

API 接口文档

支付接口文档

文档版本: 1.0

接口版本: 1.0

目錄

1	引言	2
1.1	支付模式	2
2	方案概述	2
2.1	業務實現流程（以微信為例）	2
2.2	接口調用時序圖	3
3	數據格式	3
3.1	提交數據	3
3.2	XML 數據格式	4
4	數字簽名	5
4.1	簽名原始串	5
4.2	簽名演算法	5
4.2.1	SHA256 簽名	5
4.2.2	RSA 簽名	6
5	支付接口	10
5.1	支付接口	10
5.2	訂單查詢接口	13
5.3	退款查詢接口	15
5.4	退款接口	17
5.5	撤銷接口	20
5.6	掃碼通知接口	22
5.7	綁卡接口	24
5.8	解綁接口	26
6	注意事項	28

1 引言

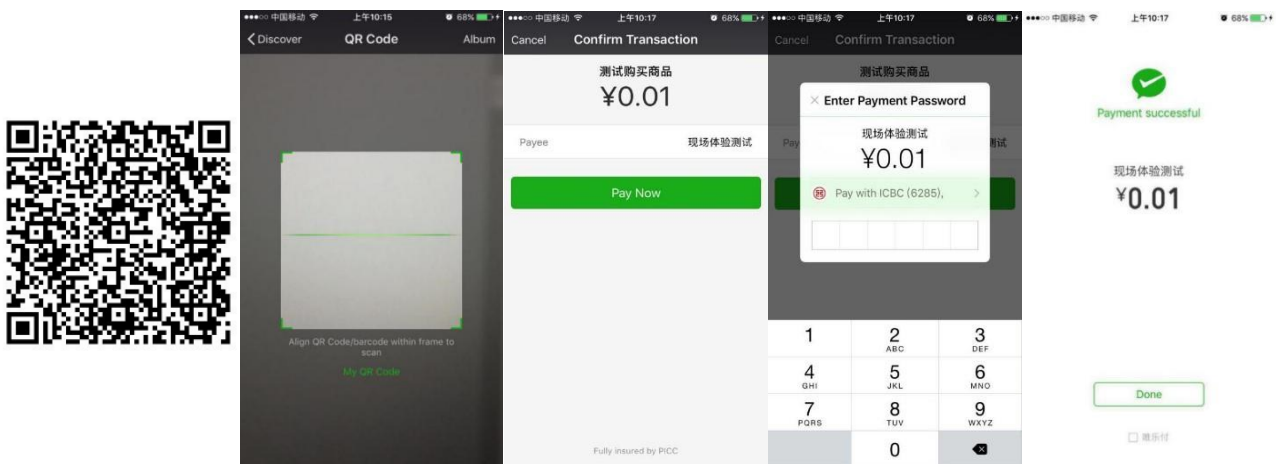
1.1 支付模式

扫码支付是商户系统按微信、支付宝、银联等钱包支付协议生成支付二维码，用户再用微信，支付宝，支持银联支付 app“扫一扫”完成支付的模式。该模式适用于 PC 网站支付、实体店单品或订单支付、媒体广告支付等场景。

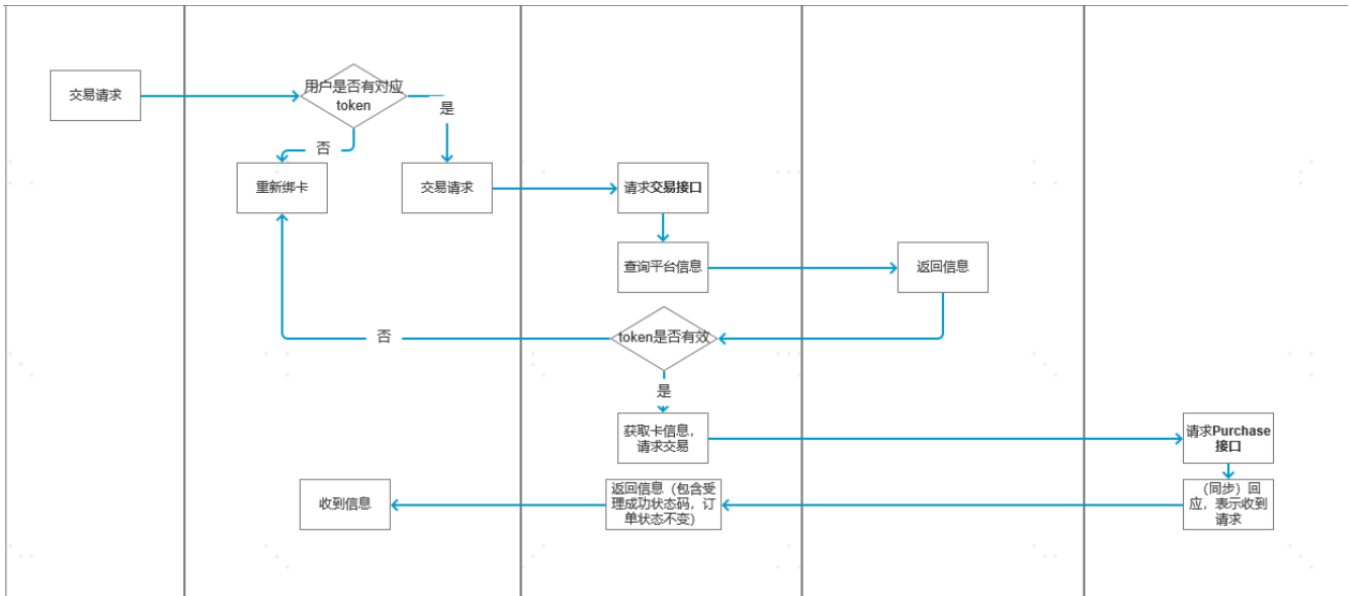
2 方案概述

2.1 業務實現流程（以微信為例）

- 步驟 1: 商戶根據第三方支付的規則，生成訂單二維碼，商戶可將生成的二維碼通過顯示屏等載體進行展示；
- 步驟 2: 用戶使用如圖 2 所示的微信掃一掃功能，獲取商品資訊，同時到商戶後臺下單；
- 步驟 3: 用戶確認訂單開始支付，輸入支付密碼；
- 步驟 4: 支付成功后，商戶後臺得到通知，進行發貨處理



2.2 接口調用時序圖



3 數據格式

3.1 提交數據

採用 HTTPS 標準的 POST 協議，為了保證接收方接收數據正確，傳輸數據必須簽名。（以微信掃碼接口為例，具體 service 字段的傳參請填寫具體接口類型）

```

<xml>
<service>pay.weixin.native </service>
<attach><![CDATA[att]]></attach>
<body><![CDATA[支付測試]]></body>
<device_info>1000</device_info>
<mch_id>10000100</mch_id>
<nonce_str>adf880d5c8986bd0deb6423c92c9d948</nonce_str>
<out_trade_no>1406046836</out_trade_no>
<spbill_create_ip>127.0.0.1</spbill_create_ip>
<total_fee>1</total_fee>
<sign><![CDATA[F53145E553092CE52E4CAA4D2B49A91C]]></sign>
</xml>
    
```

3.2 XML 數據格式

採用標準 XML 協議，所有參數只存在一級節點中，不採用多級節點嵌套，並且需要包含在 CDATA 內協議級錯誤返回：

```
<xml>
<status>500</status>
<message><![CDATA[SYSERR]]></message>
</xml>
```

