安全支付服务器端

开发指南

文件版本：2.0.2

2012-05-04

前言

1. 面向读者

本文档主要面向需要接入支付宝安全支付的商户的开发人员。

2. 读者所需技能

读者需有 web 端编程经验，支持 http 协议的 web 端开发语言，例如主流的 Java、C#、PHP 等开发语言

3. 开发环境要求

具体视开发语言所决定

目录

第一章安全支付服务简介

1.1 安全支付服务介绍

安全支付服务主要用来向第三方应用程序提供便捷、安全以及可靠的支付服务。本文主要描述安全支付服务应用开发接口的使用方法，供合作伙伴的开发者接入使用。

1.2 安全支付服务业务流程

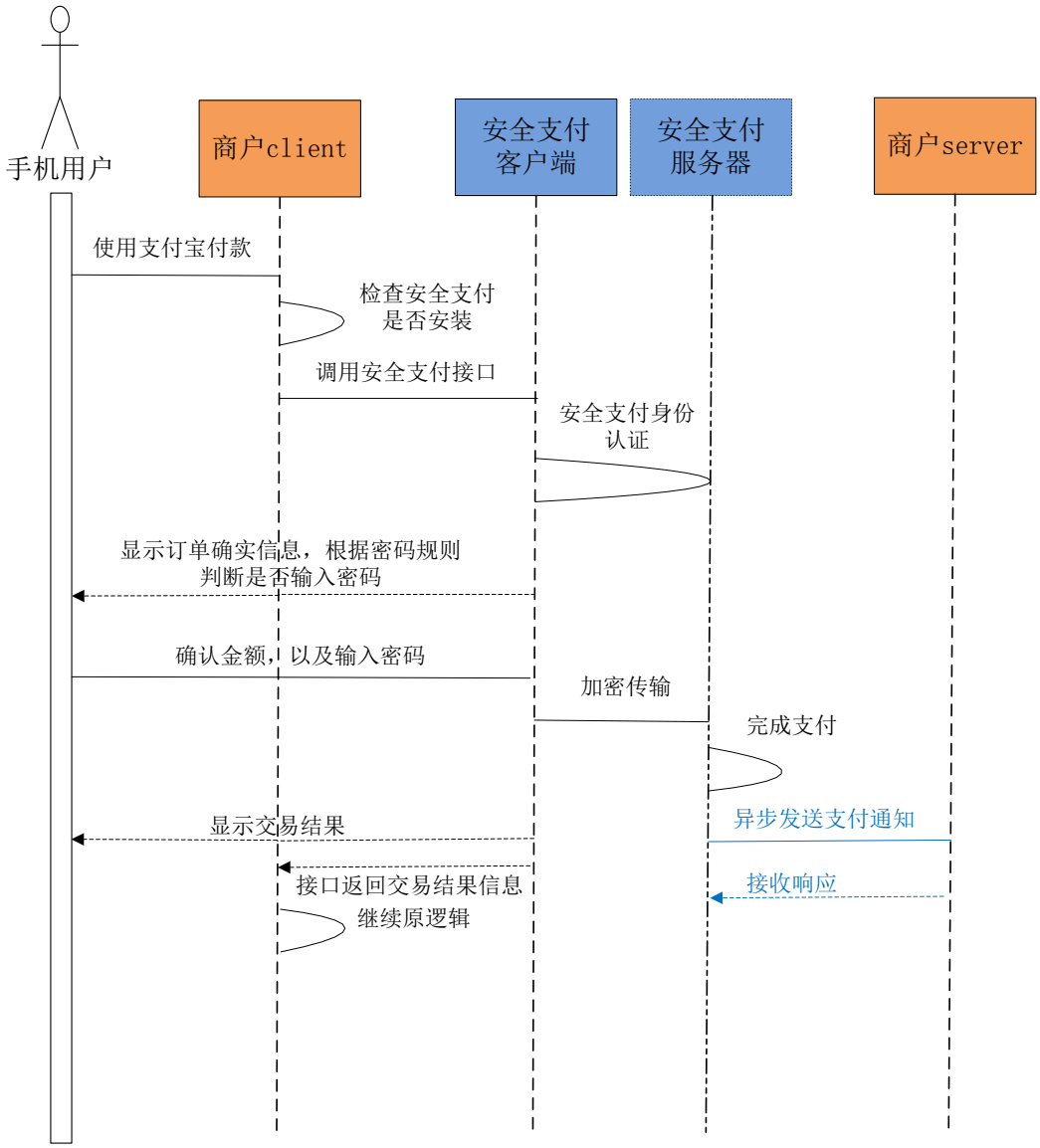

图 1-1 安全支付业务流程图

1.3 调用安全支付数据流程图

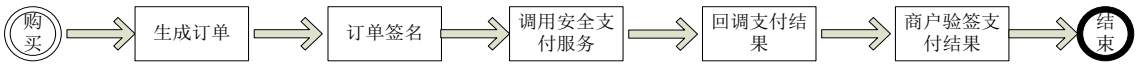

图 1-2 安全支付数据流程图

第二章安全支付接入流程

2.1 接入前期准备

接入前期准备工作包括**商户签约**和**密钥配置**，已完成商户可略过。

2.1.1 商户签约

首先，商户需要在 https://ms.alipay.com 进行注册，并签约安全支付服务。签约成功后可获取支付宝分配的合作商户 ID(PartnerID)，账户 ID(SellerID)，如图：

图 2-1 商户 ID 获取示意图

签约过程中需要任何帮助请致电：0571-88158090（支付宝商户服务专线）

2.1.2 密钥配置

签约成功后，商户可登陆 https://ms.alipay.com 获取商户账号对应的支付宝公钥，具体获取步骤请见 3.2 RSA 密钥详解

接着，商户生成商户公钥和商户私钥（具体生成步骤请见 3.2 RSA 密钥详解），并登录

https://ms.alipay.com，上传商户公钥（具体上传步骤请见 3.2 RSA 密钥详解）。

2.2Demo

