

# 钱管家系统 商户技术接口规范

汇付网络有限公司

地址：上海市虹梅路 1801 号.凯科大厦 9 层

电话: (86-021)61613999 转运行部 网站: <http://www.chinapnr.com>

## 目 录

1.	前言.....	4
1.1.	目的.....	4
1.2.	使用前须知.....	4
1.3.	使用对象.....	4
2.	商户接口特点.....	4
3.	交易流程示意图.....	5
4.	常见商户需求分析和建议.....	6
4.1.	简单支付型商户.....	6
4.2.	分帐支付型商户.....	6
4.3.	信用支付型商户.....	6
5.	交易类型概述.....	6
6.	接口规范.....	7
6.1.	业务和技术接口规范概述.....	7
6.1.1.	订单规范.....	7
6.1.2.	重复订单规范.....	7
6.1.3.	交易应答接收规范.....	8
6.1.4.	数据交换方式.....	9
6.1.5.	接口规范概述.....	9
6.2.	接口标准参数定义.....	10
6.2.1.	接口交易类型定义.....	10
6.2.2.	接口常用标准参数定义.....	10
6.3.	用户管理类接口.....	14
6.3.1.	签约.....	14
6.3.2.	签名用户登录.....	15
6.3.3.	用户开户.....	15
6.4.	交易类接口.....	16
6.4.1.	单笔订单支付.....	16
6.4.2.	订单结算.....	16
6.4.3.	订单分帐变更.....	17
6.4.4.	订单退款.....	17
6.5.	查询类接口.....	18
6.5.1.	签约查询.....	18
6.5.2.	账户余额查询.....	18
6.5.3.	查询支付状态.....	19
6.5.4.	查询订单明细.....	19
6.5.5.	退款交易状态查询.....	20
6.5.6.	查询分账明细.....	20
6.6.	签名类函数调用接口.....	21
6.6.1.	Java语言函数说明.....	21
6.6.2.	C语言函数说明.....	21
6.6.3.	Windows控件函数说明.....	23
6.6.4.	函数返回值.....	23

6.7.	页面访问URL.....	23
6.7.1.	域名.....	23
6.7.2.	页面访问路径.....	23
7.	自动出票.....	错误！未定义书签。
7.1.	接口标准参数定义.....	错误！未定义书签。
7.1.1.	接口交易类型定义.....	错误！未定义书签。
7.1.2.	接口常用标准参数定义.....	错误！未定义书签。
7.2.	用户验证接口.....	错误！未定义书签。
7.3.	查询接口.....	错误！未定义书签。
7.4.	返回接口.....	错误！未定义书签。
8.	典型插件安装说明.....	24
8.1.	插件在Java应用服务器的安装.....	32
8.2.	插件在Windows应用服务器的安装.....	32
8.2.1.	自动安装.....	32
8.2.2.	手动安装.....	32
8.3.	插件在类Unix/Linux系统上的安装.....	32
8.4.	插件在PHP应用服务器的安装.....	33
9.	标准数据附录.....	34

## 1. 前言

### 1.1. 目的

本技术规范的主要目的是，帮助商户技术人员了解钱管家系统和商户的数据交换格式、商户端插件软件包的安装，和插件函数功能及使用方法，并能方便地与商户的网上商城相集成。

让开发人员了解各种交易类型的请求和返回的数据要素，以及这些要素的属性。

让开发人员学会使用软件包接口如何对商户订单进行**签名和验证签名**的编程。

### 1.2. 使用前须知

阅读该接口规范前，必须了解业务术语或概念：商户、用户、商户和用户的关系、订单、支付、分帐、结算规则、取现规则等。

### 1.3. 使用对象

本手册的使用对象为商户会员的网上应用开发人员、维护人员和管理人员，他们应具有如下之一或者几项基本知识：

熟悉 Microsoft Windows/NT、Windows9x、Windows 2000、HP-UX、AIX、SUN Solaris、Linux、BSD 等操作系统；

熟悉上述系统上的网站设置和网页制作方法；

熟悉标准 CGI（Common Gateway Interface）或 ASP（Active Server Pages）或 ISAPI 的开发方法或 PHP 或 JAVA；

