



移动快捷支付接口

服务名称：alipay

版本号：1.2

支付宝（中国）网络技术有限公司 版权所有

目 录

1 文档说明	4
1.1 功能描述	4
1.2 阅读对象	4
1.3 业务术语	4
2 功能演示	5
2.1 快捷支付业务流程	5
2.2 快捷支付数据流程	5
3 接口调用方式	6
3.1 iOS	6
3.1.1 调用方式	6
3.1.2 回调接口	6
3.2 Android	7
4 请求参数说明	9
4.1 含义	9
4.2 列表	9
4.3 样例	11
5 同步通知参数说明	11
5.1 含义	11
5.2 列表	11

5.3 样例	12
5.4 同步通知参数获取	12
6 服务器异步通知参数说明	13
6.1 含义	13
6.2 列表	13
6.3 样例	17
6.4 服务器异步通知参数获取	18
7 商户业务处理注意事项	18
8 签名机制	19
8.1 生成待签名的字符串	19
8.1.1 需要参与签名的参数	19
8.1.2 生成待签名字符串	19
8.2 RSA 签名	20
9 建议与意见	21
10 附录	21
10.1 服务端错误码	21
10.2 客户端错误码	21
10.3 渠道类型说明与币种列表	22

1 文档说明

1.1 功能描述

快捷支付服务是安装在本地手机操作系统上的一个组件，主要用来向其它的应用程序提供便捷、安全以及可靠的支付服务。正如手机操作系统上所提供的其它服务。

通过快捷支付接口，商户可以安全快捷的通过手机客户端收款。

1.2 阅读对象

本文档面向具有一定手机客户端开发能力，了解手机客户端的开发和管理人员。

1.3 业务术语

表1-1 业务术语

术语	解释
返回	支付宝以字符串形式直接把处理结果数据返回给手机客户端。
快捷支付客户端接口 API	快捷支付客户端插件包提供的接入快捷支付的 API 接口（alipay），通过此接口其他手机 APP 应用唤起快捷支付客户端插件，不同的快捷支付业务产品调用此 API 接口传递的参数值不同，具体业务产品的参数值可以参考具体客户端业务请求参数说明。
请求	手机客户端以字符串形式把需要传输的数据发送给接收方的过程。
通知	服务器异步通知。支付宝根据得到的数据处理完成后，支付宝的服务器主动发起通知给商户的网站，同时携带处理完成的结果信息反馈给商户网站。
移动快捷支付服务端	移动快捷支付服务端是移动快捷支付产品对应的一系列后台系统，与移动快捷支付客户端相对应，为移动快捷支付整个产品提供服务。
移动快捷支付客户端	快捷支付客户端是安装在本地手机操作系统上的一个组件，主要用来向其它的应用程序提供便捷、安全以及可靠的支付服务。正如手机操作系统上所提供的其它服务。 通过快捷支付接口，商户可以安全快捷的通过手机客户端完成收款、快捷登录等业务。

