密付支付平台开发者手册

密付支付平台给游戏，直播等互联网应用提供了一个基于区块链加密货币的支付系统，方便最终用户通过加密货币給各种互联网应用支付和兑出。互联网应用采用密付完成支付和兑出，具备高效，安全，跨区域，保护隐私和降低支付成本等一些列的优势。

注册应用

开发者门户网址：https://www.cryptocurpays.com:80/

开发者输入信息

appName：应用名称。

appType：应用类型。

callbackUrl：回调地址。支付网关在支付完成或者兑出完成时，通知应用服务器时候的回调URL地址。

fiatCurrency：法币单位。

appSignAddress: 应用签名地址

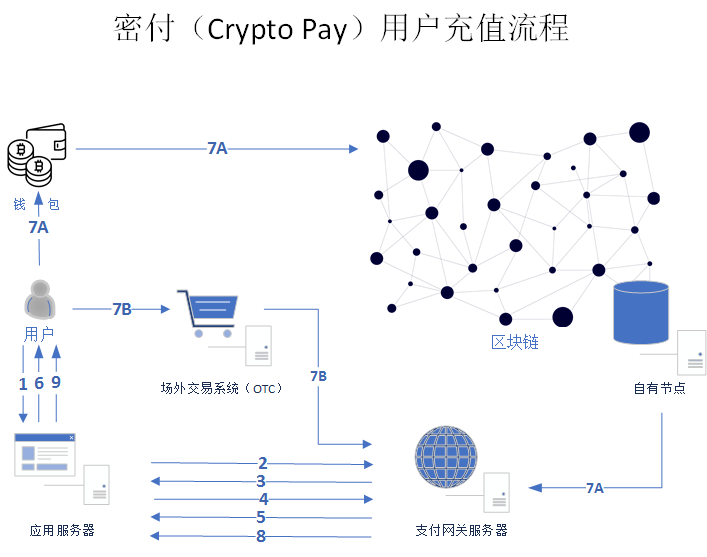
应用服务器向支付网关withdraw请求的时候，需要使用ECDSA（Elliptic Curve Cryptography Algorithm）对于请求信息进行签名。应用服务器自行生成一个私钥（Private Key），然后把私钥所对应的地址（Address），上传到开发者门户。

开发商获取信息

AppSecret：应用密钥。应用服务器发送給支付网关的部分restful请求需要附加appsecret参数来验证请求的合法性。

GatewayAddress：支付网关签名地址。支付网关发送给应用服务器的deposite成功消息，会附件签名验证。使用该地址验证签名的合法性。

支付流程

1. 用户向应用服务器请求加密币充值地址
2. 应用服务器向支付网关请求该玩家的Token
3. 支付网关返回游戏服务器该玩家的Token
4. 应用服务器请求用户的充值地址
5. 支付网关返回该玩家的充值地址
6. 应用服务器把充值地址返回給最终用户
7. 最终用户可以通过两个方式完成充值。无论那种方式，最终都是支付网关服务器得到玩家完成充值的信息：
   1. 通过钱包，向公链发送交易广播
   2. 通过OTC，购买并且充值
8. **支付网关服务器通知应用服务器该用户完成充值
9. 应用服务器在更新该用户相关数值之后，通知该用户完成充值。

REST API介绍

API 入口地址：<https://api.cryptocurpays.com/api/v1>

返回消息是JSON的格式，内容为{“type”:0,”data”:{……..}}。Type=0，代表正常的API Call和回复。否则为异常，请参考附件的异常代码。

Get Rates

因为加密货币和法币的兑换比例会随时浮动，所以最终用户使用加密货币支付可以兑换的虚拟货币也是一个浮动值。应用服务器如果需要展示当前订单需要支付加密货币的数量，可以调用下面的接口来获取最新汇率。

Type：App Server → Payment Server

URL end Point：Get: /rates/:fiatCurName

Parameters：None

Example

Request:

https://api.cryptocurpays.com/api/v1/rates/cny

Response:

{

"type": 0,

"data": {

"eth": 505.2672,

"bch": 893.3159999999999,

"btc": 6227.7222,

"usdt": 1.02

}

}

Get Token

支付网关为每个应用的每个最终用户生成一个访问令牌－Access Token。应用服务器对于最终用户的所有访问需要提供Access Token。在获取和最终用户相关的所有信息之前，应用服务器需要先获取该用户的Token。

游戏服务器向支付服务器请求某个的Access Token。游戏服务器使用本应用的App Secret来请求，支付网关下发Access Token給游戏服务器。如果支付网关发现请求的玩家是一个新玩家，则先在支付网关后台生成一个玩家的Access Token。

游戏服务器可以把这个Token下发給浏览器或者客户端应用程序，客户端或者浏览器可以使用这个Token直接访问支付网关，进行相关的操作

Type

App Server → Payment Gateway

URL End Point

Get: /:appId/token/:userId

Parameters

* App Secret: appsecret

Example

Request:

https://api.cryptocurpays.com/api/v1/122/token/user1?appsecret=GO2wW63AMEglXNmtQQeu9vYy5UwKrxmgc3DgSrpRdrf6fLY7cTqhTccQOlIdQiUvergHixziWbJsFcgaLY0zNB7oPEUXz7VBOGE0ksCPdmPHyuwVZ2yaGamYAKx33Emr46qREf9U0u3m7JFtOszsxlTIaq4ayxdYZD4WclTBMHZNtM3siqJqbbDaV4peFl5iNavxUffe

Response:

{

"type":0,

"data":{

"token":" xgY0esCYVxoY4tDw7XBN6ZOMUP8L10

"

}

}

Get Addresses

应用客户端或者应用服务器向支付网关服务器请求某应用某玩家的收款地址。

Type

App Server（or App Client） to Payment Gateway

URL end Point

Get : /{appid}/users/{userid}/address

Parameters

Token：token

Example

Request：https://api.cryptocurpays.com/api/v1/122/users/user1/address?token=xgY0esCYVxoY4tDw7XBN6ZOMUP8L10

Response:

{

"type": 0,

"data": {

"addresses": {

"eth": {

"address": "0x48091678e2102a312e4e93be98e155bad41bb3c0",

"fiatRate": 3235.2258816000003

},

"btc": {

"address": "mxStsTm3uWrwjvUx6kQQvwzg7sADFAPm9J",

"fiatRate": 39876.1052466

},

"usdt": {

"address": "1HvwaQg56VRgxp1LPBS372nMFsZWMDLDUf",

"fiatRate": 6.53106

}

}

}

}

Get Unconfirmed Deposits

如果最终用户通过自己的加密币钱包支付，加密币在提交到公链之后，有一个挖矿确认的过程，所以可能会有pending的deposits。应用客户端可以使用该API来查询最终用户是否有正在确认当中的支付交易。

URL End Point

Get : /{appid}/users/{userid}/pendingDeposits

Parameters

Token

Example

Request

https://api.cryptocurpays.com/api/v1/122/users/user1/pendingdeposits?token=xgY0esCYVxoY4tDw7XBN6ZOMUP8L10

Response:

{

"type": 0,

"data": {

"eth": {

"0x43644f650f4e500c38f634ebfda3b73f2f2824d3405fcca3f4ddd9a3eaa647f3": {

"from": "0xdeaf4aec5b133b5e827d5dfb9a37d242e5e552a9",

"value": "0.05",

"confirmations": 2,

"need": 8

},

"0x0f7f15fcbca01fa641da7f22737cf076dca9e10d43936dbf77142c45972257cf": {

"from": "0xdeaf4aec5b133b5e827d5dfb9a37d242e5e552a9",

"value": "0.05",

"confirmations": 1,

"need": 8

}

}

}

}

Post A New Deposit

当支付网关在自己管理的区块链节点上面发现可被确认的，给最终用户地址发送加密货币的transaction之后，支付网关通过该最终用户所属应用的Callback URL 通知应用服务器。