正確返回數據：

```
<xml>
<status>0</status>
<message><![CDATA[OK]]></message>
<appid><![CDATA[wx2421b1c4370ec43b]]></appid>
<mch_id><![CDATA[10000100]]></mch_id>
<device_info><![CDATA[1000]]></device_info>
<nonce_str><![CDATA[FvYSnPuFFPkAr77M]]></nonce_str>
<sign><![CDATA[63238039D6E43634297CF2A6EB5F3B72]]></sign>
<result_code>0</result_code>
<is_subscribe><![CDATA[Y]]></is_subscribe>
<trade_type><![CDATA[MICROPAY]]></trade_type>
<bank_type><![CDATA[CCB_CREDIT]]></bank_type>
<total_fee>1</total_fee>
<coupon_fee>0</coupon_fee>
<fee_type><![CDATA[CNY]]></fee_type>
<transaction_id><![CDATA[1008450740201407220000058756]]></transaction_id>
<out_trade_no><![CDATA[1406033828]]></out_trade_no>
<attach><![CDATA[att]]></attach>
<time_end><![CDATA[20140722160655]]></time_end>
</xml>
```

業務級錯誤返回：

```
<xml>
<status>0</status>
<message><![CDATA[OK]]></message>
<appid><![CDATA[wx2421b1c4370ec43b]]></appid>
<mch_id><![CDATA[10000100]]></mch_id>
<device_info><![CDATA[1000]]></device_info>
<nonce_str><![CDATA[sthBJ9QyUG6vkrjJ]]></nonce_str>
<sign><![CDATA[6277A96D7875D4FF23AA7B6A4C3046AB]]></sign>
<result_code>1</result_code>
<err_code><![CDATA[AUTHCODE_EXPIRE]]></err_code>
<err_code_des><![CDATA[二維碼已過期，請刷新再試]]></err_code_des>
```

```
</xml>
```

4 數字簽名

為了保證數據傳輸過程中的數據真實性和完整性，需要在接收簽名數據之後進行簽名校驗。

4.1 簽名原始串

无论是请求还是应答，签名原始串按以下方式组装成字符串：

1. 除 sign 字段外，所有参数按照字段名的 ascii 码从小到大排序后使用 QueryString 的格式（即 key1=value1&key2=value2...）拼接而成，空值不传递，不参与签名组串。
2. 签名原始串中，字段名和字段值都采用原始值，不进行 URL Encode。
3. 平台返回的应答或通知消息可能会由于升级增加参数，请验证应答签名时注意允许这种情况。

举例：调用某个接口，接口有如下字段：

```
<xml>
<body><![CDATA[TestPay]]></body>
<device_info><![CDATA[100]]></device_info>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[127520000042]]></mch_id>
<nonce_str><![CDATA[2209196862]]></nonce_str>
<notify_url><![CDATA[https://int.payable.lk:8088/mposcbc/callback]]></notify_url>
<out_trade_no><![CDATA[127590000128]]></out_trade_no>
<service><![CDATA[pay.weixin.native.intl]]></service>
<sign><![CDATA[84F40D5FA4BD7FE3BE9A0A88F564D789]]></sign>
<total_fee><![CDATA[250]]></total_fee>
</xml>
```

正确的签名字段排序为：

```
body=TestPay&device_info=100&mch_create_ip=127.0.0.1&mch_id=127520000042&nonce_str=2209196862&notify_url=https://int.payable.lk:8088/mposcbc/callback&out_trade_no=127590000128&service=pay.weixin.native.intl&total_fee=250
```

4.2 簽名演算法

4.2.1 SHA256 簽名

SHA256 签名计算公式：

$sign = \text{SHA256}(\text{原字符串} \& \text{key} = \text{商户密钥}). \text{toUpperCase}$

如：假设以下为 XML 传入参数

```
<xml>
<auth_code>135187250012923035</auth_code>
<body>test</body>
<charset>UTF-8</charset>
<mch_create_ip>127.0.0.1</mch_create_ip>
<mch_id>127530000052</mch_id>
<nonce_str>1542940680925</nonce_str>
<out_trade_no>1542940643087</out_trade_no>
<service>unified.trade.micropay</service>
<sign>10F2F6DC0D5E008B967CC3C86FC58179686B4EE42F4F68B4A7668501B6030C29</sign>
<sign_type>SHA256</sign_type>
<total_fee>2</total_fee>
<version>2.0</version>
</xml>
```

假設商戶密鑰為：18e0a2ad5d5571af14b855fcf33091f4

i: 經過 a 過程 URL 鍵值對字典序排序後的字串 string1 為:

```
auth_code=135187250012923035&body=test&charset=UTF-8&mch_create_ip=127.0.0.1&mch_id=127530000052
&nonce_str=1542940680925&out_trade_no=1542940643087&service=unified.trade.micropay&total_fee=1
```

ii: 經過 b 過程後得到 sign 為:

sign

```
=SHA256(string1&key=18e0a2ad5d5571af14b855fcf33091f4).toUpperCase
```

```
=SHA256(auth_code=135187250012923035&body=test&charset=UTF-8&mch_create_ip=127.0.0.1&mch_id=1275
30000052&nonce_str=1542940680925&out_trade_no=1542940643087&service=
```

```
unified.trade.micropay&total_fee=1&key=
```

```
18e0a2ad5d5571af14b855fcf33091f4).toUpperCase()
```

```
="10F2F6DC0D5E008B967CC3C86FC58179686B4EE42F4F68B4A7668501B6030C29"
```

4.2.2 RSA 簽名

RSA 演算法一直是最為使用的“非對稱加密演算法”，通過在簽名原始串後加上商戶通信 RSA 私鑰的內容，進行 RSA 運算得到結果字串即為簽名結果。

注意：簽名時將字串轉化成位元組流時指定的編碼字元集應與參數 charset 一致。RSA 密鑰對由商戶生成，需要將 public key 上傳商戶 portal，並且下載 SwiftPass public key 用於解簽 response。

RSA 簽名計算公式：

$$\text{sign} = \text{RSA}(\text{原字串}\&\text{key}=\text{商戶 RSA 私鑰}). \text{toUpperCase}$$

假設以下為 XML 傳入參數：

```
<xml>
```

```
<out_trade_no><![CDATA[6057113230875088]]></out_trade_no>
<nonce_str><![CDATA[wNzpaD0sN17KI80yBQwINNhfM0IeNqap]]></nonce_str>
<time_expire><![CDATA[]]></time_expire>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<sign_type><![CDATA[RSA_1_256]]></sign_type>
<total_fee><![CDATA[1]]></total_fee>
<notify_url><![CDATA[http://www.baidu.cn/notify.aspx]]></notify_url>
<body><![CDATA[測試購買商品]]></body>
<version><![CDATA[1.0]]></version>
<mch_id><![CDATA[102532336411]]></mch_id>
<time_start><![CDATA[]]></time_start>
<attach><![CDATA[附加資訊]]></attach>
<service><![CDATA[pay.alipay.native]]></service>
<sign>
<![CDATA[SE008JDir0uw0uBy8d48SmdfG37PyGvhtqu8pD00DpdNkgkLuegNsb6SaL/dEfzu035bwAVwiKSc9m9xBKohGJ
EMtzRm3tLNPq0BDpLLYNWQkr0JN3JAjy6wk1icSmfbjXgEvWCcx17MUe59NfI10JRR3MgFg/ySYq2cT4U/o6WJisxtIbre0
ZPM66WL815gAzosJW3Gwr+B8fkRyOckF4w64i0TM20xoSHGEa8w1ut0Fik5wvxyPHb/JcjhRyrBnb2LFZy5Rq3XzZLW6FaK1
gunWnJK5+4NiNuHDgm7CKZPp8Bp
hA/qnJtAMGfexEZ8J4z9ktyyWVK8b8VWJ5ow==]]>
</sign>
</xml>
```

