



# 收单退款接口

---

**服务名称：alipay.acquire.refund**

**版本号：1.5**

## 目 录

<b>1 文档说明</b>	<b>4</b>
1.1 功能描述	4
1.2 阅读对象	4
1.3 业务术语	4
<b>2 功能演示</b>	<b>4</b>
<b>3 数据交互</b>	<b>5</b>
<b>4 请求参数说明</b>	<b>5</b>
4.1 含义	5
4.2 列表	6
4.3 业务关联ID集合说明	7
4.4 业务扩展参数说明	8
4.5 样例	9
<b>5 xml同步返回参数说明</b>	<b>10</b>
5.1 含义	10
5.2 列表	10
5.3 退款渠道列表信息	12
5.4 样例	13

<b>6 通知返回数据处理</b>	<b>14</b>
6.1 xml同步返回特性	14
6.2 商户业务处理注意事项	15
<b>7 签名机制</b>	<b>15</b>
7.1 生成待签名的字符串	15
7.1.1 需要参与签名的参数	15
7.1.2 生成待签名字符串	16
7.2 签名	16
7.2.1 MD5 签名	16
7.2.2 DSA、RSA签名	17
7.2.3 返回参数验签特别说明	17
<b>8 附录</b>	<b>19</b>
8.1 业务错误码	19
8.2 接入错误码	20
8.3 系统错误码	20
8.4 退款渠道	21

## 1 文档说明

### 1.1 功能描述

商户可对交易进行退款。

### 1.2 阅读对象

支付宝收单退款接口文档面向具有一定的网站开发能力，了解 ASP、PHP、JAVA、ASP.NET 等开发语言中的一种及 SQL 数据库语言的网站开发、维护和管理人员。

### 1.3 业务术语

表1-1 业务术语

术语	解释
请求	通过 HTTP 协议把需要传输的数据发送给接收方的过程。
返回	支付宝根据得到的数据处理完成后，以 XML 数据形式将处理完成的结果信息反馈给商户网站。

## 2 功能演示

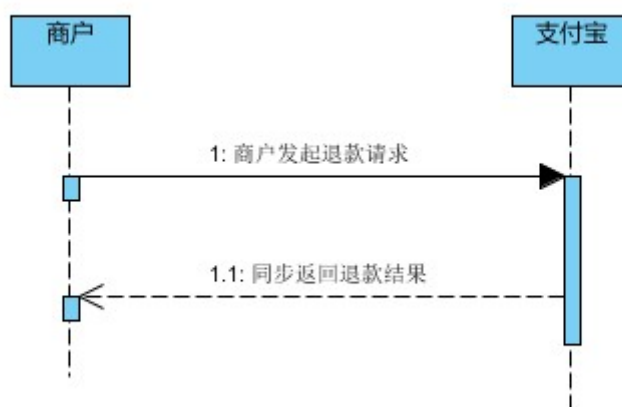


图2-1 收单退款流程

### 3 数据交互

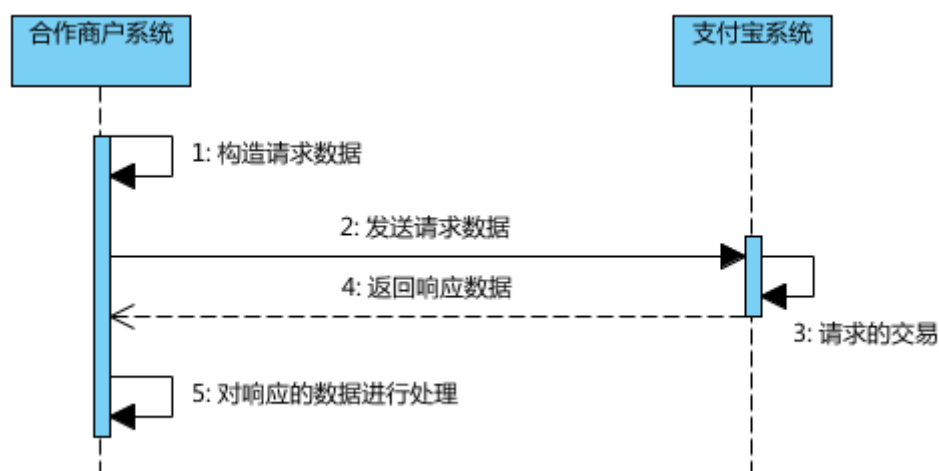


图3-1 商户系统请求/支付宝响应交互模式

#### (1) 构造请求数据

商户根据支付宝提供的接口规则，通过程序生成得到签名结果及要传输给支付宝的数据集合。

#### (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) 支付宝对请求数据进行处理

支付宝得到这些集合后，会先进行安全校验等验证，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) 返回处理的结果数据

