# 火币REST API 说明

目录

[火币REST API 说明 1](#_Toc513225735)

[REST API 简介 2](#_Toc513225736)

[安全认证 2](#_Toc513225737)

[合法请求结构 2](#_Toc513225738)

[签名运算 4](#_Toc513225739)

[请求说明 6](#_Toc513225740)

[API Reference 6](#_Toc513225741)

[行情API 7](#_Toc513225742)

[公共API 19](#_Toc513225743)

[用户资产API 21](#_Toc513225744)

[交易API 24](#_Toc513225745)

[借贷交易API 37](#_Toc513225746)

[虚拟币提现API 44](#_Toc513225747)

[错误码 49](#_Toc513225748)

[行情 API 错误码 49](#_Toc513225749)

[交易 API 错误码 50](#_Toc513225750)

## REST API 简介

火币为用户提供了一套全新的API，可以帮用户快速接入火币PRO站及HADAX站的交易系统，实现程序化交易。

| **访问地址** | **适用站点** | **适用功能** | **适用交易对** |
| --- | --- | --- | --- |
| <https://api.huobipro.com/market> | 火币PRO | 行情 | 所有Pro站交易中的交易对 |
| <https://api.huobipro.com/v1> | 火币PRO | 交易 | 同上 |
| <https://api.hadax.com/market> | HADAX hadax.com | 行情 | 所有HADAX站交易中的交易对 |
| <https://api.hadax.com/v1> | HADAX hadax.com | 交易 | 同上 |

通过API可以实现以下功能：

* 市场行情信息查询（K线、深度、实时成交、24小时行情）
* 账户资产信息查询
* 下单、撤单操作
* 订单信息查询 所有请求基于 HTTPS 协议。在使用中如果遇到问题，请加技术讨论 QQ 群: 火币网API交流群(5) 778160723（加群时请注明uid（账户管理中）和编程语言），我们将尽力帮您答疑解惑。

## 安全认证

目前关于apikey申请和修改，请在“账户 - API管理”页面进行相关操作。其中AccessKey为API 访问密钥，SecretKey为用户对请求进行签名的密钥（仅申请时可见）。Pro站和HADAX站apikey通用。

* **重要提示：这两个密钥与账号安全紧密相关，无论何时都请勿向其它人透露。**

## 合法请求结构

基于安全考虑，除行情API 外的 API 请求都必须进行签名运算。一个合法的请求由以下几部分组成：

* 方法请求地址 即访问服务器地址：api.huobi.pro或者api.hadax.com后面跟上方法名，比如api.huobi.pro/v1/order/orders。
* API 访问密钥（AccessKeyId） 您申请的 APIKEY 中的AccessKey。
* 签名方法（SignatureMethod） 用户计算签名的基于哈希的协议，此处使用 HmacSHA256。
* 签名版本（SignatureVersion） 签名协议的版本，此处使用2。
* 时间戳（Timestamp） 您发出请求的时间 **(UTC 时区)** **(UTC 时区)** **(UTC 时区)** 。在查询请求中包含此值有助于防止第三方截取您的请求。如：2017-05-11T16:22:06。再次强调是 **(UTC 时区)** 。
* 必选和可选参数 每个方法都有一组用于定义 API 调用的必需参数和可选参数。可以在每个方法的说明中查看这些参数及其含义。 请一定注意：对于**GET**请求，每个方法自带的参数都需要进行签名运算； 对于**POST**请求，每个方法自带的参数不进行签名认证，即**POST**请求中需要进行签名运算的只有AccessKeyId、SignatureMethod、SignatureVersion、Timestamp四个参数，其它参数放在body中。
* 签名 签名计算得出的值，用于确保签名有效和未被篡改。

例：

https://api.huobi.pro/v1/order/orders?

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx

&SignatureMethod=HmacSHA256

&SignatureVersion=2

&Timestamp=2017-05-11T15%3A19%3A30

&order-id=1234567890

&Signature=calculated value

## 签名运算

API 请求在通过 Internet 发送的过程中极有可能被篡改。为了确保请求未被更改，我们会要求用户在每个请求中带上签名（行情 API 除外），来校验参数或参数值在传输途中是否发生了更改。

**计算签名所需的步骤**：

1. 规范要计算签名的请求

因为使用 HMAC 进行签名计算时，使用不同内容计算得到的结果会完全不同。所以在进行签名计算前，请先对请求进行规范化处理。下面以查询某订单详情请求为例进行说明

https://api.huobi.pro/v1/order/orders?

