服务器通知说明文档 v1.9

一、接收参数

| 参数名 | 说明 | 例子 |
| --- | --- | --- |
| adv\_id | 被下载应用ID | 09f277ca386ee99cb4c910e09f562112 |
| app\_id | 调用推荐列表的应用ID | b834146fa46f5d88d4097720100b6004 |
| key | 调用推荐列表时传递的key或用户ID | [XXX@sina.com](mailto:XXX@sina.com) |
| udid | 设备唯码(ios7以下使用MAC地址,ios7以上使用idfa) |  |
| open\_udid | 设备openudid | 5d2fd12f4c011eab4cdcec66d09e429ad8d251bc |
| bill | 价格,单位元 | 2.5，如果获取到该参数值为0或null，则表示该条数据无效 |
| points | 积分,单位为服务端设置(金币) | 50，如果判断bill为0或null，则给用户积分时，points也为0. |
| ad\_name | 被激活应用名称 | %E6%B5%8B%E8%AF%95，utf-8格式转码。接收到需URLDecoder |
| status | 状态 | 1为正常激活 |
| activate\_time | 激活时间 | 用户产生激活行为时间，需解码，例 ：2014-04-25+17%3A27%3A48.0 |
| order\_id | 订单号 | 由此字段判断接收数据是否唯一，如：4E6A4EE6B8735C3AF944DC88A97A941B |
| random\_code | 随机串 | 此字段参与加密操作，值如：05141441400146701119 |
| secret\_key | 验证密钥(老) | 密钥串，md5(random\_code + order\_id)  值如：58075BF0BEE3E8F557C7804D678F51FD  可以废弃此加密方式 |
| wapskey | 验证密钥结果值(新) | 新增的验证密钥，此功能需在配置回调地址时同步添加一个密钥串。密钥加密方式：  Md5(${adv\_id}${app\_id}${key}${udid}${bill}${points}${activate\_time}${order\_id}${回调密钥})  回调密钥不会被传递，需要在服务端添加，或联系客服进行配置。  注：1.接收到的activate\_time数据如果是解码后，那么在加密数据里需先进行URLEncoder转码并大写  2.万普方发送的wapskey的值是大写字符串，开发者接收到数据后需进行对应处理 |

三、响应信息：

接收到我方的通知后，以便我方查询通知结果，请以下面格式响应，

成功：{"message":"成功接收","success":true}

失败：{"message":"无效数据","success":false}

四、Java代码示例：

**import** java.io.IOException;

**import** java.net.URLEncoder;

**import** java.security.MessageDigest;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**public** **class** ServletDemo **extends** HttpServlet {

**private** **static** **final** **long** *serialVersionUID* = 1L;

**public** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

response.setContentType("text/html");

doPost(request, response);

}

**public** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

String adv\_id = request.getParameter("adv\_id");//被下载应用ID

String app\_id = request.getParameter("app\_id");//万普平台注册应用id

String key = request.getParameter("key");//调用积分墙时传入的key或者用户id

String udid = request.getParameter("udid");//设备唯码

String openudid = request.getParameter("open\_udid"); //设备open\_udid

String bill = request.getParameter("bill");//价格

**if**(bill == **null**){bill = "";}

String points = request.getParameter("points");//积分

**if**(points == **null**){points = "";}

String ad\_name = request.getParameter("ad\_name");//下载的应用名称

String status = request.getParameter("status");//状态

**if**(status == **null**){status = "";}

String activate\_time = request.getParameter("activate\_time");//激活时间

//========以下参数为旧版验证数据的参数，可以废弃此方法加密

String order\_id = request.getParameter("order\_id"); //订单号

String random\_code = request.getParameter("random\_code");//随机串

String secret\_key = request.getParameter("secret\_key");//验证密钥

//========新版验证

String wapskey = request.getParameter("wapskey");//数据加密串

String callBackKey = "test"; //回调密钥

//当满足价格、积分不为空且不为0和状态码为1时才为有效数据

**if**(!bill.equals("0") && !points.equals("0") && status.equals("1")){

activate\_time = URLEncoder.*encode*(activate\_time, "UTF-8");//激活时间在传输时会自动解码，加密是要用url编码过的，如果没有自动解码请忽略转码

//加密并判断密钥

String plaintext = adv\_id+app\_id+key+udid+bill+points+activate\_time+order\_id+callBackKey;

System.*out*.println("加密前的密钥：" + plaintext);

String keys = *MD5*(plaintext);

System.*out*.println("解密后的密钥：" + keys);

//判断和wapskey是否相同

**if**(keys.equalsIgnoreCase(wapskey)){

//成功接收数据

System.*out*.println("验证成功");

}**else**{

System.*out*.println("验证失败");

}

}**else**{

System.*out*.println("无效数据");

}

}

**public** **static** String MD5(String s) {

**char** hexDigits[]={'0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F'};

**try** {

**byte**[] btInput = s.getBytes();

// 获得MD5摘要算法的 MessageDigest 对象

MessageDigest mdInst = MessageDigest.*getInstance*("MD5");

// 使用指定的字节更新摘要

mdInst.update(btInput);

// 获得密文

**byte**[] md = mdInst.digest();

// 把密文转换成十六进制的字符串形式

**int** j = md.length;

**char** str[] = **new** **char**[j \* 2];

**int** k = 0;

**for** (**int** i = 0; i < j; i++) {

**byte** byte0 = md[i];

str[k++] = hexDigits[byte0 >>> 4 & 0xf];

str[k++] = hexDigits[byte0 & 0xf];

}

**return** **new** String(str);

} **catch** (Exception e) {

e.printStackTrace();

**return** **null**;

}

}

}

注：1、我方会根据响应信息来判断是否重复发送，如果HTTP响应代码为非200，则会做重复通知

2、服务器交互方式仅适用于推荐列表（广告墙）

ip of waps:

219.234.85.238

219.234.85.223

219.234.85.211

219.234.85.231

219.234.85.234

以上IP用于配置访问限制