URL end Point

Post : https://gameCallbackURL/deposit

Post body Data Example

{

orderid: 'amks9172eiu1jqk2'

appId: 1,

cryptoType: 'eth',

cryptoValue: 0.01,

fee: 0.0001,

fiatCurrency: 'cny',

fiatRate: 3374.3374596,

fromAddress: '0xd54ee29702c3d11cc66ac0f0a35982bcd06b0cb4',

timestamp: 1529359814,

toAddress: '0xc4726dc0f185b621218b709e48590b9701b305fa',

txId: '0x820169b3e003777414e823e0405d9ae74674ab4329b445cae3001c20d9fae1a8',

userId: 9,

signature: 'HzPhJjzBfBPYeDl2KdxuDO9JAEknNbzI/qGgzMnIMui8Jo2QjhuR2ldbLeaccbyTkPj7KiQqwqkVJbReya0fLxU='

}

签名验证

应用服务器收到请求之后，应该使用ECDSA来验证收到消息签名的有效性。详细介绍请参照ECDSA签名和验证章节。

Expect Return

Response Status 200

{"message":"received!"}"

错误重发

如果支付网关服务器不能收到应用服务器的正确反馈，会10S发送一次。尝试10次发送失败之后，转有人工客服处理。

Check Deposit Date

应用服务器需要验证收到的支付信息的有效性的时候，可以调用这个接口

URL End Point

Get : /{appid}/tx/check/deposit

Parameters

App Secret: appsecret

TxId:txid

Example

Request

https://api.cryptocurpays.com/api/v1/122/tx/check/deposit?appsecret=GO2wW63AMEglXNmtQQeu9vYy5UwKrxmgc3DgSrpRdrf6fLY7cTqhTccQOlIdQiUvergHixziWbJsFcgaLY0zNB7oPEUXz7VBOGE0ksCPdmPHyuwVZ2yaGamYAKx33Emr46qREf9U0u3m7JFtOszsxlTIaq4ayxdYZD4WclTBMHZNtM3siqJqbbDaV4peFl5iNavxUffe&txid=xgY0esCYVxoY4tDw7XBN6ZOMUP8L10

Response:

{

"type": 0,

"data": {

"appId": "154",

"cryptoType": "ccn",

"cryptoValue": "100",

"fee": "undefined",

"fiatCurrency": "cny",

"fiatRate": "1",

"fromAddress": "otc",

"timestamp": "1543555253161",

"toAddress": "0xc2ec1a4f6f7f5894d52eadbf04b51ca4dba0de98",

"txId": "0x018faf6da6b29ada6af3b4b6515d2472dd",

"userId": "2",

"signature": "IACpEUiIA/5VY4js1e1dobhn2FWWWzOQHYSBU1SYDtW2KBOEpfOehF5mM2dUhxMpRxn0KAI7Clb06WHNBWdYyPw="

}

}

Response中的data数据和充值成功后的通知消息数据，业务内容数据一致，signature的验证方式相同，请参考加密验证模块部分

Get Balance

应用服务器向支付服务器请求该应用在支付服务器上不同加密货币的balance。

URL end Point

Get: /{appid}/balance

Parameters

* App Secret: appsecret

https://api.cryptocurpays.com/api/v1/122/balance?appsecret=GO2wW63AMEglXNmtQQeu9vYy5UwKrxmgc3DgSrpRdrf6fLY7cTqhTccQOlIdQiUvergHixziWbJsFcgaLY0zNB7oPEUXz7VBOGE0ksCPdmPHyuwVZ2yaGamYAKx33Emr46qREf9U0u3m7JFtOszsxlTIaq4ayxdYZD4WclTBMHZNtM3siqJqbbDaV4peFl5iNavxUffe

Return

{

“type”:0,

“data”:{

"eth": 30.39,

"bch": 39.01

}

}

Post New Withdraw Request

支付网关API不但可以处理最终用户向应用服务器支付的请求，而且可以处理应用服务器提取加密货币的请求。

URL end Point

Post: /{appid}/txs/withdraw/{cyptotype}

{

  cyptoType:"eth",

  toAddress:"0xc4726dc0f185b621218b709e48590b9701b305fa",

  orderId:3445666765,

  cryptoValue:0.05,

signature:"HzPhJjzBfBPYeDl2KdxuDO9JAEknNbzI/qGgzMnIMui8Jo2QjhuR2ldbLeaccbyTkPj7KiQqwqkVJbReya0fLxU="