假設商戶 RSA 私鑰為:

```
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQCfU8v4BUR81SKm/H0ahbdQZjEp08nMyk+XuYSatHwnU4//
m47R+4G2YB4Z6PHsJi4+ScfJpQutFhKrFwTXZ6TDqLvaqZDDkjq5G271g+PmrzFp7f40/E9m0qjeL64RJra0rZq123dvPW4v
VomMRgRcoP0n0YVWp+M6T5PaFgE4M8dh41MZz57gVwOdd08F99Z92f3QgZtEjI+/EXvMenXxb/aRoFNkt+Wdk2ELJ6MIP0d9
UU5v3WgLuuv5QnQYzj/RMr8GD+wrDYiNqJxsaTme/OEJggsumhD4eYY5Y1Ry2EIN504cuJYVKU1wOSZgq9oJcYnGR0aPuQW
x58IHxEtAgMBAECggEAHfEFd8qm2PTE21Tavec7F+TcgD84IUAz0dZnURtx6YI0oZ5+LH/zVG6juYLJU/Oo5RPac+iMVS68
u2JMCp7zm8Ft7B3JkrbuHLNHGuR6Q7PQuXN8PkDc0xqDmZ2kPJz14PZvBZRE0abdug+tMatGzpGAuJzrWcB/NooVIvrXp9Pn
Oqfo/Y5nxmp0FCImJppIS3AL1pftNtQz09G15CPHDYtPUBXPtD2MjJw40LxKuPrOHSWUgo6LW9XSwnXfcuK+1bzLL0Bh1WD9
IV/+yCEUEb1N87yxxfhpQFaAhXj5W+B3YsMOZuK93+XMOpYmw8EpUDM0b0nvwB0NSHURV2RUAQKBgQDT0j1nNS1e7+tjPzFt
OhGPj1uCBPAEIEHAcnPgD80bEiuJxMLCnGaAvmnTrMu4Xo0e5fAP4F7R6UD+IUsfr3CAAu7CadQ49TW+SovAvcy9AZuSVVI
wynu6QdYgFyPKe1LZYAEq5k+mB1Vh5q0RoxMNA5pGYK8+4MmsJi7X7QKBgQDAunC0qIiH128bs/1VRIhDpzuRW5Qr/SRb
02saVg5RSHn0/nGT20uxSTTkc8yrx7qd9SmAxX15kR238DhMOQ0nRBomldmVtAJuJgrdQyt0wXfeVQqshqCUaE/xhEbpScd
bPSZbKZZdp1V0y605vXIhxw+1qAvXLcxw46s3R92QQKBgQC1Q+eJywkVPDILHMwSSehwvThufkCYWYUbbCVDowp0e5AMoZid
tNju7MNjg2rLHTsCx/kBzOr+7THNw14R7kTiEmg09c0+fu5rHXepGgtig+GJukaZPZ6/bMZJvGOLgOhHmomwG/jdwpGvtIGB
Ch6BW5JZcSImT+ykIoYfvDRuQKBgCgwOHxnBGFf0RoLxEdhpsk8LT05cbueIBVuZW6UC3+8PeK82AjiBLMUy04QHupoG6D
yu3BP/1r10jd3L94PBzLBD7Gm4vJTqW0dknYo5sMXS1JrnofcKjBv7nbHXZTx3EtJSxpVa0dpcA/HpsCuCP3AH2e1yk9sZ3
wu61BYSBAoGACYM60j1CVRNSZxUNRgiwFwzS69qI1eezPc7xQEganpVBI9SZcTNp1kpDKmQikXJ4Yb5XWn12HCY/sFeBW6Su
3ruNqXvg1XiUPbH6A6nxd5B3QX0mS9+wDm60NysPLRdKbF00mdP4CeyuGPdvDIMXP4dJdLhMUL4pcJLIOB7gBE=
```

i: 經過 a 過程 URL 鍵值對字典序排序後的字串 string1 為:

attach=附加資訊&body=測試購買商品

&mch_create_ip=127.0.0.1&mch_id=102532336411&nonce_str=wNzpaD0sN17KI80yBQwINNhfM0IeNqapnotify_ur


```
l=http://www.baidu.cn/notify.aspx&out_trade_no=6057113230875088&service=pay.alipay.native&sign_type=RSA_1_256&total_fee=1&version=1.0
```

ii: 經過 b 過程後得到 sign 為:

sign

=RSA(string1, 商戶 RSA 私鑰)==RSA(attach=附加資訊&body=測試購買商品

&mch_create_ip=127.0.0.1&mch_id=102532336411&nonce_str=wNzpaD0sN17KI80yBQwINNhfM0IeNqapnotifiy_url=http://www.baidu.cn/notify.aspx&out_trade_no=6057113230875088&service=pay.alipay.native&sign_type=RSA_1_256&total_fee=1&version=1.0)

```
=MIIIEvQIBADANBgkqhkiG9w0BAQEFAASCbKcwggSjAgEAAoIBAQCfU8v4BUr81SKm/H0ahbdQZjEp08nMyk+xuYSatHwnU4/  
/m47R+4G2YB4Z6PHsJi4+ScfJpQutFhKrFwTXZ6TDqLvaqZDDkjq5G271g+PmrzFp7f40/E9m0qjeL64RJraOrZq123dvPW4  
vVomMRgRcoPOn0YVWp+M6T5PaFgE4M8dh41MZz57gVwOdd08F99Z92f3QgZtEji+/EXvMenXxb/aRofNkt+Wdk2ELJ6MIPod  
9UU5v3WgLuuNv5QnQYzj/RMr8GD+wrDYiNqJxsaTmE/OEJggsumhD4eYY5Y1Ry2EIN504cuJyVKU1wOSZgq9oJCynGR0aPuQ  
Wx58IHxEtAgMBAECggEAHfEFd8qm2PTE21TAvEc7F+TcgD84IUaZ0dZnURtx6YIOoZ5+LH/zVG6juYLJU/Oo5RPAc+iMVS6  
8u2JMCp7zm8Ft7B3JkrbuHLNHGuR6Q7PQuXN8PkDcOxqDmZ2kPjz14PZvBZRE0abdug+tMatGzpGAuJzrWcB/N0oVIvrXp9P  
n0qfo/Y5nxmp0FCImJppIS3AL1pftNtQZo9G15CPHDYtUpXPtD2MjjW40LxKuPrOHSwUgo6LW9XSwNXfcuK+1bzLL0Bh1WD  
9IV/+yCEUEb1N87yxxfhpQFaAhXj5W+B3YsMOZuK93+XMOpYmw8EpUDMOBOnvwb0NSHUrV2RUAQKBgQDTojlnNS1e7+tjPzF  
t0hGPj1uCBPAIEhAcnPg80bEiujxMLCnGaAvmmTrMu4Xo0e5fAP4F7R6UD+IUsfr3CAAu7CadQ49TW+SovAvcy9AZuSVV  
Iwynu6QdYgFyPKe1LZYAEq5k+mB1Vh5q0RoxMNA5pGYKg8+4MmmsJi7X7QKBgQDAunCOqIih128bs/1VRIhDpzuRW5Qr/SR  
b02saVg5RSHn0/nGT20uxSTTkc8yrx7qd9SmAxX15kR238DhMOQOnRBomldmVtAJuJgrdQyt0wXfeQVQqshqCUaE/xhEbpSC  
dbPSZbKZZdp1V0y605vXIhxw+1qAvXLcxw46s3R92QKKBgQC1Q+eJywkVPDILHMwSsehvwThufkCYWYUbbcVDowp0e5AMoZi  
dtNju7MNjg2rLHTsCx/kBzOr+7THNw14R7kTiEmg09c0+fu5rHXepGgtig+GJukaZPZ6/bmZJvGOLgOhHmomwG/jdwpvTIG  
BCh6BW5JZcSImT+ykIOoYfvDRuQKBgCgwoHxnBGFF0RoLxE3dhpSk8LT05cbueIBVuZW6UC3+8PeK82AjiBLMUy04QHupoG6  
Dyu3BP/1r10jd3L94PBzLBDL7Gm4vJTqWODknYo5sMXS1JrnofcKjBv7nbHXZTx3EtJSxpVa0dpcA/HpsCuCP3AH2e1yk9sZ  
3wu61BYSBAoGACYM60j1CVRNSZxUNRgiwfWzS69qIleezPc7xQEganpVBI9SZcTNp1kpDKmQikXJ4Yb5XWn12HCY/sFeBW6S  
u3ruNqxvglXiUPbH6A6nxd5B3QX0mS9+WdM60NysPLRdKbfF00mdP4CeyuGPdvDIMXP4dJdLhMUL4pcJLI0B7gBE=") = S  
E008JDir0uw0uBy8d48SmdfG37PyGvhtqu8pDT00DpdNkgkLuegNsb6SaL/dEfzu035bwAVwiKSc9m9xBKohGJEMtzRm3tLN  
npQ0BDpL1YNWQkr0JN3JAjy6wk1icSmbjXgEvWcxc17MUe59nFI10JRR3MgFg/ySYq2cT4U/o6WJisxtIbre0ZPM66WL815  
gAzosJW3Gwr+B8fkRyOckF4w64i0TM20xoSHGEa8w1utOFik5wvxyPHb/JcjhRyrBnb2LFZy5Rq3XzZLW6FaK1gunWnjK5+4  
NiNuHDgm7CKZPp8BphA/qnJtAMGfexEZ8J4z9ktyywdK8b8VWJ5ow==
```