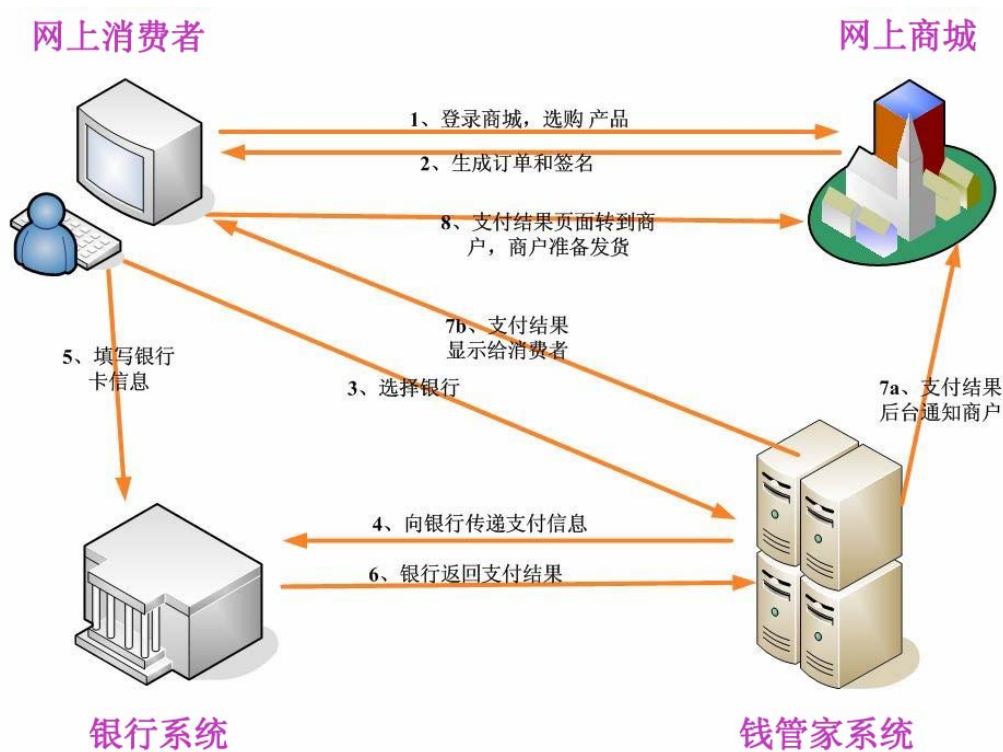
了解一般信息安全（https 知识，非对称密钥，数据签名）的基本概念。

## 2. 商户接口特点

- 每笔交易都带有签名信息，任何对交易信息的篡改都将被钱管家系统拒绝。

- 商户可以动态地将每笔交易的返回地址指向不同的交易网站，适合于多网站、多应用的商户。
- 商户在接口中可以自定义域，钱管家系统返回时，会将该域原样返回，用户商户实现特定的应用。
- 商户可以让持卡人在钱管家系统选择银行卡页面，也可以让持卡人从商户网站选择银行，持卡人不必在钱管家系统再次选择银行，直达银行页面。
- 钱管家系统通过后台方式向商户发送交易结果，通过分析商户的返回页面，可以准确地知道商户是否收到交易应答结果，从而对发送失败的交易采取自动重发。
- 对于大型的商户，支付完成后，可以对支付款项进行分账处理。

### 3. 交易流程示意图



## 4. 常见商户需求分析和建议

### 4.1. 简单支付型商户

对于只需要使用钱管家的支付功能的商户，必须而且只须实现单笔订单支付（且订单支付的分帐明细字段为空）接口即可。

建议实现订单退款、订单查询类接口。

### 4.2. 分帐支付型商户

如果是大型的企业（称为商户），下属多个分支机构（称为用户）；或者为销售代理平台（称为商户），下属多个代理机构（称为用户）。支付完成后，商户需要将支付款项付给下属的多个不同用户。

对于该类大型商户，则必须实现单笔订单支付（且订单支付的分帐明细字段不能为空）接口。

建议实现订单退款、订单查询类接口。

### 4.3. 信用支付型商户

如果是大型的企业（称为商户），下属多个分支机构（称为用户）；或者为销售代理平台（称为商户），下属多个代理机构（称为用户）。支付前，用户可以与、商户、汇付签订三方借款协议，汇付会根据用户的资产、信用状况进行信用评测，从而汇付向用户提供一定的信用额度，用户可以在授信的额度内支付。

对于该类大型商户，则必须实现单笔订单支付接口

建议实现订单退款、订单查询类接口。

## 5. 交易类型概述

钱管家支持的主要交易用途描述如下：

交易名称	交易用途简述
签约接口	用于使用平台账户绑定汇付账户。
单笔订单支付	支持简易型支付（仅商户收款），分账型支付（商户统一销售，但货款需分

	给商户下属的不同用户），信用支付（使用汇付分配信用额度进行支付）。
订单结算	支付完成后，款项何时付给商户下面的用户（卖家），需要根据付款人（买家）是否收到商品进行确认。订单结算即表示商户确认将该货款付给用户。对于订单不需要商户主动发起结算的商户，结算的发起由钱管家系统实时完成。
分帐变更	对于有分账且需要商户主动发起结算的商户，在订单未结算之前，可以对分帐进行变更。
订单退款	只能对结算后的订单进行退款。

## 6. 接口规范

### 6.1. 业务和技术接口规范概述

#### 6.1.1. 订单规范

**规范一）：**商户订单系统用订单号标识每一笔订单，对每一件产生的商品订单，应有唯一的订单号，在钱管家系统内，两个要素：商户号、订单号决定唯一一笔订单。

**规范二）：**如果商户订单在钱管家系统支付失败，则 2 天内允许让支付失败的订单被再次支付，但是商户提交给钱管家系统的订单号**应不变**；如果该笔订单在钱管家系统已经被支付成功，则钱管家系统会拒绝该订单被重复支付。

#### 6.1.2. 重复订单规范

**规范一）：**钱管家允许商户 2 天内提交重复的订单（即订单号不变）。

**规范二）：**由于网络的不稳定性等原因，相同订单的交易应答返回给商户订单系统时，**极有可能先支付完成的订单后送达商户**，对于同一笔订单收到两次支付结果，处理的原则是：a)：只要商户收到订单支付成功的应答，即使以后再次收到订单支付失败的应答，则商户订单系统不做任何处理；b)：商户订单系统对于某笔订单已经收到成功应答，有可能后续再次收到该笔订单的成功应答，则商户只需要回应钱管家系统收到成功即可，商户的账务数据不应被修改。

### 6.1.3. 交易应答接收规范

**规范一）：**钱管家系统交易应答通过页面和后台两种方式返回给商户，无论是页面还是后台，钱管家系统的交易应答数据均以表单方式返回给商户网站，商户收到该订单应答后，**一定要进行验证签名的处理**，以验证该笔交易的合法性；具体参见[接口常用标准参数定义](#)。

**规范二）：**对于页面返回方式，钱管家系统无法知道商户是否收到交易应答，对于后台返回方式，钱管家系统能够知道商户是否收到了交易应答。如果商户未收到交易应答，该交易即进入重发队列，钱管家系统则会将该交易应答重发给商户，直到重发成功为止；**商户必须实现后台和页面两种返回方式，以确保商户一定能收到钱管家系统返回的交易应答，以避免钱管家系统交易成功，而商户交易失败的情况发生。**

**规范三）：**对于后台返回方式，为了表示商户订单系统已经收到交易应答，商户必须在应答接收页面输出一段特殊的字符串，组成规则必须为：由固定字符串 **RECV\_ORD\_ID\_**和该交易的订单号组成。例如，商户收到一笔订单号为 990000034 的交易应答，**则应在该页面输出 **RECV\_ORD\_ID\_990000034** 的字符串**，钱管家系统将从商户交易接收页面中搜索该字符串（为简单处理，钱管家系统只搜索页面的前 1024 字节以内，所以该字符串尽量输出在靠近页面的前部），只有搜索到该字符串，才能确认商户已经收到该笔订单的交易返回应答，否则，钱管家系统认为商户未收到该笔订单的交易应答，进入重发队列，一直发送成功为止。