}

orderId: 代表唯一的withdraw请求。如果应用服务器提交了两次请求，两次请求使用相同的orderId，则支付网关当作相同的请求处理。

签名验证

应用服务器提交支付网关的Post Data需要包含Signature Key。使用对应的Private key使用ECDSA对下面的信息签名。详细介绍请参照ECDSA签名和验证章节。

Return

{

type:0,

data: {

orderId: "10000000000037",

withdrawId: 9

}

}

Withdraw Succeed

支付网关对于已经广播出去的Withdraw的交易Transaction，定期检测该交易被挖矿的状态，如果挖矿的区块数量大于要求的数量，则給开发商的Callback URL发送信息，通知游戏开发商这个提币的请求已经完成。

URL end Point

Post: https://callbackURL/withdraw

{

   "txId":"0xf87b8f9539d66d26fba775e05bb77349bf7f0fe5b48f6f75f51896ed5afcd635",

   "orderId":10000000044,

   "cyptoType":"eth",

   "cyptoValue":0.05,

   "fee":0.0001,

   "withdrawType":0,

   "withdrawId":10000000044,

}

Return

Response Status 200

{"message":"received!"}"

GET withdraw Status

游戏服务器或者游戏客户端向支付网关服务器更新自己请求正在Pending Withdraw的状态。

URL end Point

get: /{appid}/txs/withdraws/{cyptotype}

Parameters

* TxHash

Example

get: https://cyptoExg.net/1234434/withdraws/eth?tx\_hash=’fasdfsadf4rt34rfafsdafsdafd’

return:

   {

         "type":0,

         "data":{

          "timestamp":987653849,

          "txHash":"ujegxe4ce41d88bbb6ce734c1784ea5951f310242751df512bb6048ef2fb4ef49ec7e",

          "blockHeight":345678,

          "latestBlock":45680,

          "confirmationNeed":10,

          "cyptoValue":0.12,／／收到加密货币数量

          "fiatCur":"CNY",／／参考法币类型

          "fiatRate":0.001／／法币虚拟币汇率

}

      },

ECDSA签名和验证

签名和验证流程描述

开发商应用在支付网关注册的过程中，开发商需要生成一个随机的私钥（Private Key From 1 to 2256），该Private Key可以演算出对应的Public Key和Base58Check Address。开发商将Address上传給支付网关，而私钥仅仅个人拥有，对任何第三方保密。如果开发商希望通过Restful API withdraw加密货币，该请求的信息需要使用private key签名。支付网关会通过地址验证签名的有效性。

开发商在支付网关新建应用之后，支付网关后台会生成相应的随机私钥，Private Key。Private Key不会透露給开发商，但是Private Key对应的Base58 Check的地址会提供给开发商。支付网关确认收到一个deposit之后，会通过callback URL通知应用服务器。该信息会使用private key进行签名。应用服务器应该使用address来验证该信息是否有效。

ECDSA签名的作用

信息不能被篡改

修改交易信息的任何部分都会造成签名验证失败。因此可通过验证签名来确认交易请求在网络传输过程种，没有被篡改。

信息不可抵赖

因为Receiver并不知道Sender的Private Key，所以Receiver不能杜撰一个交易请求。Sender发出来的交易请求是不可抵赖的。

示例

Nodejs示例

使用的module

<https://github.com/bitpay/bitcore-lib>

https://github.com/bitpay/bitcore-message

安装bitcore-lib module

npm install bitcore-lib

npm install bitcore-message

在JS里面引用相关Module

var bitcore = require('bitcore-lib');

var Message = require('bitcore-message');

var Hash = require('bitcore-lib').crypto.Hash;

生成随机Private Key

var privateKey = new bitcore.PrivateKey();

计算Private Key对应的Base58check地址

var address = new Address(privateKey.toPublicKey()，bitcore.Networks.mainnet,'pubkeyhash');

生成签名

postObj.signature = Message(' This is a sample message ').sign(privateKey);

验证签名

var verified = Message(' This is a sample message ').verify(address, signature);