2 功能演示

2.1 快捷支付业务流程

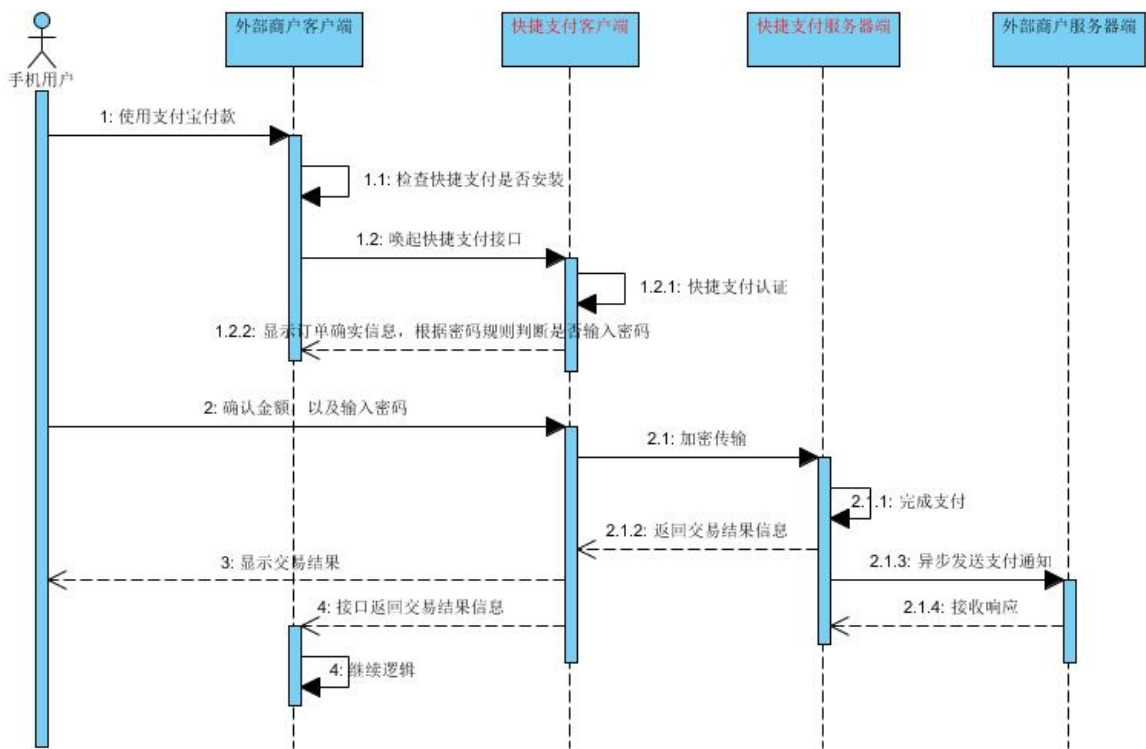


图2-1 快捷支付业务流程图

流程说明：

- “1.2:唤起快捷支付接口” 消息就是本接口所描述的“移动快捷支付接口 alipay.pay”，商户APP通过此API接口唤起快捷支付插件，具体参见“4 请求参数说明”。
- “2.1.3:异步发送支付通知”即快捷支付服务器端发送异步通知消息给商户服务器端，参见“6 服务器异步通知参数说明”。
- “4:接口返回交易结果信息” 即快捷支付客户端接口alipay.pay同步返回信息给商户客户端，参见“5 同步通知参数说明”。

2.2 快捷支付数据流程

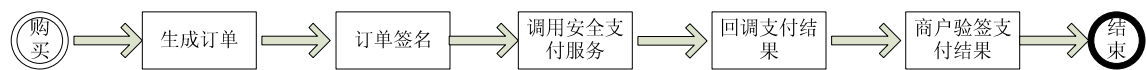


图2-2 快捷支付数据流程图

3 接口调用方式

3.1 iOS

3.1.1 调用方式

在 iOS 平台上使用移动快捷支付完成支付。使用系统 `UIApplication` 的 `openURL` 方法来唤起移动快捷支付应用，并传递参数。该方法描述如下：

方法原型	<code>-(BOOL)openURL:(NSURL *)url</code>
方法功能	通过指定 url 地址来打开一个网页或唤起应用。
参数	<p>NSURL url 主要包含移动快捷支付客户端的 <code>scheme</code> 和商户的订单信息。</p> <p>示例如下：</p> <pre>safepay://alipayclient?{"requestType":"SafePay","dataString":"","fromAppUrlScheme":"Safari"}</pre> <p>其中：</p> <p><code>dataString</code> 为商户的订单信息。字段以 <code>key = "value"</code> 形式展示，通过 <code>&</code> 连接。示例如下：</p> <pre>partner="2088002007260245"&seller="2088002007260245"&out_trade_no="5000000000006548"&subject="商品名称"&body="这是商品描述"&total_fee="30"&notify_url="http://notify.java.jpxx.org/index.jsp"&sign="kU2Fa3x6V985g8ayTozIleJ5fHtm8%2FJGeJQf9in%2BcVmRJjHaExbirnGGKJ%2F7B63drqc4Kjlk%2FSg6vtSIkOtdvVBrRDpYaKxXVqkJTzRYgUwrrpMudbIj9aMS2O3dHG0GPyL4Zb6jKDYXHabGG0aBJY3QA7JuTJ23t6SqV%2B5flxg%3D"&sign_type="RSA"</pre> <p>各个字段的含义详见“4 请求参数说明”。</p> <p><code>fromAppUrlScheme</code> 为商户应用的 <code>scheme</code>，移动快捷支付客户端在支付过程结束后，通过这个标识回调目标应用。</p>

