

行情 API WebSocket 订阅指南

1. WebSocket

WebSocket 服务仅适用于我们的白名单客户。此文档用于介绍 WebSocket 的相关规则及具体订阅频道

1.1 WebSocket 介绍

1.1.1 概述和使用限制

WebSocket 是一种新的 HTML5 协议，实现了客户端和服务端之间的全双工数据传输，允许数据在两个方向上有效传输。只需一次握手即可建立客户端和服务端之间的连接，服务器将能够根据预设规则将数据推送到客户端。其优点包括：

- 客户端和服务端之间数据传输的 WebSocket 请求头大小仅为 2 个字节。
- 客户端或服务端都可以启动数据传输。
- 无需重复创建和删除 TCP 连接，从而节省带宽和服务端资源。

连接

连接限制： 每秒 3 个请求（基于 API KEY）

订阅私有频道时，使用私有服务的地址

请求限制：

每个连接的“订阅” / “取消订阅” / “登录” 请求总数限制为 每小时 480 次。

如果出现网络问题，系统会自动关闭连接。

如果订阅未建立或超过 30 秒未推送数据，连接将自动中断。

保持连接稳定：

1. 每当收到响应消息时，设置一个 N 秒的定时器，其中 N 小于 30。
2. 如果定时器被触发，这意味着在 N 秒内没有收到新消息，则发送字符串“ping”。
3. 期望一个“pong”作为响应。如果在 N 秒内没有收到响应消息，请提出错误或重新连接。

连接计数限制

限制将设置为每个特定 WebSocket 通道每个 API KEY 30 个 WebSocket 连接。每个 WebSocket 连接由唯一的 `conId` 标识。

受此限制的 WebSocket 通道如下：

1. Price 币价频道

2. K线频道

3. 交易数据频道

如果用户通过多个参数通过同一个 WebSocket 连接订阅了同一个频道，则仅计算一次。如果用户使用相同或不同的连接订阅了列出的频道（如币价和 K 线），则不会影响计数，因为这被视为两个不同的频道。系统计算每个频道的 WebSocket 连接数。

平台将通过 `channel-conn-count` 事件消息向客户端发送活动连接数到**新频道订阅**。

连接计数更新

代码块

```
1  {
2    "event": "channel-conn-count",
3    "channel": "prices",
4    "connCount": "2",
5    "connId": "abcd1234"
6  }
```

当超过限制时，通常发送订阅请求的最新连接将被拒绝。客户端将收到通常的订阅确认，然后是来自连接的订阅已终止的 `channel-conn-count-error`。在特殊情况下，平台可能会取消订阅现有连接。

连接限制错误

代码块

```
1  {
2    "event": "channel-conn-count-error",
3    "channel": "prices",
4    "connCount": "20",
5    "connId": "a4d3ae55"
6  }
```

1.1.2 身份验证和频道订阅

登录

请求参数

参数	类型	要求	描述
op	String	Yes	操作 登陆
args	Array	Yes	登陆账户的信息
> apiKey	String	Yes	API Key
> passphrase	String	Yes	API Key 密码
> timestamp	String	Yes	Unix 时间戳, 单位是秒
> sign	String	Yes	签名字符串

响应参数

参数	类型	要求	描述
event	String	Yes	Operation 登陆 错误
code	String	No	错误码
msg	String	No	错误信息
connId	String	Yes	WebSocket 连接 ID

apiKey: 调用API 的唯一标识。需要用户手动申请一个。

密码: API 密钥密码

时间戳: Unix 时间戳, 单位是秒, 例如 1704876947

sign: 签名字符串, 签名算法如下:

首先连接 时间戳、方法、request estPath、字符串, 然后使用 HMAC SHA256方法对连接后的字符串进行SecreKey 加密, 然后进行 Base64 编码。

SecreKey: 用户申请API Key时生成的安全密钥, 例如: 22582BD0CFF14C41EDBF1AB98506286D

时间戳示例: `const时间戳 = '' + Date.now () /1,000`

其中符号示例: `sign=CryptoJS.enc.Base64.stringify (CryptoJS.HmacSHA256 (时间戳+'GET'+'/用户/自己/验证', API key 密钥))`

方法: 始终为'GET'。

请求路径: 始终为 '/users/self/verify'

请求将在时间戳后30秒到期。如果您的服务器时间与 API 服务器时间不同, 我们建议使用 RESTAPI 查询 API 服务器时间, 然后设置时间戳。

请求示例

代码块

```
1 {"op": "login", "args": [{"apiKey": "985d5b66-57ce-40fb-b714-afcf0b9787083", "passphrase": "123456", "timestamp": "1538054050", "sign": "7L+zFQ+CEgGu5rzCj4+BdV2/uUHGqddA9pI6ztsRRPs="}]}
```