**规范四）：**商户无论通过页面还是后台方式收到应答，无论收到的先后次序，都应等同处理，商户有可能收到同一笔订单的多次交易应答，后续的处理步骤列示如下。特别地，对于后台方式，以下四种状况，**商户网站都应在应答接收页面输出 **RECV\_ORD\_ID** 字样的字符串，表明商户已经收到该笔交易结果**，否则，钱管家系统一直向商户网站重发送该笔订单。

商户网站 订单状态	从钱管家接收到应 答的订单状态	商户网站的交易应答接收程序应采取的处理步骤
失败	失败	无需修改商户订单系统数据库；提示持卡人订单支付失败。
失败	成功	修改商户订单系统数据库，记录该笔订单成功；提示持卡人订单支付成功。
成功	失败	无需修改商户订单系统数据库；提示持卡人订单支付成功。
成功	成功	无需修改商户订单系统数据库；提示持卡人订单支付成



		功。
--	--	----

#### 6.1.4. 数据交换方式

钱管家系统与商户之间有两种数据交换方式：

**页面浏览器方式：**商户以构造 form 表单的方式，通过持卡人的浏览器重定向到钱管家系统，钱管家完成交易后，将交易结果回送给商户订单系统。**单笔订单支付**采用这种方式。

**后台数据流方式：**商户订单系统服务器，构造 HTTP 协议报文，直接访问钱管家系统，将 HTTP 报文发送到钱管家系统，并获取返回的满足 HTTP 报文协议的字符流。除了单笔订单支付、用户签名登陆，其它所有的交易均采用这种数据交互方式。

在数据流的方式下，商户系统以表单的方式将数据发送到钱管家系统，钱管家收到后进行相应的处理，一个简单的余额查询的接口返回例子如下：

```
CmdId=QueryBalance
RespCode=220001
ChkValue=A170E66B00D344F8CEA68C3A84F0ED2207805147793F806CBD3D3E166
B57F511ABEB7D5D7A725ECEFD6C640FDC2F7102CD8D470CB0BE18A3B4ADE870
689D7FCBD4CA55DD5C2E1D6BFC3F30514B0813D8E680708B44A1C1637780CA0D
998EF22C10B7E4B8954F304F226BB50F091A60C939F7C4DB513261FC47667757550
C7911E
ErrMsg=用户名不存在[lxy]
```

#### 6.1.5. 接口规范概述

如果参数为可选项，或者在 Form 表单中不包含该参数，或者赋值该参数为长度为零的字符串，两者均可。

所有的参数值不应该含有特殊的 HTTP 字符集，如&，‘，“等特殊符号。

钱管家系统使用 GBK 字符集,提交到钱管家系统接口的中文参数都必须以 GBK 方式编码。

所有的金额（如订单金额、可用余额）以元为单位，必须带两位小数点，如 3.50 元，

1200.00 元。

所有的日期为定长 8 的字符串，格式为 YYYYMMDD。

## 6.2. 接口标准参数定义

### 6.2.1. 接口交易类型定义

钱管家支持的交易类型定义如下：

交易类型类别	交易类型名称		交易类型取值	接口技术实现方式
用户管理类接口	签约		Sign	页面浏览器方式
	签名用户登录		Login	页面浏览器方式
交易类接口	单笔订单支付		Buy	页面浏览器方式
	订单结算		PaymentConfirm	后台数据流方式
	订单分账变更		Transfer	后台数据流方式
	订单退款	普通退款	Refund	后台数据流方式
		供应商退款给平台	Refund1	
		平台退款给分销商	Refund2	
查询类接口	签约查询		QuerySign	后台数据流方式
	账户余额查询		QueryBalance	后台数据流方式
	查询支付状态		QueryStatus	后台数据流方式
	查询订单明细		QueryOrder	后台数据流方式
	退款交易状态查询		QueryRefundStatus	后台数据流方式
	查询分账明细		QueryDivDetail	后台数据流方式

### 6.2.2. 接口常用标准参数定义

以下对接口中所使用的参数进行统一定义，各个接口中只说明是否含有该参数，具体参数的属性如无特别说明，均以此处定义为准；与标准参数定义有差别的，在每个接口定义中，会特别说明。

变量名	名称	长度	说明
Version	版本号	定长 2 位	目前固定为 10。如版本升级，能向前兼容。
CmdId	消息类型	变长	每一种消息类型代表一种交易，具体交易类型代码见 <a href="#">接口交易类型定义</a>
MerId	商户号	定长 6 位	由钱管家系统分配的 6 位数字代码，商户的唯一标识
UsrId	用户号	变长 40 位	商户下的平台用户号，在每个商户下唯一
MerUsrId	用户号	变长 40 位	含义同 UsrId
OperId	管理员号	变长 40 位	汇付账户的管理员号
ChkValue	签名	定长 256 位	各接口所列有的请求参数和返回参数如无个别说明，都需要参与签名，参与签名的数据体为：按照每个接口中包含的参数值（不包含参数名）的顺序（按接口表格中从左到右，从上到下的顺序）进行字符串相加。如果参数为可选项并且为空，则该参数值不参与签名。
RespCode	应答返回码	定长 6 位	000000 代表交易成功 其它代表失败，具体参见 <a href="#">标准数据附录</a>
ErrMsg	应答返回码	变长 40 位	如果 RespCode 为失败，则该域为具体的错误信息描述
OrdId	订单号	变长 10~20 位	由商户的系统生成，必须保证唯一。失败的订单 2 天内可以被重复支付。
OrdAmt	订单金额	变长 12 位	订单的总金额，应大于或等于各分账金额总和
CurCode	币种	定长 3 位	固定填写 RMB
Pid	商品编号	变长 20 位	由商户系统产生，含有商品的信息，不作为订单的唯一标识