微信签名认证流程:

请使用此工具验证: https://pay.weixin.qq.com/wiki/doc/api/jsapi.php?chapter=20_1

1. 请打开网址 https://pay.weixin.qq.com/wiki/doc/api/jsapi.php?chapter=20_1, 然后在“XML 源串”中输入 xml;
2. 请将您的签名密钥放入“商户 key”;
3. 单击“校验签名”按钮; 4. 我们可以在“新 Sing 值”中获得正确的 Sing key

签名类型: MD5 HMAC-SHA256

校验方式: XML deeplink 自定义参数

XML源串:

商户Key:

表 1 签名校验工具

5 支付接口

5.1 支付接口

一、支付接口注意事项

1. 由于支付通道的特性，发起支付后，需要主动发起查询或者等待通知结果才能确定支付的结果。

注意：綁卡后才可以發起支付

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL：https://gateway.wepayez.com/pay/gateway

參數名	必填	類型	說明
service	是	String(32)	接口类型：pay.expresspay.pay
version	否	String(8)	版本号，version 默认值是 2.0
charset	否	String(8)	字元集，取值：UTF-8
sign_type	否	String(12)	MD5 签名：MD5（默认） SHA256 签名：SHA256 RSA 签名：RSA_1_256
mch_id	是	String(32)	商户唯一标识码，由平台分配
out_trade_no	是	String(32)	商户系统内部的订单号，5 到 32 个字元、只能包含字母数字或者下划线，区分大小写，确保在商户系统唯一
device_info	否	String(32)	终端设备号，商户自定义。
device_type	否	String(32)	设备类型
body	是	String(127)	商品描述
attach	否	String(127)	商户附加资讯，可做扩展参数
total_fee	是	Int	金额，以支付货币为准，以分为单位。如支付货币为港币，1000 表示 HKD10.00。测试商户号最大限额 100 分/笔（即 1 元/笔）。
mch_create_ip	是	String(16)	订单生成的机器 IP
notify_url	是	String(255)	接收平台通知的 URL，需给绝对路径，255 字元内格式如： http://wap.tenpay.com/tenpay.asp ，确保平台能通过互联网访问该地址
time_start	否	String(14)	订单生成时间，格式为 yyymmddhhmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010。时区为 GMT+8 beijing。
nonce_str	是	String(32)	随机字串，不长于 32 位
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”
channel_type	是	String	INTERNET MOBILE 支付 in app 交易用 08

Token_id	是	String	Token_id
phone_no	否	String	手机号 eg: 852-11112222
cvn2	否	String(3)	卡 cvn2

Example

```
<xml><body><![CDATA[測試商品]]></body>
<channel_type><![CDATA[INTERNET]]></channel_type>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608383533]]></nonce_str>
<notify_url><![CDATA[www.baidu.com]]></notify_url>
<out_trade_no><![CDATA[1608383533]]></out_trade_no>
<service><![CDATA[pay.expresspay.pay]]></service>
<sign><![CDATA[DE3A9D99534203D64D03E351B6D453DC]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<token_id><![CDATA[100550002127Cz95tz8]]></token_id>
<total_fee><![CDATA[100]]></total_fee>
</xml>
```

返回參數:

參數名	必填	類型	說明
version	否	String(8)	版本号, version 默认值是 2.0。
charset	是	String(8)	字元集, 取值: UTF-8
sign_type	是	String(16)	MD5 签名: MD5; SHA256 签名: SHA256 RSA 签名: RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识, 非交易标识, 交易是否成功需要查看 result_code 来判断
message	否	String(128)	返回资讯, 如非空, 为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 有返回

result_code	是	String(16)	0 表示成功非 0 处理中或失败, 请调用查询接口获取最终状态
mch_id	是	String(32)	商户唯一标识码, 由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字串, 不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String(128)	错误信息
sign	是	String(344)	签名结果, 详见“第 4 章 签名规则”
trade_type	是	String(32)	pay.expresspay.pay
pay_result	是	Int	支付结果: 0—成功; 其他—处理中或失败, 请调用查询

接口获取最终状态			
transaction_id	是	String(32)	平台交易号
out_transaction_id	是	String(32)	第三方订单号
out_trade_no	是	String(32)	商户系统内部的订单号 ,5 到 32 个字符、只能包含字母数字或者下划线, 区分大小写, 确保在商户系统唯一
total_fee	是	Int	订单总金额, 币种为通道币种
fee_type	否	String(8)	订单通道币种, E.g AUD
attach	否	String(127)	商家数据包, 原样返回
bank_type	否	String(16)	支付类型
time_end	是	String(14)	支付完成时间, 格式为 yyyyMMddhhmmss, 如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。时区为 GMT+8 beijing。
uuid	否	String(64)	订单 uuid
local_fee_type	否	Int	商户标价币种, E.g AUD
local_total_fee	否	String(16)	商户标价金额
order_fee	是	Int	订单金额 (币种为商户标价币种)
need_query	否	String	是否需要调用查询接口确定订单状态 Y:需要 N: 不需要
accepted	是	String	表示请求是否受理成功, 0:成功 1:失败

Example

```

<xml><bank_type><![CDATA[07]]></bank_type>
<charset><![CDATA[UTF-8]]></charset>
<fee_type><![CDATA[AUD]]></fee_type>
<local_fee_type><![CDATA[AUD]]></local_fee_type>
<local_total_fee><![CDATA[100]]></local_total_fee>
<mch_id><![CDATA[100550002127]]></mch_id>
<need_query><![CDATA[Y]]></need_query>
<nonce_str><![CDATA[1608383533]]></nonce_str>
<order_fee><![CDATA[100]]></order_fee>
<out_trade_no><![CDATA[1608383533]]></out_trade_no>
<out_transaction_id><![CDATA[902012192112152302758]]></out_transaction_id>
<pay_result><![CDATA[1]]></pay_result>
<result_code><![CDATA[1]]></result_code>
<sign><![CDATA[6A859C769308854FB895F441B620FE4C]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<time_end><![CDATA[20201219211215]]></time_end>
<total_fee><![CDATA[0]]></total_fee>
<trade_type><![CDATA[pay.expresspay.pay]]></trade_type>
<transaction_id><![CDATA[100550002127202012191076647141]]></transaction_id>
<uuid><![CDATA[0068d9b6b84808dece1f08800ba16c06a]]></uuid>
<version><![CDATA[2.0]]></version>

```

```
</xml>
```

5.2 訂單查詢接口

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL：https://gateway.wepayez.com/pay/gateway

