# T-chain - carbon wallet development interface

directory

[1. Overview of the T-chain 2](#_Toc3184)

[2. Instructions for using functions 2](#_Toc11606)

[3. Basic contract interface 3](#_Toc30066)

[1、 List of managers 3](#_Toc11625)

[2、 List of managers to be confirmed 4](#_Toc25234)

[3、 Increase the number of platform managers 4](#_Toc16919)

[4、 Eliminate platform managers 5](#_Toc9206)

[5、 Inquire about the list of tokens 5](#_Toc17599)

[6、 List of tokens to be confirmed 6](#_Toc30757)

[7、 Deploy tokens 6](#_Toc8676)

[8、 Upgrade tokens 7](#_Toc2371)

[9、 List of minting addresses 7](#_Toc21505)

[10、 A list of minting addresses to be confirmed 8](#_Toc9508)

[11、 Add minting address 8](#_Toc16855)

[12、 Delete the minting address 9](#_Toc3797)

[4. ToC token interface 9](#_Toc1098)

[1. Displays the amount of tokens you own 9](#_Toc32142)

[2. Transfers 10](#_Toc30729)

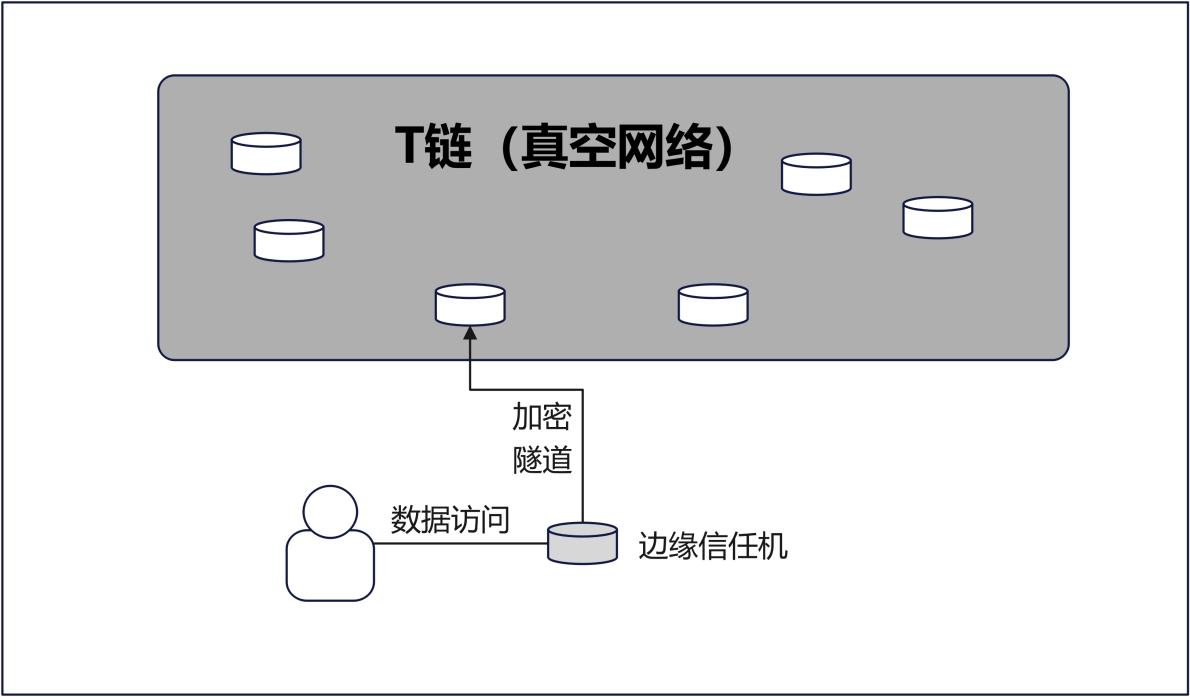
|  |  |  |  |
| --- | --- | --- | --- |
| version | Released | content | author |
| 1.00 | 2024.6.20 | Carbon ecological smart contract API WEB interface | Zoning |

1. **Overview of the T-chain**

T chain (Trust node blockchain network) technical architecture is a historic moment in the direction of blockchain development, the blockchain mainnet network is a standard privacy computing business, it is unimaginable to build its own privacy computing network in the early days of blockchain invention, and to run privacy computing business on the public network, it has also achieved one of the four basic technologies of the blockchain, Byzantine fault tolerance technology, the other three basic technologies are mature technologies invented in the early history, In the later period, the development history of blockchain became the development history of Byzantine fault-tolerant technology, and the T chain brought a trusted privacy computing network, and the entire mainnet node operates in a decentralized network vacuum where humans cannot intervene and interfere, and the network is completely isolated inside and outside, and the only communication channel is to communicate through the encrypted channel of the edge trust machine that can be independently verified.

1. **Instructions for using functions**

T-chain communication architecture:



Edge trust machine download path:

Windows ：

http:/www.xxx.com/download/tmac\_win.rar

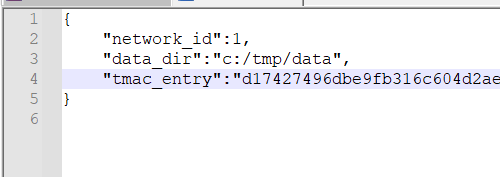
Linux

http:/www.xxx.com/download/tmac\_linux.tar

Directions of use:

First, configure the connection parameters:

chain.conf



Data\_dir: The data directory that exists, there is no space requirement, and it hardly takes up much space

tmac\_entry: Vacuum node trust machine access information

Windws Startup Method:

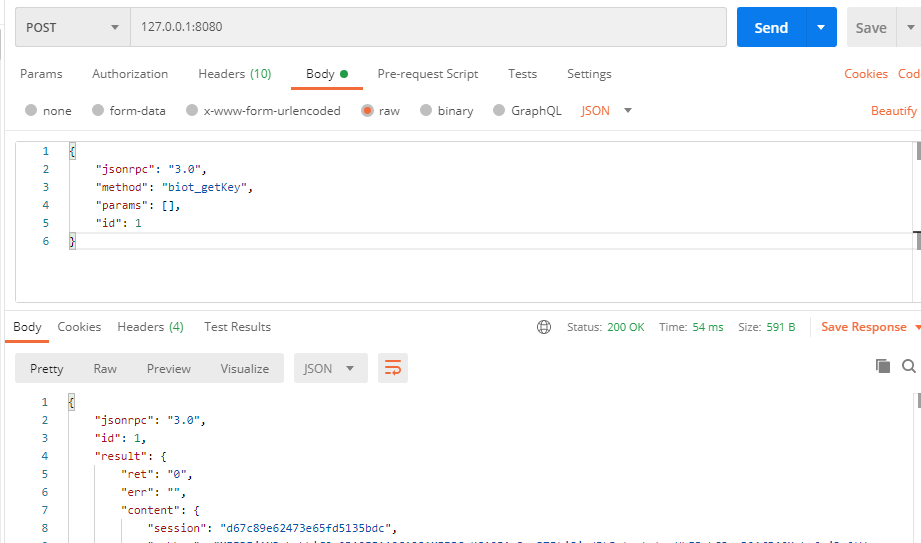
Dirver:/parhname/guard\_tmac.exe

Linux boot mode:

Nohup /pathname/guard\_tmac.exe &

**Postman Operational Interface Methods:**

API Address for Local Machine：<http://127.0.0.1:8080>



1. **The underlying contract interface**
2. List of managers

**parameter**

Addr contract address

**Return value**

addr manager address

name

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showManager&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "result":[{"addr":"","name":""}]  }}} |

1. List of managers to be confirmed

**parameter**

Addr contract address

**Return value**

addr manager address

name Name

At 0: Add 1:Remove

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showPendingManager&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {   "result":[{"addr":"","name":"","op":0}]  }}} |

1. Increase the number of platform managers

**parameter**

Addr contract address

Addr2 administrative address

Name

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=addManager&addr=0x... &addr2=0x... &name=ccc", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. Eliminate platform managers

**parameter**

Addr contract address

Addr2 administrative address

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=removeManager&addr=0x... &addr2=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. Inquire about the list of tokens

**parameter**

Addr contract address

**Return value**

Addr token contract address

Name

Type token type

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showToken&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {  "result":[{"addr":"","name":"","type":0}]  }}} |

1. List of tokens to be confirmed

**parameter**

Addr contract address

**Return value**

Addr token contract address

Name

Type

On deploying or Upgrading ( 0 or 1)

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showPengindToken&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {  "result":[{"addr":"","name":"","type":0,,"op":0}]  }}} |

1. Deploy tokens

**parameter**

Addr contract address

Addr2 token contract address

Name

Type

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=deployToken&addr=0x... &addr2=0x... &name=&type=0", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. Upgrade tokens

**parameter**

Addr contract address

Addr2 token contract address

Name

Type

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=updateToken&addr=0x... &addr1=0x... &name=&type=0", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. List of minting addresses

**parameter**

Addr contract address

**Return value**

Addr2 minting address

Name

Type token type

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showCreater&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {  "result":[{"addr":"","name":"","type":0}]  }}} |

1. A list of minting addresses to be confirmed

**parameter**

Addr contract address

**Return value**

Addr2 minting address

Name

Type token type

On Add or Remove ( 0 or 1)

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=showPendingCreater&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "result":[{"addr":"","name":"","type":0,"op":0}]  }}} |

1. Add minting address

**parameter**

Addr contract address

Addr2 minting address

Name

Type token type

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=addCreater&addr=0x... &addr2=0x... &name=... &type=0", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. Delete the minting address

**parameter**

Addr contract address

Addr2 minting address

**Return value**

txhash: If the submission is successful, the transaction hash will be returned, and the transaction result needs to be obtained regularly (once per second).

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=remvoeCreater&addr=0x... &addr2=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |

1. **ToC token interface**
   1. **Displays the amount of tokens owned**

**parameter**

Addr contract address

**Return value**

Amount Amount of tokens owned

Total Total amount of tokens

**Sample code**

request

|  |
| --- |
| {  "jsonrpc": "3.0",  "method": "chain\_tvm",  "params":["opcode=carbon&subcode=TokenInfo&addr=0x...", "encryp=none"],  "id": "aa80aa0c6bb0c7d81ac7e5b3"} |

Response:

|  |
| --- |
| {    "jsonrpc": "3.0",      "id": 1,      "result": {          "right": "0",          "err": "",          "content": {              amount:0,  total:=0  }}} |

* 1. **Transfer**

**parameter**

to: Recipient

value： amount

**Return value**

bone: Transaction hash

**Sample code**

request

|  |
| --- |
| { "jsonrpc": "3.0",  "method": "chain\_sendTransaction",  "params": [{  "from": "0xc3b6472d6370eaf4eb58fff19ce1724c9d61892c",  "to": "0x6268af5542c4b0ccbb650a0aaccbfcdfbabf6b52",  "value": "0x2E90EDD00"  }, "", "encryp=none"],  "id": "d2c8fe196f9f8d8ace952e9e"} |

Response:

|  |
| --- |
| {      "jsonrpc": "3.0",      "id": "d2c8fe196f9f8d8ace952e9e",      "result": {          "right": "0",          "err": "",          "content": {              "txhash": "0x6885997ca51fa08ffded573088b0c5c294652ec842ac7f3182d32e30ce261025"  }}} |