福清市医院
医院统一支付平台(腕带付)

{JAVA-SDK 使用说明}

易联众信息技术股份有限公司
2016 年 8 月

变更日志

变更日期	版本	变量内容	变更人
			陈俊海
			孙志伟
			陈俊海
			陈俊海

目录

一、 接入说明
二、 接口规则

1. 接口规定

1.1 交易金额要求

交易金额默认为人民币交易，接口中参数支付金额单位为【元】，对账单中的交易金额单位为【元】。

1.2 交易时间戳要求

【yyyyMMddHHmmss】。

1.3 商户订单号要求

商户支付、退款的外部订单号由商户自定义生成，并要求商户订单号保持唯一性（建议根据当前系统时间加随机序列来生成订单号）。重新发起一笔支付要使用原订单号，避免重复支付，如该笔订单已支付过或已调用退款、撤销（请见后文的 API 列表）的订单号不能将支付失败。

1.4 交易类型字典(tradeType)

交易类型字典	交易类型名称
hop.account.info	查询账户信息
hop.trade.recharge	充值
hop.trade.refund	退款
hop.trade.query	查询交易结果
hop.trade.details	查询交易明细

1.5 充值方式字典（depositeType）

充值方式入参	充值方式名称
CASHIER	收银台
ONE_QR	一码付
CASH	现金
POS	POS 刷卡
POS_OFF	离线 POS（HIS 脱机交易）
TRANS_ UNIT	单位转账
ALI_QR	支付宝二维码支付

ALI_BAR	支付宝条形码支付
ALI_APP	支付宝移动 APP 支付
ALI_WEB	支付宝 PC 网页支付
ALI_WAP	支付宝手机 WAP 支付
WX_QR	微信二维码支付
WX_PUB	微信公共号支付
WX_BAR	微信条形码支付
WX_APP	微信移动 APP 支付
UP_WEB	银联网页
UP_WAP	银联 WAP
JKT_APP	健康通 APP

1.6 退款方式字典（withdrawMode）

支付渠道入参	支付渠道名称
0	现金退款
1	原路退回
~~2~~	~~银企直连~~
~~3~~	~~医保自费部份退款~~
4	离线 POS（HIS 脱机交易）
5	单位转账

1.7 交易状态字典(result)

交易类型字典	交易类型名称
init	待交易
succ	交易成功
fail	交易失败
exp	交易异常
ing	正在交易
closed	交易关闭
notfound	交易记录不存在

1.8 账户类型字典(accountType)

账户类型字典	类型名称
01	门诊账户
02	住院账户

1.9 卡类型字典(cardType)

卡类型字典	类型名称
01	医（社）保卡
02	健康卡
03	就诊卡
04	身份证
05	住院病案号
06	临时卡
07	腕带编码

1.10 性别字典(idType)

性别字典	类型名称
1	男
2	女

1.11 账户状态字典(status)

账户状态字典	类型名称
<<门诊账户状态>>	
01	正常
02	挂失
03	冻结
04	注销
<<住院账户状态>>	
01	在院
02	出院证明（出院证明才能行结算操作）
03	出院（出院不能进行充值操作）

1.12 证件类型字典(idType)

证件类型字典	类型名称
01	身份证

1.13 渠道类型字典(channelType)

渠道类型字典	类型名称
01	窗口
02	自助终端
04	手机 APP
07	微信小程序
08	支付宝服务窗

2. 错误码说明

00000 交易成功

……

3. 安全规范

1.1 签名算法

支付平台提供的 Server SDK 已将 MD5 签名验签方法封装，开发者只需要调用 sdk 配置业务入参即可，用 sdk 封装的方法发送请求到开放平台时，sdk 会自动签名。

如开发者不用 sdk，可根据 MD5 规则自己拼写签名方法。以下是结合开放平台业务对自主签名进行简单说明：

请求参数签名

1. 筛选

获取所有请求参数，不包括字节类型参数，如文件、字节流，剔除 sign 字段。

2. 排序

将筛选的参数按照第一个字符的键值 ASCII 码递增排序（字母升序排序），如果遇到相同字符则按照第二个字符的键值 ASCII 码递增排序，以此类推。

3. 拼接

将排序后的参数与其对应值，组合成“参数=参数值”的格式，并且把这些参数用&字符连接起来，最后拼接上应用密钥在“…参数=参数值&key=你的密钥”，此时生成的字符串为待签名字符串，将待签名字符串 MD5 运算，即是签名（sign）的值。