支付宝会把处理完成的结果数据以 XML 的形式返回给商户。

#### (5) 对获取的返回结果数据进行处理

商户利用编程方法来模拟 http 请求远程解析 XML，获取支付宝返回的结果数据后，可以结合自身网站的业务逻辑进行数据处理。

## 4 请求参数说明

### 4.1 含义

请求参数是商户在与支付宝进行数据交互时，提供给支付宝的请求数据，以便支付宝根据这些数据进一步处理。

## 4.2 列表

表4-1 请求参数说明

参数	参数名称	类型（字节长度）	参数说明	是否可为空	样例
基本参数					
service	接口名称	String	接口名称。	不可空	alipay.acquire.refund
partner	合作者身份ID	String(16)	签约的支付宝账号对应的支付宝唯一用户号。 以 2088 开头的 16 位纯数字组成。	不可空	2088101125328850
_input_charset	参数编码字符集	String	商户网站使用的编码格式，如 utf-8、gbk、gb2312 等。	不可空	utf-8
sign_type	签名方式	String	DSA、RSA、MD5 三个值可选，必须大写。	不可空	MD5
sign	签名	String	请参见“7 签名机制”。	不可空	363a752668ee2b3b8f124c623bd028ad
alipay_ca_request	签名类型	String	签名类型。 <ul style="list-style-type: none"> <li>1：证书签名</li> <li>2：其他密钥签名</li> </ul> 如果为空，默认为 2。	可空	2
业务参数					
out_trade_no	商户网站唯一订单号	String(64)	支付宝合作商户网站唯一订单号。	不可空	HZ0120131127001
refund_amount	退款金额	number(9,2)	退款金额不能大于订单金额，全额退款必须与订单金额一致。	不可空	200.00
trade_no	支付宝交易号	String(64)	该交易在支付宝系统中的交易流水号。 最短 16 位，最长 64 位。 如果同时传了 out_trade_no 和 trade_no，则以 trade_no 为准。	可空	2013112611001004680073956707

参数	参数名称	类型（字节长度）	参数说明	是否可为空	样例
out_request_no	商户退款请求单号	String(64)	商户退款请求单号，用以标识本次交易的退款请求。 如果不传入本参数，则以 out_trade_no 填充本参数的值。同时，认为本次请求为全额退款，要求退款金额和交易支付金额一致。	可空	HZ01RF001
operator_type	操作员类型	String(1)	操作员的类型： <ul style="list-style-type: none"> <li>0：支付宝操作员</li> <li>1：商户的操作员</li> </ul> 如果传入其它值或者为空，则默认设置为 1。	可空	1
operator_id	操作员号	String(28)	卖家的操作员 ID。	可空	OP001
refund_reason	退款原因	String(128)	退款原因说明。	可空	正常退款
ref_ids	业务关联 ID 集合	String(256)	业务关联ID集合，用于放置商户的退款单号、退款流水号等信息，json格式，具体请参见“4.3 业务关联ID集合说明”。	可空	[{"id_type":"orig_out_request_no","id":"HZ0001"}, {"id_type":"orig_out_order_no","id":"HZ0001"}]
extend_params	业务扩展信息	String(128)	用于商户退款业务信息的传递，只有商户与支付宝约定了传递此参数且约定了参数含义，此参数才有效。 比如可传递条码支付场景下的门店ID等信息，以json格式传输，请参见“4.4 业务扩展参数说明”。	可空	{"STORE_ID":"BJ_ZZ_001","STORE_TYPE":"0"}

#### 说明：

部分参数类型为 String，未指明长度范围，表明系统不校验该参数的长度。

## 4.3 业务关联ID集合说明

业务关联 ID 集合参数为 json 格式，包含以下参数：

表4-2 业务关联 ID 集合参数说明

参数	参数名称	类型（字节长度）	参数说明	是否可为空	样例
id_type	扩展 ID 类型	String	扩展 ID 类型： <ul style="list-style-type: none"><li>orig_out_order_no: 商户原始退款单号</li><li>orig_out_request_no: 商户原始退款请求流水号</li></ul>	不可空	orig_out_order_no
id	扩展 ID 值	String	扩展 ID 值	不可空	HZ0001