首先需要先配置 Demo 中的配置文件，填上所有必填的商户信息后方能测试。

本 Demo 是需要配合客户端进行测试，首先是由客户端请求服务端，然后服务端生成签名和其他数据返回给客户端，由客户端最后调用安全支付提交给支付宝。为了安全起见，所以商户的密钥及其他敏感信息推荐保存在服务端，签名和验签方法也在服务端实现。

第三章 RSA 详解

*以下内容加*号为重点*

3.1 RSA 和 OpenSSL 介绍

3.1.1 什么是 RSA

RSA 是一种非对称的签名算法，即签名密钥（私钥）与验签密钥（公钥）是不一样的，私钥用于签名，公钥用于验签。

在与支付宝交易中，会有 2 对公私钥，即商户公私钥，支付宝公私钥。

3.1.2 为什么要用 RSA

使用这种算法可以起到防止数据被篡改的功能，保证支付订单和支付结果不可抵赖(商户私钥只有商户知道)。

3.1.3 什么是 OpenSSL

一句话概括：OpenSSL 是基于众多的密码算法、公钥基础设施标准以及 SSL 协议安全开发包。

3.1.4 为什么要用 OpenSSL

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。

3.2 RSA 密钥详解*

3.2.1 找到生成 RSA 密钥工具

1. 下载开发指南和集成资料，如下图，您能看到此文档说明指南和集成包已经下载了。

图 3- 1 文档下载

2. 解压下载的压缩包(WS_SECURE_PAY)，找到并解压 openssl-0.9.8k_WIN32(RSA 密钥生成工具).zip 工具包

图 3- 2openssl

3.2.2 生成密钥并获取支付宝公钥

(1) 生成 RSA 私钥 PEM 文件

假设您解压后的目录在 C:\alipay 目录下，命令行执行"*openssl genrsa -out rsa_private_key.pem 1024*"命令生成 rsa_private_key.pem 文件，该文件会生成在 C:\alipay\bin 文件夹下