變數名	必填	類型	說明
service	是	String(32)	接口类型：unified.trade.query
version	否	String(8)	版本号，version 默认值是 2.0。
charset	否	String(8)	取值：UTF-8。
sign_type	否	String(12)	MD5 签名：MD5（默认）； SHA256 签名：SHA256 RSA 签名：RSA_1_256
mch_id	是	String(32)	商户唯一标识码，由平台分配
out_trade_no	否	String(32)	商户系统内部的订单号，out_trade_no 和 transaction_id 至少一个必填，同时存在时 transaction_id 优先
transaction_id	否	String(32)	平台交易号，out_trade_no 和 transaction_id 至少一个必填，同时存在时 transaction_id 优先。
nonce_str	是	String(32)	随机字符串，不长于 32 位
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”

Example

```
<xml><body><![CDATA[测试订单查询]]></body>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608383774]]></nonce_str>
<out_trade_no><![CDATA[1608383533]]></out_trade_no>
<service><![CDATA[unified.trade.query]]></service>
<sign><![CDATA[838BB3FABAC3B980E850A2E7C6EB06EB]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
</xml>
```

返回參數：

變數名	必填	類型	說明
version	是	String(8)	版本号，version 默认值是 2.0。
charset	是	String(8)	取值：UTF-8。
sign_type	是	String(16)	MD5 签名：MD5（默认）；

			SHA256 签名: SHA256 RSA 签名: RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识, 非交易标识, 交易是否成功需要查看 trade_state 来判断
message	否	String(128)	返回资讯, 如非空, 为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回			
result_code	是	String(16)	0 表示成功非 0 表示失败
mch_id	是	String(32)	商户唯一标识码, 由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字符串, 不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String (128)	错误信息
sign	是	String(344)	签名结果, 详见“第 4 章 签名规则”
reverse_id	否	String	当商家调用撤销接口后返回的参数
reverse_status	否	String	表明撤销操作是否成功
trade_state	是	String	订单的状态
以下字段在 trade_state 為 SUCCESS 的時候有返回			
trade_type	是	String (32)	pay.expresspay.pay
trade_state_desc	否	String(32)	交易状态描述
transaction_id	是	String(32)	平台交易号
out_transaction_id	是	String(32)	第三方交易号
out_trade_no	是	String(32)	商户系统内部的订单号 ,5 到 32 个字元、 只能包含字母数字或者下划线, 区分大小写, 确保在商户系统唯一
total_fee	是	Int	总金额, 以分为单位, 不允许包含任何字、符号
fee_type	否	String(8)	货币类型, 符合 ISO 4217 标准的三位字母代码。
attach	否	String(127)	商家数据包, 原样返回
bank_type	否	String(16)	支付类型
time_end	是	String(14)	支付完成时间, 格式为 yyyyMMddhhmmss, 如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。时区为 GMT+8 beijing。该时间取自平服务器
local_fee_type	否	String(8)	商户标价币种, E.g AUD
local_total_fee	否	Int	商户比价金额
order_fee	是	Int	订单金额 (标价币种)

Example

```
<xml><bank_type><![CDATA[07]]></bank_type>
<charset><![CDATA[UTF-8]]></charset>
<fee_type><![CDATA[AUD]]></fee_type>
<local_fee_type><![CDATA[AUD]]></local_fee_type>
<local_total_fee><![CDATA[100]]></local_total_fee>
<mch_id><![CDATA[100550002127]]></mch_id>
```

```

<nonce_str><![CDATA[1608383774]]></nonce_str>
<order_fee><![CDATA[100]]></order_fee>
<out_trade_no><![CDATA[1608383533]]></out_trade_no>
<out_transaction_id><![CDATA[902012192112152302758]]></out_transaction_id>
<pay_result><![CDATA[0]]></pay_result>
<result_code><![CDATA[0]]></result_code>
<sign><![CDATA[EE17A8318E6A4DBA353421E78AF64451]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<time_end><![CDATA[20201219211215]]></time_end>
<total_fee><![CDATA[100]]></total_fee>
<trade_state><![CDATA[SUCCESS]]></trade_state>
<trade_type><![CDATA[pay.expresspay.pay]]></trade_type>
<transaction_id><![CDATA[100550002127202012191076647141]]></transaction_id>
<version><![CDATA[2.0]]></version>
</xml>

```

5.3 退款查询接口

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL: <https://gateway.wepayez.com/pay/gateway>

變數名	必填	類型	說明
service	是	String(32)	接口类型: unified.trade.refundquery
version	否	String(8)	版本号, version 默认值是 2.0。
charset	否	String(8)	取值: UTF-8。
sign_type	否	String(12)	MD5 签名: MD5 (默认); SHA256 签名: SHA256 RSA 签名: RSA_1_256
mch_id	是	String(32)	商户唯一标识码, 由平台分配
out_trade_no	否	String(32)	商户系统内部的订单号, out_trade_no 和 transaction_id 至少一个必填, 同时存在时 transaction_id 优先
transaction_id	否	String(32)	平台交易号, out_trade_no 和 transaction_id 至少一个必填, 同时存在时 transaction_id 优先。
refund_id	否	String(32)	平台退款单号, 必须和 out_trade_refund 至少一个必填
out_refund_no	否	String(32)	商户退款单号, 必须和 refund_id 至少一个必填
nonce_str	是	String(32)	随机字串, 不长于 32 位
sign	是	String(344)	签名结果, 详见“第 4 章 签名规则”

Example


```

<xml><body><![CDATA[测试订单查询]]></body>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608536843]]></nonce_str>
<out_refund_no><![CDATA[1608536636]]></out_refund_no>
<out_trade_no><![CDATA[1608536491]]></out_trade_no>
<service><![CDATA[unified.trade.refundquery]]></service>
<sign><![CDATA[92105838ADB1AE8F55057E090B814FCC]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
</xml>

```

返回結果：

變數名	必填	類型	說明
version	是	String(8)	版本号，version 默认值是 2.0。
charset	是	String(8)	取值：UTF-8
sign_type	是	String(16)	MD5 签名：MD5； SHA256 签名：SHA256 RSA 签名：RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识，非交易标识，交易是否成功需要查看 result_code 来判断
message	否	String(128)	返回资讯，如非空，为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回

result_code	是	String(16)	0 表示成功非 0 表示失败
mch_id	是	String(32)	商户唯一标识码，由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字串，不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String(128)	错误信息
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”

以下字段在 status 和 result_code 都為 0 的時候有返回

transaction_id	是	String(32)	平台交易号
out_trade_no	是	String(32)	商户系统内部的订单号
out_refund_no_0	是	String(32)	商户退款单号
refund_id_0	是	String(32)	平台退款单号
refund_channel_0	是	String(16)	取值：ORIGINAL
refund_fee_0	是	Int	退款总金额,单位为分,可以做部分退款
total_fee	否	Int	订单通道金额
fee_type	否	String(8)	订单通道币种

order_fee	否	Int	订单金额
local_total_fee	否	Int	商户标价金额
local_fee_type	否	String(8)	商户标价币种, E.g AUD
out_transaction_id	否	String(32)	第三方商户号
trade_type	否	String(32)	pay.expresspay.pay
refund_time_0	是	String(32)	退款时间
refund_status_0	是	String	退款状态
trans_time	否	String	退款时间
refund_count	否	String	Expresspay 只能查一笔退款记录

