

支付宝发货信息同步使用手册

修订历史

版本号	作者	修订章节	修订原因	修订日期	审阅人
1.0	技术支持团队	全部	起草	2010-7-28	

版权信息

本手册中所有的信息为支付宝公司提供。未经过支付宝公司书面同意，接收本手册的人不能复制，公开，泄露手册的部分或全部的内容。

1	概要	2
2	请求参数.....	3
3	返回参数.....	3
4	签名机制.....	5
4.1	步骤一：哪些是要加密的参数？	5
4.2	步骤二：排序与连接.....	6
4.3	步骤三：加密.....	6
5	发货流程.....	7
5.1	进入商家网站后台支付宝交易发货页面.....	7
5.2	发货.....	7
5.3	发货完成.....	7
6	数据传输流程.....	7
6.1	商户系统请求/支付宝响应交互模式.....	7
7	XML返回	8
7.1	特性.....	8
7.2	如何实现发货信息同步.....	9

1 概要

本手册是以“确认发货(send_goods_confirm_by_platform).pdf”文档为根本，在此基础

上做详细说明。本手册仅包含发货信息同步功能，因此部分请求与通知返回参数在本手册中不做直接体现。

2 请求参数

参数名	功能介绍
必填参数——不需要去更改值的参数	
service	用来标明当前接口类型。支付宝为了辨识每种接口，用名称以示区分。例如： <code>send_goods_confirm_by_platform</code> 为支付宝发货信息同步接口服务名称。
sign_type	参数签名加密方式。值为： <code>MD5</code>
_input_charset	编码格式。一般网页右键可看到编码，打开扩展箭头后发现，默认的原点在哪种编码上，那么就能知道该网站使用的是什么编码。两个值可选： <code>utf-8</code> 、 <code>gb2312</code> （GBK），支付宝默认为 <code>GBK</code> 编码方式，不区分大小
必填参数——特殊参数	
sign	加密结果。把所有待签名参数，每个参数以“参数名=参数值”为一个字符串的形式，通过a到z的顺序排序以后，用&字符连接而成的字符串，并对这个字符串加密后得出的 32 位加密结果。（示例见 签名机制 ）
必填参数——需要设置的参数	
partner	合作伙伴 ID。商户签约合同后，商家可登陆到支付宝里面，在“商家服务”栏目中找到，以 2088 开头，纯 16 位数字。
trade_no	支付宝交易号。以 8 位日期开头的纯数字唯一订单号。
logistics_name	物流公司名称。
invoice_no	物流发货单号
transport_type	发货时的运输类型。三个固定值可选择： <code>EXPRESS</code> （快递）、 <code>EMS</code> （EMS）、 <code>POST</code> （平邮）
选填参数	
seller_ip	发货人的客户端 IP 地址

3 返回参数

返回格式：**XML 格式**

参数名	功能介绍
输出反馈回来的处理结果，在根节点 aipay 中	
is_success	标示是否请求成功。成功，则返回“T”，不成功则返回“F”。
输出请求时的参数，在节点 request 中	
_input_charset	编码格式。一般网页右键可看到编码，打开扩展箭头后发现，默认的原点在哪种编码上，那么就能知道该网站使用的是什么编码。两个值可选： <code>utf-8</code> 、 <code>gb2312</code> （GBK）。对应请求时参数 <code>_input_charset</code> ，原样输出。
partner	它是签约合同后，商户可以得到一个标示自己的唯一编号。以 2088 开头，纯 16 位数字。商家可登陆到支付宝里面，在“商家服务”栏目中找到。对