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx

&SignatureMethod=HmacSHA256

&SignatureVersion=2

&Timestamp=2017-05-11T15:19:30

&order-id=1234567890

1. 请求方法（GET 或 POST），后面添加换行符\n。

GET\n

1. 添加小写的访问地址，后面添加换行符\n。

api.huobi.pro\n

1. 访问方法的路径，后面添加换行符\n。

/v1/order/orders\n

1. 按照**ASCII码的顺序对参数名进行排序**(**使用 UTF-8 编码**，**且进行了 URI 编码**，**十六进制字符必须大写**，如‘:’会被编码为'%3A'，空格被编码为'%20')。

例如，下面是请求参数的原始顺序，进行过编码后。

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx

order-id=1234567890

SignatureMethod=HmacSHA256

SignatureVersion=2

Timestamp=2017-05-11T15%3A19%3A30

这些参数会被排序为：

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx

SignatureMethod=HmacSHA256

SignatureVersion=2

Timestamp=2017-05-11T15%3A19%3A30

order-id=1234567890

按照以上顺序，将各参数使用字符’&’连接。

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx&SignatureMethod=HmacSHA256&SignatureVersion=2&Timestamp=2017-05-11T15%3A19%3A30&order-id=1234567890

组成最终的要进行签名计算的字符串如下：

GET\n

api.huobi.pro\n

/v1/order/orders\n

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx&SignatureMethod=HmacSHA256&SignatureVersion=2&Timestamp=2017-05-11T15%3A19%3A30&order-id=1234567890

1. 计算签名，将以下两个参数传入加密哈希函数：

* 要进行签名计算的字符串

GET\n

api.huobi.pro\n

/v1/order/orders\n

AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx&SignatureMethod=HmacSHA256&SignatureVersion=2&Timestamp=2017-05-11T15%3A19%3A30&order-id=1234567890

* 进行签名的密钥（SecretKey）

b0xxxxxx-c6xxxxxx-94xxxxxx-dxxxx

得到签名计算结果并进行 Base64编码

4F65x5A2bLyMWVQj3Aqp+B4w+ivaA7n5Oi2SuYtCJ9o=

1. 将上述值作为参数Signature的取值添加到 API 请求中。 将此参数添加到请求时，**必须将该值进行 URI 编码**。
2. 最终，发送到服务器的 API 请求应该为：

https://api.huobi.pro/v1/order/orders?AccessKeyId=e2xxxxxx-99xxxxxx-84xxxxxx-7xxxx&order-id=12345678

## 请求说明

1. 访问地址

Pro 站： 行情： <https://api.huobipro.com/market> 交易： <https://api.huobipro.com/v1>

HADAX 站： 行情： <https://api.hadax.com/market> 交易： <https://api.hadax.com/v1>

1. POST请求头信息中必须声明 Content-Type:application/json;GET请求头信息中必须声明 Content-Type:application/x-www-form-urlencoded。(汉语用户建议设置 Accept-Language:zh-cn)
2. 所有请求参数请按照 API 说明进行参数封装。
3. 将封装好参数的 API 请求通过 POST 或 GET 的方式提交到服务器。
4. 火币网处理请求，并返回相应的 JSON 格式结果。
5. 请使用 https 请求。
6. 限制频率（每个接口，只针对交易api，行情api不限制）为10秒100次。
7. 查询资产详情方法调用顺序：查询当前用户的所有账户->查询指定账户的余额
8. 支持所有Pro站上交易中的交易对，上新币保持与网站同步。

# API Reference

请务必在header中设置user agent为 'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.71 Safari/537.36'

symbol 规则： 基础币种+计价币种。如BTC/USDT，symbol为btcusdt；ETH/BTC， symbol为ethbtc。以此类推

### 行情API

在调用行情接口时，请添加get参数，key为AccessKeyId ，value为网页上申请的apikey的accesskey 。

例：

https://api.huobipro.com/market/history/kline?period=1day&size=200&symbol=btcusdt&AccessKeyId=fff-xxx-ssss-kkk

#### GET /market/history/kline 获取K线数据

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |
| period | true | string | K线类型 |  | 1min, 5min, 15min, 30min, 60min, 1day, 1mon, 1week, 1year |
| size | false | integer | 获取数量 | 150 | [1,1000] |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| status | true | string | 请求处理结果 | "ok" , "error" |
| ts | true | number | 响应生成时间点，单位：毫秒 |  |
| tick | true | object | KLine 数据 |  |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.kline.$period |  |

data 说明:

"data": [

{

"id": K线id,

"amount": 成交量,

"count": 成交笔数,

"open": 开盘价,

"close": 收盘价,当K线为最晚的一根时，是最新成交价

"low": 最低价,

"high": 最高价,

"vol": 成交额, 即 sum(每一笔成交价 \* 该笔的成交量)

}

]

请求响应示例:

/\* GET /market/history/kline?period=1day&size=200&symbol=btcusdt \*/

{

"status": "ok",

"ch": "market.btcusdt.kline.1day",

"ts": 1499223904680,

“data”: [

{

"id": 1499184000,

"amount": 37593.0266,

"count": 0,

"open": 1935.2000,

"close": 1879.0000,

"low": 1856.0000,

"high": 1940.0000,

"vol": 71031537.97866500

},

// more data here

]

}

/\* GET /market/history/kline?period=not-exist&size=200&symbol=ethusdt \*/

{

"ts": 1490758171271,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid period"

}

/\* GET /market/history/kline?period=1day&size=not-exist&symbol=ethusdt \*/

{

"ts": 1490758221221,

"status": "error",

"err-code": "bad-request",

"err-msg": "invalid size, valid range: [1,2000]"