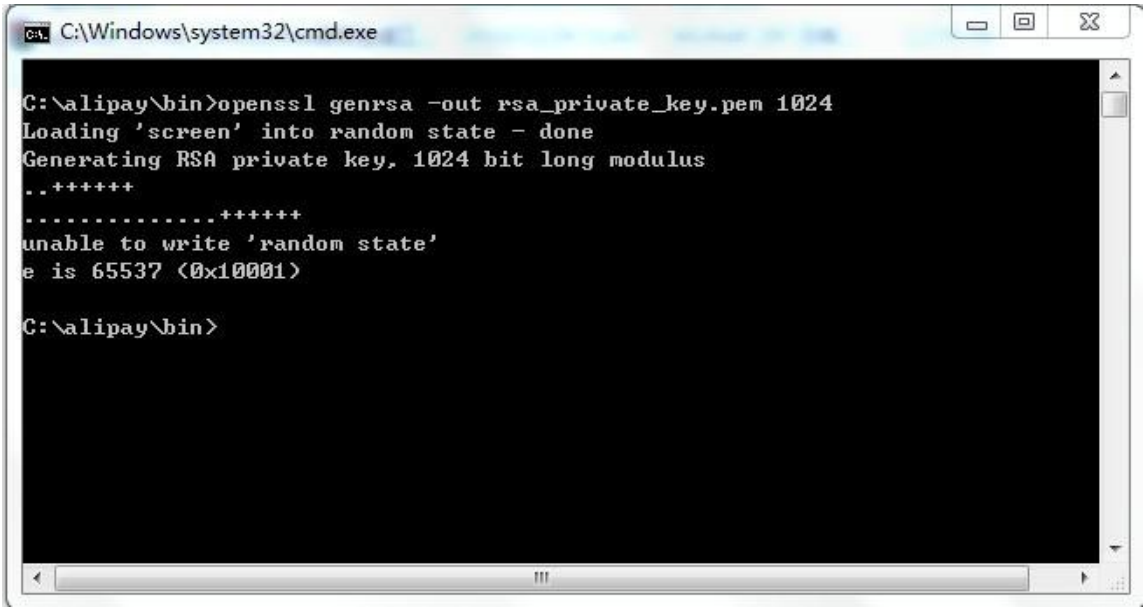

图 3-3 商户私钥生成

(2) 生成公钥，命令行执行"*openssl rsa -in rsa_private_key.pem -pubout -out rsa_public_key.pem*"命令生成 rsa_public_key.pem 文件，该文件会生成在 C:\alipay\bin 文件夹下。

(3) 将 RSA 私钥转换成 PKCS8 格式，命令行执行"*openssl pkcs8 -topk8 -inform PEM -in rsa_private_key.pem -outform PEM -nocrypt*"命令得到下图红色方框内的私钥，并 COPY 出来保管好，在支付的时候需要使用到。

PHP 服务端语言读取私钥不需要 PKCS8 转换，该步骤可以省略。

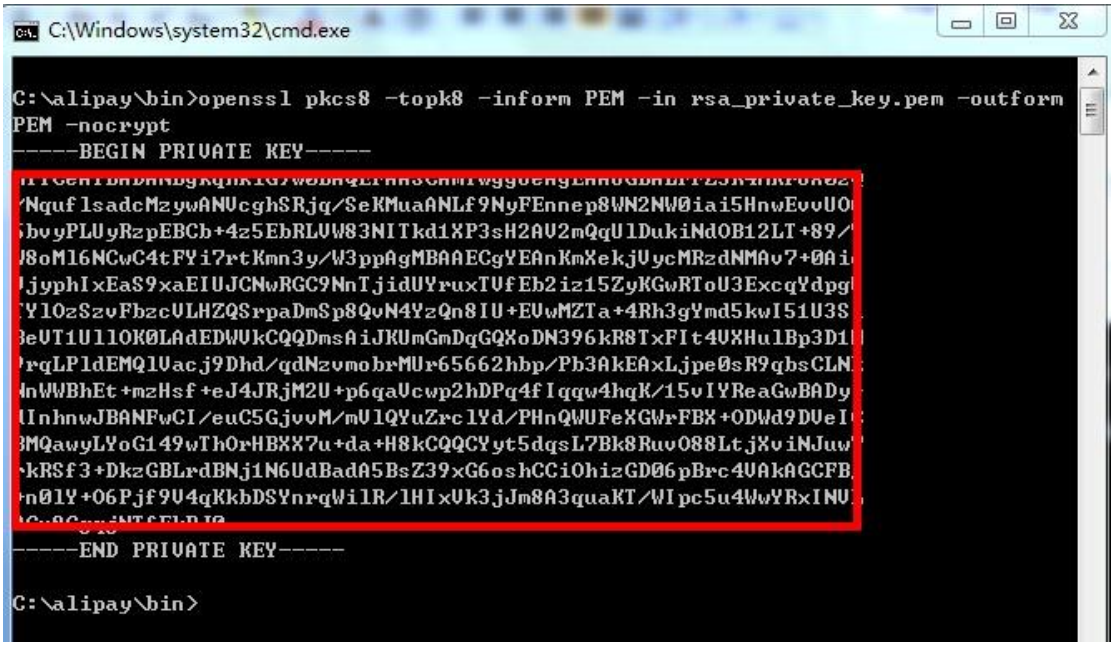

图 3- 4 pkcs8 转换

上面(2)和(3)生成的公钥和私钥是商户自己的公钥和私钥，将商户的公钥复制到 TXT 文本文件中，删除文件头”-----BEGIN PUBLIC KEY-----“与文件尾”-----END PUBLIC KEY-----“还有空格、换行，如下图变成一行字符串并保存该 TXT 文件