“SIGN”与“ENCRYPTDATA”参数不参与签名

例如下面的请求示例，参数值都是示例，开发者参考格式即可：


```
{
    "appId": "1BN7VC8VT0007518A8C000006602D80D",
    "encryptType": "AES",
    "param": {
        "ghfy00": "202",
        "bcsfcs": "1",
        "bqbm02": "000003000003",
        "bqbm01": "000003000002",
        "jslx00": "01",
        "bqbm00": "000003000001",
        "sfydgh": "N",
        "yydjh0": "YY10150011",
        "yszjhm": "350626198510201017",
        "sfyjs0": "Y",
        "bcghcs": "2",
        "ysxm00": "医生",
```

```
        "cfxms0": "1",
        "zdlist": "[{\"zdbh00\":\"ZZXY10\",\"zdms00\":\"心阴不足证\"}]",
        "fyzje0": "2950",
        "sfygwd": "Y",
        "ghksmc": "YLZ",
        "tsbzbm": "T00003000001",
        "sftjgh": "Y",
        "mzlb00": "101",
        "yszjdj": "111",
        "sfsj00": "101234",
        "cardno": "DB0748478",
        "yyksbm": "YLZ",
        "mxlist": "[{\"sptxbm\":\"3333333333\",\"xmdw00\":\"瓶\",\"ysxm00\":\"帅\",\"xmdj00\":\"2950.00\",
\"qyzldw\":\"*\",\"fpxmmc\":\"fp0\",\"yppl00\":\"0080\",\"ypyl00\":\"1 粒/次\",\"cfhao0\":\"*\",\"jx0000\
":\"*\",\"xmsl00\":\"1\",\"xmmc00\":\"乐力\",\"sfybxm\":\"N\",\"ylts00\":\"1\",\"xmbh00\":\"10703044195932
330501\",\"gytj00\":\"门诊\",\"qyzl00\":\"*\",\"dcyydw\":\"*\",\"ypts00\":\"1\",\"xmje00\":\"2950\",\"xmgg
00\":\"1 克/30 粒\",\"fpxmbh\":\"01\"}]",
        "sfrq00": "20170524"
    },
    "sign": "15DA8F79D870FA4DCF8031D54253FC35",
    "signType": "MD5",
    "timestamp": "20170831092820",
    "transType": "mmp.upload.medical.detail",
    "version": "H.F.0.1"
}
```


组成的待签名字符串为：


```
1  appId=1BN7VC8VT0007518A8C000006602D80D&encryptType=AES&param={"bcghcs":"2","bcsfcs":"1
","bqbm00":"000003000001","bqbm01":"000003000002","bqbm02":"000003000003","cardno":"DB0748478","cfxm
s0":"1","fyzje0":"2950","ghfy00":"202","ghksmc":"YLZ","jslx00":"01","mxlist":"[{\"sptxbm\":\"3333333
333\",\"xmdw00\":\"瓶\",\"ysxm00\":\"帅\",\"xmdj00\":\"2950.00\",\"qyzldw\":\"*\",\"fpxmmc\":\"fp0\
",\"yppl00\":\"0080\",\"ypyl00\":\"1 粒/次\",\"cfhao0\":\"*\",\"jx0000\":\"*\",\"xmsl00\":\"1\",\"xmm
c00\":\"乐力\",\"sfybxm\":\"N\",\"ylts00\":\"1\",\"xmbh00\":\"10703044195932330501\",\"gytj00\":\"门
诊\",\"qyzl00\":\"*\",\"dcyydw\":\"*\",\"ypts00\":\"1\",\"xmje00\":\"2950\",\"xmgg00\":\"1 克/30 粒\",
\"fpxmbh\":\"01\"}]","mzlb00":"101","sfrq00":"20170524","sfsj00":"101234","sftjgh":"Y","sfydgh":"N",
"sfygwd":"Y","sfyjs0":"Y","tsbzbm":"T00003000001","ysxm00":"医生","yszjdj":"111","yszjhm":"350626198
510201017","yydjh0":"YY10150011","yyksbm":"YLZ","zdlist":"[{\"zdbh00\":\"ZZXY10\",\"zdms00\":\"心阴
不足证\"}]"}&signType=MD5&timestamp=20170831092820&transType=mmp.upload.medical.detai
l&version=H.F.0.1&key=1BN7VC8VU0017518A8C0000083776CB9
```