Example

```

<xml><charset><![CDATA[UTF-8]]></charset>
<fee_type><![CDATA[AUD]]></fee_type>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608536843]]></nonce_str>
<out_refund_no_0><![CDATA[1608536636]]></out_refund_no_0>
<out_trade_no><![CDATA[1608536491]]></out_trade_no>
<out_transaction_id><![CDATA[662012211541542749008]]></out_transaction_id>
<refund_channel_0><![CDATA[ORIGINAL]]></refund_channel_0>
<refund_count><![CDATA[1]]></refund_count>
<refund_fee_0><![CDATA[10]]></refund_fee_0>
<refund_id_0><![CDATA[100550002127202012211276647223]]></refund_id_0>
<refund_status_0><![CDATA[SUCCESS]]></refund_status_0>
<refund_time_0><![CDATA[20201221154711]]></refund_time_0>
<result_code><![CDATA[0]]></result_code>
<sign><![CDATA[9F571EF77919C0F7EA4FAFFB0B383953]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<total_fee><![CDATA[10]]></total_fee>
<trade_type><![CDATA[pay.expresspay.pay]]></trade_type>
<trans_time><![CDATA[20201221154711]]></trans_time>
<transaction_id><![CDATA[100550002127202012211276647221]]></transaction_id>
<version><![CDATA[2.0]]></version>
</xml>

```

5.4 退款接口

商戶針對某一個已經成功支付的訂單發起退款，操作結果在同一會話中同步返回。

一、退款方式

目前只支持原路返回退款

說明：退款至銀行卡，由於每個銀行的處理時間不同，一般退款請求發起後 7 個工作日內到賬。

同一筆單的部分退款需要設置相同的訂單號和不同的 out_refund_no。一筆退款失敗後重新提交，要採用原來

的 out_refund_no。總退款金額不能超過用戶實際支付金額(現金券金額不能退款)

二、退款限制

商戶在退款操作時應該注意退款限制，避免發起不會成功的退款請求，下麵是主要的退款限制：

在平臺中，退款申請單號（退款接口中有此參數）唯一確定一次退款，而不是交易單號確定一次退款。退款申請單號由商戶生成，所以商戶一定要保證退款申請單的唯一性。

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL: <https://gateway.wepayez.com/pay/gateway>

變數名	必填	類型	說明
service	是	String(32)	接口类型: unified.trade.refund
version	否	String(8)	版本号, version 默认值是 2.0。
charset	否	String(8)	取值: UTF-8
sign_type	否	String(12)	MD5 签名: MD5 (默认); SHA256 签名: SHA256 RSA 签名: RSA_1_256
mch_id	是	String(32)	商户唯一标识码, 由平台分配
out_trade_no	否	String(32)	商户系统内部的订单号, out_trade_no 和 transaction_id 至少一个必填, 同时存在时 transaction_id 优先
transaction_id	否	String(32)	平台单号, out_trade_no 和 transaction_id 至少一个必填, 同时存在时 transaction_id 优先
out_refund_no	是	String(32)	商户退款单号, 32 个字节内、可包含字母, 确保在商户系统唯一。同个退款单号多次请求, 平台当一个单处理, 只会退一次款。如果出现退款不成功, 请采用原退款单号重新发起, 避免出现重复退款。
total_fee	是	Int	订单总金额, 单位为分
refund_fee	是	Int	退款总金额, 单位为分, 可以做部分退款
op_user_id	是	String(32)	操作员帐号, 默认为商户号
refund_channel	否	String(16)	取值: ORIGINAL
nonce_str	是	String(32)	随机字符串, 不长于 32 位
sign	是	String(344)	签名结果, 详见“第 4 章 签名规则”

Example

```
<xml><mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608384472]]></nonce_str>
<op_user_id><![CDATA[11]]></op_user_id>
<out_refund_no><![CDATA[1608384472]]></out_refund_no>
<out_trade_no><![CDATA[1608384211]]></out_trade_no>
<refund_channel><![CDATA[ORIGINAL]]></refund_channel>
<refund_fee><![CDATA[2]]></refund_fee>
<service><![CDATA[unified.trade.refund]]></service>
```

```
<sign><![CDATA[721E2E68744E9D0966A36294B5CB583B]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<total_fee><![CDATA[10]]></total_fee>
</xml>
```

返回結果：

變數名	必填	類型	說明
version	是	String(8)	版本号，version 默认值是 2.0。
charset	是	String(8)	取值：UTF-8
sign_type	是	String(16)	MD5 签名：MD5； SHA256 签名：SHA256 RSA 签名：RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识，非交易标识，交易是否成功需要查看 result_code 来判断
message	否	String(128)	返回资讯，如非空，为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回

result_code	是	String(16)	0 表示成功非 0 表示失败
mch_id	是	String(32)	商户唯一标识码，由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字符串，不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String(128)	错误信息
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”

以下字段在 status 和 result_code 都為 0 的時候有返回

transaction_id	是	String(32)	平台交易号
out_trade_no	是	String(32)	商户系统内部的订单号
out_refund_no	是	String(32)	商户退款单号
refund_id	是	String(32)	平台退款单号
refund_channel	是	String(16)	取值：ORIGINAL
refund_fee	是	Int	退款总金额,单位为分,可以做部分退款
total_fee	否	Int	订单通道金额
fee_type	否	String(8)	订单通道币种
order_fee	否	Int	订单金额
local_total_fee	否	Int	商户标价金额
local_fee_type	否	String(8)	商户标价币种，E.g AUD
out_transaction_id	否	String(32)	第三方商户号（Alipay/WeChat/unionPay/etc）
trade_type	否	String(32)	pay.expresspay.pay

Example

```

<xml><charset><![CDATA[UTF-8]]></charset>
<fee_type><![CDATA[AUD]]></fee_type>
<local_fee_type><![CDATA[AUD]]></local_fee_type>
<local_total_fee><![CDATA[10]]></local_total_fee>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608384472]]></nonce_str>
<order_fee><![CDATA[2]]></order_fee>
<out_refund_no><![CDATA[1608384472]]></out_refund_no>
<out_trade_no><![CDATA[1608384211]]></out_trade_no>
<out_transaction_id><![CDATA[902012192124322311508]]></out_transaction_id>
<refund_channel><![CDATA[ORIGINAL]]></refund_channel>
<refund_fee><![CDATA[2]]></refund_fee>
<refund_id><![CDATA[100550002127202012191276647145]]></refund_id>
<result_code><![CDATA[0]]></result_code>
<sign><![CDATA[AA5A8F0787FDD6BB6F044CB8BBBDF5EC]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<total_fee><![CDATA[10]]></total_fee>
<trade_type><![CDATA[pay.expresspay.pay]]></trade_type>
<transaction_id><![CDATA[100550002127202012191076647144]]></transaction_id>
<version><![CDATA[2.0]]></version>
</xml>

```

5.5 撤销接口

一、撤销接口注意事项

1. 撤销接口只能在支付订单支付后 5 分钟内调用，只允许全额撤销，后续需要退款请调用退款接口。

注意：只有支付成功才能發起撤銷

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL: <https://gateway.wepayez.com/pay/gateway>

變數名	必填	類型	說明
service	是	String(32)	接口类型: unified.micropay.reverse
version	否	String(8)	版本号, version 默认值是 2.0。
charset	否	String(8)	取值: UTF-8。
sign_type	否	String(12)	MD5 签名: MD5 (默认); SHA256 签名: SHA256 RSA 签名: RSA_1_256

mch_id	是	String(32)	商户唯一标识码，由平台分配
out_trade_no	否	String(32)	商户系统内部的订单号, out_trade_no 和 transaction_id 至少一个必填，同时存在时 transaction_id 优先
transaction_id	否	String(32)	平台交易号, out_trade_no 和 transaction_id 至少一个必填，同时存在时 transaction_id 优先。
nonce_str	是	String(32)	随机字符串，不长于 32 位
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”

Example

```
<xml><body><![CDATA[测试撤销查询]]></body>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[100550002127]]></mch_id>
<nonce_str><![CDATA[1608385032]]></nonce_str>
<out_trade_no><![CDATA[1608385012]]></out_trade_no>
<service><![CDATA[unified.micropay.reverse]]></service>
<sign><![CDATA[18A42328AE9FF8B6B0586A65E77DC22F]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
</xml>
```