RetUrl	返回 url	变长 120 位	交易完成后,钱管家系统把交易结果通过页面方式, 发送到该地址上
MerPriv	商户私有域	变长 128 位	为商户的自定义字段, 该字段在交易完成后由钱管家系统原样返回
GateId	银行 ID	定长 2 位	支付的银行号, 具体参见 <a href="#">标准数据附录</a> 。 <b>强烈建议商户置该域为空</b> , 否则新增钱管家系统银行支付渠道, 商户系统也需要作相应改动。如果为空, 则银行的选择页面在钱管家系统完成; 否则, 跳过银行选择页面, 直接跳转到网银系统
UsrMp	购买者手机号	定长 11 位	钱管家系统提供按照手机号查询订单的功能
DivDetails	分账明细	变长 128 位	<p>对于支付: 如果商户仅仅实现支付功能, 无分帐需求, 该域为空; 否则填写的格式为 “角色: 用户名:金额;”, 如果多笔分帐时使用分号分割, 其中角色固定填写 Agent, 用户名为商户下属的用户, 例如订单金额为 1000 元, 分账明细为: Agent:user1:50.00;Agent:user2:800.00 表示分给 user1, 50 元, 分给 user2, 800 元, 剩余的 150 元分给商户本身。</p> <p>对于退款: 填写退款的分账信息, 其格式和支付时填写的一样。对于 CmdId=Refund, 为可选;</p>
OrderType	订单类型	变长 20 位	由商户系统产生, 含有商品的信息, 不作为订单的唯一标识
PayUsrId	付费用户号	变长 40 位	对于航空公司,该域为必填。
TrxId	钱管家交易唯一标识	定长 16 位	组成规则为: 8 位钱管家日期+8 位系统流水号
RetType	返回类型	Char(1)	1: 页面浏览器方式返回, 钱管家通过页面浏览

			<p>器重定向返回。</p> <p>2: 后台数据流方式返回，钱管家系统通过后台数据流的方式发送给商户，对于该方式，商户成功收到之后，必须在页面上输出 RECV_ORD_ID_+订单号的字符串；</p>
PnrNum	Pnr 号	Char(5)	Pnr 号
RefAmt	退款总金额	变 长 12 位	订单退款的总金额，应小于或等于各分账退款金额总和
OldOrdId	原始订单号	变 长 20 位	需要操作（订单退款，分账变更）的原支付成功的订单号
TransAmt	交易金额	变 长 12 位	泛指交易金额，如支付、退款金额
AvlBal	可用余额	变 长 12 位	账户可以支取的余额
TmpBal	临时余额	变 长 12 位	对于主动发起结算的商户，对应于冻结余额
SepBal	专用余额	变 长 12 位	专用账户余额，一般用作退款用途
LiqBal	待结算余额	变 长 12 位	暂不使用
AcctBal	账户余额	变 长 12 位	账户资金余额，该余额能真正反映账户的资金量
ProcStat	处理状态	定长 1 位	<p>0: 支付交易不存在</p> <p>1: 交易已成功支付，并且已经结算</p> <p>2: 交易未支付 或支付失败</p> <p>3: 交易已成功支付，但未结算</p> <p>5: 退款交易，已扣款，系统处理中</p> <p>6: 退款交易，处理失败</p> <p>7: 退款交易，未处理</p>

			8: 退款交易已成功
RefCnt	退款次数	变长 2 位	订单被退款的次数，如次数为 12，则该值为 12
TransType	交易类型	定长 1 位	‘P’ – 支付 ‘R’ – 退款
MerDate	商户日期	定长 8 位	格式为：YYYYMMDD
MerTime	商户时间	定长 8 位	格式为：HHMMSS
BgRetUrl	订单支付时， 商户后台应答 地址	变长 120 位	该域为必填
IsBalance	是否结算	定长 1 位	‘Y’ – 结算 ‘N’ – 不结算

## 6.3. 用户管理类接口

### 6.3.1. 签约

**接口功能说明：**用于在平台上绑定供应商在汇付的账户。请求会跳转到汇付的签约页面，汇付会验证汇付管理员的交易密码，验证通过，在汇付显示成功信息，同时后台应答发送给商户。商户实现后台方式时候，需要在页面上打印 RECV\_ORD\_ID\_和 MerId。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
UsrId	必须	OperId	可选	MerDate	必须
MerTime	必须	BgRetUrl	必须	ChkValue	必须

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	MerId	必须	UsrId	必须
OperId	必须	RespCode	必须	ErrMsg	必须
ChkValue	必须				

### 6.3.2. 签名用户登录

**接口功能说明：**该接口可以实现从商户网站链接过来的用户登录请求

**特别说明：**

登录通用账户用户管理平台的 Web 路径为：/gau/usr\_login.do，其中变量名取 UsrId，

值为必须；

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
MerUsrId	必须	MerDate	必须	ChkValue	必须

返回参数列表：无

### 6.3.3. 用户开户

**接口功能说明：**使用该接口，可以在钱管家系统下开设银账户，用户名必须是手机号。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
UsrId	必须	UsrPwd	必须	UsrName	必须
UsrNameShort	必须	UsrEmail	必须	UsrTel	可选
UsrMp	必须	PwdQues	可选	PwdAns	可选
UsrType	可选	MerGrpId	必须	SaveAcctType	可选
SaveFeeType	可选	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	CustId	必须
RandStr	必须	ErrMsg	可选	ChkValue	必须

## 6.4. 交易类接口

### 6.4.1. 单笔订单支付

**接口功能说明：**该接口实现商户的支付功能，如果商户无分账需求，则参数 DivDetails 为空；

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	OrdAmt	必须	CurCode	可选
Pid	可选	RetUrl	必须	MerPriv	可选
GateId	可选	UsrMp	可选	DivDetails	可选
OrderType	可选	PayUsrId	可选	PnrNum	可选
BgRetUrl	必须	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	MerId	必须	RespCode	必须
TrxId	必须	OrdAmt	必须	CurCode	可选
Pid	可选	OrdId	必须	MerPriv	可选
RetType	必须	DivDetails	可选	GateId	必须
ChkValue	必须				

### 6.4.2. 订单结算

**接口功能说明：**订单支付成功后，需要结算后才能支取。目前，结算有两种方式：第一种，需要由商户主动发起结算；第二种，不需要由商户主动发起结算。该接口实现第一种方式下由商户发起的订单结算，接口中的订单号为原**支付成功的订单号**。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	ChkValue	必须		



