c1eb3486957b5d9&currencyCode=CNY&userName=122648700&params=CP&products=%5B%7B%22externalProductId%22%3A%22123456789%22%2C%22quantity%22%3A1%2C%22sku%22%3A%22e28e0292-7116-43a9-ba66-d48fc8f0ef66%22%2C%22total%22%3A%220%22%7D%5D&appKey=221018gc
```

解析并移除 sign 参数生成的数组为：

```
[price=0.01,pxNumber=f052123c14d141c29c1eb3486957b5d9,currencyCode=CNY,userName=122648700,params=CP,products=[{"externalProductId":"123456789","quantity":1,"sku":"e28e0292-7116-43a9-ba66-d48fc8f0ef66","total":"0"}],appKey=221018gc]
```

将数组按照字典升序排列：

```
[appKey=221018gc,currencyCode=CNY,params=CP,price=0.01,products=[{"externalProductId":"123456789","quantity":1,"sku":"e28e0292-7116-43a9-ba66-d48fc8f0ef66","total":"0"}],pxNumber=f052123c14d141c29c1eb3486957b5d9,userName=122648700]
```

拼接成的字符串为：

```
appKey=221018gccurrencyCode=CNYparams=CPprice=0.01products=[{"externalProductId":"123456789","quantity":1,"sku":"e28e0292-7116-43a9-ba66-d48fc8f0ef66","total":"0"}]pxNumber=f052123c14d141c29c1eb3486957b5d9userName=122648700
```

将回调 URL 的查询参数之前的部分(也就是 “?” 之前的部分)和上一步得到字符串以及约定的密钥拼接在一起

(urlPath + paramsString +secretKey) , 将拼接后的字符串做 URLEncode 操作 , 使用 UTF-8 编码.

例如 : 上面的回调 URL 中查询参数之前的部分为 http://www.stv.com/. 连接上一步生成的字符串,

以及约定的密钥(4c9bdf56ffac4197aeee0b40es07913d)得到的结果为

```
http://www.stv.com/appKey=221018gccurrencyCode=CNYPparams=CPprice=0.01products=[{"externalPr  
oductId":"123456789","quantity":1,"sku":"e28e0292-7116-43a9-ba66-d48fc8f0ef66","total":"0"}]pxNumb  
er=f052123c14d141c29c1eb3486957b5d9userName=12264870054d65f31d388450988e8827cb1e2218g
```

进行 URLEncode 后的结果为 :

```
http%3A%2F%2Fwww.stv.com%2FappKey%3D221018gccurrencyCode%3DCNYparams%3DCPprice%3D0  
.01products%3D%5B%7B%22externalProductId%22%3A%22123456789%22%2C%22quantity%22%3A1  
%2C%22sku%22%3A%22e28e0292-7116-43a9-ba66-d48fc8f0ef66%22%2C%22total%22%3A%220%22%  
7D%5DpxNumber%3Df052123c14d141c29c1eb3486957b5d9userName%3D12264870054d65f31d38845  
0988e8827cb1e2218g
```

将上一部生成的字符串进行 MD5 计算 , 将计算的结果与 sign 参数的值进行对比 , 如果一致则为合法回调 , 否则为非法回调.

例如 : 将上一步生成的字符串计算 MD5 结果为 5f5a8044dc03c02a4658fb3ce0c4b0c0, 与 sign 参数的值一致 , 为合法请求.

注意事项 :

- 被签名字符串中的关键信息需要按照 key 值做升序排列。
- 空值 (空字符串或 null 值) 不参与签名运算 , 报文中也不传递。
- 签名时 , 对于没有子域的字段 value 保持原始值 , 不做 URL 编码 ;
- callback_url 中的参数在 URL 编码后返回。

创建订单时支付 SDK 的 callback_url 会做如下处理:

```
callback_url=URLEncoder.encode("http://yoururl/function?params=your_param_string","utf-8")
```