格式结构：

```
[  
{扩展 ID 类型:扩展 ID 类型 1, 扩展 ID 值:扩展 ID 值 1}  
,  
{扩展 ID 类型:扩展 ID 类型 2, 扩展 ID 值:扩展 ID 值 2}  
,  
...  
]
```

示例：

- 业务需求：传递商户退款订单号和商户订单退款流水号
- 参数赋值：  

```
ref_ids=[{"id_type":"orig_out_request_no","id":"HZ0001"}, {"id_type":"orig_out_order_no","id":"HZ0001"}]
```

## 4.4 业务扩展参数说明

业务扩展参数为 json 格式，包含以下参数：



表4-3 业务扩展参数说明

参数	参数名称	类型（字节长度）	参数说明	是否可为空	样例
STORE_TYPE	门店类型	String	门店类型。 • 0: 支付宝门店, 指支付宝体系内的门店; • 1: 商户门店, 指商户自己的门店。	可空	0
STORE_ID	门店编号	String	门店的编号, 需要和门店类型一起使用。 商户需保证传入的门店编号在相应门店类型下是存在的。	可空	BJ_ZZ_001
TERMINAL_ID	终端设备号	String	提交请求使用的终端设备ID。 说明: 如果要使用优惠立返活动, 该参数必须传入。	可空	A80001

**格式结构:**

{门店类型:门店类型值,门店编号:门店编号值,终端设备号:终端设备号值}

**说明:**

业务扩展参数只支持传入一条门店信息。

**示例:**

- 业务需求: 传递门店 ID、门店类型
- 参数赋值:  

```
extend_params={"STORE_ID":"BJ_ZZ_001","STORE_TYPE":"0"}
```

## 4.5 样例

[https://mapi.alipay.com/gateway.do?operator\\_id=OP001&sign\\_type=MD5&out\\_trade\\_no=HZ0120131127001&operator\\_type=1&sign=363a752668ee2b3b8f124c623bd028ad&input\\_charset=UTF-8&service=alipay.acquire.refund&out\\_request\\_no=HZ01RF001&partner=2088101125328850&refund\\_reason=%E6%AD%A3%E5%B8%B8%E9%80%80%E6%AC%BE&refund\\_amount=200&alipay\\_request=2](https://mapi.alipay.com/gateway.do?operator_id=OP001&sign_type=MD5&out_trade_no=HZ0120131127001&operator_type=1&sign=363a752668ee2b3b8f124c623bd028ad&input_charset=UTF-8&service=alipay.acquire.refund&out_request_no=HZ01RF001&partner=2088101125328850&refund_reason=%E6%AD%A3%E5%B8%B8%E9%80%80%E6%AC%BE&refund_amount=200&alipay_request=2)

说明：

本样例仅供参考，支付宝网关为<https://mapi.alipay.com/gateway.do>。

## 5 xml同步返回参数说明

### 5.1 含义

支付宝对商户提供的请求数据进行处理后，返回给商户结果数据，以便商户根据这些数据进一步处理。

返回结果分两种场景，一种为业务正常受理的结果，一种为系统级异常或者接入数据错误。对于业务正常受理，商户需要解析响应码以判断业务是否处理成功。

### 5.2 列表

表5-1 xml 同步返回参数说明

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
基本参数					
is_success	请求是否成功	String	请求是否成功。请求成功不代表业务处理成功。 <ul style="list-style-type: none"><li>T 代表成功</li><li>F 代表失败</li></ul>	不可空	T
sign_type	签名方式	String	DSA、RSA、MD5 三个值可选，必须大写。	可空	MD5
sign	签名	String	请参见“7 签名机制”。	可空	fc209d86b57b2adb23254a897ebaa7d9
error	错误代码	String	<ul style="list-style-type: none"><li>请求成功时，不存在本参数；</li><li>请求失败时，本参数为错误代码，参见“8.2 接入错误码”和“8.3 系统错误码”。</li></ul>	可空	ILLEGAL_SIGN