返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	ErrMsg	必须
ChkValue	必须				

#### 6.4.3. 订单分帐变更

**接口功能说明：**对于有分账且需要商户主动发起结算的订单，在订单未结算之前，可以对分帐进行变更。接口中的订单号为原**支付成功的订单号**，分账明细为**新的分账变更明细**。

**注：**参数 IsBalance 如果为“Y”，则在分账变更后直接结算；如果为“N”或不填，则只做分账变更。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
DivDetails	必须	OrdId	必须	IsBalance	可选
ChkValue	必须				

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	ErrMsg	必须
ChkValue	必须				

#### 6.4.4. 订单退款

**接口功能说明：**只有在订单结算之后，才能进行订单退款。如果 CmdId= Refund，为普通退款；CmdId=Refund1，为供应商退款给平台；CmdId=Refund2，为平台退款给分销商。接口中的 OrdId 作为一个退款交易的本身订单号，**特别指出，该订单号的最大长度为 30**，OldOrdId 为原**支付成功的订单号**，分账明细为**分账退款明细**。返回参数中，OrdId 和 OldOrdId 对于后台方式才返回，页面方式不返回这两个参数。商户实现后台方式时候，需要在页面上打印 RECV\_ORD\_ID\_和原**支付成功的订单号**。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
DivDetails	可选	RefAmt	必须	OrdId	必须
OldOrdId	必须	BgRetUrl	可选	ChkValue	必须

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	OrdId	可选	OldOrdId	可选
RespCode	必须	ErrMsg	必须	ChkValue	必须

## 6.5. 查询类接口

### 6.5.1. 签约查询

**接口功能说明：**用于查询平台供应商在汇付的账户

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
UsrId	必须	OperId	可选	MerDate	必须
MerTime	必须	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	MerId	必须	UsrId	必须
OperId	必须	RespCode	必须	ErrMsg	必须
ChkValue	必须				

### 6.5.2. 账户余额查询

**接口功能说明：**该接口可以实现用户收款账户的各种余额的查询

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须

UsrId	必须	ChkValue	必须		
-------	----	----------	----	--	--

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	AvlBal	必须
TmpBal	必须	SepBal	必须	LiqBal	必须
AcctBal	必须	ErrMsg	必须	ChkValue	必须

### 6.5.3. 查询支付状态

**接口功能说明：**该接口可以查询订单被支付的状态。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	ProcStat	必须
ErrMsg	必须	ChkValue	必须		

### 6.5.4. 查询订单明细

**接口功能说明：**该接口可以查询订单的比较详细信息。此处版本号为 20。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	UsrId	可选	ChkValue	必须

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	TrxId	必须
OrdAmt	必须	CurCode	可选	Pid	可选

OrdId	必须	DivDetails	可选	MerPriv	可选
RefCnt	必须	RefAmt	必须	ErrMsg	必须
ChkValue	必须				

#### 6.5.5. 退款交易状态查询

接口功能说明：该接口可以查询退款交易状态信息。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	ProcStat	必须
ErrMsg	必须	ChkValue	必须		

#### 6.5.6. 查询分账明细

接口功能说明：该接口可以查询订单分账信息。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	CmdId	必须	MerId	必须
OrdId	必须	ChkValue	必须		

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
CmdId	必须	RespCode	必须	OrdId	必须
DivDetails	可选	ErrMsg	必须	ChkValue	必须

## 6.6. 签名类函数调用接口

### 6.6.1. Java 语言函数说明

功能	数据签名，订单在提交给钱管家之前，需要用商户的私钥产生签名。
类名	SecureLink
函数原型	<code>public int SignMsg( String MerId , String MerKeyFile , String MsgData )</code>
输入项	<p><b>MerId</b> : 商户号，6 位数字，由汇付分配</p> <p><b>MerKeyFile</b> : 商户密钥文件，包含路径和文件名，由汇付分配</p> <p><b>MsgData</b> : 被签名的数据体。</p>
输出项	通过 <code>public String getChkValue()</code> 方法得到签名结果 <b>ChkValue</b> 。
返回值	返回值为零，表示接口调用成功。否则失败，具体的失败返回代码参见 <a href="#">函数返回值说明</a>
备注	这里需用商户的私钥签名

功能	验证数据签名，钱管家将交易应答返回给商户网站后，商户需要使用钱管家的公钥验证签名，以验证该数据是否被篡改过。
类名	SecureLink
函数原型	<code>public int VeriSignMsg ( String PgKeyFile , String MsgData, String ChkValue )</code>
输入项	<p><b>PgKeyFile</b> : 钱管家公钥文件，包含路径和文件名，由汇付提供</p> <p><b>MsgData</b>: 被签名的数据体</p> <p><b>ChkValue</b>: 需要验证的签名</p>
返回值	返回值为零，表示接口调用成功。否则失败，具体的失败返回代码参见 <a href="#">函数返回值说明</a>
备注	这里需用公钥验证签名。

### 6.6.2. C 语言函数说明

功能	数据签名，订单在提交给钱管家之前，需要用商户的私钥产生签名。
----	--------------------------------

库名	对于 Linux 或 Unix 环境，库名为 libnpc.a 对于 Windows 环境的，库名为 pnrpay.dll，控件名为 chinapnr.dll
函数原型	int SignMsg(char *MerId, char *MerKeyFile, char * MsgData, int MsgLen, char*ChkValue )
输入项	<b>MerId</b> ：商户号，6 位数字，由汇付分配 <b>MerKeyFile</b> ：商户密钥文件，包含路径和文件名，由汇付分配 <b>MsgData</b> ：被签名的数据体 <b>MsgLen</b> ：被签名的数据体长度
输出项	签名结果由字符串指针 <b>ChkValue</b> 带回。
返回值	返回值为零，表示接口调用成功。否则失败，具体的失败返回代码参见 <a href="#">函数返回值说明</a>
备注	这里需用商户的私钥签名

