

开启 API

开启 API 权限

用户的 API 权限在网站的安全中心->API 处获取。点击设置，邮箱验证后即可获得，其中 API Key 是 tokok 提供给 API 用户的访问密钥，secretKey 用于对请求参数签名的密钥。

注意： 请勿向任何人泄露这两个密钥，它们关乎您账号的安全。

签名认证

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。

所有需要签名的接口，其请求头都必须包含以下内容：

ACCESS-KEY 字符串类型的 API Key。

ACCESS-SIGN 使用 base64 编码签名(请参阅签名)。

ACCESS-TIMESTAMP 发起请求的时间戳，Unix 时间戳的十进制秒数格式。

如何对请求参数进行签名

用户提交的参数、请求头中的 ACCESS-KEY 及 ACCESS-TIMESTAMP，都参与签名。

首先，将待签名字符串要求按照参数名的字典顺序进行排序。

例如：

ACCESS-KEY=09c2c831-6737-49db-879e-0b21416a54f6

ACCESS-TIMESTAMP=1543057116

参数如下：

string[]

```
parameters={"symbol=tok_eth","type=1","entrustPrice=680","entrustCount=100"};
```

生成待签名的字符串

```
ACCESS-KEY=09c2c831-6737-49db-879e-0b21416a54f6&ACCESS-TIMESTAMP=1543057116&entrustCount=100&entrustPrice=1.03&symbol=tok_eth&type=1
```

最后，根据 secretKey 使用 HMAC SHA256 算法，对最终待签名字符串进行签名，计算结果采用 base64 编码(该字符串赋值于请求头 ACCESS-SIGN)。

REST API 参考

币市行情 API

行情接口

1. Get /api/v1/tickers 获取全部交易对行情数据

URL <https://www.tokok.com/api/v1/tickers>

示例

```
# Request
GET https://www.tokok.com/api/v1/tickers
# Response
{
  "timestamp":1534755107946,
  "ticker":[{"symbol":"NXT_ETH",
    "buy":"33.15",
    "high":"34.15",
    "last":"33.15",
```

```
        "low": "32.05",
        "sell": "33.16",
        "vol": "10532696.39199642"
    }, {
        "symbol": "TOK_ETH",
        "buy": "33.15",
        "high": "34.15",
        "last": "33.15",
        "low": "32.05",
        "sell": "33.16",
        "vol": "10532696.39199642"
    }
}
```

返回值说明

timestamp: 返回数据时服务器时间

symbol: 交易对

buy: 买一价

high: 最近 24 小时最高价

last: 最新成交价

low: 最近 24 小时最低价

sell: 卖一价

vol: 最近 24 小时成交量

2. Get /api/v1/ticker 获取行情数据

URL https://www.tokok.com/api/v1/ticker?symbol=tok_eth

示例

Request

GET https://www.tokok.com/api/v1/ticker?symbol=tok_eth

Response

```
{
    "timestamp": 1534755107946,
    "ticker": {
```

```
        "buy": "33.15",
        "high": "34.15",
        "last": "33.15",
        "low": "32.05",
        "sell": "33.16",
        "vol": "10532696.39199642"
    }
}
```

返回值说明

timestamp: 返回数据时服务器时间
buy: 买一价
high: 最近 24 小时最高价
last: 最新成交价
low: 最近 24 小时最低价
sell: 卖一价
vol: 最近 24 小时成交量

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth

深度接口

3. Get /api/v1/depth 获取市场深度

URL https://www.tokok.com/api/v1/depth?symbol=tok_eth

示例

```
# Request
GET https://www.tokok.com/api/v1/depth?symbol=tok_eth
# Response
```

```
{
  "asks": [
    ["0.159","111"],
    ["0.16","111"],
    ["0.17","111"],
    ["0.18","111"]
  ],
  "bids": [
    ["0.1582","111"],
    ["0.1581","111"],
    ["0.158","100"]
  ]
}
```

返回值说明

asks :卖方深度[[价格, 数量],...]

bids :买方深度[[价格, 数量],...]

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
size	Integer	否(默认 50)	value: 1-50

最新成交记录接口

4. Get /api/v1/trades 获取最新成交记录

URL https://www.tokok.com/api/v1/trades?symbol=tok_eth&size=50

示例

Request

GET https://www.tokok.com/api/v1/trades?symbol=tok_eth&size=50

Response

```
[
  {
    "timestamp": 1367130137000,
    "price": "787.71",
    "amount": "0.003",
    "side": "sell"
  },
  {
    "timestamp": 1367130137000,
    "price": "787.71",
    "amount": "0.003",
    "side": "sell"
  }
]
```

返回值说明

timestamp: 交易时间

price: 交易价格

amount: 交易数量

side: buy/sell (成交方向)