	应请求时参数 partner ，原样输出。
service	用来标明当前接口类型。支付宝为了辨识每种接口，用名称以示区分。例如： send_goods_confirm_by_platform 。
trade_no	支付宝交易号。以 8 位日期开头的纯数字唯一订单号。
logistics_name	物流公司名称。
invoice_no	物流发货单号
transport_type	请求时选择的发货运输类型
seller_ip	发货人的客户端 IP 地址
输出返回信息，在节点 response 中	
buyer_account	买家账号，以 2088 开头 20 位纯数字的用户 ID
buyer_actions	买家下一步操作，固定值： [REFUND, CONFIRM_GOODS] ，表示申请退款或确认收货。
buyer_login_email	买家登陆支付宝账号
buyer_type	买家类型。个人账号或公司账号。
buyer_user_id	买家唯一用户号。以 2088 开头 16 位纯数字的用户 ID
channel	操作通道。固定值： interface/escrow ，表示接口
create_time	交易创建时间
currency	货币类型。固定值： 156 ，表示人民币
gathering_type	商品收款类型。默认值： 1 。与担保交易中的参数 payment 对应。
last_modified_time	最后修改时间。即发货时间
operator_role	操作角色。固定值： B ，代表买家类型。
out_trade_no	商户网站唯一订单号
partner_id	商户合作身份者 ID。以 2088 开头 16 位纯数字的用户 ID
seller_account	卖家账号，以 2088 开头 20 位纯数字的用户 ID
seller_actions	卖家下一步操作，固定值： [EXTEND_TIMEOUT] ，表示延长超时，需要买家操作完成以后方可操作。
seller_login_email	卖家登陆支付宝账号。
seller_type	卖家类型。个人账号或公司账号。
seller_user_id	卖家唯一用户号。以 2088 开头 16 位纯数字的用户 ID
service_fee	服务费用。
service_fee_ratio	服务费率
stop_timeout	停止超时状态。 F ：停止。 T ：不停止。
total_fee	交易总金额
trade_from	交易来源。 INST_PARTNER ，表示外部商家
trade_no	支付宝交易号。以 8 位日期开头的纯数字唯一订单号。
trade_status	交易状态。固定值： WAIT_BUYER_CONFIRM_GOODS ，表示卖家已发货，等待买家确认收货。
trade_type	交易类型。固定值： S ，代表担保交易的含义。
输出返回的校验参数，在根节点 alipay 中	
sign_type	它是影响参数签名加密使用的方式。
sign	输出返回时支付宝提供的加密结果。是把所有要加密的返回参数，每个参数以“参数名=参数值”为一个字符串的形式，通过 a 到 z 的顺序排序以后，

用&字符连接而成的字符串,并对这个字符串加密后得出的 32 位加密结果。

由于没有通知、返回页 (参数 notify_url、return_url), 所以返回的方式不是通知也不是自动跳转回商户网站地址, 而是自动跳转到结果页面 (即 XML 格式的支付宝页面), 如图:

```
<?xml version="1.0" encoding="gb2312" ?>
- <alipay>
  <is_success>T</is_success>
  - <request>
    <param name="seller_ip">192.168.1.2</param>
    <param name="partner">2088002892920338</param>
    <param name="logistics_name">物流公司名称</param>
    <param name="trade_no">2010080627505335</param>
    <param name="invoice_no">物流发货单号06</param>
    <param name="_input_charset">gb2312</param>
    <param name="service">send_goods_confirm_by_platform</param>
    <param name="transport_type">EXPRESS</param>
  </request>
  - <response>
    - <tradeBase>
      <buyer_account>20880023967123540156</buyer_account>
      <buyer_actions>[REFUND, CONFIRM_GOODS]</buyer_actions>
      <buyer_login_email>[REDACTED]@[REDACTED].com</buyer_login_email>
      <buyer_type>PRIVATE_ACCOUNT</buyer_type>
      <buyer_user_id>2088002396712354</buyer_user_id>
      <channel>interface/escrow</channel>
      <create_time>2010-08-06 19:57:29</create_time>
      <currency>156</currency>
      <gathering_type>1</gathering_type>
      <last_modified_time>2010-08-09 11:32:40</last_modified_time>
      <operator_role>B</operator_role>
      <out_trade_no>20100806195713</out_trade_no>
      <partner_id>2088002892920338</partner_id>
      <seller_account>20880028929203380156</seller_account>
      <seller_actions>[EXTEND_TIMEOUT]</seller_actions>
      <seller_login_email>[REDACTED]@[REDACTED].com</seller_login_email>
      <seller_type>PRIVATE_ACCOUNT</seller_type>
      <seller_user_id>2088002892920338</seller_user_id>
      <service_fee>0.00</service_fee>
      <service_fee_ratio>0.0</service_fee_ratio>
      <stop_timeout>F</stop_timeout>
      <total_fee>0.01</total_fee>
      <trade_from>INST_PARTNER</trade_from>
      <trade_no>2010080627505335</trade_no>
      <trade_status>WAIT_BUYER_CONFIRM_GOODS</trade_status>
      <trade_type>S</trade_type>
    </tradeBase>
  </response>
  <sign>f70a6d06181ba6f05ebecb03aa704a8a</sign>
  <sign_type>MD5</sign_type>
</alipay>
```

4 签名机制

4.1 步骤一: 哪些是要加密的参数?