功能	验证数据签名，钱管家将交易应答返回给商户网站后，商户需要使用钱管家的公钥验证签名，以验证该数据是否被篡改过。
库名	对于 Linux 或 Unix 环境，库名为 libnpc.a 对于 Windows 环境的，库名为 pnrpay.dll，控件名为 chinapnr.dll
函数原型	int VeriSignMsg(char *PgKeyFile, char *MsgData, int MsgLen, char *ChkValue)
输入项	<b>PgKeyFile</b> ：钱管家公钥文件，包含路径和文件名，由汇付提供 <b>MsgData</b> ：被签名的数据体 <b>MsgLen</b> ：被签名的数据体长度 <b>ChkValue</b> ：需要验证的签名
返回值	返回值为零，表示接口调用成功。否则失败，具体的失败返回代码参见 <a href="#">函数返回值说明</a>
备注	这里需用网关的公钥验证签名。

### 6.6.3. Windows 控件函数说明

Windows 控件与 C 语言的函数说明原型基本一样，只是函数的名字分别为，SignMsg0 和 VeriSignMsg0，其它完全一样，控件文件名为 chinapnr.dll。

### 6.6.4. 函数返回值

钱管家提供的签名函数（包括 Java 和 C）都具有统一的返回值，返回值如下：

返回值	返回值说明	可能的原因和需要采取的措施
0	本次调用成功	
-100	环境变量"NPCDIR"未设置	需要在 Linux 或 Unix 上设置环境变量 NPCDIR
-101	商户密钥文件不存在或无法打开，	设置密钥文件为可写方式。
-102	密钥文件格式错误	可能商户的密钥文件在传输途中有误。
-103	参数错误	请检查参与签名的参数是否正确。
-112	公钥文件不存在或无法打开	设置密钥文件为可写方式。
-113	公钥文件格式错误	可能公钥文件在传输途中有误。
-301	验证签名出错	请检查参与签名的参数是否正确。
-401	参与签名的数据体超过规定长度	

## 6.7. 页面访问 URL

页面访问 URL 为域名+页面访问路径。域名和页面访问路径请查看 6.7.1 和 6.7.2。

### 6.7.1. 域名

钱管家生产系统页面的域名：<https://mas.chinapnr.com>

钱管家生产系统后台的域名：<http://mas.chinapnr.com>

钱管家测试系统页面和后台的域名：<http://116.228.159.51>

### 6.7.2. 页面访问路径

➤ 接口访问路径

接口名称	生产页面访问路径	测试页面访问路径
签约	/gau/UnifiedServlet	/gau/UnifiedServlet
签名用户登录	/gau/usr_login.do	/gau/usr_login.do
单笔订单支付	/gar/RecvMerchant.do	/gar/RecvMerchant.do
订单结算	/gao/entry.do	/gar/entry.do
订单分账变更	/gao/entry.do	/gar/entry.do
订单退款	/gao/entry.do	/gar/entry.do
签约查询	/gau/UnifiedServlet	/gau/UnifiedServlet
账户余额查询	/gaq/entry.do	/gar/entry.do
查询支付状态	/gaq/entry.do	/gar/entry.do
查询订单明细	/gaq/entry.do	/gar/entry.do
退款交易状态 查询	/gaq/entry.do	/gar/entry.do
查询分账明细	/gaq/entry.do	/gar/entry.do

- 商户操作人员登陆钱管家系统的管理控制台路径为/gam
- 用户登陆钱管家系统的用户控制台路径为/gau

## 7. 自动出票

平台商与钱管家对接以后，我们为合作的平台供应商提供高效的出票方式—实现平台 B2B 网电票的自动出票功能。

自动出票的实现需要和汇付的产品—支付窗相配合，以 C/S 结构的方式实现，支付窗为客户端，竞价平台为服务器端，客户端和服务端之间需要有 3 个交互接口：

- 供应商用户验证；
- 支付窗向竞价平台获取需要出票的 PNR 号及相关信息的列表，即获取待出票的 PNR 号；
- 支付窗向竞价平台返回 PNR 号出票结果列表，即返回 PNR 号出票结果。

这 3 个接口都是客户端（PNR 支付窗）主动向服务器（竞价平台）发送请求，并获得服务器



（竞价平台）的应答。

## 7.1. 接口标准参数定义

### 7.1.1. 接口交易类型定义

接口定义如下：

接口类型	接口类型名称	接口技术实现方式
用户验证接口	供应商用户验证	HTTP 协议
查询接口	获取待出票的 PNR 号	HTTP 协议
返回接口	返回 PNR 号的出票结果	HTTP 协议

### 7.1.2. 接口常用标准参数定义

以下对接口中所使用的参数进行统一定义，各个接口中只说明是否含有该参数，具体参数的属性如无特别说明，均以此处定义为准；与标准参数定义有差别的，在每个接口定义中，会特别说明。

变量名	名称	长度	说明
Version	版本号	定长 2 位	目前固定为 10。如版本升级，能向前兼容。
ReqType	自动出票的请求类型	定长 1 位	该域为必填， 'C' — 验证供应商身份 'Q' — 获取待出票的 PNR 号 'R' — 返回出票结果
ReqDate	请求日期	定长 8 位	格式为：YYYYMMDD
ReqTime	请求时间	定长 8 位	格式为：HHMMSS
Username	供应商用户名	变长 40 位	供应商在竞价平台上的登录用户名
Password	供应商密码	定长 32 位	供应商在竞价平台上的登录密码的 MD5 编码
AuthCode	验证码	变长 8 位	验证码
Airlines	航空公司列表	变长 40	已经登录的航空公司 2 位码列表，可以有多个，

