

口红机主机程序

author	data	desc
rqz	2018-08-03	修改app与game的通讯方式

目录

- 1. [基本定义](#)
- 2. [游戏发包命名规则](#)
- 3. [口红机主机与游戏之间通讯方式](#)
- 4. [口红机主机与游戏接入及通讯流程](#)
- 5. [重新定义接口](#)
- 6. [修改sdk消息格式](#)
- 7. [协议](#)

1. 基本定义

- 口红机主机程序（主要功能）（M）
 - 与服务器交互，获取支付订单信息
 - 与串口通讯，控制硬件
 - 与游戏接口
 - 管理游戏包安装
 - 管理游戏启动及处理游戏返回结果
- 游戏(G)
 - 使用主机程序参数启动游戏
 - 将游戏结果返回主机程序

2.游戏发包命名规则

- 游戏包命名使用以下格式

```
${pkg}-${versionName}-${versionCode}-release.apk
```

如：

参数	数值	说明
pkg	com.cloudpoint.friendwealth	包名
versionCode	370	版本号
versionName	2.1.2.9	版本名

游戏发包名称：

```
com.cloudpoint.friendwealth-2.1.2.9-370-release.apk
```

3. 口红机主机与游戏之间通讯方式

- 使用基于intent-broadcastreceiver的方式进行进程间通讯，目前已开发了个android类库
- 调用demo

```
https://github.com/cpsdk/simple-ipc-demo
```

- 启动游戏的方式（提供两种方式）
 - 口红机主机以intent方式启动activity(要求游戏设置gameid为category并获取相应加解密钥)
 - 口红机主机以intent方式请求service(游戏自启动service，要求游戏获取相应加解密钥)
- 数据传输
 - 数据传输格式
 - intent.action
 - intent.putByteArrayExtra('data_json',data)
 - intent.putStringExtra('data_md5',md5)
 - intent.addCategory('game_id') **activity方式**
 - **data**
 - 3des.encrypt(Message.getBytes('utf-8'))
 - Message (GameStart,GameStartState,GameEnd,GameEndState)
 - **md5**
 - md5(data)
 - 主机程序至游戏(M->G)
 - 请求

- `intent.action="GameStart"`
- 响应
 - `intent.action="GameStartState"`

4. 口红机主机与游戏接入及通讯流程

接入

- 口红机提供游戏id及加解密密钥与接入库包
 - **game_id** 游戏标识
 - **des_key** 3Des加解密钥
 - 接入库包
- 游戏
 - 使用`gameid,deskey`
 - 游戏以activity方式启动时，将`game_id`加入`intent-filter`
 - 游戏以服务方式启动时，需要游戏提供服务启动方式

5.重新定义接口

口红机App 与 GameApp

- 口红机app启动GameApp,并传入配置参数
- 口红机app通知GameApp用户按箱状态
- 口红机app通知GameApp用户启动游戏，及游戏参数
- GameApp 通知口红机app，游戏结束，及游戏状态

6.修改sdk消息格式

6.1 消息类型（MessageType）

发启	消息	接收
LipstickApp	--- LGRequest -->	GameApp
	<-- LGResponse ---	
GameApp	--- GLRequest -->	LipstickApp
	<-- GLResponse ---	

6.2 消息格式（MessageForamt）

[GLILG]Request

参数	类型	说明
cmd	string	指令
seq_id	string	指令序号
create_time	long	创建时间
device_id	string	设备id
game_id	string	游戏id
data	Req	请求消息内容

[GLILG]Request

参数	类型	说明
seq_id	string	指令序号
update_time	long	返回时间
state	int	状态
msg	string	返回消息
data	Req	返回消息内容

7.协议（Protocol）

7.1 LG0001 (启动游戏app参数或服务器参数更新时,对应游戏主界面)

LG0001Req

参数	类型	说明
backgroud	string	首界面背景图地址（已下载）
qr	string	二维码地址（已下载）
pay_icons	string[]	支付方式icon(已下载)
device_id	string	设备id
phone	string	电话
box_meta	BoxMeta[]	口红箱体信息

BoxMeta

参数	类型	说明
id	int	箱子id
image	string	口红图片景
brand	string	口红品牌
color_serial	string	口红色序
description	string	口红描述
price	int	价格（分）

LG0001Resp (无)

7.2 LG0002 （用户点击箱体按钮）

LG0002Req

参数	类型	说明
boxid	int	用户按箱体id
state	int	箱体状态 0: 无货， 1: 有货

LG0002Resp (无)

7.3 LG0003 （启动动游戏）

LG0003Req

参数	类型	说明
order_id	string	订单号
timeout	int	游戏时间
probability	int	游戏难度

LG0003Resp(无)

7.4 GL0001 (返回游戏结果)

GL0001Req

参数	类型	说明
order_id	string	订单号
state	int	是否完成 0：未完成，1：已完成

GL0001Resp(无)

应用交互流程

- *LipstickApp* 口红机程序
- *GameApp* 游戏程序

主要流程

1. *LipstickApp*发送GameStart启动游戏程序
2. *LipstickApp*发送 LG0001传递参数
 - 电话
 - 设备编号
 - 口红机各箱体参数
 - id
 - icon
 - brand
 - color serial

- 3. 用户点击箱体按钮，LipstickApp发送LG0002主界面显示口红机箱体参数
- 4. 用户支付扫码，LipstickApp发送LG0003传递游戏参数，启动游戏
- 5. GameApp完成游戏传出GL0001

废掉以下流程

通讯流程（activity）

- 口红机根据用户支付及服务器参数设置游戏时间及游戏难度，并使用启动activity
- 游戏根据设置参数启动游戏
- 游戏结束，调用intent发出游戏结果

通讯流程（service）

- 口红机启动游戏服务
- 根据用户支付及服务器参数设置游戏时间及游戏难度，将发出intent
- 游戏根据设置参数启动游戏
- 游戏结束，调用intent发出游戏结果

主要消息(Message)

- 启动游戏 (M->G)
 - 请求 **GameStart**

```
| 参数 | 类型 | 说明 |
| --- | --- | --- |
| order_id | string | 订单号 |
| game_id | string | 游戏序号 |
| timeout | int | 游戏时间 |
| probility | int | 游戏难度 |
| timestamp | long | 时间戳 |
```

- 响应

****GameState****

```
| 参数 | 类型 | 说明 |
| --- | --- | --- |
| state | int | 状态 |
| msg | string | 信息 |
| order_id | string | 订单号 |
| game_id | string | 游戏序号 |
| timestamp | long | 时间戳 |
```

- 游戏结束返回结果 (G->M)

- 请求 **GameEnd**

参数	类型	说明
---	---	---
order_id	string	订单号
game_id	string	游戏序号
succeed	int	游戏是否完成
time	int	游戏玩的时间
timestamp	long	时间戳

- 响应

****GameState****

参数	类型	说明
---	---	---
state	int	状态
msg	string	信息
order_id	string	订单号
game_id	string	游戏序号
timestamp	long	时间戳