返回参数：

變數名	必填	類型	說明
version	是	String(8)	版本号，version 默认值是 2.0。
charset	是	String(8)	取值： UTF-8。
sign_type	是	String(16)	MD5 签名： MD5（默认）； SHA256 签名： SHA256 RSA 签名： RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识，非交易标识，交易是否成功需要查看 trade_state 来判断
message	否	String(128)	返回资讯，如非空，为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回

result_code	是	String(16)	0 表示成功非 0 处理中或失败，请调用查询接口获取最终状态
mch_id	是	String(32)	商户唯一标识码，由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字符串，不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String (128)	错误信息
sign	是	String(344)	签名结果，详见“第 4 章 签名规则”
transaction_id	是	String(32)	平台交易号
out_transaction_id	是	String(32)	第三方交易号
out_trade_no	是	String(32)	商户系统内部的订单号 ,5 到 32 个字节、 只能包含字母数

			字或者下划线，区分大小写，确保在商户系统唯一
accepted	是	String	表示请求是否受理成功，0:成功 1:失败

以下字段在 status 和 result_code 都为 0 的时候有返回			
trade_state	是	String	订单状态

<p>Example</p> <pre> <xml><accepted><![CDATA[0]]></accepted> <charset><![CDATA[UTF-8]]></charset> <mch_id><![CDATA[100550002127]]></mch_id> <nonce_str><![CDATA[1608563554]]></nonce_str> <out_trade_no><![CDATA[1608563490]]></out_trade_no> <out_transaction_id><![CDATA[162012212311560835268]]></out_transaction_id> <result_code><![CDATA[0]]></result_code> <sign><![CDATA[A1C7839C1D604C7EC94B6D7074014B8D]]></sign> <sign_type><![CDATA[MD5]]></sign_type> <status><![CDATA[0]]></status> <trade_state><![CDATA[REVERSE]]></trade_state> <transaction_id><![CDATA[100550002127202012211276647265]]></transaction_id> <version><![CDATA[2.0]]></version> </xml> </pre>
--

5.6 掃碼通知接口

通知 URL 是 5.1 节中提交的参数 `notify_url`，支付完成后，平台会把相关支付和用户资讯发送到该 URL，商户需要接收处理资讯。对后台通知交互时，如果平台收到商户的应答不是纯字符串 `success` 或超过 5 秒后返回时，平台认为通知失败，平台会通过一定的策略（通知频率为 0/15/15/30/180 /1800/1800/1800/1800/3600，单位：秒）间接性重新发起通知，尽可能提高通知的成功率，但不保证通知最终能成功。

由于存在重新发送后台通知的情况，因此同样的通知可能会多次发送给商户系统。商户系统必须能够正确处理重复的通知。推荐的做法是，当收到通知进行处理时，首先检查对应业务数据的状态，判断该通知是否已经处理过，如果没有处理过再进行处理，如果处理过直接返回结果成功。在对业务数据进行状态检查和处理之前，要采用数据锁进行并发控制，以避免函数重入造成的数据混乱。

特别注意：商户后台接收到通知参数后，要对接收到通知参数里的订单号 `out_trade_no` 和订单金额 `total_fee` 和自身业务系统的订单和金额做校验，校验一致后才更新资料库订单状态。后台通知通过请求中的 `notify_url` 进行，`post` 方式给商户系统（通知参数内容为 xml 的字符串）

返回格式：XML

Swiftpass 通知參數：如下

變數名	必填	類型	說明
version	否	String(8)	版本号，version 默认值是 2.0
charset	否	String(8)	取值：UTF-8
sign_type	否	String(12)	MD5 签名：MD5（默认）； SHA256 签名：SHA256 RSA 签名：RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失败。此字段是通信标识，非交易标识，交易是否成功需要查看 result_code 来判断
message	否	String(128)	返回资讯，如非空，为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回			
result_code	是	String(16)	0 表示成功非 0 表示失败
mch_id	是	String(32)	商户号，由平台分配
device_info	否	String(32)	终端设备号
nonce_str	是	String(32)	随机字符串，不长于 32 位
err_code	否	String(32)	错误码
err_msg	否	String(128)	错误信息
sign	是	String(32)	MD5 签名结果，详见“第 4 章 MD5 签名规则”

以下字段在 status 和 result_code 都為 0 的時候有返回			
trade_type	是	String(32)	pay.expresspay.pay
pay_result	是	Int	支付结果：0—成功；其他—失败
pay_info	否	String(64)	支付结果资讯，支付成功时为空
transaction_id	是	String(32)	平台交易单号
out_transaction_id	是	String(34)	第三方订单号
out_trade_no	是	String(32)	商户系统内部的定单号，32 个字节内、可包含字母、数字、下划线
total_fee	是	Int	总金额，以分为单位，不允许包含任何字、符号
fee_type	是	String(8)	货币类型，符合 ISO 4217 标准的三位字母代码
attach	否	String(127)	商家数据包，原样返回
bank_type	是	String(16)	支付类型
time_end	是	String(14)	支付完成时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。时区为 GMT+8 beijing。
local_total_fee	否	Int	商户标价金额
local_fee_type	否	String(8)	商户标价币种，E.g AUD

Example
<xml>


```

<bank_type><![CDATA[CFT]]></bank_type>
<charset><![CDATA[UTF-8]]></charset>
<device_info><![CDATA[SPAY_AND]]></device_info>
<discount_detail><![CDATA[["discountAmt":"30.00","discountNote":"Uplan
discount"],["discountAmt":"10.00","discountNote":"Instant Discount"]]]>
</discount_detail>

<fee_type><![CDATA[AUD]]></fee_type>
<local_fee_type><![CDATA[AUD]]></local_fee_type>
<local_total_fee><![CDATA[21850]]></local_total_fee>
<mch_id><![CDATA[102520000212]]></mch_id>
<nonce_str><![CDATA[1573186381348]]></nonce_str>
<order_fee><![CDATA[21850]]></order_fee>
<out_trade_no><![CDATA[10252000021270741546158683729]]></out_trade_no>
<out_transaction_id><![CDATA[4200000437201911083557508581]]></out_transaction_id>
<pay_result><![CDATA[0]]></pay_result>
<platform_rate><![CDATA[2040399.9184]]></platform_rate>
<rate><![CDATA[699040000]]></rate>
<result_code><![CDATA[0]]></result_code>
<sign><![CDATA[0FEC037A2D48654C0FC8AEA96F1C4B55]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<time_end><![CDATA[20191108121300]]></time_end>
<total_fee><![CDATA[446]]></total_fee>
<trade_type><![CDATA[pay.expresspay.pay]]></trade_type>
<transaction_id><![CDATA[102520000212201911081086849655]]></transaction_id>
<unsettled_discount_fee><![CDATA[3000]]></unsettled_discount_fee>
<version><![CDATA[2.0]]></version>
</xml>

```

後臺通知結果回饋

平臺伺服器發送通知，post 發送 XML 數據流，商戶 notify_Url 地址接收通知結果，接收方法 demo 有寫（如 php 中的 callback 方法，c#中的 notify.aspx 檔，java 中的 TestPayResultServlet 方法），商戶做業務處理後，需要以純字串的形式回饋處理結果，內容如下：

返回结果	返回结果说明
success	处理成功，平台收到此结果后不再进行后续通知
fail 或其他字元	处理不成功，平台收到此结果或者没有收到任何结果，系统通过补单机制（详见第 5 节）再次通知

5.7 綁卡接口

請求類型：POST

請求格式: XML

請求參數: 如下

請求 URL: <https://gateway.wepayez.com/pay/gateway>

變數名	必填	類型	說明
service	是	String(32)	接口類型: pay.card.bindcard
version	否	String(8)	版本號, version 默認值是 2.0。
charset	否	String(8)	取值: UTF-8。
sign_type	否	String(12)	MD5 簽名: MD5 (默認); SHA256 簽名: SHA256 RSA 簽名: RSA_1_256
mch_id	是	String(32)	商戶唯一标识码, 由平臺分配
nonce_str	是	String(32)	隨機字串, 不長於 32 位
sign	是	String(344)	簽名結果, 詳見“第 4 章 簽名規則”
pan	是	String(19)	卡号
expired	是	String	卡的过期时间 MM/yy
request_time	是	String	请求时间 yyyyMMdd
token_expired	否	String	Token 过期时间 yyyyMMdd