4. 签名结果

使用各自语言对应的 MD5 签名函数，对拼接商户私钥得出的待签名字符串进行 MD5 签名后，再将字节码转换成 16 进制字符串，并对转换后的字符串转换成大写，即是签名结果，如 B773BC783F1319B384B242BF874A1F94。

返回参数验证签名

(与请求签名类似)

1. 筛选

获取所有请求参数，不包括字节类型参数，如文件、字节流，剔除 sign 字段。

2. 排序

将筛选的参数按照第一个字符的键值 ASCII 码递增排序（字母升序排序），如果遇到相同字符则按照第二个字符的键值 ASCII 码递增排序，以此类推。

3. 拼接

将排序后的参数与其对应值，组合成“参数=参数值”的格式，并且把这些参数用&字符连接起来，最后拼接上应用密钥在“…参数=参数值&key=你的密钥”，此时生成的字符串为待签名字符串，将待签名字符串 MD5 运算，即是签名（sign）的值。

“SIGN”与“ENCRYPTDATA”参数不参与签名

例如下面的请求示例，参数值都是示例，开发者参考格式即可：

如微信支付请求的返回内容为：

```
{
    "param": {
        "chargeNo": "201609221543522572",
        "credential": {
            "WX_APP": {
                "appid": "wxe4fa81d8fd121a22",
                "nonceStr": "4otknlnyu3v1kddlkagsymp6zojmxvhf",
                "package": "Sign=WXPay",
                "partnerid": "1270560301",
                "prepayid": "wx20160922154537da4a45a89c0584844458",
                "sign": "F8598C909881E6EB42631DB7087E84A6",
                "timeStamp": "1475126810"
            }
        },
        "outChargeNo": "201609221055122142"
    },
    "respCode": "000000",
    "respMsg": "处理成功",
    "sign": "A9085E68A4D545628A009065AE0ECFE2",
    "signType": "MD5",
    "encryptType": “DES”,
    "encryptData": "XXXXX",
    "timestamp": "20160929132650301"
}
```

则待验签内容为：

```
encryptType=AES&param={"chargeNo":"201609221543522572","credential":{"WX_APP":{"appid":"wxe4fa81d8fd121a22","nonceStr":"4otknlnyu3v1kddlkagsymp6zojmxvhf","package":"Sign=WXPay","partnerid":"1270560301","prepayid":"wx20160922154537da4a45a89c0584844458","sign":"F8598C909881E6EB42631DB7087E84A6","
```

```
timeStamp":"1475126810"}},"outChargeNo":"201609221055122142"}&respCode=000000&respMsg=处理成功
&signType=MD5&timestamp=20160929132650301&key=1A3VL0KVE0010B010A0A0000277BDC91
```

4. 签名结果

调用签名函数：使用各自语言对应的 MD5 签名方法，对拼接商户私钥得出的待签名字符串进行 MD5 签名后，再将字节码转换成 16 进制字符串，并对转换后的字符串转换成大写，即是签名结果，如 B773BC783F1319B384B242BF874A1F94，再比较与返回报文里的 sign 字段是否相等，根据比较结果判定是否验签通过。

1.2 加密算法

支付平台提供 JAVA 版本 Server SDK 已将加解密方法封装，开发者只需要调用 sdk 配置业务入参即可，用 sdk 封装的方法发送请求到开放平台时，sdk 会自动加解密。

如开发者不用 sdk，可根据 AES 或 DES 加密算法，自己加解密 encryptData 参数。以下是结合开放平台业务对自主加解密进行简单说明：

请求报文加密

1. 组装请求报文

根据 API 列表定义参数，整理请求报文

```
{
    "appId": "1A3VL0KVK0000B020A0A0000CC3F48AD",
    "encryptType": "AES",
    "param": {
        "channel": "WX_QR",
        "chargeAmt": 1,
        "chargeGoodsDetail": [
            {
                "chargeAmt": 1,
                "goodsInfo": "明细信息",
                "goodsName": "明细标题",
                "orgPrice": 1,
                "price": 1,
                "quantity": "1",
                "remark": "备注"
            }
        ],
        "extra": { },
        "outChargeNo": "OC12260001",
        "outChargeTime": "20161226093148",
        "returnUrl": "http://120.42.37.94:1301/onepay-test-web/returnUrl",
        "showUrl": "http://www.ylzinfo.com",
        "subject": "易惠-支付"
    },
    "sign": "FA57D29EBC64622116F5B9272AB407AF",
```

```
    "signType": "MD5",
    "timestamp": "20161226093149",
    "transType": "op.trade.charge"
}
```