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
size	Integer	否	如: 50

K 线接口

5. Get /api/v1/kline 获取 K 线数据

URL https://www.tokok.com/api/v1/kline?symbol=tok_eth&size=20&type=1min

示例

```
# Request
GET https://www.tokok.com/api/v1/kline?symbol=tok_eth&size=20&type=1min
# Response
[
  [
    "1417536000000",
    "2339.11",
    "2383.15",
    "2322",
    "2369.85",
    "83850.06"
  ],
  [
    "1417536000000",
    "2339.11",
    "2383.15",
    "2322",
    "2369.85",
    "83850.06"
  ],
]
```

返回值说明

```
[
  "1417536000000", 时间戳
  "2370.16",        开
  "2380",           高
  "2352",           低
```

```
"2367.37",      收  
"17259.83"     交易量  
]
```

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
type	String	是	
size	Integer	否	指定获取数据的条数

type 1min/5min/15min/30min/60min/1day/1week/1mon

交易对信息接口

6. Get /api/v1/exchangeInfo 获取交易对信息

URL <https://www.tokok.com/api/v1/exchangeInfo>

示例

```
# Request  
GET https://www.tokok.com/api/v1/exchangeInfo  
# Response  
[  
  {  
    "symbol": "eth_btc",  
    "status": "trading",  
    "baseAsset": "eth",  
    "baseAssetPrecision": 8,  
    "quoteAsset": "btc",  
    "quoteAssetPrecision": 8
```



```
    },  
    {  
      "symbol": "eth_btc",  
      "status": "trading",  
      "baseAsset": "eth",  
      "baseAssetPrecision": 8,  
      "quoteAsset": "btc",  
      "quoteAssetPrecision": 8  
    }  
  ]
```

返回值说明

symbol:交易对

status: 交易对状态（目前仅有 trading 交易中一种状态）

baseAsset: 基础货币

baseAssetPrecision: 基础货币精度

quoteAsset: 计价货币

quoteAssetPrecision: 计价货币精度

币币交易 API

用于币币交易

币币账户信息

7. POST /api/v1/accounts 获取用户所有账户信息

URL <https://www.tokok.com/api/v1/accounts> 访问频率 6 次/2 秒

示例

Request

POST <https://www.tokok.com/api/v1/accounts>

Response

```
{
  "data": [
    {
      "hotMoney": 8897.319,
      "coldMoney": "0",
      "coinCode": "ARDR"
    },
    {
      "hotMoney": "7003.556618",
      "coldMoney": "0",
      "coinCode": "AE"
    }
  ],
  "result": true
}
```

返回值说明

hotMoney:账户可用余额

coldMoney:账户冻结余

coinCode:币种代码

请求参数

参数名	参数类型	必填	描述
无			

币对账户信息

8. POST /api/v1/singleAccount 获取用户币对账户信息

URL <https://www.tokok.com/api/v1/singleAccount> 访问频率 6次/2秒

示例

Request

POST https://www.tokok.com/api/v1/singleAccount

Response

```
{
  "data": [
    {
      "hotMoney": 8897.319,
      "coldMoney": "0",
      "coinCode": "TOK"
    },
    {
      "hotMoney": "7003.556618",
      "coldMoney": "0",
      "coinCode": "ETH"
    }
  ],
  "result": true
}
```

返回值说明

hotMoney:账户可用余额

coldMoney:账户冻结余

coinCode:币种代码

请求参数

参数名	参数类型	必填	描述
symbol	String	是	交易对, 如 tok_eth

下单交易

9. POST /api/v1/trade 下单交易

URL https://www.tokok.com/api/v1/trade 访问频率 20 次/2 秒

示例

```
# Request
POST https://www.tokok.com/api/v1/trade
# Response
{"result":true,"data":"181114210709001779"}
```

返回值说明

```
result: true 代表下单请求成功
data: 订单流水号
```

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
type	String	是	买卖类型: 限价单(1 买/2 卖)
entrustPrice	Double	是	下单价格
entrustCount	Double	是	交易数量
openTok	Integer	否	是否用 TOK 抵扣交易手续费 (1 是; 0 否, 默认值)

批量下单

10. POST /api/v1/batchTrade 批量下单

URL <https://www.tokok.com/api/v1/batchTrade> 访问频率 20 次/2 秒

示例

```
# Request
```

POST https://www.tokok.com/api/v1/batchTrade

Response

```
{"result":true,"data":[{"order_id":"181115095219001580"},{"order_id":"181115095219002234"}]}
```

返回值说明

result: true 代表下单请求成功（只要有一个下单成功则返回 true）

order_id: 订单流水号，下单失败则为-1

error_code: 下单失败代码

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
type	String	是	买卖类型：限价单(1 买/2 卖)
openTok	Integer	否	是否用 TOK 抵扣交易手续费（1 是 0 否，默认值）
orders_data	String	是	示例： [{"amount":"100","price":"0.02","type":1},{ "amount":"200","price":"0.03","type":1}] 最大下单量为 5，price 和 amount 参数参考 trade 接口中的说明，最终买卖类型由 orders_data 中 type 为准，如 orders_data 不设定 type 则由上面 type 设置为准。