4.1.1 加密参数组是什么？

支付宝提供的接口案例代码中对应的参数组所在位置如下：

ASP: index.asp 里 para= Array 括号中包含的参数集合

PHP: index.php 里 \$parameter = array 括号中包含的参数集合

ASP.NET: default.aspx.cs 里 AlipayService aliService = new AlipayService 括号中包含的参数集合

JAVA: src/com/alipay/util/AlipayService.java 里 CreateUrl 函数中的 Map sPara = new HashMap();包含的参数集合

4.1.2 哪些是要加密的参数？

- sign、sign_type、key 这三个参数是不在加密参数组中。
- 当编码格式是 GB2312 或 GBK 的时候，_input_charset 可参与加密也可以不参与加密，当编码格式是 utf-8 的时候，_input_charset 是必须参与加密的。
- 二选一的参数，则只能有一个在加密参数组中，如：seller_email 与 seller_id；total_fee 与 price、quantity；buyer_email 与 buyer_id。
- 当想增加扩展功能时，扩展功能参数必须全部参与加密，加入加密参数组中。
- 所有参与加密的参数，绝对不能为空值。

4.2 步骤二：排序与连接

所有参数连同携带的值，组合成“参数=参数值”的字符串形式，存放进加密参数组，例：

```
string[] para ={
    "service=send_goods_confirm_by_platform",
    "_input_charset=gb2312",
    "partner=2088002892920338",
    "trade_no=2010080627505335",
    "logistics_name=物流公司名称",
    "invoice_no=物流发货单号 06",
    "transport_type=EXPRESS",
};
```

对数组里的每一个值从 a 到 z 的顺序排序，若遇到相同首字母，则看第二个字母，以此类推。排序后的顺序把每个值以“&”字符连接起来，如：

```
_input_charset=gb2312&invoice_no= 物 流 发 货 单 号 06&logistics_name= 物 流 公 司 名 称
&partner=2088002892920338&service=send_goods_confirm_by_platform&trade_no=2010080627505335&transport_type=EXPRESS&sign=2da824a0f75198e1c0f8b2897798576f&sign_type=MD5
```

4.3 步骤三：加密

在众多参数中有一个参数是并没有在请求参数列表中，这个参数是 key。一个 partner 对应一个 key，它就是俗称的私钥。以英文字母和数字组成的 32 位字符串。商家可登录到支付宝里面，在“商家服务”栏目中找到，安装数字证书即可查看。

它是加密时才需要用上的参数。

当拿到排序后用“&”字符连接起来的字符串后，再把参数 key 的值与该字符串拼接起

来形成新的字符串，如：

```
_input_charset=gb2312&invoice_no= 物 流 发 货 单 号 06&logistics_name= 物 流 公 司 名 称
&partner=2088002892920338&service=send_goods_confirm_by_platform&trade_no=2010080627505335&trans
port_type=EXPRESS&sign=2da824a0f75198e1c0f8b2897798576f&sign_type=MD5wop2rabt5bvcetdsg8wj6jjhzx
n9jf0a
```

把这个新的字符串放入加密函数中，得到 32 位加密结果字符串。

此时，整个签名算法完成。

这个 32 位加密结果字符串就是参数 sign 的值。

5 发货流程

5.1 进入商家网站后台支付宝交易发货页面

打开在商家网站后台的支付宝交易发货功能界面，如图：

支付宝发货通道

支付宝交易号：	<input type="text"/>
发货类型：	快递 <input type="button" value="v"/>
物流公司名称：	<input type="text"/>
物流发货单号：	<input type="text"/>
<input type="button" value="发 货"/>	