}

/\* GET /market/history/kline?period=1day&size=200&symbol=not-exist \*/

{

"ts": 1490758171271,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid symbol"

}

#### GET /market/detail/merged 获取聚合行情(Ticker)

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| status | true | string | 请求处理结果 | "ok" , "error" |
| ts | true | number | 响应生成时间点，单位：毫秒 |  |
| tick | true | object | K线数据 |  |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.detail.merged |  |

tick 说明:

"tick": {

"id": K线id,

"amount": 成交量,

"count": 成交笔数,

"open": 开盘价,

"close": 收盘价,当K线为最晚的一根时，是最新成交价

"low": 最低价,

"high": 最高价,

"vol": 成交额, 即 sum(每一笔成交价 \* 该笔的成交量)

"bid": [买1价,买1量],

"ask": [卖1价,卖1量]

}

请求响应示例:

/\* GET /market/detail/merged?symbol=ethusdt \*/

{

"status":"ok",

"ch":"market.ethusdt.detail.merged",

"ts":1499225276950,

"tick":{

"id":1499225271,

"ts":1499225271000,

"close":1885.0000,

"open":1960.0000,

"high":1985.0000,

"low":1856.0000,

"amount":81486.2926,

"count":42122,

"vol":157052744.85708200,

"ask":[1885.0000,21.8804],

"bid":[1884.0000,1.6702]

}

}

/\* GET /market/detail/merged?symbol=not-exist \*/

{

"ts": 1490758171271,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid symbol”

}

#### GET /market/depth 获取 Market Depth 数据

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |
| type | true | string | Depth 类型 |  | step0, step1, step2, step3, step4, step5（合并深度0-5）；step0时，不合并深度 |

* 用户选择“合并深度”时，一定报价精度内的市场挂单将予以合并显示。合并深度仅改变显示方式，不改变实际成交价格。

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| status | true | string |  | "ok" 或者 "error" |
| ts | true | number | 响应生成时间点，单位：毫秒 |  |
| tick | true | object | Depth 数据 |  |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.depth.$type |  |

tick 说明:

"tick": {

"id": 消息id,

"ts": 消息生成时间，单位：毫秒,

"bids": 买盘,[price(成交价), amount(成交量)], 按price降序,

"asks": 卖盘,[price(成交价), amount(成交量)], 按price升序

}

请求响应示例:

/\* GET /market/depth?symbol=ethusdt&type=step1 \*/

{

"status": "ok",

"ch": "market.btcusdt.depth.step1",

"ts": 1489472598812,

"tick": {

"id": 1489464585407,

"ts": 1489464585407,

"bids": [

[7964, 0.0678], // [price, amount]

[7963, 0.9162],

[7961, 0.1],

[7960, 12.8898],

[7958, 1.2],

[7955, 2.1009],

[7954, 0.4708],

[7953, 0.0564],

[7951, 2.8031],

[7950, 13.7785],

[7949, 0.125],

[7948, 4],

[7942, 0.4337],

[7940, 6.1612],

[7936, 0.02],

[7935, 1.3575],

[7933, 2.002],

[7932, 1.3449],

[7930, 10.2974],

[7929, 3.2226]

],

"asks": [

[7979, 0.0736],

[7980, 1.0292],

[7981, 5.5652],

[7986, 0.2416],

[7990, 1.9970],

[7995, 0.88],

[7996, 0.0212],

[8000, 9.2609],

[8002, 0.02],

[8008, 1],

[8010, 0.8735],

[8011, 2.36],

[8012, 0.02],

[8014, 0.1067],

[8015, 12.9118],

[8016, 2.5206],

[8017, 0.0166],

[8018, 1.3218],

[8019, 0.01],

[8020, 13.6584]

]

}

}

/\* GET /market/depth?symbol=ethusdt&type=not-exist \*/

{

"ts": 1490759358099,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid type"

}

#### GET /market/trade 获取 Trade Detail 数据

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| status | true | string |  | "ok" 或者 "error" |
| ts | true | number | 响应生成时间点，单位：毫秒 |  |
| tick | true | object | Trade 数据 |  |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.trade.detail |  |

tick 说明：

"tick": {

"id": 消息id,

"ts": 最新成交时间,

"data": [

{

"id": 成交id,

"price": 成交价钱,

"amount": 成交量,

"direction": 主动成交方向,

"ts": 成交时间

}

]

}

请求响应例子:

/\* GET /market/trade?symbol=ethusdt \*/

{

"status": "ok",

"ch": "market.btcusdt.trade.detail",

"ts": 1489473346905,

"tick": {

"id": 600848670,

"ts": 1489464451000,

"data": [

{

"id": 600848670,

"price": 7962.62,

"amount": 0.0122,

"direction": "buy",

"ts": 1489464451000

}

]

}

}

/\* GET /market/trade?symbol=not-exist \*/

{

"ts": 1490759506429,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid symbol"

}

#### GET /market/history/trade 批量获取最近的交易记录

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |
| size | false | integer | 获取交易记录的数量 | 1 | [1, 2000] |