成功响应示例

代码块

```
1 {  
2   "event": "login",  
3   "code": "0",  
4   "msg": "",  
5   "connId": "a4d3ae55"  
6 }
```

故障响应示例

代码块

```
1 {  
2   "event": "error",  
3   "code": "600009",  
4   "msg": "Login failed.",  
5   "connId": "a4d3ae55"  
6 }
```

订阅

订阅说明

WebSocket频道都是 `private` 频道。

`private` --包括币价频道、K 线频道和交易频道等--需要 API key 登录。

用户可以选择订阅一个或多个频道，多个频道的总长度不能超过 64KB。

下面以订阅参数为例，各渠道对订阅参数的要求不同，具体可参考各渠道规范。

请求参数

参数	类型	要求	描述
op	String	Yes	运营 订阅
args	Array	Yes	订阅频道列表
> channel	String	Yes	频道名称
> chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex
> tokenContractAddress	String	Yes	令牌合约地址（例如，0x382bb369d343125bfb2117af9c1

返回参数

参数	类型	要求	描述
event	String	Yes	事件 订阅 错误
arg	Object	No	订阅频道
> channel	String	Yes	频道名称
> chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
> tokenContractAddress	String	Yes	令牌合约地址（例如，0x382bb369d343125bfb2117af9c14
code	String	No	错误代码
msg	String	No	错误信息
connId	String	Yes	WebSocket连接ID

请求格式说明

代码块

```
1  {"op": "subscribe","args": ["<SubscriptionTopic>"]}
```

请求示例

代码块

```
1  {"op":"subscribe","args":[{"channel":"price","chainIndex":  
  "1","tokenContractAddress":"0x382bb369d343125bfb2117af9c149795c6c65c50"}]}
```

响应示例

代码块

```
1  {
```

```
2      "event": "subscribe",
3      "arg": {
4          "channel": "price",
5          "chainIndex": "1",
6          "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
7      },
8      "connId": "accb8e21"
9  }
```

取消订阅

取消订阅一个或多个频道。

请求参数

参数	类型	要求	描述
op	String	Yes	操作 取消订阅
args	Array	Yes	取消订阅的频道列表
> channel	String	Yes	频道名称
> chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
> tokenContractAddress	String	Yes	代币合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）

响应参数

参数	类型	要求	描述
event	String	Yes	事件 取消订阅 错误
arg	Object	No	取消订阅的频道
> channel	String	Yes	频道名称
> chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
> tokenContractAddress	String	Yes	代币合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）
code	String	No	错误代码
msg	String	No	错误信息

请求格式说明

代码块

```
1  {"op": "unsubscribe", "args": ["< SubscriptionTopic> "]}
```

请求示例

代码块

```
1 {"op": "unsubscribe", "args": [{"channel": "price", "chainIndex":  
  "1", "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"}]}
```

响应示例

代码块

```
1 {  
2   "event": "unsubscribe",  
3   "arg": {  
4     "channel": "price",  
5     "chainIndex": "1",  
6     "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"  
7   },  
8   "connId": "d0b44253"  
9 }
```

通知

WebSocket引入了一种新的消息类型（event= `notice`）。

客户将在以下情况下收到信息：

- Websocket断开服务升级

WebSocket服务升级前30秒，将向用户发送通知消息，提示连接即将断开。鼓励用户建立新的连接，以防止断开连接造成的任何中断。

响应示例

代码块

```
1 {  
2   "event": "notice",  
3   "code": "64008",  
4   "msg": "The connection will soon be closed for a service upgrade. Please  
  reconnect.",  
5   "connId": "a4d3ae55"  
6 }
```

1.2 WebSocket 频道

1.2.1 币价

WS价格频道

检索代币的最新价格数据。每当有更新时都会推送数据。

请求URL

wss: [//wsdex.okx.com/ws/v5/dex](https://wsdex.okx.com/ws/v5/dex)

请求参数

参数	类型	要求	描述
op	String	Yes	操作，订阅或者取消订阅
args	Array	Yes	订阅频道列表
channel	String	Yes	频道名称
chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
tokenContractAddress	String	Yes	代币合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）

响应参数

参数	类型	要求	描述
event	String	Yes	事件，订阅，取消订阅错误
arg	Object	No	订阅频道
channel	String	yes	频道名称 价格
chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
tokenContractAddress	String	Yes	代币合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）
code	String	No	错误码
msg	String	No	错误信息