		位	<p>之间用 “,” 分隔</p> <p>CA: 中国国际航空</p> <p>MU: 东方航空</p> <p>HU: 海南航空</p> <p>ZH: 深圳航空</p> <p>FM: 上海航空</p> <p>SC: 山东航空</p> <p>3U: 四川航空</p> <p>HO: 吉祥航空</p> <p>KN: 中国联合航空</p> <p>PN: 西部航空</p> <p>JD: 金鹿航空</p> <p>GS: 天津航空</p>
Suppliers	供应商列表	变长 120 位	<p>供应商在竞价平台上的用户名，因为供应商在同一个平台商可以对应多个用户名，所以这个字段是供应商的列表，每个供应商之间用 “,” 分隔。</p> <p>每个供应商要包含两个字段，这两个字段之间用 “^” 分隔：</p> <p>供应商用户名；</p> <p>供应商用户名 和 供应商在验证接口中返回的 “SupplierKey” 相加后的 MD5 编码</p>
Supplier	供应商	变长 60 位	<p>每个供应商要包含两个字段，这两个字段之间用 “^” 分隔：</p> <p>供应商用户名；</p> <p>供应商用户名 和 供应商在验证接口中返回的 “SupplierKey” 相加后的 MD5 编码</p>
GUID	此 PNR 号对应的唯一编码	变长 20 位	<p>因为 PNR 号可以重复使用，不能作为支付窗和平台交互的唯一索引值，需要平台指定一个唯</p>

			一编码，这个唯一编码可以是平台数据库中的唯一索引值
DrawResult	出票结果	定长 6 位	<p>出票结果的返回码</p> <p>00000: 出票成功</p> <p>00001: 竞价平台上非汇付支付</p> <p>00002: 航空公司网站的用户名不存在</p> <p>00003: 航空公司网站的登录密码错误</p> <p>00004: 有多个不同的票面价</p> <p>00005: 入库失败 +（具体的失败描述）</p> <p>00006: 支付失败 +（具体的失败描述）</p> <p>00007: 出票失败 +（具体的失败描述）</p> <p>这个字段是开放的，可以不断扩展</p>
DrawResultDesc	出票结果的具体描述	变长 没有限制	<p>对应以上结果码的具体描述。</p> <p>若出票成功，为票号信息，格式为“票号+旅客姓名”，多个票号之间用“;”分隔；</p> <p>若出票失败，为错误的具体描述。</p>
Rsv1	保留字段	变长 128 位	保留字段，供不同平台的自己需要特殊使用的字段
Rsv2	保留字段	变长 128 位	保留字段，供不同平台的自己需要特殊使用的字段
ChkValue	签名	定长 16 位	<p>各接口所列有的请求参数和返回参数如无个别说明，都需要参与签名，参与签名的数据体为：按照每个接口中包含的参数值（不包含参数名）的顺序（按接口表格中从左到右，从上到下的顺序）进行字符串相加。</p> <p>然后用一个平台和支付窗共同维护的一个 KEY 与以上相加后的字符串再相加，最后再用 MD5 算法编码。</p>
RespCode	应答返回码	定长 6 位	000000 代表成功

			其它表示失败
SupplierKey	供应商验证码	定长 16 位	用户验证时，如果 RespCode 为成功，返回一个供应商验证码，这个验证码在查询待出票 PNR 号时的验证口令
ErrMsg	应答返回码	变长 40 位	如果 RespCode 为失败，则该域为具体的错误信息描述

## 7.2. 用户验证接口

**接口功能说明：**在打开支付窗时，支付窗要向竞价平台发送供应商的身份验证信息，这样才能绑定支付窗的供应商用户与竞价平台上的该供应商的出票信息；

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	ReqType	必须	ReqDate	必须
ReqTime	必须	Username	必须	Password	必须
AuthCode	可选	Rsv1	可选	ChkValue	必须

返回参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
ReqType	必须	Username	必须	RespCode	必须
SupplierKey	必须	ErrMsg	必须	ChkValue	必须

```
<?xml version="1.0" encoding="GB2312" ?>
<result>
<ReqType>C</ReqType>
<Username>b2bauto</Username>
<RespCode>000000</RespCode>
<SupplierKey>C1B2D76FAD792E473271C38C80532FA0</SupplierKey>
<ErrMsg />
<ChkValue>EA30F7DFD4E56FB4EF567D6F7BEDDB48</ChkValue>
</result>
```

## 7.3. 查询接口

**接口功能说明：**支付窗主动向竞价平台发起请求，向平台查询等待出票的 PNR 号；

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	ReqType	必须	ReqDate	必须
ReqTime	必须	Airlines	必须	Suppliers	必须
Rsv1	可选	Rsv2	可选	ChkValue	必须

返回说明：

每次支付窗向竞价平台发送“获取待出票的 PNR 号”请求后，竞价平台根据查询请求中的航空公司、供应商名查询数据库，返回未出票的 PNR 号列表。返回的结果集为 XML 格式的文件，每次可以返回多个待出票的 PNR 号，一次可以返回多个 PNR 号，具体的个数可以由我们合作的平台上来指定。已经被返回的 PNR 号，在竞价平台的数据库中标识为“已被获取”。文件的格式如下

```
<?xml version="1.0" encoding="GBK" ?>

<PNRsRoot>

  <ResultSet>

    <Result>

      <GUID>12342354356</GUID>

      <PnrNo>KKCFW</PnrNo>

      <Pnr>DGCET</Pnr>

      <Airline>CA</Airline>

      <CPnROderID>HF12345678</CPnROderID>

      <AgencyPct>6</AgencyPct>

      <FaceValue>1000.00</FaceValue>

      <Build>50</Build>

      <Oil>50</Oil>
```

```
</Result>

<Result>

  <GUID>12342354356</GUID>

  <PnrNo>FTF4M</PnrNo>

  <Pnr>DGCET</Pnr>

  <Airline>CA</Airline>

  <CPnROderID>HF12345679</CPnROderID>

  <AgencyPct>6</AgencyPct>

  <FaceValue>1000.00</FaceValue>

  <Build>50</Build>

  <Oil>50</Oil>

</Result>

</ResultSet>

<ResultCn>2</ResultCn>

<Date>20090721</Date>

<Time>091002</Time>

</PNRsRoot>
```