响应数据:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| status | true | string |  |  | ok, error |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.trade.detail |  |  |
| ts | true | integer | 发送时间 |  |  |
| data | true | object | 成交记录 |  |  |

data 说明：

"data": {

"id": 消息id,

"ts": 最新成交时间,

"data": [

{

"id": 成交id,

"price": 成交价,

"amount": 成交量,

"direction": 主动成交方向,

"ts": 成交时间

}

]

}

请求响应例子:

/\* GET /market/history/trade?symbol=ethusdt \*/

{

"status": "ok",

"ch": "market.ethusdt.trade.detail",

"ts": 1502448925216,

"data": [

{

"id": 31459998,

"ts": 1502448920106,

"data": [

{

"id": 17592256642623,

"amount": 0.04,

"price": 1997,

"direction": "buy",

"ts": 1502448920106

}

]

}

]

}

#### GET /market/detail 获取 Market Detail 24小时成交量数据

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| status | true | string |  | "ok" 或者 "error" |
| ts | true | number | 响应生成时间点，单位：毫秒 |  |
| tick | true | object | Detail 数据 |  |
| ch | true | string | 数据所属的 channel，格式： market.$symbol.depth.$type |  |

tick 说明:

"tick": {

"id": 消息id,

"ts": 24小时统计时间,

"amount": 24小时成交量,

"open": 前推24小时成交价,

"close": 当前成交价,

"high": 近24小时最高价,

"low": 近24小时最低价,

"count": 近24小时累积成交数,

"vol": 近24小时累积成交额, 即 sum(每一笔成交价 \* 该笔的成交量)

}

请求响应例子

/\* GET /market/detail?symbol=ethusdt \*/

{

"status": "ok",

"ch": "market.btcusdt.detail",

"ts": 1489473538996,

"tick": {

"amount": 4316.4346,

"open": 8090.54,

"close": 7962.62,

"high": 8119.00,

"ts": 1489464451000,

"id": 1489464451,

"count": 9595,

"low": 7875.00,

"vol": 34497276.905760

}

}

/\* GET /market/detail?symbol=not-exists \*/

{

"ts": 1490759594752,

"status": "error",

"err-code": "invalid-parameter",

"err-msg": "invalid symbol"

}

### 公共API

#### GET /v1/common/symbols 查询Pro站支持的所有交易对及精度

#### GET /v1/hadax/common/symbols 查询HADAX站支持的所有交易对及精度

请求参数: (无)

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| base-currency | true | string | 基础币种 |  |
| quote-currency | true | string | 计价币种 |  |
| price-precision | true | string | 价格精度位数（0为个位） |  |
| amount-precision | true | string | 数量精度位数（0为个位） |  |
| symbol-partition | true | string | 交易区 | main主区，innovation创新区，bifurcation分叉区 |

请求响应例子:

/\* GET /v1/common/symbols \*/

{

"status": "ok",

"data": [

{

"base-currency": "eth",

"quote-currency": "usdt",

"symbol": "ethusdt"

}

{

"base-currency": "etc",

"quote-currency": "usdt",

"symbol": "etcusdt"

}

]

}

#### GET /v1/common/currencys 查询Pro站支持的所有币种

#### GET /v1/hadax/common/currencys 查询HADAX站支持的所有币种

请求参数:

(无)

响应数据:

currency list

请求响应例子:

/\* GET /v1/common/currencys \*/

{

"status": "ok",

"data": [

"usdt",

"eth",

"etc"

]

}

#### GET /v1/common/timestamp 查询系统当前时间

请求参数:

(无)

响应数据:

系统时间戳

请求响应例子

/\* GET /v1/common/timestamp \*/

{

"status": "ok",

"data": 1494900087029

}

### 用户资产API

#### GET /v1/account/accounts 查询当前用户的所有账户(即account-id)，Pro站和HADAX account-id通用

请求参数:

无

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| id | true | long | account-id |  |
| state | true | string | 账户状态 | working：正常, lock：账户被锁定 |
| type | true | string | 账户类型 | spot：现货账户 |

请求响应例子:

/\* GET /v1/account/accounts \*/

{

"status": "ok",

"data": [

{

"id": 100009,

"type": "spot",

"state": "working",

"user-id": 1000

}

]

}

#### GET /v1/account/accounts/{account-id}/balance 查询Pro站指定账户的余额

#### GET /v1/hadax/account/accounts/{account-id}/balance 查询HADAX站指定账户的余额

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| account-id | true | string | account-id，填在 path 中，可用 GET /v1/account/accounts 获取 |  |  |

* 如果不知道自己的账户ID，请使用 GET /v1/account/accounts 查询

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| id | true | long | 账户 ID |  |
| state | true | string | 账户状态 | working：正常 lock：账户被锁定 |
| type | true | string | 账户类型 | spot：现货账户 |
| list | false | Array | 子账户数组 |  |

list字段说明

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| balance | true | string | 余额 |  |
| currency | true | string | 币种 |  |
| type | true | string | 类型 | trade: 交易余额，frozen: 冻结余额 |