2. 待加密串

将 param 字段值，转换为 JSON 字符串 jStr：

{"channel":"WX_QR","chargeAmt":1,"chargeGoodsDetail":[{"chargeAmt":1,"goodsInfo":"明细信息","goodsName":"明细标题","orgPrice":1,"price":1,"quantity":"1","remark":"备注"}],"extra":{},"outChargeNo":"OC12260001","outChargeTime":"20161226093148","returnUrl":"http://120.42.37.94:1301/onepay-test-web/returnUrl","showUrl":"http://www.ylzinfo.com","subject":"易惠-支付"}

3. 报文加密密钥

根据 encryptType 声明加密算法 AES *(AES 算法加密参数见末尾备注)*，截取 appId 前 16 位作为 AES 密钥，如 appId.substring(0, 16)加密 appSecret，**并将加密结果转 16 进制，再将 16 进制串转换为大写**，即获得*报文加密密钥*;

（appId 密钥需根据不同的加密算法取不同的长度做密钥，如 AES 截取前 16 位，DES 截取前 8 位）

如：String newPassword = AESUtil.encrypt(appSecret, appId.substring(0, 16));
// 待加密内容 appSecret
// 加密密钥 appId.substring(0, 16)

4. 获取密文

根据 encryptType 声明加密算法 AES，使用 3 获得的报文加密密钥，截取 newPassword 前 16 位作为 AES 密钥，如 newPassword.substring(0, 16)加密 jStr 字符串，**并将加密结果转 16 进制，再将 16 进制串转换为大写**，获得加密密文 encData:

0FE9C0E3DA4A94F7014DBA93C34D481218862944C2A84D69EC4E9B73E5EE6DAE169CF174A8F674D670883BC939CDC10D7E9459D1C2BFE22843A8B3B5B443FBD6A279E86038DCE8244BB147FF4CAC75310596AEE3898626EDC61E354EEADDA0B2FF4F3A1FA9DC138665CDA9725F81316095FC6B1EFD96D4C0DA111135EE25BF1D41CCE243B8925C48FF145530711494E199B30377432AD9266A594D38C8E1725C9DFE14E4E0EF2BD94F8B0FED295CD15FEF2E489238230AEDBA1784C8D2105E76C0B3FB4E63FD8D55BA57399E3A04C22FB3660D7B9DBA4DD4A8F53032236C8D9534D3594F38EA1D7650124CE5D9DF09BCC27BFDC06681E06E666B14A5C5C85F44C7A22357E44604BD65E9E47DFF68101054EC9C427D5C6897CA947ADA16C72A68F1CF38B566A9CA54B2E8C66278CCCDD9629CAA45F38EAE2F81755ACAFADC946118D4ACB0FE8251B015FB6DCF4F83BEFB2598BCFAAF7F53115CA5B586374C1B488280E3778ED5B0385F30DD8C8AEC59238B50CE9CEBB14AACC09061256F02645A