文件格式说明：

XML 的根键为 PNRsRoot，有 3 个子键

➤ ResultSet：返回的 PNR 号集合，若没有 PNR 号返回，没有子键；

ResultSet 的子键为 Result，每个 Result 包含一个 PNR 号的相关信息，Result 包含的子键如下：

1、GUID：该 PNR 号对应的唯一编码，因为 PNR 号不是唯一的，竞价平台必须给出一

个唯一码，才能与出票完成后的结果对应起来；

2、PnrNo：待出票的 PNR 号，为大编号；

3、Pnr：待出票的 PNR 号，为小编号；

3、Airline：该 PNR 号对应的航空公司；

4、CPnrOrderID：该 PNR 号在平台上通过汇付支付的订单号，可以为空；

5、AgencyPct：因为竞价平台上不维护 B2B 网站上的入库扣点，所以该扣点只能为供应商在竞价平台上发放的扣点。这个值在支付窗出票时，只能作为一个不是很严格的参考，根据代理人自己的设定给出判断方法，比如：入库时得到的最大扣点小于这个值时，将中断操作。

6、FaceValue：该 PNR 号在竞价平台上的票面价；

7、Build：该 PNR 号在竞价平台上的机场建设费；

8、Oil：该 PNR 号在竞价平台上的燃油费加费。

- ResultCn：返回 PNR 号的个数，若没有 PNR 号返回，值为 0。
- RetDate：本次请求的应答日期，格式为“YYYYMMDD”。
- RetTime：本次请求的应答时间，格式为“HHNNSS”。

#### 7.4. 返回接口

**接口功能说明：**支付窗向竞价平台返回出票结果，每出票完成一个 PNR 号就返回一个出票结果给平台。

请求参数列表：

变量名	是否必须	变量名	是否必须	变量名	是否必须
Version	必须	ReqType	必须	ReqDate	必须
ReqTime	必须	GUID	必须	DrawResult	必须
DrawResultDesc	必须	Supplier	可选	Rsv1	可选
Rsv2	可选	ChkValue	必须		

返回说明：

变量名	是否必须	变量名	是否必须	变量名	是否必须
ReqType	必须	GUID	必须	RespCode	必须
ErrMsg	必须	ChkValue	必须		

- 1、支付窗由于网络的原因，没有拿到平台返回的 PNR 号信息；
- 2、由于网络原因，平台没有接受到支付窗返回的出票结果信息；
- 3、由于平台或者支付窗异常，导致平台没能接受到出票结果信息。

## 8. 典型插件安装说明

### 8.1. 插件在 Java 应用服务器的安装

基于 Java 的应用服务器的安装比较简单，只需要在操作系统的 CLASSPATH 环境变量或者应用服务器的 JAR 配置中包含 chinapnr.jar 的路径即可。

### 8.2. 插件在 Windows 应用服务器的安装

#### 8.2.1. 自动安装

将安装包解开后运行，安装程序会将指定文件复制到指定目录下。

#### 8.2.2. 手动安装

如果安装程序运行后出错（可能是由于系统环境异常导致），可采用手工安装的方法，具体步骤如下：

将核心文件 copy 到指定的目录下；

注册 chinapnr.dll（在命令行里键入 regsvr32 [gensign 的路径]，此控件需要 VC 环境。建议是用 Dependency Walker 看一下系统是否包含有所需的 DLL）；

### 8.3. 插件在类 Unix/Linux 系统上的安装

下载 NetPayClient 的 UNIX/Linux 版，解包后，即得 libnpc.a（库文件），netpayclient.h。

UNIX/Linux 版本下 JAVA 的实现请参考[插件在 Java 应用服务器的安装](#)，PHP 的实现请参考[插件在 PHP 应用服务器的安装](#)。



## 8.4. 插件在 PHP 应用服务器的安装

1. 下载并解包 Apache 及 PHP，此处假设 Apache 解包后的源文件目录为 `apache_src`，PHP 解包后的源文件目录为 `php_src`。
2. 按照标准流程安装 Apache 和 PHP，并能保证其可正常运行
3. 将文件 `php_crypto.h`, `php_crypto.c`, `netpayclient.h` 拷贝到 `php_src/ext/standard/` 目录下
4. 将文件 `libnpc.a` 拷贝到 `/usr/local/lib` 目录下
5. 修改 `php_src/ext/standard/` 中的文件 `basic_functions.c`:

在 `function_entry basic_functions[] = {` 中加入以下两行：

```
    PHP_FE(SignMsg,  NULL)

    PHP_FE(VeriSignMsg, NULL)

    .
    .
    .
}
```

修改 `php_src/ext/standard/` 中的文件 `basic_functions.h`:

加入一行：`#include "php_crypto.h"`

6. 修改 `php_src/` 中的文件 `Makefile`，有两处需要修改：
  - a) 找到 `EXTRA_LIBS`，并在该行的最后加入 `-lnpc`，注意加入内容要用空格隔开
  - b) 找到 `basic_functions.lo` 的声明处，类似下面代码：

```
ext/standard/basic_functions.lo: php_src/ext/standard/basic_functions.c

$(LIBTOOL) --mode=compile $(CC) -Iext/standard/ -I php_src/ext/standard/
$(COMMON_FLAGS) $(CFLAGS_CLEAN) $(EXTRA_CFLAGS) -prefer-non-pic -c
php_src/ext/standard/basic_functions.c -o ext/standard/basic_functions.lo
```

加入对 `php_crypto.c` 的依赖，即：

```
ext/standard/basic_functions.lo: php_src/ext/standard/basic_functions.c
php_src/ext/standard/php_crypto.c

$(LIBTOOL) --mode=compile $(CC) -Iext/standard/ -I php_src/ext/standard/
```

```
$(COMMON_FLAGS) $(CFLAGS_CLEAN) $(EXTRA_CFLAGS) -prefer-non-pic -c  
php_src/ext/standard/basic_functions.c php_src/ext/standard/php_crypto.c -o  
ext/standard/basic_functions.lo
```

7. 在 **php\_src**/下, make , make install

8. 运行如下命令:

```
/usr/local/bin/php -r "echo SignMsg("880001", "key", "000001", "6", "20050916");"
```

有错误码-101 返回, 说明安装成功。

## 9. 标准数据附录

为方便商户接口的开发, 支付银行代号, 开户银行代号, 省份代号, 地区代号, 接口返回码、交易报表等标准数据均另外以 Excel 的文件格式提供给商户, 在此文档中不再提供。