5.2 发货

填写支付宝交易号、发货类型、物流公司名称、物流发货单号等发货信息，点击“发货”按钮。

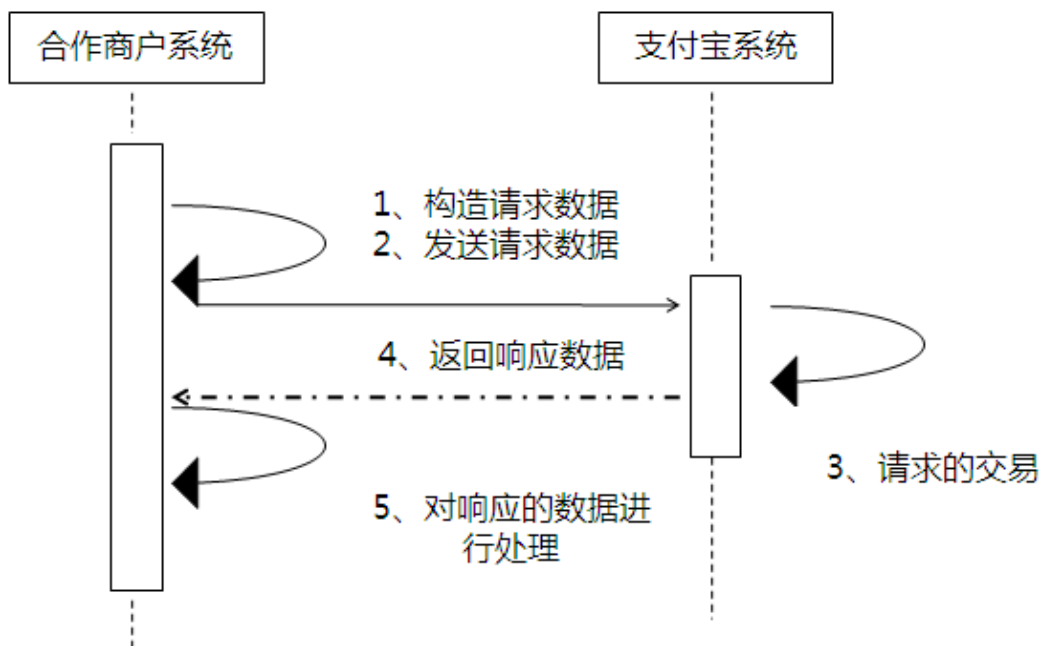
5.3 发货完成

点击“发货”后，当前录页面自动刷新，提示发货成功。该页面可由商家自己来制定、页面美化、数据处理等工作（相关业务逻辑实现见[7.2](#)）如图：

XML返回	
是否发货成功：	T
订单号：	20100806195713
支付宝交易号：	2010080627505335
交易状态：	WAIT_BUYER_CONFIRM_GOODS
发货时间：	2010-08-09 11:32:40

6 数据传输流程

6.1 商户系统请求/支付宝响应交互模式



1、构造请求数据

商户通过提供的接口代码示例，通过代码示例的规则，程序构造与运算得到 sign 加密结果以及构造后的请求给支付宝的数据集合。GET 方式下是 URL 地址链接，POST 方式下是支付宝网关地址及参数集合。

2、发送请求数据

把构造完成的数据集合，通过页面链接跳转或表单提交的方式传递给支付宝。

3、请求的交易

支付宝得到这些集合后，会先做安全校验等验证，一系列验证通过后便会处理完成这次发送过来的数据请求。

4、返回相应数据

支付宝对商家发来的请求处理完成，程序上自动进行重新构造 URL 地址链接，以自动跳转的方式跳回商家在请求时设定好的页面路径地址。

5、对相应的数据进行处理。

在 XML 返回类型的接口中，商家的请求页面（sendgoods 页面文件）远程解析并得到支付宝反馈回来的 XML 格式的数据，把这些数据结合自身网站情况，进行数据处理（如：订单更新）。

7 XML 返回

7.1 特性

支付宝对请求处理完成后，当前页面立刻自动跳转返回处理的结果数据，这些数据是以 XML 的格式呈现出来。这类的返回称之为 XML 返回。

它具有以下特性：

- 1、点击发货以后，即请求给支付宝数据后，支付宝立刻返回处理结果；
- 2、它仅仅只在点击发货以后（即请求给支付宝数据后）自动跳转回来，因此只会跳转回来一次；
- 3、XML 返回在本机电脑就可调试；

- 4、返回 URL 只有一分钟的有效, 超过一分钟该链接地址会失效, 验证则会失败。建议不要模拟返回数据;
- 5、该页面中获得返回数据的时需要远程 XML 解析;
- 6、本地电脑或商户网站服务器需要支持远程 XML 解析, 如: 需要支持 SSL;
- 7、若未做远程 XML 解析, 支付宝处理后的结果数据仅停留在支付宝的页面中 (<https://www.alipay.com/cooperate/gateway.do> 路径页面), 本地电脑或商户服务器无法或获取处理的结果数据; 若有做远程 XML 解析, 支付宝处理后的结果数据则停留在请求的页面 (sendgoods 文件), 此时获取数据的页面就是请求的页面文件 (sendgoods 文件), 因此相关的业务逻辑程序代码需要在请求的页面文件 (sendgoods 文件) 中编写。

7.2 如何实现发货信息同步

请求页 (sendgoods 文件) 中处理流程:

```
'nodeOut_trade_no 是远程 XML 解析后获取到的商家网站订单系统中的唯一订单号
dim sOld_trade_status = GetOldTradeStatus(sOut_trade_no) '从商户数据库中查询得到该笔交易当前的交易状态, sOld_trade_status 是数组, sOld_trade_status(0)是交易状态的序号, sOld_trade_status(1)是交易状态名称。
假设:
    交易状态序号 0 表示订单未处理;
    交易状态序号 1 表示买家已在支付宝交易管理中产生了交易记录, 但没有付款
    交易状态序号 2 表示买家已在支付宝交易管理中产生了交易记录且付款成功, 但卖家没有发货
    交易状态序号 3 表示卖家已经发了货, 但买家还没有做确认收货的操作
    交易状态序号 4 表示买家已经确认收货, 这笔交易完成
if sOld_trade_status = 2 then      '判断该笔交易状态目前是否是 WAIT_SELLER_SEND_GOODS 状态
    UpdateOrder(nodeOut_trade_no);      '根据订单号更新订单, 把订单处理成卖家已发货, 等待
    买家确认收货状态
end if
```