请求响应例子:

/\* GET /v1/account/accounts/{account-id}/balance \*/

{

"status": "ok",

"data": {

"id": 100009,

"type": "spot",

"state": "working",

"list": [

{

"currency": "usdt",

"type": "trade",

"balance": "500009195917.4362872650"

},

{

"currency": "usdt",

"type": "frozen",

"balance": "328048.1199920000"

},

{

"currency": "etc",

"type": "trade",

"balance": "499999894616.1302471000"

},

{

"currency": "etc",

"type": "frozen",

"balance": "9786.6783000000"

}

{

"currency": "eth",

"type": "trade",

"balance": "499999894616.1302471000"

},

{

"currency": "eth”,

"type": "frozen",

"balance": "9786.6783000000"

}

],

"user-id": 1000

}

}

## 交易API

#### POST /v1/order/orders/place Pro站下单

#### POST /v1/hadax/order/orders/place HADAX站下单

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| account-id | true | string | 账户 ID，使用accounts方法获得。币币交易使用‘spot’账户的accountid；借贷资产交易，请使用‘margin’账户的accountid |  |  |
| amount | true | string | 限价单表示下单数量，市价买单时表示买多少钱，市价卖单时表示卖多少币 |  |  |
| price | false | string | 下单价格，市价单不传该参数 |  |  |
| source | false | string | 订单来源 | api，如果使用借贷资产交易，请填写‘margin-api’ |  |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |
| type | true | string | 订单类型 |  | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | false | string | 订单ID |  |

请求响应例子:

/\* POST /v1/order/orders/place {

"account-id": "100009",

"amount": "10.1",

"price": "100.1",

"source": "api",

"symbol": "ethusdt",

"type": "buy-limit"

} \*/

{

"status": "ok",

"data": "59378"

}

#### POST /v1/order/orders/{order-id}/submitcancel 申请撤销一个订单请求

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| order-id | true | string | 订单ID，填在path中 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | true | string | 订单 ID |  |

请求响应例子:

/\* POST /v1/order/orders/{order-id}/submitcancel \*/

{

"status": "ok",//注意，返回OK表示撤单请求成功。订单是否撤销成功请调用订单查询接口查询该订单状态

"data": "59378"

}

#### POST /v1/order/orders/batchcancel 批量撤销订单

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| order-ids | true | list | 撤销订单ID列表 |  | 单次不超过50个订单id |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | false | map | 撤单结果 |  |

请求响应例子:

/\* POST /v1/order/orders/batchcancel \*/

{

"order-ids": [

"1", "2", "3"

]

}

-----

{

"status": "ok",

"data": {

"success": [

"1",

"3"

],

"failed": [

{

"err-msg": "记录无效",

"order-id": "2",

"err-code": "base-record-invalid"

}

]

}

}

#### GET /v1/order/orders/{order-id} 查询某个订单详情

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| order-id | true | string | 订单ID，填在path中 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| account-id | true | long | 账户 ID |  |
| amount | true | string | 订单数量 |  |
| canceled-at | false | long | 订单撤销时间 |  |
| created-at | true | long | 订单创建时间 |  |
| field-amount | true | string | 已成交数量 |  |
| field-cash-amount | true | string | 已成交总金额 |  |
| field-fees | true | string | 已成交手续费（买入为币，卖出为钱） |  |
| finished-at | false | long | 最后成交时间 |  |
| id | true | long | 订单ID |  |
| price | true | string | 订单价格 |  |
| source | true | string | 订单来源 | api |
| state | true | string | 订单状态 | pre-submitted 准备提交, submitting , submitted 已提交, partial-filled 部分成交, partial-canceled 部分成交撤销, filled 完全成交, canceled 已撤销 |
| symbol | true | string | 交易对 | btcusdt, bchbtc, rcneth ... |
| type | true | string | 订单类型 | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |

请求响应例子:

/\* GET /v1/order/orders/{order-id} \*/

{

"status": "ok",

"data": {

"id": 59378,

"symbol": "ethusdt",

"account-id": 100009,

"amount": "10.1000000000",

"price": "100.1000000000",

"created-at": 1494901162595,

"type": "buy-limit",

"field-amount": "10.1000000000",

"field-cash-amount": "1011.0100000000",

"field-fees": "0.0202000000",

"finished-at": 1494901400468,

"user-id": 1000,

"source": "api",

"state": "filled",

"canceled-at": 0,

"exchange": "huobi",

"batch": ""

}

}

#### GET /v1/order/orders/{order-id}/matchresults 查询某个订单的成交明细

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| order-id | true | string | 订单ID，填在path中 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| created-at | true | long | 成交时间 |  |
| filled-amount | true | string | 成交数量 |  |
| filled-fees | true | string | 成交手续费 |  |
| id | true | long | 订单成交记录ID |  |
| match-id | true | long | 撮合ID |  |
| order-id | true | long | 订单 ID |  |
| price | true | string | 成交价格 |  |
| source | true | string | 订单来源 | api |
| symbol | true | string | 交易对 | btcusdt, bchbtc, rcneth ... |
| type | true | string | 订单类型 | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |

请求响应例子:

/\* GET /v1/order/orders/{order-id}/matchresults \*/

{

"status": "ok",

"data": [

{

"id": 29553,

"order-id": 59378,

"match-id": 59335,

"symbol": "ethusdt",

"type": "buy-limit",

"source": "api",

"price": "100.1000000000",

"filled-amount": "9.1155000000",

"filled-fees": "0.0182310000",

"created-at": 1494901400435

}

]

}

#### GET /v1/order/orders 查询当前委托、历史委托

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  | btcusdt, bchbtc, rcneth ... |
| types | false | string | 查询的订单类型组合，使用','分割 |  | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |
| start-date | false | string | 查询开始日期, 日期格式yyyy-mm-dd |  |  |
| end-date | false | string | 查询结束日期, 日期格式yyyy-mm-dd |  |  |
| states | true | string | 查询的订单状态组合，使用','分割 |  | pre-submitted 准备提交, submitted 已提交, partial-filled 部分成交, partial-canceled 部分成交撤销, filled 完全成交, canceled 已撤销 |
| from | false | string | 查询起始 ID |  |  |
| direct | false | string | 查询方向 |  | prev 向前，next 向后 |
| size | false | string | 查询记录大小 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| account-id | true | long | 账户 ID |  |
| amount | true | string | 订单数量 |  |
| canceled-at | false | long | 订单撤销时间 |  |
| created-at | true | long | 订单创建时间 |  |
| field-amount | true | string | 已成交数量 |  |
| field-cash-amount | true | string | 已成交总金额 |  |
| field-fees | true | string | 已成交手续费（买入为币，卖出为钱） |  |
| finished-at | false | long | 最后成交时间 |  |
| id | true | long | 订单ID |  |
| price | true | string | 订单价格 |  |
| source | true | string | 订单来源 | api |
| state | true | string | 订单状态 | pre-submitted 准备提交, submitting , submitted 已提交, partial-filled 部分成交, partial-canceled 部分成交撤销, filled 完全成交, canceled 已撤销 |
| symbol | true | string | 交易对 | btcusdt, bchbtc, rcneth ... |
| type | true | string | 订单类型 | submit-cancel：已提交撤单申请 ,buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |

请求响应例子:

/\* GET /v1/order/orders \*/

{

"status": "ok",

"data": [

{

"id": 59378,

"symbol": "ethusdt",

"account-id": 100009,

"amount": "10.1000000000",

"price": "100.1000000000",

"created-at": 1494901162595,

"type": "buy-limit",

"field-amount": "10.1000000000",

"field-cash-amount": "1011.0100000000",

"field-fees": "0.0202000000",

"finished-at": 1494901400468,

"user-id": 1000,

"source": "api",

"state": "filled",

"canceled-at": 0,

"exchange": "huobi",

"batch": ""

}

]

}

#### GET /v1/order/matchresults 查询当前成交、历史成交

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 | btcusdt, bchbtc, rcneth ... |  |
| types | false | string | 查询的订单类型组合，使用','分割 |  | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |
| start-date | false | string | 查询开始日期, 日期格式yyyy-mm-dd |  |  |
| end-date | false | string | 查询结束日期, 日期格式yyyy-mm-dd |  |  |
| from | false | string | 查询起始 ID |  |  |
| direct | false | string | 查询方向 |  | prev 向前，next 向后 |
| size | false | string | 查询记录大小 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| created-at | true | long | 成交时间 |  |
| filled-amount | true | string | 成交数量 |  |
| filled-fees | true | string | 成交手续费 |  |
| id | true | long | 订单成交记录ID |  |
| match-id | true | long | 撮合ID |  |
| order-id | true | long | 订单 ID |  |
| price | true | string | 成交价格 |  |
| source | true | string | 订单来源 | api |
| symbol | true | string | 交易对 | btcusdt, bchbtc, rcneth ... |
| type | true | string | 订单类型 | buy-market：市价买, sell-market：市价卖, buy-limit：限价买, sell-limit：限价卖, buy-ioc：IOC买单, sell-ioc：IOC卖单 |

请求响应例子:

/\* GET /v1/orders/matchresults \*/

{

"status": "ok",

"data": [

{

"id": 29555,

"order-id": 59378,

"match-id": 59335,

"symbol": "ethusdt",

"type": "buy-limit",

"source": "api",

"price": "100.1000000000",

"filled-amount": "0.9845000000",

"filled-fees": "0.0019690000",

"created-at": 1494901400487

}

]

}

## 借贷交易API

**（重要：如果使用借贷资产交易，请在下单接口/v1/order/orders/place请求参数source中填写‘margin-api’）**

目前仅支持 USDT 交易区和 BTC 交易区部分交易对

#### POST /v1/dw/transfer-in/margin 现货账户划入至借贷账户

#### POST /v1/dw/transfer-out/margin 借贷账户划出至现货账户

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  |  |
| currency | true | string | 币种 |  |  |
| amount | true | string | 金额 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | true | long | 划转ID |  |