5. 设置密文

将加密结果 encData，赋值 encryptData，并清空 param 明文 JSON，将新获取报文发送服务器

如：String encryptData = AESUtil.encrypt(jStr, newPassword.substring(0, 16));
// 待加密内容 data

```
// 加密密钥 newPassword.substring(0, 16)
```

```
{
    "appId": "1A3VL0KVK0000B020A0A0000CC3F48AD",
    "encryptData": "0FE9C0E3DA4A94F7014DBA93C34D481218862944C2A84D69EC4E9B73E5EE6DAE169CF174A8F674D670883BC939CDC10D7E9459D1C2BFE22843A8B3B5B443FBD6A279E86038DCE8244BB147FF4CAC75310596AEE3898626EDC61E354EEADDA0B2FF4F3A1FA9DC138665CDA9725F81316095FC6B1EFD96D4C0DA111135EE25BF1D41CCE243B8925C48FF145530711494E199B30377432AD9266A594D38C8E1725C9DFE14E4E0EF2BD94F8B0FED295CD15FEF2E489238230AEDBA1784C8D2105E76C0B3FB4E63FD8D55BA57399E3A04C22FB3660D7B9DBA4DD4A8F53032236C8D9534D3594F38EA1D7650124CE5D9DF09BCC27BFDC06681E06E666B14A5C5C85F44C7A22357E44604BD65E9E47DFF68101054EC9C427D5C6897CA947ADA16C72A68F1CF38B566A9CA54B2E8C66278CCCDD9629CAA45F38EAE2F81755ACAFADC946118D4ACB0FE8251B015FB6DCF4F83BEFB2598BCFAAF7F53115CA5B586374C1B488280E3778ED5B0385F30DD8C8AEC59238B50CE9CEBB14AACC09061256F02645A",
    "encryptType": "AES",
    "sign": "FA57D29EBC64622116F5B9272AB407AF",
    "signType": "MD5",
    "timestamp": "20161226093149",
    "transType": "op.trade.charge"
}
```

返回报文解密

1. 获取响应报文

```
{
    "encryptData": "0FE9C0E3DA4A94F7014DBA93C34D48123FDDCA34998537D8F12883A9C92D5CC608955AB6E7815BBABC711363373684A75CB30986C2639571FA46A715C86025C7EA1B10ED135AADEFCD007B7EAE65BC402C19EC50CC4ABF9A7C6972F25AB773F4111312C183ACFF6163728B5AA6AE56663FCEE0F6A98AC1AC899608D3BC33E5C67116EA21A53450CD9C8229FE555F045A6C6944B8A3338EFC1B6FD9FCECF5F2B5",
    "encryptType": "AES",
    "respCode": "000000",
    "respMsg": "处理成功",
    "sign": "17122A2041288920CA16A62752FB38CB",
    "signType": "MD5",
    "timestamp": "20161226093147927"
}
```

2. 获取加密密文数据 encryptData

0FE9C0E3DA4A94F7014DBA93C34D48123FDDCA34998537D8F12883A9C92D5CC608955AB6E7815BBABC711363373684A75CB30986C2639571FA46A715C86025C7EA1B10ED135AADEFCD007B7EAE65BC402C19EC50CC4ABF9A7C6972F25AB773F4111312C183ACFF6163728B5AA6AE56663FCEE0F6A98AC1AC899608D3BC33E5C67116EA21A53450CD9C8229FE555F045A6C6944B8A3338EFC1B6FD9FCECF5F2B5

3. 报文解密密钥

根据 encryptType 声明加密算法，使用 appId 密钥加密 appSecret，并将加密结果转换为大写，再根据不同加密算法截取长度（AES 截取前 16 位、DES 截取前 8 位），即获得报文解密密钥；

（appId 密钥是根据不同的加密算法取不同的长度做密钥，如 AES 截取前 16 位，DES 截取前 8 位）

4. 获取明文

根据 encryptType 声明加密算法，使用 3.获取的报文解密密钥，解密 encryptData 获取 JSON 字符串明文 jStr

{"channel":"WX_QR","chargeNo":"201612260931476856","chargeStatus":"1","outChargeNo":"OC12260001","qrcode":"weixin://wxpay/bizpayurl?pr=75gzLi6"}

5. 设置明文

将 jStr 转换为 JSON 赋值 param，获取解密后返回报文


```
{
    "encryptType": "AES",
    "param": {
        "channel": "WX_QR",
        "chargeNo": "201612260931476856",
        "chargeStatus": "1",
        "outChargeNo": "OC12260001",
        "qrcode": "weixin://wxpay/bizpayurl?pr=75gzLi6"
    },
    "respCode": "000000",
    "respMsg": "处理成功",
    "sign": "17122A2041288920CA16A62752FB38CB",
    "signType": "MD5",
    "timestamp": "20161226093147927"
}
```