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
业务参数					
result_code	响应码	String(32)	退款处理结果响应码。 <ul style="list-style-type: none"> <li>SUCCESS: 退款成功</li> <li>FAIL: 退款失败</li> <li>UNKNOWN: 结果未知</li> </ul>	不可空	SUCCESS
trade_no	支付宝交易号	String(64)	该交易在支付宝系统中的交易流水号。 最短 16 位，最长 64 位。	可空	2013112011001004330000121536
out_trade_no	商户网站唯一订单号	String(64)	对应商户网站的订单系统中的唯一订单号，非支付宝交易号。 需保证在商户网站中的唯一性。是请求时对应的参数，原样返回。	可空	6823789339978248
buyer_user_id	买家支付宝用户号	String(24)	买家支付宝账号对应的支付宝唯一用户号。 以 2088 开头的纯 16 位数字。	可空	2088102122524333
buyer_logon_id	买家支付宝账号	String(100)	买家支付宝账号，可以是 Email 或手机号码。	可空	159****5620
fund_change	本次退款请求是否发生资金变动	String(1)	对同一个商户退款请求，如果该笔退款已退款过，则直接返回上一次的退款结果。同时，返回本次请求是否发生了资金变动的标识。 <ul style="list-style-type: none"> <li>Y: 本次退款请求发生资金变动；</li> <li>N: 本次退款请求未发送资金变动。</li> </ul>	可空	Y
refund_fee	退款金额	Number	实际退款金额，单位为元。	可空	1.00
gmt_refund_pay	退款时间	Date	退款时间，格式为 yyyy-MM-dd HH:mm:ss。	可空	2013-11-27 15:45:57
refund_detail_item_list	退款渠道列表	List	退款金额的退回渠道，xml 格式，可返回多个退款渠道子节点<TradeFundBill>，该节点包含的参数请参见“5.3 退款渠道列表信息”。	可空	参见“5.3 退款渠道列表信息”

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
detail_error_code	详细错误码	String(48)	对返回响应码进行原因说明，请参见“8.1 业务错误码”。当 result_code 响应码为 SUCCESS 时，不返回该参数。	可空	TRADE_STATUS_ERROR
detail_error_des	详细错误描述	String(64)	对详细错误码进行文字说明。当 result_code 响应码为 SUCCESS 时，不返回该参数。	可空	交易状态不合法

#### 说明：

同步返回的参数随支付宝服务端的演化，可能会返回更多的节点，需要程序使用方对文档说明以外的节点予以忽略。

## 5.3 退款渠道列表信息

表5-2 TradeFundBill 节点参数列表

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
fund_channel	退款渠道	String	退款渠道，参见“8.4 退款渠道”。	可空	ALIPAYACCOUNT
amount	退款金额	String	使用指定退款渠道退还的金额，单位为元。	可空	70.00

#### 退款单据列表信息示例：

```
<refund_detail_item_list>
  <TradeFundBill>
    <amount>70.00</amount>
    <fund_channel>ALIPAYACCOUNT</fund_channel>
  </TradeFundBill>
  <TradeFundBill>
    <amount>20.00</amount>
    <fund_channel>COUPON</fund_channel>
  </TradeFundBill>
  <TradeFundBill>
```

```
<amount>10.00</amount>
<fund_channel>MCARD</fund_channel>
</TradeFundBill>
</refund_detail_item_list>
```

## 5.4 样例

- 业务正常受理并退款成功:

```
<?xml version="1.0" encoding="utf-8"?>
<alipay>
  <is_success>T</is_success>
  <request>
    <param name="sign">8c066dc2cb92a9c2c8070d935917fa73</param>
    <param name="_input_charset">UTF-8</param>
    <param name="sign_type">MD5</param>
    <param name="service">alipay.acquire.refund</param>
    <param name="out_request_no">daiqing121</param>
    <param name="partner">2088101125936190</param>
    <param name="refund_reason">退款接口</param>
    <param name="refund_amount">1.00</param>
    <param name="out_trade_no">6823789339978248</param>
  </request>
  <response>
    <alipay>
      <result_code>SUCCESS</result_code>
      <buyer_logon_id>159***5620</buyer_logon_id>
      <buyer_user_id>2088102122524333</buyer_user_id>
      <fund_change>Y</fund_change>
      <out_trade_no>6823789339978248</out_trade_no>
      <trade_no>2013112011001004330000121536</trade_no>
    </alipay>
  </response>
  <sign>fc209d86b57b2adb23254a897ebaa7d9</sign>
  <sign_type>MD5</sign_type>
</alipay>
```