请求示例

代码块

```
1  {
2    "op": "subscribe",
3    "args": [
4      {
5        "channel": "price",
6        "chainIndex": "66",
7        "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
8      }
9    ]
10 }
```


响应示例

成功的响应示例

代码块

```
1  {
2    "event": "subscribe",
3    "arg": {
4      "channel": "price",
5      "chainIndex": "66"
6      "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
7    },
8    "connId": "a4d3ae55"
9  }
```

故障响应示例

代码块

```
1  {
2    "event": "error",
3    "code": "60012",
4    "msg": "Invalid request: {\\"op\\": \\"subscribe\\", \\"argss\\": [{ \\"channel\\" :
5    \\"price\\", \\"chainIndex\\" : \\"1\\", \\"tokenContractAddress\\" :
6    \\"0x382bb369d343125bfb2117af9c149795c6c65c50\\"}]}",
7    "connId": "a4d3ae55"
8  }
```

推送数据参数

参数	类型	描述
arg	Object	成功订阅频道
> channel	String	频道名称
> chainIndex	String	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
> tokenContractAddress	String	令牌合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）
data	Array	订阅数据
> time	String	价格的时间戳，Unix时间戳格式，单位毫秒
> price	String	最新代币价格

推送数据示例

```
1  {
2    "arg": {
3      "channel": "price",
4      "chainIndex": "1"
5      "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
6    },
7    "data": [
8      {
9        "time": "1716892020000",
10       "price": "26.458143090226812",
11      }
12    ]
13  }
```

1.2.2 K 线频道

K 线频道

检索代币的 K 线数据。最快的推送频率为每秒1次推送。

请求URL

wss: [//wsdex.okx.com/ws/v5/dex](https://wsdex.okx.com/ws/v5/dex)

请求参数

参数	类型	要求	描述
op	String	Yes	运营，订阅退订
args	Array	Yes	订阅频道列表
channel	String	Yes	频道名称 dex-token-candle4H dex-token-candle2H dex-token-candle30m dex-token-candle15m dex-token-candle5m token-candle1m dex-token-candle1s dex-token-candle3M dex-token-candle1M dex-token-candle1D dex-token-candle12H dex-token-candle6H dex-token-candle3Mutc dex-token-candle1Mutc dex-token-candle1Dutc dex-token-candle12Hutc dex-token-candle12H
chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex
tokenContractAddress	String	Yes	代币合约地址（例如，0x382bb369d343125bfb2117af9

响应参数

参数	类型	要求	描述
event	String	Yes	事件，订阅取消订阅错误
arg	Object	No	订阅频道
channel	String	yes	频道名称
chainIndex	String	Yes	链的唯一标识符。（例如，以太坊为1。请参见 ChainIndex ）
tokenContractAddress	String	Yes	令牌合约地址（例如，0x382bb369d343125bfb2117af9c149795c6c65c50）
code	String	No	错误代码
msg	String	No	错误信息

请求示例

代码块

```
1  {
2    "op": "subscribe",
3    "args": [
4      {
5        "channel": "candle1D",
6        "chainIndex": "66",
7        "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
8      }
9    ]
10 }
```

响应示例

成功的响应示例

代码块

```
1  {
2    "event": "subscribe",
3    "arg": {
4      "channel": "candle1D",
5      "chainIndex": "66"
6      "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
7    },
8    "connId": "a4d3ae55"
9  }
```

故障响应示例

代码块

```

1  {
2    "event": "error",
3    "code": "60012",
4    "msg": "Invalid request: {\\"op\\": \\"subscribe\\", \\"argss\\": [{ \\"channel\\": \\"
    \\"candle1D\\", \\"chainIndex\\": \\"1\\", \\"tokenContractAddress\\": \\"0x382bb369d343125bfb2117af9c149795c6c65c50\\"} ]}",
5    "connId": "a4d3ae55"
6  }

```

推送数据参数

Parameter	Type	Description	Note
arg	Object	成功订阅频道	
> channel	String	频道名称	
> chainIndex	String	链的唯一标识 (e.g., 1 for Ethereum. See ChainIndex)	
> tokenContractAddress	String	代币合约地址 (e.g., 0x382bb369d343125bfb2117af9c149795c6c65c50)	
data	Array	已订阅的数据	
> ts	String	K 线的产生时间, Unix 时间戳, 毫秒级 e.g. 1597026383085	
> o	String	开盘价	
> h	String	最高价	
> l	String	最低价	
> c	String	收盘价	
> vol	String	交易量, 按代币最小单位展示	
> volUsd	String	交易量的美金价值	
> confirm	String	K线的状态, 0:代表完成, 1代表未完成	