请求响应例子:

/\* POST /v1/dw/transfer-in/margin

{

"symbol": "ethusdt",

"currency": "eth",

"amount": "1.0"

} \*/

{

"status": "ok",

"data": 1000

}

#### POST /v1/margin/orders 申请借贷

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  |  |
| currency | true | string | 币种 |  |  |
| amount | true | string | 金额 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | true | long | 订单号 |  |

请求响应例子:

/\* POST /v1/margin/orders {

"amount": "10.1",

"symbol": "ethusdt",

"currency": "eth"

} \*/

{

"status": "ok",

"data": 59378

}

#### POST /v1/margin/orders/{order-id}/repay 归还借贷

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| order-id | true | long | 借贷订单 ID，写在path中 |  |  |
| amount | true | string | 还款量 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | true | long | 订单号 |  |

请求响应例子:

/\* POST /v1/margin/orders/59378/repay {

"amount": "10.1"

}\*/

{

"status": "ok",

"data": 59378

}

#### GET /v1/margin/loan-orders 借贷订单

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  |  |
| start-date | false | string | 查询开始日期, 日期格式yyyy-mm-dd |  |  |
| end-date | false | string | 查询结束日期, 日期格式yyyy-mm-dd |  |  |
| states | false | string | 状态 |  |  |
| from | false | string | 查询起始 ID |  |  |
| direct | false | string | 查询方向 |  | prev 向前，next 向后 |
| size | false | string | 查询记录大小 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| id | true | long | 订单号 |  |
| user-id | true | long | 用户ID |  |
| account-id | true | long | 账户ID |  |
| symbol | true | string | 交易对 |  |
| currency | true | string | 币种 |  |
| loan-amount | true | string | 借贷本金总额 |  |
| loan-balance | true | string | 未还本金 |  |
| interest-rate | true | string | 利率 |  |
| interest-amount | true | string | 利息总额 |  |
| interest-balance | true | string | 未还利息 |  |
| created-at | true | long | 借贷发起时间 |  |
| accrued-at | true | long | 最近一次计息时间 |  |
| state | true | string | 订单状态 | created 未放款，accrual 已放款，cleared 已还清，invalid 异常 |

请求响应例子:

/\* GET /v1/margin/loan-orders?symbol=btcusdt

\*/

{

"status": "ok",

"data": [

{

"loan-balance": "0.100000000000000000",

"interest-balance": "0.000200000000000000",

"interest-rate": "0.002000000000000000",

"loan-amount": "0.100000000000000000",

"accrued-at": 1511169724531,

"interest-amount": "0.000200000000000000",

"symbol": "ethbtc",

"currency": "btc",

"id": 394,

"state": "accrual",

"account-id": 17747,

"user-id": 119913,

"created-at": 1511169724531

}

]

}

#### GET /v1/margin/accounts/balance 借贷账户详情

请求参数

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| symbol | false | string | 交易对，作为get参数 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| symbol | true | string | 交易对 |  |
| state | true | string | 账户状态 | working,fl-sys,fl-mgt,fl-end |
| risk-rate | true | object | 风险率 |  |
| fl-price | true | string | 爆仓价 |  |
| list | true | array | 子账户列表 |  |

请求响应例子:

/\* GET /v1/margin/accounts/balance?symbol=btcusdt

\*/

{

"status": "ok",

"data": [

{

"id": 18264,

"type": "margin",

"state": "working",

"symbol": "btcusdt",

"fl-price": "0",

"fl-type": "safe",

"risk-rate": "475.952571086994250554",

"list": [

{

"currency": "btc",

"type": "trade",

"balance": "1168.533000000000000000"

},

{

"currency": "btc",

"type": "frozen",

"balance": "0.000000000000000000"

},

{

"currency": "btc",

"type": "loan",

"balance": "-2.433000000000000000"

},

{

"currency": "btc",

"type": "interest",

"balance": "-0.000533000000000000"

},

{

"currency": "usdt",

"type": "trade",//借贷账户可用

"balance": "1313.534000000000000000"

},

{

"currency": "usdt",

"type": "frozen",//借贷账户冻结

"balance": "0.000000000000000000"

},

{

"currency": "usdt",

"type": "loan",//已借贷

"balance": "-140.234099999999999920"

},

{

"currency": "usdt",

"type": "interest",//usdt待还利息

"balance": "-0.931206660000000000"

},

{

"currency": "btc",

"type": "transfer-out-available",//可转btc

"balance": "1163.872174670000000000"

},

{ "currency": "usdt",

"type": "transfer-out-available",//可转usdt

"balance": "1313.534000000000000000"

},

{

"currency": "btc",

"type": "loan-available",//可借btc

"balance": "8161.876538350676000000"

},

{

"currency": "usdt",

"type": "loan-available",//可借usdt

"balance": "49859.765900000000000080"

}

]

}

]