- 请求成功，业务处理失败:

```
<?xml version="1.0" encoding="utf-8"?>
<alipay>
  <is_success>T</is_success>
  <request>
    <param name="sign">8c066dc2cb92a9c2c8070d935917fa73</param>
    <param name="_input_charset">UTF-8</param>
```

```

<param name="sign_type">MD5</param>
<param name="service">alipay.acquire.refund</param>
<param name="out_request_no">daiqing121</param>
<param name="partner">2088101125936190</param>
<param name="refund_reason">退款原因</param>
<param name="refund_amount">1.00</param>
<param name="out_trade_no">6823789339978248</param>
</request>
<response>
  <alipay>
    <result_code>FAIL</result_code>
    <detail_error_code>TRADE_STATUS_ERROR</detail_error_code>
    <detail_error_des>交易状态不合法</detail_error_des>
  </alipay>
</response>
<sign>fc209d86b57b2adb23254a897ebaa7d9</sign>
<sign_type>MD5</sign_type>
</alipay>

```

- 请求失败或者接入数据错误时输出：

```

<?xml version="1.0" encoding="utf-8"?>
<alipay>
  <is_success>F</is_success>
  <error>ILLEGAL_SIGN</error>
</alipay>

```

## 6 通知返回数据处理

### 6.1 xml同步返回特性

- (1) 支付宝处理完请求的数据后，当前界面会自动刷新，立刻显示处理结果；
- (2) 处理一次请求，当前界面只跳转回来一次，即只返回一次处理结果；
- (3) 由于没有参数 `return_url` 来设置页面跳转同步通知页面文件路径，因此显示处理结果的页面文件所在位置是在支付宝的服务器中，即当前页面路径地址是以支付宝的域名为开头；
- (4) 该步骤可在本机电脑调试，无需到服务器上；
- (5) 处理结果显示的格式是 XML 格式；
- (6) 可对这个 XML 格式的处理结果进行 XML 远程解析，使之能在支付宝处理结果时获取到相关处理结果数据，并对这个结果界面进行商户的业务逻辑处理；
- (7) 本地电脑或商户服务器的配置上需支持 XML 远程解析，如：需要支持 SSL；

- (8) 页面跳转后，浏览器地址栏中的链接仅在一分钟内有效，超过一分钟该链接地址会失效，导致该页面中商户编写的业务逻辑程序无法完成处理。

## 6.2 商户业务处理注意事项

当支付宝处理完成后，支付宝会以 XML 同步返回数据的形式，把数据反馈给商户网站。商户可以利用编程方法来模拟 http 请求远程解析 XML 获得处理的结果数据，增加商户的业务逻辑处理程序。此时，**商户必须判断商户网站中是否已经对该次的结果数据做过同样处理**。如果不判断，存在潜在的风险，商户自行承担因此而产生的所有损失。

# 7 签名机制

## 7.1 生成待签名的字符串

### 7.1.1 需要参与签名的参数

- 在请求参数列表中，除去 sign、sign\_type 两个参数外，其他需要使用到的参数皆是要签名的参数。（个别接口中参数 sign\_type 也需要参与签名。）
- 在 XML 返回参数列表中，一个节点作为一个参数，节点名代表参数名，节点值代表参数值。

正常返回的情况下，需要签名的参数是节点<response>的子节点<alipay>下的所有子节点。

发生错误的情况下，需要签名的参数只有<error>节点。

如果参数值中出现 HTML 转义字符，那么需要将这些转义字符调整成正常的字符，如：“&lt;”代表的就是“<”。

举例说明：

表7-1 xml 返回参数签名示例

示例	参数名	参数值
<pre>&lt;response&gt;   &lt;alipay&gt;     &lt;a&gt;111&lt;/a&gt;     &lt;b&gt;222&lt;/b&gt;   &lt;/alipay&gt; &lt;/response&gt;</pre>	a	111
	b	222

### 7.1.2 生成待签名字符串

对于如下的参数数组：

```
string[] parameters={
    "service=alipay.acquire.refund",
    "partner=2088101568338364",
    "_input_charset=utf-8",
    "refund_amount=200",
    "out_trade_no=HZ0120131127001"
};
```

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。

排序完成之后，再把所有数组值以 “&” 字符连接起来，如：

```
_input_charset=utf-8&out_trade_no=HZ0120131127001&partner=2088101568338364&refund_amount=200&service=alipay.acquire.refund
```