Example
<pre><xml> <expired><![CDATA[12/33]]></expired> <mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip> <mch_id><![CDATA[100550002127]]></mch_id> <nonce_str><![CDATA[1608537335]]></nonce_str> <pan><![CDATA[625094*****0014]]></pan> <request_time><![CDATA[20201221]]></request_time> <service><![CDATA[pay.card.bindcard]]></service> <sign><![CDATA[5E0D9024DDE72F8B29B2A479E85DBF72]]></sign> <sign_type><![CDATA[MD5]]></sign_type> </xml></pre>

返回結果:

變數名	必填	類型	說明
version	是	String(8)	版本號, version 默認值是 2.0。
charset	是	String(8)	取值: UTF-8
sign_type	是	String(16)	MD5 簽名: MD5; SHA256 簽名: SHA256 RSA 簽名: RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失敗。此字段是通信標識, 非交易標識, 交易是否成功需要查看 result_code 來判斷
message	否	String(128)	返回資訊, 如非空, 為錯誤原因簽名失敗參數格式校驗錯

誤

以下字段在 status 為 0 的時候有返回			
-------------------------	--	--	--

result_code	是	String(16)	0 表示成功非 0 表示失敗
mch_id	是	String(32)	商戶唯一标识码，由平臺分配
nonce_str	是	String(32)	隨機字串，不長於 32 位
err_code	否	String(32)	錯誤碼
err_msg	否	String (128)	錯誤信息
sign	是	String(344)	簽名結果，詳見“第 4 章 簽名規則”

以下字段在 status 和 result_code 都為 0 的時候有返回			
--	--	--	--

pan	是	String	卡号
expired	是	String	卡的过期时间
request_time	是	String	请求时间 yyyyMMdd
token_expired	是	String	Token 过期时间 yyyyMMdd
token_id	是	String	tokenId

Example

```
<xml><charset><![CDATA[UTF-8]]></charset>
<expired><![CDATA[12/33]]></expired>
<mch_id><![CDATA[100550002127]]></mch_id>
<msg><![CDATA[成功]]></msg>
<nonce_str><![CDATA[1608537335]]></nonce_str>
<pan><![CDATA[625094*****0014]]></pan>
<request_time><![CDATA[20201221]]></request_time>
<sign><![CDATA[9FC1390DB98C52778FA1A272DBF5B946]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<status_code><![CDATA[000006]]></status_code>
<token_expired><![CDATA[20210621]]></token_expired>
<token_id><![CDATA[100550002127GCca9vO]]></token_id>
<version><![CDATA[2.0]]></version>
</xml>
```

5.8 解绑接口

請求類型：POST

請求格式：XML

請求參數：如下

請求 URL：https://gateway.wepayez.com/pay/gateway

變數名	必填	類型	說明
service	是	String(32)	接口類型：pay.card.unbindcard
version	否	String(8)	版本號，version 默認值是 2.0。
charset	否	String(8)	取值：UTF-8。
sign_type	否	String(12)	MD5 簽名：MD5（默認）； SHA256 簽名：SHA256 RSA 簽名：RSA_1_256
mch_id	是	String(32)	商戶唯一标识码，由平臺分配
nonce_str	是	String(32)	隨機字串，不長於 32 位
sign	是	String(344)	簽名結果，詳見“第 4 章 簽名規則”
pan	是	String	卡号
expired	是	String	卡的过期时间 MM/yy
token_id	是	String	

Example
<pre><xml> <expired><![CDATA[12/33]]></expired> <mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip> <mch_id><![CDATA[100550002127]]></mch_id> <nonce_str><![CDATA[1608537247]]></nonce_str> <pan><![CDATA[625*****00014]]></pan> <service><![CDATA[pay.card.unbindcard]]></service> <sign><![CDATA[3F7F4F96CCA6BE0BCD95827FF719D770]]></sign> <sign_type><![CDATA[MD5]]></sign_type> <token_id><![CDATA[100550002127Cz95tz1]]></token_id> </xml></pre>

返回結果：

變數名	必填	類型	說明
version	是	String(8)	版本号，version 默认值是 2.0。
charset	是	String(8)	取值：UTF-8
sign_type	是	String(16)	MD5 簽名：MD5； SHA256 簽名：SHA256 RSA 簽名：RSA_1_256
status	是	String(16)	0 表示成功非 0 表示失敗。此字段是通信，非交易标志，交易是否成功需要查看 result_code 來判斷
message	否	String(128)	返回资讯，如非空，为错误原因签名失败参数格式校验错误

以下字段在 status 為 0 的時候有返回

result_code	是	String(16)	0 表示成功非 0 表示失敗
mch_id	是	String(32)	商戶唯一标识码，由平台分配
nonce_str	是	String(32)	随机字符串，不長於 32 位
err_code	否	String(32)	错误码
err_msg	否	String (128)	错误信息
sign	是	String(344)	签名結果，详情请看第四章签名规则

以下字段在 status 和 result_code 都為 0 的時候有返回			
pan	是	String	卡号
expired	是	String	卡的过期时间 MM/yy
token_status	是	String	Token 的状态
status_code	是	String	解绑的状态
msg	是	String	status_code 的描述

Example			
<pre> <xml><charset><![CDATA[UTF-8]]></charset> <expired><![CDATA[12/33]]></expired> <mch_id><![CDATA[100550002127]]></mch_id> <msg><![CDATA[成功]]></msg> <nonce_str><![CDATA[1608537247]]></nonce_str> <pan><![CDATA[625094*****0014]]></pan> <sign><![CDATA[984BFAD030DF6CEF779950549ED055FD]]></sign> <sign_type><![CDATA[MD5]]></sign_type> <status><![CDATA[0]]></status> <status_code><![CDATA[000006]]></status_code> <token_status><![CDATA[UNBINDING]]></token_status> <version><![CDATA[2.0]]></version> </xml> </pre>			

6 注意事項

- 1、所有涉及到金額的單位都是分，最小的單位是 1 分，不能有小數出現。
- 2、notify_url 是平臺伺服器從後臺直接發起請求到商戶伺服器，商戶處理時不能檢查用戶的 cookie 或 session；
商戶更新 DB 等發貨流程需要在 notify_url 完成後，以確保掉單時，平臺補單能成功補上
- 3、notify_url 有可能重複通知，商戶需要做去重處理，避免多次發貨

- 4、notify_url 收到的通知，商戶處理成功或檢查訂單已經處理，需要返回處理成功的標誌純字串 success，字串 success 不區分大小寫；如果我們沒有收到返回的 success，我方伺服器繼續向你們發送通知，在三小時後將不再通知；假設所有訂單都沒有返回 success，將增加我們伺服器的通知負載，最壞的情況，可能就是導致正常通知商戶的通知發生延遲；另外我們會催促你們完善，如果長期不改善，研發或運維技術將對貴司開通的支付接口採取控制措施。
- 5、返回參數中必填為是的參數是一定會返回的，必填為否的參數則不一定返回，因升級或配置等情況實際返回參數可能不會跟文檔完全一致，必須以實際接收到的參數為準

6、其他注意事項

(1) 參數大小寫問題

請留意文檔中要求的字元大小寫問題，如“加簽運算後，字串的字元要轉換為大寫”。

(2) 參數格式問題

所有傳入參數，請注意文檔中各處的具體要求。

(3) 時間戳問題

請使用 Linux 時間戳，注意為字串格式。

(4) 同一商戶訂單號支付問題

商戶的 out_trade_no 必須全局唯一，調試和生產環境，都需要使用唯一的訂單號