# 穿山甲小程序SDK文档

注释：接入SDK前H5游戏需要先向SDK（穿山甲）申请SDK接入需要的参数game\_id、app\_key。游戏研发方需要提供小程序必要的游戏地址。（小程序打开游戏，必须配置白名单，游戏地址和游戏内iframe嵌入的地址 并且必须使用HTTPS协议）。

## 一、接入必读

游戏必须引入穿山甲JS SDK文件：

<script type="text/javascript" src="https://h5.ichuanshanjia.com/public/weixin/chuanshanjia.js"></script>

## 二、游戏登陆

从小程序跳转到游戏时，通过小程序的用户直接登录游戏

**链接携带参数：**

| **参数名** | **必选** | **类型** | **说明** | **签名** |
| --- | --- | --- | --- | --- |
| gameId | 是 | int | 游戏ID | 是 |
| uid | 是 | int | 用户ID | 是 |
| sign | 是 | string | 签名参数 | 否 |
| timestamp | 是 | string | 时间戳从 | 是 |

假设游戏地址为：[https://xxx.com/game（必须为https）](https://xxx.com/game（必须为https）" \t "_blank)携带参数后最终地址为：  
<https://xxx.com/game?gameId=123&uid=10023&timestamp=1490840662&sign=4753DCE3AE736E7F894EBCC6CD3CFF7A>

## 三、游戏支付

游戏通过调用JS方法实现

初始化：Chuanshanjia.init(game\_id);

支付：Chuanshanjia.pay(obj);

支付之前必须先初始化init(),game\_id为穿山甲分配的游戏id

**obj参数：**

| **参数名** | **必选** | **类型** | **说明** | **签名** |
| --- | --- | --- | --- | --- |
| uid | 是 | int | 用户ID | 是 |
| gameOrderId | 是 | sting | 游戏订单号 | 是 |
| payFee | 是 | int | 充值金额，单位为分 | 是 |
| notifyUrl | 是 | string | 异步通知地址 | 是 |
| syncUrl | 否 | string | 同步跳转地址 | 否 |
| roleName | 是 | string | 角色名称 | 是 |
| areaName | 是 | string | 游戏服名 | 是 |
| ext | 否 | string | 扩展字段 | 是 |
| body | 是 | string(64) | 商品名称 | 是 |
| sign | 是 | string | 签名（见下面） | 否 |

**同步返回**  
（由于是调用小程序支付，页面会跳转到小程序，因此，用户完成支付动作后，可以返回到调起支付的页面，也可以跳转到传入的同步链接地址页面）

如果syncUrl没传入，则从小程序支付完成后页面回到原始的H5页面

如果syncUrl传入，则跳转到改地址页面  
例如地址为： <https://xxx.com/payResult>  
则跳转:<https://xxx.com/payResult?orderId=xxx&isPay=1&ext=xxx>

跳转参数说明

| **参数名** | **类型** | **说明** |
| --- | --- | --- |
| orderId | string | sdk平台订单号 |
| isPay | int | 0，支付失败 1 支付完成 （注意：这里的支付完成指的是用户完成支付动作，支付的结果需要根据异步通知结果，处理相关业务） |
| ext | string | 扩展字段 |

## 四、支付通知

**简要描述：**

平台充值成功后通知游戏服务器 请求接口

**请求URL：**

即充值结果通知地址，CP提供。在下单接口中传入。

**请求方式：**

POST

**接口说明：**

1. 请求方为SDK 服务器，响应方为CP服务器。
2. 用户在游戏中提交购买请求后，游戏平台会异步执行充值购买，在购买操作完成后，游戏平台通过该接口将充值结果发送给“游戏服务器”。
3. 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。
4. 游戏方需要严格校验下单时候的金额、订单号、充值uid等确定无误之后方可发货、并且做好订单重复请求机制处理、游戏方将直接输出返回：SUCCESS，平台将作为是否发货成功作为标识。失败直接输出原因

**请求方：**

SDK 服务器

**响应放：**

游戏服务器

**参数：**

| **参数名** | **必选** | **签名** | **类型** | **说明** |
| --- | --- | --- | --- | --- |
| gameId | 是 | 是 | int | 游戏ID |
| uid | 是 | 是 | int | 用户id |
| gameOrderId | 是 | 是 | int | 游戏订单号 |
| orderId | 是 | 是 | string | 平台订单号 |
| payFee | 是 | 是 | int | 充值金额，整数，单位为分 |
| payTime | 是 | 是 | int | Linux时间戳 |
| roleName | 否 | 是 | string | 角色名称 |
| areaName | 否 | 是 | string | 游戏服名称 |
| ext | 否 | 是 | string | 扩展字段，下单时传入，原样返回 |
| sign | 是 | 否 | string | 签名参数 |

**返回示例**

1. SUCCESS

## 五、新增角色上报

**简要描述：**

用户进入游戏创建角色后，数据上报

**请求URL：**

https://h5.ichuanshanjia.com/h5/role/create

**请求方式：**

POST (utf-8)

**参数：**

| **参数名** | **必选** | **类型** | **说明** |
| --- | --- | --- | --- |
| gameId | 是 | int | 游戏ID |
| uid | 是 | int | 用户ID |
| createtime | 是 | string | Linux时间戳 |
| roleName | 是 | string | 游戏角色名称 （64字符） |
| roleId | 是 | string | 游戏角色ID （64字符） |
| sign | 是 | string | 签名字段字段 |

**返回示例**

1. {
2. "code": 1,
3. "msg": "成功"
4. }
5. **返回参数说明**

| **参数名** | **类型** | **说明** |
| --- | --- | --- |
| code | int | 1成功 0失败 |
| msg | string | 信息提示 |

## 六、签名

第一步，设所有发送或者接收到的数据为集合M，将集合M内参数值的参数按照参数名ASCII码从小到大排序（字典序），使用URL键值对的格式（即key1=value1&key2=value2…）拼接成字符串stringA。

第二步，在stringA最后拼接上key得到stringSignTemp字符串，并对stringSignTemp进行MD5运算，再将得到的字符串所有字符转换为大写，得到sign值signValue。

例如：md5( gameId=xxx&token=xxx&uid=xxx&key=xxx) 转大写  
得到签名结果，如：4753DCE3AE736E7F894EBCC6CD3CFF7A。

key值为app\_key sdk后台提供。