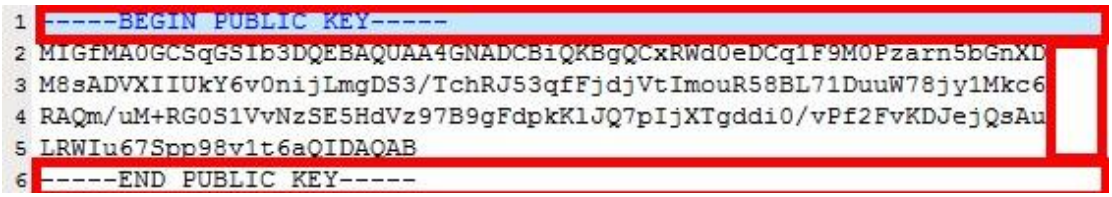

图 3- 5 公钥格式修改

(4) 将该 TXT 文件提交到支付宝无线签约平台，详细步骤是浏览器访问 https://ms.alipay.com/index.htm 并用签约帐号登录，点击菜单栏”我的产品”，右侧点击”密钥管理”，见下图红色框内

图 3- 6 公钥上传

(5) 浏览本地 TXT 文件并上传，并且复制出支付宝公钥保存，见下图

备注：获取支付宝公钥时字符串中含有空格，程序配置中需先把空格删除

图 3- 7 支付宝公钥获取

3.3 RSA 签名和验签 *

建议：签名和验签尽量在商户服务器端进行，同时一些敏感数据（如公私钥等）也应存储在服务器端，避免可能的安全隐患。

3.3.1 RSA 签名

将商品信息拼接成字符串

例子：出售商品（*subject*）“Iphone4”，价格（*total_fee*）“1”元，外部交易号（*out_trade_no*）“zzzz”，商品描述（*body*）为“秒杀”，订单支付完成通知 URL（*notify_url*）为“http://notify.java.jpxx.org/index.jsp”

则生成如下商品信息字串：

```
partner="xxxx"&seller="yyyy"&out_trade_no="zzzz"&subject="Ipone4"&body="秒杀"&total_fee="1"&notify_url="http%3A%2F%2Fnotify.java.jpxx.org%2Findex.jsp"
```

备注：notify_url 的值需要进行 URLEncode 编码。

对商品信息进行 *RSA* 签名：

将签名字符串和商品信息字符串以及签名类型按格式拼接，调用 demo 中的签名方法。

待签名的字符串数据(下图中红色部分)

签名需要 base64 编码以及 UrlEncode 处理（下图中蓝色部分）

签名后的订单字符串示例：

```
partner="xxxx"&seller="yyyy"&out_trade_no="zzzz"&subject="Ipone4"&body="秒杀"&total_fee="1"&notify_url="http%3A%2F%2Fnotify.java.jpxx.org%2Findex.jsp"&sign_type="RSA"&sign="O0I1APPVQcK5bbSgdeFx9HB3Yu/U2+akTZ3T0/P7v3g7XD7TsQCprb6O9Nybr8CDIrztdUseQN/TCXuEvCU2cvCt1xX9UUyI6fOxXxQFlDWx7IE2S7Zo5wOeVWmMBnCQCV8iDjcNxGHwhtCT09bVVf0wbaOiHXvAYzWlvPhyR+0="
```

3.3.2 RSA 验签（同步返回示例）

验签可以参照demo中的函数实现。

安全支付返回的待验签字符串(下图中红色部分)

安全支付返回的字符串(下图中蓝色部分)

以下是一个订单支付成功完整信息的示例：

```
resultStatus={9000};memo={交易成功};result={partner="2088002007260245"&seller="2088002007260245"&out_trade_no="600000000006891"&subject="商品名称"&body="这是商品描述"&total_fee="1"&notify_url="http%3A%2F%2Fnotify.java.jpxx.org%2F
```

```
index.jsp"&success="true"&sign_type="RSA"&sign="O0I1APPVQcK5bbSgdeF
x9HB3Yu/U2+akTZ3T0/P7v3g7XD7TsQCprb6O9Nybr8CDIrztdUseQN/TCXuEvCU2cvC
t1xX9UUyI6fOxXxQFlDWx7IE2S7Zo5wOeVWmMBnCQCV8iDjcNxGHwhtCT09bVVf0wbaO
iHXvAYzWlvPhyR+0="}
```

备注：安全支付服务返回的URL解析前需要*UrlDeCode*。*resultStatus*状态码请参考附录A 错误代码列表

第四章通知结果

4.1 同步返回通知

参考 3.3.2 RSA 验签（同步返回示例）
建议：同步通知仅仅作为支付成功后通知用户，修改订单交易状态还是请以异步返回通知为准，并且同步返回的参数也并不完整，仅部分参数，异步通知返回的参数是完整的。