签名和验证签名的消息

支付网关会首先对于消息体进行sha256sha256的哈希计算，然后把计算出来的签名增加到消息体。在验证签名的时候，首先会把消息体里面的signature这个key取出来，然后把删除signature key的消息，进行sha256sha256的哈希计算。对于计算结果进行签名验证。

let message = Hash.sha256sha256(Buffer.from(JSON.stringify(postObj))).toString;

如何跳转加密货币购买OTC平台

游戏服务器需要向支付网关请求一个只针对当前游戏玩家的，打开购币OTC的链接地址，请求方式如下

URL end Point

get: /{appId}/otcurl/{userId}

Parameters

Token：token

UserName：userName(玩家在游戏中的昵称)

return:

 {

"type": 0,

"data": "http://www.cryptocurbuy.com:8080/#/buyer?appId=MTAwMDEz&userId=Mjk5OQ%3D%3D&acttime=1536921584948&userNmae=hello&address=%7B%22eth%22%3A%7B%22address%22%3A%220xe1a7a1f90a09eb9ea19790dd5c885722dbcf9038%22%7D%2C%22bch%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%2C%22usdt%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%2C%22btc%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%7D&sign=8af796f0b3adb99c991c9a7ba93c00c9"

}

data中的链接，为登陆链接。

可以支持本用户登录购币充值平台进行购币操作，生成一次url地址，可供用户15分钟以内登录使用，过期需要获取新的登录地址信息。

如果需要引导用户购买对应币种或者币值的加密货币 可以再登陆链接中，添加参数

matchone=192(意向引导的购买金额)

matchtype=eth(意向购买的加密货币种类)

payways=[1,0](意向购买支付渠道 0 银行卡 1 支付宝 2 微信)

backopenurl=Imh0dHBzOi8vd3d3LmJhaWR1LmNvbT9tPTEyMyZuPTk4NzIi(为保证回跳链接附带参数的完整性，url地址请进行base64编码，此地址实际是https://www.baidu.com，用户下单完成后的跳回的商家界面地址，可选，如果不填，otc购币充值平台将跳过回跳逻辑。地址采用标准网站url地址格式，回跳时，除backopenurl完整地址，不附带其他形式的参数)

orderid=‘201810241650’， 开发者平台附带的订单编号，订单交易完成后，会随着deposit通知消息，一同通知到开发者后台。

例如：

http://www.cryptocurbuy.com:8080/#/buyer?appId=MTAwMDEz&userId=Mjk5OQ%3D%3D&acttime=1536921584948&userNmae=hello&address=%7B%22eth%22%3A%7B%22address%22%3A%220xe1a7a1f90a09eb9ea19790dd5c885722dbcf9038%22%7D%2C%22bch%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%2C%22usdt%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%2C%22btc%22%3A%7B%22address%22%3A%221GjiF9KfEUHsjJrqgcr2QaYh1DYo3stuqR%22%7D%7D&sign=8af796f0b3adb99c991c9a7ba93c00c9&matchone=192&matchtype=eth&backopenurl=aHR0cHM6Ly93d3cuYmFpZHUuY29tP209MTIzJm49OTg3Mg&orderid=201810241650

错误代码

SYSTEM\_BUSY:9001,//系统繁忙

ERRORPARAMETER:9007,//错误的参数 api

API\_DATABASES\_ERROR: 10001,//'The database service is unavailable!'

API\_MORE\_APP\_DATA:10002,//More than one record in application table has this appId:

API\_INCORREVT\_AOOID:10003,//appId is incorrect! appId:

API\_NOT\_EXIST\_USER:10004,//

API\_USERTOKEN\_ERROR:10005,//

API\_RATE\_UNAVAILABLE:10006,//The rate service is temporarily unavailable

API\_CRYPTO\_UNAVAILABLE:10007,//The crypto currency you asked to withdraw is not available in CryptoPay!

API\_ERROR\_ADDRESS:10008,//错误的 地址信息

API\_NOT\_EXIST\_APP:10009,//不存在应用信息

API\_WITHDRAW\_ERROR\_VALUE:10010,//错误的withdraw 金额参数

API\_APP\_BALANCE\_ERROR:10011,//账户库存不足

API\_TXID\_ERR:10016,//无效的txid信息