

# Enable API

## Enable API

'api\_key' is the users' access key provided by TOKOK, 'secretKey' is used to request parameter signature. To get api\_key and secretKey, go to Account center- Security center and click Create API Key.

Note: DO NOT disclose these keys to anyone. Otherwise your account might be unsecure.

## Parameter Signature

Web service requests are sent across the Internet and are vulnerable to tampering. For security reasons, TOKOK requires a signature as part of every request except for ticker API.

All request formatted for signature must contain:

ACCESS-KEY - that distributed when users created API Key

ACCESS-SIGN- base64 encoded signature

ACCESS-TIMESTAMP- the time at which you make the request. Unix timestamp format decimal in seconds (Ex: 1543057116)

## How to sign

All parameters submitted by users, ACCESS-KEY and ACCESS-TIMESTAMP must to be signed.

The parameters must be re-ordered alphabetically according to the initials of the parameter name.

For example,

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

```
ACCESS-TIMESTAMP=1543057116
```

If the parameters are

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

The result string is

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

```
TIMESTAMP=1543057116&entrustCount=100&entrustPrice=1.03&symbol=tok_eth&type=1
```

'secretKey' is required to generate HMAC SHA256 signature. Use HMAC SHA256 encryption function to sign the string. Pass the encrypted string, which is base64 encoded, to 'ACCESS-SIGN' parameter.

## REST API Reference

### Price API

#### Get ticker

1. Get /api/v1/tickers Used to get price tickers for all markets.

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

Example

```
# 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"}
  ]
}
```

## Response Details

timestamp: server time for returned data  
symbol: trading pair  
buy: bid price  
high: highest trade price of the last 24 hours  
last: the price at which the last order executed  
low: lowest trade price of the last 24 hours  
sell: ask price  
vol: trading volume of the last 24 hours

2.Get /api/v1/ticker Used to get the current tick values for a market.

URL [https://www.tokok.com/api/v1/ticker?symbol=tok\\_eth](https://www.tokok.com/api/v1/ticker?symbol=tok_eth)

## Example

```
# 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"
  }
}
```

## Response Details

timestamp: server time for returned data  
buy: bid price  
high: highest trade price of the last 24 hours  
last: the price at which the last order executed  
low: lowest trade price of the last 24 hours  
sell: ask price  
vol: trading volume of the last 24 hours

## Parameters

Parameters	Type	Required	Description
symbol	String	Yes	Ex: tok_eth

## Get market depth

3. Get /api/v1/depth Used to get market depth of a symbol

URL [https://www.tokok.com/api/v1/depth?symbol=tok\\_eth](https://www.tokok.com/api/v1/depth?symbol=tok_eth)

Example

```
# 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"]
  ]
}
```

### Response Details

asks : ask array [[price, volume],... ]

bids : bid array [[price, volume],... ]

### Parameters

Parameters	Type	Required	Description
symbol	String	Yes	Ex: tok_eth
size	Integer	Optional(default 50)	value: 1-50

## Get recent trades

### 4. Get /api/v1/trades Get recently 50 trades for a symbol

URL [https://www.tokok.com/api/v1/trades?symbol=tok\\_eth&size=50](https://www.tokok.com/api/v1/trades?symbol=tok_eth&size=50)

#### Example

```
# 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"  
  }  
]
```

#### Response Details

timestamp: time

price: price

amount: amount

side: buy/sell 'sell' or 'buy'

#### Parameters

Parameters	Type	Required	Description
symbol	String	Yes	Ex: tok_eth
size	Integer	Optional	Ex: 50

## Get K line

### 5. Get /api/v1/kline Get candlestick data

URL [https://www.tokok.com/api/v1/kline?symbol=tok\\_eth&size=20&type=1min](https://www.tokok.com/api/v1/kline?symbol=tok_eth&size=20&type=1min)

Example

```
# 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"
  ],
]
```

```
]
```

## Response Details

```
[  
  "1417536000000", timestamp  
  "2370.16",      opening  
  "2380",        high  
  "2352",        low  
  "2367.37",    closing  
  "17259.83"     volume  
]
```

## Parameters

Parameters	Type	Required	Description
symbol	String	Yes	Ex: tok_eth
type	String	Yes	
size	Integer	Optional	Limit the amount of data returned

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

## Get exchange info

6. Get /api/v1/exchangeInfo Used to get info of a symbol

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

### Example

```
# 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
  }
]
```

## Response Details

symbol: trading pair

status: status (At present, only status is in trading)

baseAsset: base currency

baseAssetPrecision: decimal number of base currency

quoteAsset: quote currency

quoteAssetPrecision: decimal number of quote currency

# Trade API

Place orders on exchange

## Get user info

7.POST /api/v1/accounts Used to get user account info for all funds

URL <https://www.tokok.com/api/v1/accounts> Request frequency 6 times/2s

### Example

```
# 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
}
```

### Response Details

hotMoney: available fund  
coldMoney: frozen fund  
coinCode: currency symbol

### Parameters

Parameters	Type	Required	Description
N/A			

## Get pair info

8.POST /api/v1/singleAccount Used to get account info for one pair

URL <https://www.tokok.com/api/v1/singleAccount> Request frequency 6 times/2s

### Example

```
# 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
}
```

### Response Details

hotMoney: available balance

coldMoney: frozen balance

coinCode: currency symbol

### Parameters

Parameters	Type	Required	Description
------------	------	----------	-------------