}

## 虚拟币提现API

**仅支持提现到【Pro站提币地址列表中的提币地址】**

#### POST /v1/dw/withdraw/api/create 申请提现虚拟币

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| address | true | string | 提现地址 |  |  |
| amount | true | string | 提币数量 |  |  |
| currency | true | string | 资产类型 |  | btc, ltc, bch, eth, etc ...(火币Pro支持的币种) |
| fee | false | string | 转账手续费 |  |  |
| addr-tag | false | string | 虚拟币共享地址tag，XRP特有 |  | 格式, "123"类的整数字符串 |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | false | long | 提现ID |  |

请求响应例子:

/\* POST /v1/dw/withdraw/api/create\*/

{

"address": "0xde709f2102306220921060314715629080e2fb77",

"amount": "0.05",

"currency": "eth",

"fee": "0.01"

}

{

"status": "ok",

"data": 700

}

#### POST /v1/dw/withdraw-virtual/{withdraw-id}/cancel 申请取消提现虚拟币

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| withdraw-id | true | long | 提现ID，填在path中 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| data | false | long | 提现 ID |  |

请求响应例子:

/\* POST /v1/dw/withdraw-virtual/{withdraw-id}/cancel \*/

{

"status": "ok",

"data": 700

}

#### GET /v1/query/deposit-withdraw 查询虚拟币充提记录

请求参数:

| **参数名称** | **是否必须** | **类型** | **描述** | **默认值** | **取值范围** |
| --- | --- | --- | --- | --- | --- |
| currency | true | string | 币种 |  |  |
| type | true | string | 'deposit' or 'withdraw' |  |  |
| from | false | string | 查询起始 ID |  |  |
| size | false | string | 查询记录大小 |  |  |

响应数据:

| **参数名称** | **是否必须** | **数据类型** | **描述** | **取值范围** |
| --- | --- | --- | --- | --- |
| id | true | long |  |  |
| type | true | long | 类型 | 'deposit' 'withdraw' |
| currency | true | string | 币种 |  |
| tx-hash | true | string | 交易哈希 |  |
| amount | true | long | 个数 |  |
| address | true | string | 地址 |  |
| address-tag | true | string | 地址标签 |  |
| fee | true | long | 手续费 |  |
| state | true | string | 状态 | 状态参见下表 |
| created-at | true | long | 发起时间 |  |
| updated-at | true | long | 最后更新时间 |  |

###### 虚拟币提现状态定义：

| **状态** | **描述** |
| --- | --- |
| submitted | 已提交 |
| reexamine | 审核中 |
| canceled | 已撤销 |
| pass | 审批通过 |
| reject | 审批拒绝 |
| pre-transfer | 处理中 |
| wallet-transfer | 已汇出 |
| wallet-reject | 钱包拒绝 |
| confirmed | 区块已确认 |
| confirm-error | 区块确认错误 |
| repealed | 已撤销 |

###### 虚拟币充值状态定义：

| **状态** | **描述** |
| --- | --- |
| unknown | 状态未知 |
| confirming | 确认中 |
| confirmed | 确认中 |
| safe | 已完成 |
| orphan | 待确认 |

请求响应例子:

/\* GET /v1/query/deposit-withdraw?currency=xrp&type=deposit&from=5&size=12 \*/

{

"status": "ok",

"data": [

{

"id": 1171,

"type": "deposit",

"currency": "xrp",

"tx-hash": "ed03094b84eafbe4bc16e7ef766ee959885ee5bcb265872baaa9c64e1cf86c2b",

"amount": 7.457467,

"address": "rae93V8d2mdoUQHwBDBdM4NHCMehRJAsbm",

"address-tag": "100040",

"fee": 0,

"state": "safe",

"created-at": 1510912472199,

"updated-at": 1511145876575

},

...

]

}

## 错误码

### 行情 API 错误码

| **错误码** | **描述** |
| --- | --- |
| bad-request | 错误请求 |
| invalid-parameter | 参数错 |
| invalid-command | 指令错 |
| code 的具体解释, 参考对应的 err-msg. |  |

### 交易 API 错误码

| **错误码** | **描述** |
| --- | --- |
| base-symbol-error | 交易对不存在 |
| base-currency-error | 币种不存在 |
| base-date-error | 错误的日期格式 |
| account-transfer-balance-insufficient-error | 余额不足无法冻结 |
| bad-argument | 无效参数 |
| api-signature-not-valid | API签名错误 |
| gateway-internal-error | 系统繁忙，请稍后再试 |
| security-require-assets-password | 需要输入资金密码 |
| audit-failed | 下单失败 |
| ad-ethereum-addresss | 请输入有效的以太坊地址 |
| order-accountbalance-error | 账户余额不足 |
| order-limitorder-price-error | 限价单下单价格超出限制 |
| order-limitorder-amount-error | 限价单下单数量超出限制 |
| order-orderprice-precision-error | 下单价格超出精度限制 |
| order-orderamount-precision-error | 下单数量超过精度限制 |
| order-marketorder-amount-error | 下单数量超出限制 |
| order-queryorder-invalid | 查询不到此条订单 |
| order-orderstate-error | 订单状态错误 |
| order-datelimit-error | 查询超出时间限制 |
| order-update-error | 订单更新出错 |