3.1.2 回调接口

在支付过程结束后，移动快捷支付客户端会通过商户在 `openURL` 接口中传入的 `fromAppUrlScheme` 回调商户的应用。

- 在 iOS 4.2 之前，回调的函数是：

```
-(BOOL)application : (UIApplication)application handleOpenURL : (NSURL *)url;
```

- 在 iOS 4.2 之后，回调的函数是：

```
-(BOOL)application : (UIApplication)application openURL : (NSURL *)url  
sourceApplication : (NSString *)sourceApplication annotation:(id)annotation
```

返回内容放在url中，详细信息详见“5 同步通知参数说明”。

3.2 Andriod

在 Android 上使用移动快捷支付的 **IAlixPay** 接口完成支付。移动快捷支付实现为一个 **Service**，为其他应用提供支付服务。接口描述如下：

方法名称	方法描述
Pay	为商户提供订单支付功能。

Pay 方法信息描述如下：

方法原型	String Pay(String info);
方法功能	商户订单支付。
参数	<p>String info 主要包含商户的订单信息。字段以 key = "value"形式展示，通过&连接。示例如下：</p> <pre>partner="2088002007260245"&seller="2088002007260245"&out_trade_no="500000000006548"&subject=" 商 品 名 称 "&body=" 这 是 商 品 描 述 "&total_fee="30"&notify_url="http://notify.java.jpaxx.org/index.jsp"&sign="kU2Fa3x6V985g8ayTozIleJ5fHtm8%2FJGeJQf9in%2BcVmRJjHaExbirnGGKJ%2F7B63drqc4Kjlk%2FSg6vtSikOtdvVBrRDpYaKxXVqkJTzRYgUwrrpMudbIj9aMS2O3dHG0GPyL4Zb6jKDYXHabGG0aBJY3QA7JuTJ23t6SqV%2B5flxg%3D"&sign_type="RSA"</pre> <p>各个字段的含义详见“4 请求参数说明”。</p>
返回值	本方法调用的结果。字符串格式。