Parameters	Type	Required	Description
symbol	String	yes	Trading pair like: tok_eth

## Place orders

9.POST /api/v1/trade Used to place orders

URL <https://www.tokok.com/api/v1/trade> Request frequency 20 times/2s

Example

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

Response details

```
result: true means order placed successfully
data: order ID
```

## Parameters

Parameters	Type	Requires	Description
symbol	String	Yes	Ex: tok_eth
type	String	Yes	order type: limit order(1 buy/2 sell)
entrustPrice	Double	Yes	order price

Parameters	Type	Requires	Description
entrustCount	Double	Yes	order amount
openTok	Integer	Optional	Using Tok to pay for fees (1 yes; 0 no, default)

## Batch trade

10.POST /api/v1/batchTrade Used to batch trade

URL <https://www.tokok.com/api/v1/batchTrade> Request frequency 20 times/2s

### Example

```
# Request
POST https://www.tokok.com/api/v1/batchTrade
# Response
{"result":true,"data":[{"order_id":"181115095219001580"},{"order_id":"181115095219002234"}]}
```

### Response Details

result: true means order successfully placed(return true if any one order is placed successfully)  
order\_id: order ID, if fail to place order: order\_id is -1  
error\_code: error code

### Parameters

Parameters	Type	Required	Description
symbol	String	yes	Ex: tok_eth
type	String	yes	order type: limit orders (1 buy/2 sell)

Parameters	Type	Required	Description
openTok	Integer	Optional	Using TOK to pay for fees (1 yes, 0 no, default)
orders_data	String	Yes	Ex: [{"amount":"100","price":"0.02","type":1},{ "amount":"200","price":"0.03","type":1}] max order number is 5, for 'price' and 'amount' parameter, refer to trade/API. Final order type is decided primarily by 'type' field within 'orders_data' and subsequently by 'type' field (if no 'type' is provided within 'orders_data' field)

## Cancel order

11.POST /api/v1/cancelEntrust Used to cancel orders

URL <https://www.tokok.com/api/v1/cancelEntrust> Request frequency 20 times/2s

Example

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

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

## Response Details

result : true means order cancelled successfully, wait to be processed; false means fail to cancel order

code: error code

## Parameters

Parameters	Type	Requires	Description
------------	------	----------	-------------

Parameters	Type	Requirements	Description
order_id	String	Yes	Order ID

## Batch cancel

12. POST /api/v1/batchCancel Used to batch cancel

URL <https://www.tokok.com/api/v1/batchCancel> Request frequency 20 times/2s

Example

```
# Request
POST https://www.tokok.com/api/v1/batchCancel
# Response
{"result":true,"code":0}
```

## Response Details

```
result : true means order cancelled successfully, wait to be processed; false means fail to cancel
order
code: error code
```

## Parameters

Parameters	Type	Required	Description
symbol	String	yes	Ex: tok_eth
order_ids	String	yes	Order Ids, Ex: [181114210459002003,181114210459001083] Maximum number is 5

## Get orders info

13. POST /api/v1/order/orders Used to get order information in batch

URL <https://www.tokok.com/api/v1/order/orders> Request frequency 20 times/2s

### Example

```
# 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
}
```

### Response Details

data:	order info
-------	------------



entrustCount: order amount  
 processedPrice: average transaction price  
 entrustTime\_long: order time  
 surplusEntrustCount: unfilled amount  
 entrustNum: order ID  
 entrustPrice: order price  
 status: 0:unfilled 1:partially filled 2: fully filled 3:partially cancelled 4:cancelled  
 type: order type: limit order(1 buy/2 sell)  
 result: true means request successfully handled  
 total: record numbers

## Parameters

Parameters	Type	Required	Description
symbol	String	Yes	Ex: tok_eth
status	Integer	Yes	query status: 0 for unfilled orders, 1 for filled orders (maximum 10)
page	Integer	yes	Current page number
pageSize	Integer	Yes	number of orders returned per page, maximum 50

## Get order info

14.POST /api/v1/order/orderInfo Used to get specific order info

URL <https://www.tokok.com/api/v1/order/orderInfo> Request frequency 20 times/2s

### Example

```

# 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"
}
}

```

## Response Details

entrustCount: order amount  
 processedPrice: average transation price  
 entrustTime\_long: order time  
 surplusEntrustCount: unfilled amount  
 entrustNum: order ID  
 entrustPrice: order price  
 status: 0:unfilled 1:partially filled 2:fully filled 3:partially cancel 4:cancelled  
 type: order type: limit order (1 buy/2 sell)  
 result: true means request successfully handled

## Parameters

Parameters	Type	Requires	Description
order_id	String	Yes	order ID

## Error code

Error code	Description
80101	Request frequency too high
80102	This IP is not allowed to access
82101	No account information
82102	Fail to get pair info
83001	Please enable SMS or GOOGLE authentication
83002	Please enable Real-name authentication
83103	Account blocked
83104	Account is prohibited from trading
83005	Trading pair can not be null
83006	Trading pair format error
83007	Incorrect order type
83008	Incorrect order price
83009	Incorrect order amount
83110	Place order failed
83011	Incorrect order ID
83112	Cancel order failed
83013	Order ID can not be null
83114	No order
83015	Required parameters of batch trade can not be null

---

83016	Exceed maximum order number
84001	Lack of timestamp
84002	Lack of signature
84003	Lack of parameters
84004	Signature error
84005	Invalid signature
84006	Lack of api_key
84007	Invalid API key

---