备注：

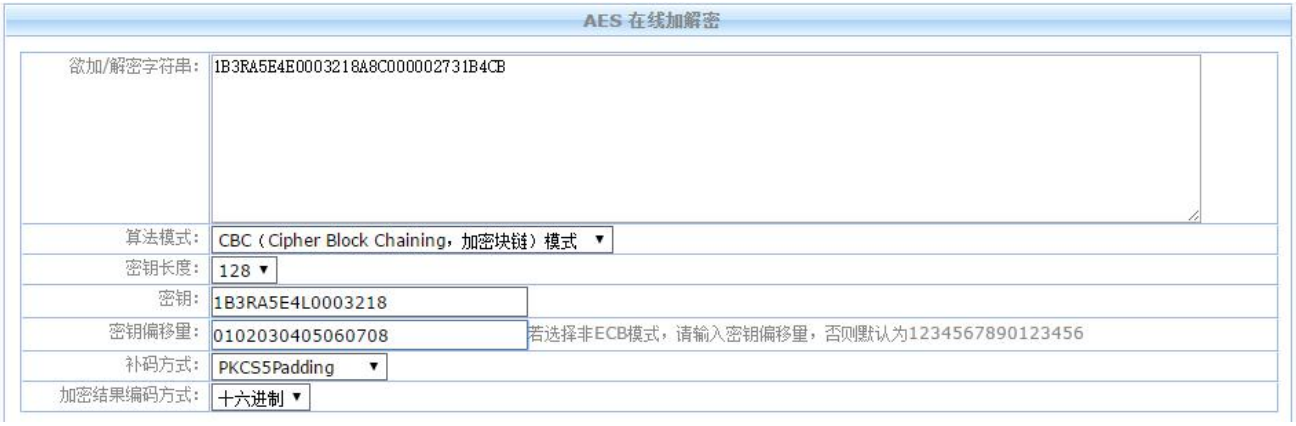
AES 加密参数

加密算法 AES/CBC/PKCS5Padding

加密模式 1

初始向量（密钥偏移量） 0102030405060708

字符编码 UTF-8



在线加解密工具地址：http://www.seacha.com/tools/aes.html

三、 平台接口说明

1. 查询腕带账户信息

入出参：

用途说明	查询腕带账户信息				
请求地址	http://127.0.0.1:8080/onepay-web/oneweb/oneapi				
	portal.wb.queryAccount				
请求入参（param）	名称	类型长度	可空	说明	备　注
	key	是	String	腕带码编号	
响应出参（param）	名称	类型长度		说明	备　注
	balance	是	string	账户余额	balance
	idNo	是	string	证件号	idNo
	userName	否	string	账户姓名	userName

简要描述：

- 查询腕带账户信息

请求 URL：

- http://127.0.0.1:8080/onepay-web/oneweb/oneapi

请求方式：

- POST

交易类型(transType)

- portal.wb.queryAccount

请求报文加密后

```
{
    "appId": "1CLT55NO6001790BA8C00000B14B2D07",
    "encryptData": "DBADBF1C20303C1A70B737F722CCC740FEDCC3DF937BD7A4863E9CF38DE1566A",
    "encryptType": "AES",
    "sign": "56D9153A64F1C30CCB610D832C0B57E4",
    "signType": "MD5",
    "timestamp": "20181109172804",
    "transType": "portal.wb.queryAccount"
}
```

响应报文示例

```
{
    "termNo": null,
```

```
    "respCode": "000000",
    "respMsg": "处理成功",
    "param": {
        "idNo": "66",
        "cardNo": "D12345678",
        "idType": "01",
        "appId": "1CLT55NO6001790BA8C00000B14B2D07",
        "operatorName": "腕带付",
        "operatorId": "123",
        "hospName": "福清医院吊炸天",
        "accountType": "02",
        "cardType": "01"
    },
    "sign": "6F854E0C94F35F52AD49711E501814A2",
    "timestamp": "20181109172805942",
    "pageParams": null,
    "billContent": null,
    "signType": "MD5",
    "encryptType": "AES",
    "version": null,
    "encryptData": null,
    "transType": null,
    "appId": null,
    "externalParam": null
}
```


2. His 生成腕带码

入出参项：

用途说明	His 生成腕带码				
请求地址	http://127.0.0.1:8080/onepay-web/oneweb/oneapi				
	portal.wb.getWristbandCode				
入参说明	名称	类型长度	可空	说明	备　注
	cardType			卡类型	
	cardNo			卡号	
	idNo			身份证号	
	idType			身份证类型	
	operatorId			操作员 id	
	operatorName			操作员姓名	
	accountType			账号类型	
出参说明	名称	类型长度		说明	备　注