移动快捷支付的调用流程和Android系统操作**Service**的流程是一致的，具体流程如图 3-1所示。

(1) 绑定快捷支付服务

绑定快捷支付服务，初始化 **IAlixPay** 接口。

(2) 注册回调（可选）

移动快捷支付通过调用特定的回调来保证其展示的收银台界面和调用方的 **Activities** 存在于同一个 **Back stack**，此步骤为可选。

(3) 调用支付接口

调用 **IAlixPay.Pay** 接口完成支付，此接口为同步接口，支付结果会通过返回值返回给调用方。详细信息参见“5 同步通知参数说明”。

(4) 反注册回调，解除快捷支付服务绑定。

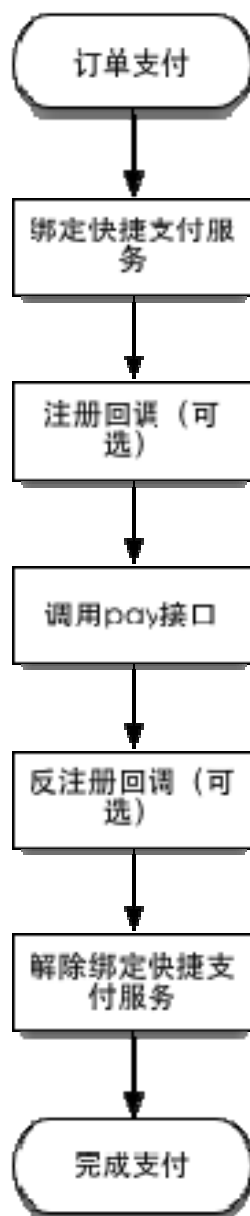


图3-1 移动快捷支付调用流程

4 请求参数说明

4.1 含义

请求参数是商户在与支付宝客户端进行数据交互时，提供给支付宝客户端的请求数据，以便支付宝客户端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					
partner	合作者身份ID	String(16)	商户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。 商户可用签约支付宝账号登录 https://ms.alipay.com 获取。	不可空	2088002007260245
sign_type	签名方式	String	签名类型，目前仅支持 RSA。	不可空	RSA
sign	签名	String	请参见“8 签名机制”。注意此参数要做urlencode。	不可空	kU2Fa3x6V985g8ayTo z11eJ5fHtm8%2FJGeJ Qf9in%2BcVmRJjHaEx birnGGKJ%2F7B63drq c4KjdvVBrdblj9aMS2O 3dHG0GPyL4Zb6jKDY XHabGG0aBJY3QA7Ju TJ23t6SqV%2B5f1xg% 3D
notify_url	服务器异步通知页面路径	String(255)	支付宝服务器主动通知商户网站里指定的页面 http 路径。	不可空	http://notify.java.jpxx.org/index.jsp
业务参数					

参数	参数名称	类型(字节长度)	参数说明	是否可为空	样例
seller	商户收款支付宝用户号	String(16)	<p>商户用于收款的支付宝账号（订单付款成功后，交易款项会转到该账号中）对应的支付宝唯一用户号。</p> <p>以 2088 开头的 16 位纯数字组成。</p> <p>商户可以用签约支付宝账号登录https://ms.alipay.com获取。</p>	不可空	2088002007260245
out_trade_no	商户网站唯一订单号	String(64)	<p>支付宝合作商户网站唯一订单号，商户自定义。</p> <p>最长 64 位字母、数字和下划线组成。</p>	不可空	20120910-0001
subject	商品名称	String(128)	<p>商品的标题/交易标题/订单标题/订单关键字等。</p> <p>由商户定义，可重复，支持中文，最长 64 位字符（即 128 字节）。</p>	不可空	羽毛球拍
body	商品描述	String(2048)	<p>商品的具体描述信息，支持中文，最长 1024 位字符（即 2048 字节）。</p>	不可空	正品纳米科技台湾产
total_fee	交易金额	String	<p>该笔订单的资金总额，单位为 RMB-Yuan。</p> <p>大于 0 的数字，精确到小数点后两位。</p>	不可空	1.5
extern_token	访问令牌	String	<p>如果商户入驻了钱包开放平台，且在调用移动快捷支付接口时，需要用到此访问令牌时传入。该令牌可以指定手机登录时的支付宝账号。</p>	可空	appopenBb759968d131f4594a158d97906c89d56

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；
- subject、body、notify_url 参数的值中，不能存在影响请求或返回的数据结构的特殊字符，如：“”、“&”、“{”、“}”、“+”、“\”等。

4.3 样例

```
partner="2088002007260245"&seller="2088002007260245"&out_trade_no="20120910-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"&sign="kU2Fa3x6V985g8ayTozIleJ5fHtm8%2FJGeJQf9in%2BcVmRJjHaExbirnGGKJ%2F7B63drqc4Kjlk%2FSg6vtSIkOtdvVBrRDpYaKxXVqkJTzRYgUwrrpMudbIj9aMS2O3dHG0GPyL4Zb6jKDYXHabGG0aBJY3QA7JuTJ23t6SqV%2B5f1xg%3D"&sign_type="RSA"
```

5 同步通知参数说明

5.1 含义

支付宝对支付宝客户端提供的请求数据进行处理后，返回给支付宝客户端结果数据，商户从支付宝客户端中获取结果数据进行验证签名，之后商户根据这些数据进一步处理。

5.2 列表

表5-1 客户端同步返回参数说明

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
resultStatus	状态代码	String	本次操作的状态返回值，标识本次调用的结果，参见“10 附录”。	不可空	9000