这串字符串便是待签名字符串。



注意：

- 没有值的参数无需传递，也无需包含到待签名数据中；
- 签名时将字符转化成字节流时指定的字符集与\_input\_charset 保持一致；
- 如果传递了\_input\_charset 参数，这个参数也应该包含在待签名数据中；
- 根据 HTTP 协议要求，传递参数的值中如果存在特殊字符（如：&、@等），那么该值需要做 URL Encoding，这样请求接收方才能接收到正确的参数值。这种情况下，待签名数据应该是原始值而不是 encoding 之后的值。例如：调用某接口需要对请求参数 email 进行数字签名，那么待签名数据应该是 email=test@msn.com，而不是 email=test%40msn.com。

## 7.2 签名

### 7.2.1 MD5 签名

在MD5 签名时，需要私钥参与签名。MD5 的私钥是以英文字母和数字组成的 32 位字符串。商户可登录到商户服务中心（<https://b.alipay.com>），安装数字证书，在“技术服务”栏目中点击“交易安全校验码”，即可查看。

- 请求时签名



当拿到请求时的待签名字符串后，需要把私钥直接拼接对待签名字符串后面，形成新的字符串，利用 MD5 的签名函数对这个新的字符串进行签名运算，从而得到 32 位签名结果字符串（该字符串赋值于参数 sign）。

- **通知返回时验证签名**

当获得到通知返回时的待签名字符串后，同理，需要把私钥直接拼接对待签名字符串后面，形成新的字符串，利用 MD5 的签名函数对这个新的字符串进行签名运算，从而得到 32 位签名结果字符串。此时这个新的字符串需要与支付宝通知返回参数中的参数 sign 的值进行验证是否相等，来判断签名是否验证通过。

## 7.2.2 DSA、RSA 签名

在 DSA 或 RSA 的签名时，需要私钥和公钥一起参与签名。私钥与公钥皆是客户通过 OPENSSL 来生成得出的。客户把生成出的公钥与支付宝技术人员配置好的支付宝公钥做交换。因此，在签名时，客户要用到的是客户的私钥及支付宝的公钥。

- **请求时签名**

当拿到请求时的待签名字符串后，把待签名字符串与客户的私钥一同放入 DSA 或 RSA 的签名函数中进行签名运算，从而得到签名结果字符串。

- **通知返回时验证签名**

当获得到通知返回时的待签名字符串后，把待签名字符串、支付宝提供的公钥、支付宝通知返回参数中的参数 sign 的值三者一同放入 DSA 或 RSA 的签名函数中进行非对称的签名运算，来判断签名是否验证通过。

## 7.2.3 返回参数验签特别说明

同步返回参数中，如果返回了 refund\_detail\_item\_list 参数，在验签拼接字符串时，需要将整个 refund\_detail\_item\_list 作为节点的值放入。

例如：同步返回参数如下。

```
<?xml version="1.0" encoding="GBK"?>
<alipay>
  <is_success>T</is_success>
  <request>
    <param name="trade_no">2015031621001004530000075612</param>
    <param name="operator_id">8888</param>
    <param name="sign_type">MD5</param>
    <param name="notify_url">http://api.test.alipay.net/atinterface/
receive_notify.htm</param>
    <param name="out_trade_no">wavepay20150316172615</param>
    <param name="operator_type">1</param>
    <param name="sign">196701af9bdc17838aeecce433d3b4ab</param>
```

```

<param name="_input_charset">GBK</param>
<param name="service">alipay.acquire.refund</param>
<param name="out_request_no">rdm6652015-03-16172624</param>
<param name="partner">2088101145404570</param>
<param name="refund_reason">退款测试</param>
<param name="refund_amount">60</param>
<param name="alipay_ca_request">2</param>
</request>
<response>
  <alipay>
    <buyer_logon_id>ash***@rayy.me</buyer_logon_id>
    <buyer_user_id>2088102145405536</buyer_user_id>
    <fund_change>Y</fund_change>
    <gmt_refund_pay>2015-03-16 17:26:26</gmt_refund_pay>
    <out_trade_no>wavepay20150316172615</out_trade_no>
    <refund_detail_item_list>
      <TradeFundBill>
        <amount>10.00</amount>
        <fund_channel>ALIPAYACCOUNT</fund_channel>
      </TradeFundBill>
      <TradeFundBill>
        <amount>50.00</amount>
        <fund_channel>MCARD</fund_channel>
      </TradeFundBill>
    </refund_detail_item_list>
    <refund_fee>60.00</refund_fee>
    <result_code>SUCCESS</result_code>
    <trade_no>2015031621001004530000075612</trade_no>
  </alipay>
</response>
<sign>09cfdc67ec89b4f0a7ebbc23de570a68</sign>
<sign_type>MD5</sign_type>
</alipay>