推送数据示例

代码块

```

1  {
2    "arg": {
3      "channel": "candle1D",
4      "chainIndex": "1"
5      "tokenContractAddress": "0x382bb369d343125bfb2117af9c149795c6c65c50"
6    },
7    "data": [
8      [
9        "1597026383085",
10       "8533.02",
11       "8553.74",

```

```
12     "8527.17",
13     "8548.26",
14     "529.5858061",
15     "226348.0482",
16     "0"
17 ]
18 ]
19 }
```

1.2.3 交易频道

WS交易频道

检索最近的交易数据。只要有交易，就会推送数据。

请求URL

```
wss: //wsdex.okx.com/ws/v5/dex
```

请求参数

Parameter	Type	Required	Description
op	String	Yes	操作，订阅或者不订阅
args	Array	Yes	订阅的详细信息
channel	String	Yes	频道名称 trades
chainIndex	String	Yes	链的唯一标识. (e.g., 1 for Ethereum. See ChainIndex)
tokenContractAddress	String	Yes	代币合约地址 (e.g., 0x382bb369d343125bfb2117af9c14

响应参数

请求示例

代码块

```
1  {
2    "op": "subscribe",
3    "args": [
4      {
5        "channel": "trades",
6        "chainIndex": "501",
7        "tokenContractAddress": "HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC"
8      }
9    ]
10 }
```

响应示例

成功的响应示例

代码块

```
1  {
2    "event": "subscribe",
3    "arg": {
4      "channel": "trades",
5      "chainIndex": "501",
6      "tokenContractAddress": "HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC"
7    },
8    "connId": "a4d3ae55"
9  }
```

故障响应示例

代码块

```
1  {
2    "event": "error",
3    "code": "60012",
4    "msg": "Invalid request: {\\"op\\": \\"subscribe\\", \\"argss\\": [{ \\"channel\\" : \\"trades\\", \\"chainIndex\\" : \\"501\\", \\"tokenContractAddress\\" : \\"HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC\\"} ]}",
5    "connId": "a4d3ae55"
6  }
```

推送数据参数

推送数据示例

代码块

```
1  {
2    "arg": {
3      "channel": "trades",
4      "chainIndex": "501"
5      "tokenContractAddress": "HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC"
6    },
7    "data": [
8      {
9        "id": "1739439633000!@#120!@#14731892839",
10       "chainIndex": "501",
11       "tokenContractAddress": "HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC",
```

```
12      "txHashUrl":  
    "https://solscan.io/tx/zgDzoiVG4XuDgQcoEg9vhpRyfyk5thNUQuTeTCeF289Qec5iraeCrUzP  
    LyiE2UCviox2ebbTcsagGvzYF7M5uqs",  
13      "userAddress": "2kCm1RHGJjeCKL4SA3ZJCLyXqUD7nEJ7GMtVaP7c6jQ8",  
14      "dexName": "Orca Whirlpools",  
15      "poolLogoUrl": "https://static.okx.com/cdn/wallet/logo/dex_orcaswap.png",  
16      "type": "sell",  
17      "changedTokenInfo": [  
18      {  
19      "amount": "100.396595878",  
20      "tokenSymbol": "ai16z",  
21      "tokenContractAddress": "HeLp6NuQkmYB4pYWo2zYs22mESHXPQYzXbB8n4V98jwC"  
22      },  
23      {  
24      "amount": "2.482831",  
25      "tokenSymbol": "SOL",  
26      "tokenContractAddress": "So1111111111111111111111111111111111111112"  
27      }  
28      ]  
29      "price": "26.458143090226812",  
30      "volume": "519.788163",  
31      "time": "1739439633000",  
32      "isFiltered": "0"  
33      }  
34      ]  
35      }
```

1.3 Websocket 错误代码

Error Code	Error Message
60004	Invalid timestamp
60005	Invalid apiKey
60006	Timestamp request expired
60007	Invalid sign
60011	Please log in
60012	Invalid request
60013	Invalid args
60014	Requests too frequent
60018	Wrong URL or {0} doesn't exist. Please use the correct URL, channel and parameters referring to API document.
60024	Wrong passphrase
60029	Only users who are in the whitelist are allowed to subscribe to this channel.
60031	The WebSocket endpoint does not allow multiple or repeated logins.
60032	API key doesn't exist.
63999	Login failed due to internal error. Please try again later.