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
result	结果数据	String	<p>本次操作返回的订单支付结果数据。</p> <p>字符串格式，形式一般如下：</p> <p>partner=""&seller=""&out_trade_no=""&subject=""&body=""&total_fee=""&notify_url=""&success="true"&sign_type="RSA"&sign="xxx"</p> <p>其中：</p> <ul style="list-style-type: none"> • &success="true"&sign_type="RSA"&sign="xxx" 之前的部分为商户的原始请求数据，原样返回。 • success 用来标识本次支付结果（true 或 false）。 • sign="xxx" 为支付宝对本次支付结果（红色部分）的签名，商户可以使用签约时支付宝提供的公钥进行验证。 	不可空	<p>partner="2088002007260245"&seller="2088002007260245"&out_trade_no="20120910-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"&success="true"&sign_type="RSA"&sign="O011APPVQcK5bbSgdeFx9HB3Yu/U2+akTZ3T0/P7v3g7XD7TsQCprb6O9Nybr8CDlrztdUseQN/TCXuEvCU2c"</p>

5.3 样例

```
resultStatus={9000};result={partner="2088002007260245"&seller="2088002007260245"&out_trade_no="20120910-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"&success="true"&sign_type="RSA"&sign="O011APPVQcK5bbSgdeFx9HB3Yu/U2+akTZ3T0/P7v3g7XD7TsQCprb6O9Nybr8CDlrztdUseQN/TCXuEvCU2c"}
```

5.4 同步通知参数获取

手机客户端通过 AliXpay.pay() 接口方法得到的返回。

6 服务器异步通知参数说明

6.1 含义

支付宝对商户的请求数据处理完成后，会将处理的结果数据通过服务器主动通知的方式通知给商户网站。这些处理结果数据就是服务器异步通知参数。

6.2 列表

表6-1 服务器异步通知参数说明

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					
sign	签名	String	请参见“8 签名机制”。 签名类型为固定值 RSA。	不可空	ZPZULntRpJwFmGNIV KwjLEF2Tze7bqs60rxQ 22CqT5J1UlvGo575QK 9z/+p+7E9cOoRoWzqR 6xHZ6WVv3dloyGKDR 0btvrdqPgUAoeaX/YO WzTh00vwcQ+HBtXE+ vPTfAqjCTxiiSJE0Y7A TCF1q7iP3sfQxhS0nD Uug1LP3OLk=
业务参数					

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
notify_data	通知业务数据	String	通知业务数据，xml格式，请参考“表 6-2 notify_data参数说明”。	不可空	<pre> <notify><partner>20882016519021111</partner> <discount>0.00</discount><payment_type>1</payment_type><subject>10000 起点币 </subject><trade_no>201311070318218701</trade_no><buyer_email>15800000000</buyer_email><gmt_create>2013-07-03 09:27:32</gmt_create> <quantity>1</quantity><out_trade_no>398521</out_trade_no><notify_reg_time>2013-07-03 09:32:43.000</notify_reg_time><seller_id>2088002007018916</seller_id> <out_channel_type>OPTIMIZED_MOTO</out_channel_type><trade_status>TRADE_FINISHED</trade_status><is_total_fee_adjust>N</is_total_fee_adjust><total_fee>100.00</total_fee><gmt_payment>2013-07-03 09:27:34</gmt_payment> <seller_email>xxxx@xxxx.com.cn</seller_email><gmt_close>2013-07-03 09:27:34</gmt_close><price>100.00</price><buyer_id>2088002007013600</buyer_id><out_channel_amount>100.00</out_channel_amount> <use_coupon>N</use_coupon></notify> </pre>

表6-2 notify_data 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
trade_status	交易状态	String	交易当前所处的状态，取值范围： <ul style="list-style-type: none"> TRADE_FINISHED：表示交易成功完成 WAIT_BUYER_PAY：表示等待付款 	不可空	TRADE_FINISHED
total_fee	交易金额	String	该笔订单的资金总额，单位为 RMB-Yuan。 大于 0 的数字，精确到小数点后两位。	不可空	100.00
subject	商品名称	String	商品的标题/交易标题/订单标题/订单关键字等。 商户请求数据，原样返回。	不可空	10000 起点币
out_trade_no	商户网站唯一订单号	String	支付宝合作商户网站唯一订单号，商户自定义。 商户请求数据，原样返回。	不可空	398521
notify_reg_time	通知时间	String	通知的发送时间。 格式为“年-月-日时:分:秒.毫秒”，如 “2012-11-1814:02:43.000”。	不可空	2013-07-0309:32:43.000
trade_no	支付宝交易号	String	该交易在支付宝系统中的交易流水号。 最短 16 位，最长 64 位。	不可空	201311070318218701
payment_type	支付类型	String	<ul style="list-style-type: none"> 1：商品购买 8：机票购买 	不可空	1
price	商品单价	String	如果请求时使用是 total_fee，那么 price 等于 total_fee；如果请求时使用是 price，那么对应请求时的 price 参数，原样通知回来。	不可空	100.00