```

针对以上同步返回参数，验签时针对 refund\_detail\_item\_list 节点的值应该是：

```

refund_detail_item_list=<refund_detail_item_list><TradeFundBill><amount>10.00</amount><fund_channel>ALIPAYACCOUNT</fund_channel></TradeFundBill><TradeFundBill><amount>50.00</amount><fund_channel>MCARD</fund_channel></TradeFundBill></refund_detail_item_list>

```

最终验签的字符串是：

```

buyer_logon_id=ash***@rayy.me&buyer_user_id=2088102145405536&fund_change=Y&gmt_refund_pay=2015-03-16
17:26:26&out_trade_no=wavepay20150316172615&refund_detail_item_list=<re

```

```
fund_detail_item_list><TradeFundBill><amount>10.0</amount><fund_channel>
ALIPAYACCOUNT</fund_channel></TradeFundBill><TradeFundBill><amount>
50.00</amount><fund_channel>MCARD</fund_channel></TradeFundBill></ref
und_detail_item_list>&result_code=SUCCESS&refund_fee=60.00&trade_no=20
15031621001004530000075612
```

## 8 附录

### 8.1 业务错误码

表8-1 业务错误码

详细错误代码（detail_error_code）	含义（detail_error_des）
INVALID_PARAMETER	参数无效
TRADE_ROLE_ERROR	没有该笔交易的退款或撤销权限
DISCORDANT_REPEAT_REQUEST	同一笔退款或撤销单号金额不一致
TRADE_HAS_CLOSE	交易已经关闭
REASON_TRADE_BEEN_FREEZEN	交易已经被冻结
BUYER_ERROR	买家不存在
SELLER_ERROR	卖家不存在
TRADE_NOT_EXIST	交易不存在
TRADE_STATUS_ERROR	交易状态不合法
TRADE_HAS_FINISHED	交易已结束
REFUND_AMT_NOT_EQUAL_TOTAL	撤销或退款金额与订单金额不一致
SELLER_BALANCE_NOT_ENOUGH	卖家余额不足

## 8.2 接入错误码

表8-2 接入错误码

错误代码 (error_code)	含义
ILLEGAL_SIGN	签名不正确
ILLEGAL_DYN_MD5_KEY	动态密钥信息错误
ILLEGAL_ENCRYPT	加密不正确
ILLEGAL_ARGUMENT	参数不正确
ILLEGAL_SERVICE	Service 参数不正确
ILLEGAL_USER	用户 ID 不正确
ILLEGAL_PARTNER	合作伙伴 ID 不正确
ILLEGAL_EXTERFACE	接口配置不正确
ILLEGAL_PARTNER_EXTERFACE	合作伙伴接口信息不正确
ILLEGAL_SECURITY_PROFILE	未找到匹配的密钥配置
ILLEGAL_AGENT	代理 ID 不正确
ILLEGAL_SIGN_TYPE	签名类型不正确
ILLEGAL_CHARSET	字符集不合法
HAS_NO_PRIVILEGE	无权访问
INVALID_CHARACTER_SET	字符集无效

## 8.3 系统错误码



注意：

当出现系统错误提示时，请联系支付宝技术支持协助处理。

表8-3 系统错误码

错误代码 (error_code)	含义
SYSTEM_ERROR	支付宝系统错误
SESSION_TIMEOUT	session 超时
ILLEGAL_TARGET_SERVICE	错误的 target_service
ILLEGAL_ACCESS_SWITCH_SYSTEM	partner 不允许访问该类型的系统
EXTERFACE_IS_CLOSED	接口已关闭

## 8.4 退款渠道

表8-4 退款渠道

退款渠道代码	退款渠道
ALIPAYACCOUNT	支付宝余额
POINT	支付宝积分
COUPON	支付宝红包
BANKCARD	银行卡
DISCOUNT	支付宝优惠
PCARD	预付卡
FINANCEACCOUNT	支付宝理财专户
TMALLFUND	天猫宝
MCARD	商户资金卡
MDISCOUNT	商户优惠券
MCOUPON	商户红包