(param)					
	key			腕带 key	
	Url			腕带地址	

简要描述：

- His 生成腕带码

请求 URL：

- http://127.0.0.1:8080/onepay-web/oneweb/oneapi

请求方式：

- POST

交易类型(transType)

- portal.wb.getWristbandCode

请求报文示例：

```
{
    "appId": "1CLT55NO6001790BA8C00000B14B2D07",
    "encryptData":
"{\"accountType\":\"02\",\"appId\":\"1CLT55NO6001790BA8C00000B14B2D07\",\"cardNo\":\"D12345678\",\"cardType\":
\"01\",\"hospName\":\"福清医院吊炸天
\",\"idNo\":\"66\",\"idType\":\"01\",\"operatorId\":\"123\",\"operatorName\":\"腕带付\"}",
    "encryptType": "Plain",
    "param": {
        "accountType": "02",
        "appId": "1CLT55NO6001790BA8C00000B14B2D07",
        "cardNo": "D12345678",
        "cardType": "01",
        "hospName": "福清医院吊炸天",
        "idNo": "66",
        "idType": "01",
        "operatorId": "123",
        "operatorName": "腕带付"
    },
    "sign":
"appId=1CLT55NO6001790BA8C00000B14B2D07&encryptType=Plain&param={\"accountType\":\"02\",\"appId\":\"1CLT55NO60
01790BA8C00000B14B2D07\",\"cardNo\":\"D12345678\",\"cardType\":\"01\",\"hospName\":\"福清医院吊炸天
\",\"idNo\":\"66\",\"idType\":\"01\",\"operatorId\":\"123\",\"operatorName\":\"腕带付
\"}&signType=Plain&timestamp=20181109162604&transType=portal.wb.getWristbandCode&key=1CLT55NN7008790BA8C000009
D9060C4",
    "signType": "Plain",
    "timestamp": "20181109162604",
    "transType": "portal.wb.getWristbandCode"
}
```

响应报文示例

```
{
  "termNo": null,
  "respCode": "000000",
  "respMsg": "处理成功",
  "param": {
    "url": "http://msypay.com:1802/onepay-web/wb/recharge/k4WtYZ",
    "key": "k4WtYZ"
  },
  "sign": "55836DCE01654429709553113DDEB6B8",
  "timestamp": "20181109170047076",
  "pageParams": null,
  "billContent": null,
  "signType": "MD5",
  "encryptType": "Plain",
  "version": null,
  "encryptData": null,
  "transType": null,
  "appId": null,
  "externalParam": null
}
```


3.取消腕带付

入出参项：

用途说明	His 生成腕带码
请求地址	http://127.0.0.1:8080/onepay-web/oneweb/oneapi
	portal.wb.cancelQrCode

入参说明	名称	类型长度	可空	说明	备　注
	key			卡类型	

出参说明(param)	名称	类型长度		说明	备　注

简要描述：

- 取消腕带码

请求 URL：

- http://127.0.0.1:8080/onepay-web/oneweb/oneapi

请求方式：

- POST

交易类型(transType)

- portal.wb.cancelQrCode

请求报文示例

参数名	必选	类型	说明
腕带码编号	是	string	腕带码编号


```
{
    "appId": "1CLT55NO6001790BA8C00000B14B2D07",
    "encryptData": "{\"key\":\"k4WtYn\"}",
    "encryptType": "Plain",
    "param": {
        "key": "k4WtYn"
    },
    "sign": "38480B05E2C196CE36A76886B2619E03",
    "signType": "MD5",
    "timestamp": "20181109170334",
    "transType": "portal.wb.cancelQrCode"
}
```


响应报文示例


```
{
  "termNo": null,
  "respCode": "000000",
  "respMsg": "处理成功",
  "param": {},
  "sign": "D1F97EAB8EF62A1E193185F6B8D7BF4A",
  "timestamp": "20181109170334480",
  "pageParams": null,
  "billContent": null,
  "signType": "MD5",
  "encryptType": "Plain",
  "version": null,
  "encryptData": null,
  "transType": null,
  "appId": null,
  "externalParam": null
}
```