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
quantity	购买数量	String	如果请求时使用的是 total_fee, 那么 quantity 等于 1; 如果请求时使用的是 quantity, 那么对应请求时的 quantity 参数, 原样通知回来。	不可空	1
seller_id	卖家支付宝账户号	String	卖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。	不可空	2088002007018916
seller_email	卖家支付宝账号	String	卖家支付宝账号, 可以是 email 和手机号码。	不可空	xxxx@xxxx.com.cn
buyer_id	买家支付宝账户号	String	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。	不可空	2088002007013600
buyer_email	买家支付宝账号	String	买家支付宝账号, 可以是 email 和手机号码。	不可空	15800000000
discount	折扣	String	折扣金额。支付宝系统会把 discount 的值加到交易金额上, 如果需要折扣, 本参数为负数。	可空	0.00
use_coupon	是否使用红包	String	是否在交易过程中使用了红包。	可空	N
is_total_fee_adjust	交易金额是否修改过	String	该交易是否调整过价格。	可空	N
gmt_create	交易创建时间	String	该笔交易创建的时间。 格式为 yyyy-MM-dd HH:mm:ss。	可空	2013-07-03 09:27:32
gmt_payment	买家付款时间	String	该笔交易的买家付款时间。 格式为 yyyy-MM-dd HH:mm:ss。	可空	2013-07-03 09:27:34
gmt_close	交易结束时间	String	交易关闭时间。 格式为 yyyy-MM-dd HH:mm:ss。	可空	2013-07-03 09:27:34

参数	参数名称	类型（长度范围）	参数说明	是否可为空	样例
out_channel_type	支付渠道组合信息	String	该笔交易所使用的支付渠道。 格式为：渠道 1 渠道 2 ...，如果有多个渠道，用“ ”隔开。 取值范围请参见附录“10.3 渠道类型说明与币种列表”。	可空	OPTIMIZED_MOTO
out_channel_amount	支付金额组合信息	String	该笔交易通过使用各支付渠道所支付的金额。 格式为：金额 1 金额 2 ...，如果有多个支付渠道，各渠道所支付金额用“ ”隔开。	可空	100.00

6.3 样例

```
http://xxxx.com/Alipay/Notify.ashx?notify_data=<notify><partner>20882016519021111</partner><discount>0.00</discount><payment_type>1</payment_type><subject>10000 起点币</subject><trade_no>201311070318218701</trade_no><buyer_email>15800000000</buyer_email><gmt_create>2013-07-03 09:27:32</gmt_create><quantity>1</quantity><out_trade_no>398521</out_trade_no><notify_reg_time>2013-07-03 09:32:43.000</notify_reg_time><seller_id>2088002007018916</seller_id><out_channel_type>OPTIMIZED_MOTO</out_channel_type><trade_status>TRADE_FINISHED</trade_status><is_total_fee_adjust>N</is_total_fee_adjust><total_fee>100.00</total_fee><gmt_payment>2013-07-03 09:27:34</gmt_payment><seller_email>xxxx@xxxx.com.cn</seller_email><gmt_close>2013-07-03 09:27:34</gmt_close><price>100.00</price><buyer_id>2088002007013600</buyer_id><out_channel_amount>100.00</out_channel_amount><use_coupon>N</use_coupon></notify>&sign=ZPZULntRpJwFmGNIVKwjLEF2Tze7bqs60rxQ22CqT5J1UlvGo575QK9z/+p+7E9cOoRoWzqR6xHZ6WVv3dloyGKDR0btvrdqPgUAoeaX/YOWzTh00vwcQ+HBtXE+vPTfAqjCTxiisJEOY7ATCF1q7iP3sfQxhS0nDUug1LP30Lk=
```