撤销订单

11. POST /api/v1/cancelEntrust 撤销订单

URL https://www.tokok.com/api/v1/cancelEntrust 访问频率 20 次/2 秒

示例

Request

POST https://www.tokok.com/api/v1/cancelEntrust

Response

```
{"result":true,"code":0}
```

返回值说明

result : true 撤单请求成功, 等待系统执行撤单; false 撤单失败
code: 撤单失败代码

请求参数

参数名	参数类型	必填	描述
order_id	String	是	订单流水号

批量撤销

12. POST /api/v1/batchCancel 撤销订单

URL https://www.tokok.com/api/v1/batchCancel 访问频率 20 次/2 秒

示例

Request

POST https://www.tokok.com/api/v1/batchCancel

Response

```
{"result":true,"code":0}
```

返回值说明

result : true 撤单请求成功, 等待系统执行撤单; false 撤单失败
code: 撤单失败代码

请求参数

参数名	参数类型	必填	描述
-----	------	----	----

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
order_ids	String	是	订单流水号, 如: [181114210459002003,181114210459001083] 最大撤单量为 5

获取订单列表

13. POST /api/v1/order/orders 获取当前与历史委托订单信息

URL <https://www.tokok.com/api/v1/order/orders> 访问频率 20 次/2 秒

示例

```
# Request
POST https://www.tokok.com/api/v1/order/orders

# Response
{
  "data": [
    {
      "entrustNum": "181114210459002003",
      "coinCode": "TOK",
      "fixPriceCoinCode": "ETH",
      "type": 1,
      "status": 0,
      "entrustPrice": "0.03",
      "entrustCount": "200",
      "entrustSum": "6",
      "surplusEntrustCount": "200",
      "transactionSum": "0",
      "entrustTime_long": 1542200699132,
      "transactionFee": "0",
      "processedPrice": "0",
```

```

        "openTokFee": 0,
        "symbol": "TOK_ETH"
    },
]
"result": true,
"total": 3
}

```

返回值说明

data:	委托详细信息
entrustCount:	委托数量
processedPrice:	平均成交价
entrustTime_long:	委托时间
surplusEntrustCount:	剩余数量
entrustNum:	订单流水号
entrustPrice:	委托价格
status:	0:未成交 1:部分成交 2:完全成交 3:部分撤销 4:全部撤单
type:	买卖类型: 限价单(1 买/2 卖)
result:	true 代表成功返回
total:	记录数

请求参数

参数名	参数类型	必填	描述
symbol	String	是	币对如 tok_eth
status	Integer	是	查询状态 0: 未完成的订单 1: 已经完成的订单 (最多查询 10 条)
page	Integer	是	当前页数
pageSize	Integer	是	每页数据条数, 最多不超过 50

获取订单详情

14. POST /api/v1/order/orderInfo 获取用户的订单信息

URL <https://www.tokok.com/api/v1/order/orderInfo> 访问频率 20 次/2 秒

示例

```
# Request
POST https://www.tokok.com/api/v1/order/orderInfo
# Response
{
  "result": true,
  "data": {
    "entrustNum": "181114201745001862",
    "type": 1,
    "status": 0,
    "entrustPrice": "1.02",
    "entrustCount": "50",
    "entrustSum": "51",
    "surplusEntrustCount": "50",
    "entrustTime_long": 1542197865000,
    "transactionFee": "0",
    "processedPrice": "0",
    "openTokFee": 0,
    "symbol": "TOK_ETH"
  }
}
```

返回值说明

entrustCount:	委托数量
processedPrice:	平均成交价
entrustTime_long:	委托时间
surplusEntrustCount:	剩余数量
entrustNum:	订单流水号
entrustPrice:	委托价格

status: 0:未成交 1:部分成交 2:完全成交 3:部分撤销 4:全部撤单
type: 买卖类型: 限价单(1 买/2 卖)
result: true 代表成功返回

请求参数

参数名	参数类型	必填	描述
order_id	String	是	订单流水号

错误代码

错误代码	详细描述
80101	超过频率限制
80102	非用户绑定的 ip
82101	查询不到账户信息
82102	查询币对账户失败
83001	交易时必须绑定手机或谷歌验证
83002	是否开启交易需要实名认证
83103	用户被禁用
83104	禁止用户交易

83005	交易对为空
83006	交易对格式错误
83007	交易下单类型不正确
83008	交易下单价格不正确
83009	交易下单数量不正确
83110	交易下单失败
83011	订单号错误
83112	撤单失败
83013	订单 id 为空
83114	查询不到此订单
83015	批量下单的订单参数为空
83016	批量下单的订单数量超过限制
84001	缺少时间戳
84002	缺少签名
84003	缺少参数

84004 签名错误

84005 签名无效

84006 缺少 api_key

84007 API key 无效