4.2 异步返回通知（Notify_url 地址）

4.2.1 什么是 Notify_url

支付宝通过访问商户提供的地址的形式，将交易状态信息发送给商户服务器。商户通过支付宝的通知判断交易是否成功，具体如下：

商户地址：提供一个 http 的 URL(例:http://www.partnertest.com/servlet/NotifyReceiver)，支付宝将以 POST 方式调用该地址。

通知触发条件：交易状态发生改变，如交易从“创建”到“成功”或“关闭”。

商户返回信息：商户服务器收到通知后需返回纯字符串“success”，不能包含其他任何 HTML 等语言的文本。

通知重发：若支付宝没有收到商户返回的“success”，将对同一笔订单的通知进行周期性重发 (间隔时间为：2 分钟,10 分钟,10 分钟,1 小时,2 小时,6 小时,15 小时共 7 次)。

交易判断条件：收到 trade_status=TRADE_FINISHED（如果签有高级即时到帐协议则 trade_status=TRADE_SUCCESS）的请求后才可判定交易成功（其它 trade_status 状态请求可以不作处理）

4.2.2Notify_url 接收数据示例

```
notify_data=<notify><trade_status>TRADE_FINISHED</trade_status><total_fee>0.90</total_fee><subject>123456</subject><out_trade_no>1118060201-7555</out_trade_no><notify_reg_time>2010-11-1814:02:43.000</notify_reg_time><trade_no>2010111800209965</trade_no></notify>&sign=ZPZULntRpJwFmGNIVKwjLEF2Tze7bqs60rxQ22CqT5J1UlvGo575QK9z/+p+7E9cOoRoWzqR6xHZ6WVv3dloyGKDR0btvrdqPgUAoeaX/YOWzTh00vwcQ+HBtXE+vPTfAqjCTxiiSJEOY7ATCF1q7iP3sfQxhS0nDUug1LP3OLk=
```

注意：在调用验签方法时，需要将"*notify_data*="引号内的几个字符加上，一并验签，以上红色部分

Notify_data 参数说明

参数名	说明
trade_status	用于判断交易状态，值有： TRADE_FINISHED：表示交易成功完成 WAIT_BUYER_PAY：表示等待付款
total_fee	交易金额
subject	商品名称
out_trade_no	外部交易号（商户交易号）
notify_reg_time	通知时间
trade_no	支付宝交易号

第五章常见问答

1. **客户端验签，报“订单信息被篡改”是什么问题？**

   可能有以下2种情况

   a) 有可能数据在传输过程中被黑客截取和篡改

   b) 检查 (待签名的字符串)中是否有以下四个符号，如果参数当中包含了这四个字符也会报“订单信息被篡改”：

   +加号

   &连接符

   “双引号

   =等号

2. **客户端调用安全支付时对 body 和 subject 进行 URLEncode 会报签名错误，到底哪些需**

要 URLEncode？

a) 调用安全支付接口时，只需要对参数sign进行URLEncode，其他参数都不能URLEncode，安全支付服务插件会对所有参数进行URLEncode，所以不用担心中文乱码

3. 上传商户公钥报格式错误怎么办？

a) 首先确认上传的位置是否是RSA的下面，注意不要是DSA，无线目前不支持DSA加密；另外请检查上传的文件中是否去除注释、空格、换行等，必须是一行的字符串

4. Notify 通知收不到

a) 首先这个 url 地址必须是公网能够访问

b) 防火墙、白名单的问题（建议暂时关闭防火墙试试，或者配置下白名单）

c) 该页面是异步等待支付宝通知，无法调试，可以通过记录日志方式查看支付宝是否 post 该 url

d) 通过模拟 http post 请求商户 Notify_url 地址，看是否有些异常错误没有处理，导致接收通知异常

e) ASP.NET 的服务端可以配置下以下信息

Notify_url.aspx 加 ValidateRequest="false"指令

web.config 中要在<system.web></system.web>节点中配置
<httpRuntime requestValidationMode="2.0" />

附录 A 错误代码列表

以下为安全支付服务所定义的错误代码：（同步返回）

表 A-1 系统定义的错误代码表

错误代码	含义
9000	操作成功。
4000	系统异常。
4001	数据格式不正确。
4003	该用户绑定的支付宝账户被冻结或不允许支付。
4004	该用户已解除绑定。
4005	绑定失败或没有绑定。
4006	订单支付失败。

4010	重新绑定账户。
6000	支付服务正在进行升级操作。
6001	用户中途取消支付操作。