说明：

本样例仅供参考，实际网关为商户域名。

6.4 服务器异步通知参数获取

- (1) 必须保证服务器异步通知页面（`notify_url`）上无任何字符，如空格、HTML 标签、开发系统自带抛出的异常提示信息等；
- (2) 支付宝是用 **POST** 方式发送通知信息，因此该页面中获取参数的方式，如：
`request.Form("out_trade_no")`、`$_POST['out_trade_no']`；
- (3) 支付宝主动发起通知，该方式才会被启用；
- (4) 服务器间的交互，不像页面跳转同步通知可以在页面上显示出来，这种交互方式是不可见的；
- (5) 程序执行完后必须打印输出“**success**”（不包含引号）。如果商户反馈给支付宝的字符不是 **success** 这 7 个字符，支付宝服务器会不断重发通知，直到超过 24 小时 22 分钟。
一般情况下，25 小时以内完成 8 次通知（通知的间隔频率一般是：
`2m,10m,10m,1h,2h,6h,15h`）；
- (6) 程序执行完成后，该页面不能执行页面跳转。如果执行页面跳转，支付宝会收不到 **success** 字符，会被支付宝服务器判定为该页面程序运行出现异常，而重发处理结果通知；
- (7) **cookies**、**session** 等在此页面会失效，即无法获取这些数据；
- (8) 该方式的调试与运行必须在服务器上，即互联网上能访问；
- (9) 该方式的作用主要防止订单丢失，即页面跳转同步通知没有处理订单更新，它则去处理；
- (10) 当商户收到服务器异步通知并打印出 **success** 时，服务器异步通知参数 **notify_id** 才会失效。也就是说在支付宝发送同一条异步通知时（包含商户并未成功打印出 **success** 导致支付宝重发数次通知），服务器异步通知参数 **notify_id** 是不变的。

7 商户业务处理注意事项

当支付宝处理完成后会把数据结果反馈给商户。商户获得这些数据时，必须依据服务器异步通知，且判断是否已经对该次的通知结果数据做过同样处理。未按此规范处理，商户需自行承担因此而产生的所有损失。

通过 **OpenSSL** 生成的签名和内置的算法可以做到跨平台，这样在不同的开发语言中均可以签名和验签。

8 签名机制

8.1 生成待签名的字符串

8.1.1 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。
- 在返回参数列表中，只对 `result` 参数中 `partner`、`seller`、`out_trade_no`、`subject`、`body`、`total_fee`、`notify_url`、`success` 这几个参数做签名。
- 在通知参数列表中，除去 `sign` 参数外，凡是通知返回回来的参数皆是要签名的参数，本接口中只有 `notify_data` 这一个参数。

8.1.2 生成待签名字符串

1. 请求

对于如下的参数集合：

```
Map<String, String> parameters = new HashMap<String, String>()
parameters.put("partner", "2088002007260245");
parameters.put("seller", seller);
parameters.put("out_trade_no", out_trade_no);
parameters.put("subject", subject);
parameters.put("body", body);
parameters.put("total_fee", total_fee);
parameters.put("notify_url", notify_url);
```

不需要对集合里的所有参数做排序，这些集合里的参数也没有顺序要求。

直接把集合里的所有元素按照“**key 值=value 值**”的格式拼接起来，并且把这些拼接以后的元素以“&”字符再连接起来，如：

```
partner="2088002007260245"&seller="2088002007260245"&out_trade_no="201209
10-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产
"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"
```

这段字符串即是待签名字符串。

2. 客户端同步返回

获取返回参数 `result` 的值，如：

```
{partner="2088002007260245"&seller="2088002007260245"&out_trade_no="20120
910-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产
```

```
"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"&success="true"&sign_type="RSA"&sign="O011APPVQcK5bbSgdeFx9HB3Yu/U2+akTZ3T0/P7v3g7XD7TsQCprb6O9Nybr8CDI rztdUseQN/TCXuEvCU2c" }
```

从中截取出参数 **success** 之前的所有数据，且去掉特殊符号 “{”、“}”，如下：

```
partner="2088002007260245"&seller="2088002007260245"&out_trade_no="20120910-0001"&subject="羽毛球拍"&body="正品纳米科技台湾产"&total_fee="1.5"&notify_url="http://notify.java.jpxx.org/index.jsp"
```

这段字符串即是待签名字符串。也就是说截取出来的数据不需要做任何处理，如排序、拼接等。

3. 服务器异步通知

获取返回参数 **notify_data** 的信息，如：

```
notify_data=<notify><trade_status>TRADE_FINISHED</trade_status><total_fee>1.5</total_fee><subject>羽毛球拍</subject><out_trade_no>20120910-0001</out_trade_no><notify_reg_time>2012-11-18 14:02:43.000</notify_reg_time><trade_no>2010111800209965</trade_no></notify>
```

这串字符串便是待签名字符串。也就是说直接对“**notify_data=其值**”做验证签名，该处不需要做任何处理，如排序、拼接等。



注意：

- **key 值="value 值"**格式中双引号不能省略；
- 没有值的参数无需传递，也无需包含到待签名数据中；
- 签名时需要符合 url 编码规则，并且字符编码为 UTF-8。

8.2 RSA签名

在 RSA 签名或验证签名时，需要私钥或公钥参与。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，在签名或验证签名时，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 RSA 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 **RSA 验证签名函数** 中进行非对称的验证签名运算，来判断签名是否验证通过。

9 建议与意见

如果您在阅读技术文档时有任何建议和意见，请点击下面的链接跟帖回复，我们将根据您的反馈完善文档内容。

<http://club.alipay.com/read.php?tid=8684674&fid=703>

10 附录

10.1 服务端错误码

表10-1 服务端错误码

状态代码（resultStatus）	状态描述
9000	登录成功
4000	系统异常

10.2 客户端错误码

表10-2 客户端错误码

状态代码（resultStatus）	状态描述
9000	登录成功
4000	系统异常
6001	用户中途取消

10.3 渠道类型说明与币种列表

表10-3 渠道类型说明与币种列表

渠道类型	描述	卡种
BALANCE	余额	无
CREDIT_PAY	信用支付支付方式类型	无
CASH	现金支付方式类型	无
CONSUMER_CARD	消费卡支付方式类型	无
COUPON	红包支付方式类型	无
VOUCHER	购物券支付方式类型	无
POINT	积分支付方式类型	无
WANG_HUI_E	网汇 E 支付方式类型	无
PREPAID_CARD	预存卡支付方式类型	无
INTERNATIONAL_CREDIT_CARD	国际卡类型	无
PREPAY	预付卡支付方式类型	无
RT_DISCOUNT	实时优惠支付方式类型	无
CARTOON	借记卡卡通支付方式类型	借记卡
SC_DEBIT_CARTOON	结算中心借记卡卡通支付方式类型	借记卡
B2C_EBANK	借记卡 B2C 网银支付方式类型	借记卡
B2B_EBANK	B2B 网银支付方式类型	借记卡
WANGDIAN_DEBIT_CARD	网点借记卡刷卡支付方式类型	借记卡
DEBIT_EXPRESS	借记卡快捷支付方式类型	借记卡
CREDIT_CARTOON	普通信用卡卡通支付方式类型	信用卡
BIGAMOUNT_CREDIT_CARTOON	大额信用卡卡通支付方式类型	信用卡

渠道类型	描述	卡种
VISA	VISA 网银支付方式类型	信用卡
CREDIT_CARD_EBANK	信用卡网关支付方式类型	信用卡
MOTO_CREDIT_CARD	moto 信用卡支付方式类型	信用卡
OPTIMIZED_MOTO	信用卡快捷支付类型	信用卡
CREDIT_EXPRESS_INSTALLMENT	信用卡快捷分期支付方式类型	信用卡
WANGDIAN_CREDIT_CARD	网点信用卡刷卡支付方式类型	信用卡
INTERNATIONAL_CREDIT_CARD_VISA	VISA 支付方式类型	信用卡
INTERNATIONAL_CREDIT_CARD_MASTER	MASTER 支付方式类型	信用卡
INTERNATIONAL_CREDIT_CARD_JCB	JCB 支付方式类型	信用卡
MIXED_B2C_EBANK	混合卡 B2C 网银支付方式类型	混合卡