

Table Of Contents

- Configuration
- Chain API
 - `get_info`
 - `get_block`
 - `get_account`
 - `get_code`
 - `get_table_rows`
 - `abi_json_to_bin`
 - `abi_bin_to_json`
 - `push_transaction`
 - `push_transactions`
 - `get_required_keys`
- Wallet API
 - `wallet_create`
 - `wallet_open`
 - `wallet_lock`
 - `wallet_lock_all`
 - `wallet_import_key`
 - `wallet_list`
 - `wallet_list_keys`
 - `wallet_get_public`
 - `wallet_set_timeout`
 - `wallet_sign_tx`

```
{
  "previous": "0000000445a9f27898383fd7de32835d5d6a978cc14ce40d9f327b5329de796b",
  "timestamp": "2017-07-18T20:16:36",
  "transaction_merkle_root": "0000000000000000000000000000000000000000000000000000000000000000",
  "producer": "initf",
  "producer_changes": [ ],
  "producer_signature": "204cb94b3186c3b4a7f88be4e9db9f8af2ffdb7ef0f27a065c8177a5fcfacf876f684e59c39fb009903c0c59220b147bb07f1144df1c65c",
  "cycles": [ ],
  "id": "000000050c0175cbf218a70131ddc3c3fab8b6e954edef77e0bfe7c36b599b1d",
  "block_num": 5,
  "ref_block_prefix": 27728114
}
```

get_account

Get information related to an account.

Example get_account Usage

```
$ curl http://127.0.0.1:8888/v1/chain/get_account -X POST -d '{"account_name":"inita"}'
```

Example get_account Result

```
{
  "name": "inita",
  "eos_balance": "999998.9574 EOS",
  "staked_balance": "0.0000 EOS",
  "unstaking_balance": "0.0000 EOS",
  "last_unstaking_time": "2106-02-07T06:28:15",
  "permissions": [
    {
      "name": "active",
      "parent": "owner",
      "required_auth": {
        "threshold": 1,
        "keys": [
          {
            "key": "EOS6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV",
            "weight": 1
          }
        ]
      },
      "accounts": [ ]
    },
    {
      "name": "owner",
      "parent": "owner",
      "required_auth": {
        "threshold": 1,
        "keys": [
          {
            "key": "EOS6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV",
            "weight": 1
          }
        ]
      },
      "accounts": [ ]
    }
  ]
}
```

get_code

Fetch smart contract code.

Example get_code Usage

```
$ curl http://127.0.0.1:8888/v1/chain/get_code -X POST -d '{"account_name":"currency"}'
```

Example get_code Result

```
{
  "name": "currency",
  "code_hash": "a1c8c84b4700c09c8edb83522237439e33cf011a4d7ace51075998bd002e04c9",
  "wasm": "(module\\n (type $0 (func (param i64 i64 i32) (result i32)))\\n ...truncated",
  "abi": {
    "types": [
      {
        "new_type_name": "account_name",
        "type": "name"
      }
    ],
    "structs": [
      {
        "name": "transfer",
        "base": "",
        "fields": [
          { "name": "from", "type": "account_name" },
          { "name": "to", "type": "account_name" },
          { "name": "quantity", "type": "uint64" }
        ]
      },
      {
        "name": "account",
        "base": "",
        "fields": [
          { "name": "key", "type": "name" },
          { "name": "balance", "type": "uint64" }
        ]
      }
    ],
    "actions": [
      {
        "name": "transfer",
        "type": "transfer"
      }
    ],
    "tables": [
      {
        "name": "account",
        "type": "account",
        "index_type": "i64",
        "key_names": [ "key" ],
        "key_types": [ "name" ]
      }
    ]
  }
```

```
}
}
```

get_table_rows

Fetch smart contract data from an account.

get_table_rows Usage

```
$ curl http://127.0.0.1:8888/v1/chain/get_table_rows -X POST -d '{"scope":"inita", "code":"currency", "table":"account", "json": true}'
$ curl http://127.0.0.1:8888/v1/chain/get_table_rows -X POST -d '{"scope":"inita", "code":"currency", "table":"account", "json": true, "lower_bound":0, "upper_bound":-1, "limit":10}'
```

Example get_table_rows Result

```
{
  "rows": [
    {
      "account": "account",
      "balance": 1000
    }
  ],
  "more": false
}
```

abi_json_to_bin

Serialize json to binary hex. The resulting binary hex is usually used for the data field in [push_transaction](#).

Example abi_json_to_bin Usage

```
$ curl http://127.0.0.1:8888/v1/chain/abi_json_to_bin -X POST -d '{"code":"currency", "action":"transfer", "args":{"from":"initb", "to":"initc", "quantity":1000}}'
```

Example abi_json_to_bin Result

```
{
  "binargs": "000000008093dd74000000000094dd74e803000000000000",
  "required_scope": [],
  "required_auth": []
}
```

abi_bin_to_json

Serialize back binary hex to json.

Example abi_bin_to_json Usage

```
$ curl http://127.0.0.1:8888/v1/chain/abi_bin_to_json -X POST -d '{"code":"currency", "action":"transfer", "binargs":"000000008093dd7400000000000094dd74e803000000000000"}'
```

Example abi_bin_to_json Result

```
{
  "args": {
    "from": "initb",
    "to": "initc",
    "quantity": 1000
  },
  "required_scope": [],
  "required_auth": []
}
```

push_transaction

This method expects a transaction in JSON format and will attempt to apply it to the blockchain,

Success Response

On success it will return HTTP 200 and the transaction ID.

```
{
  'transaction_id' : "..."
}
```

Just because the transaction is pushed locally does not mean that the transaction has been incorporated into a block.

Error Response

If an error occurs it will return either HTTP 400 (Invalid arguments) or 500 (Internal Server Error)

```
HTTP/1.1 500 Internal Server Error
Content-Length: 1466
...error message...
```

Example push_transaction Usage

This example assumes a transfer operation. The ref_block_num and ref_block_prefix here are provided as a result of /v1/chain/get_block of the last irreversible_block. You also need to use /v1/wallet/sign_transaction to get the right signature.

```
curl http://localhost:8888/v1/chain/push_transaction -X POST -d '{
  "ref_block_num":100,"ref_block_prefix":"137469861","expiration":"2017-09-25T06:28:49","scope":["initb","initc"],"actions":
  [{"code":"currency","type":"transfer","recipients":["initb","initc"],"authorization":
  [{"account":"initb","permission":"active"}],"data":"000000000041934b00000000008041934be803000000000000"}],"signatures":
  [],"authorizations":[]}'
```

push_transactions

This method push multiple transactions at once.

Example push_transactions Usage

```
curl http://localhost:8888/v1/chain/push_transaction -X POST -d
'[{ "ref_block_num": "101", "ref_block_prefix": "4159312339", "expiration": "2017-09-25T06:28:49", "scope":
[{"initb", "initc"}, {"actions": [{"code": "currency", "type": "transfer", "recipients": [{"initb", "initc"}], "authorization":
[{"account": "initb", "permission": "active"}], "data": "000000000041934b00000000008041934be803000000000000"}, {"signatures":
[{"authorizations": []}], {"ref_block_num": "101", "ref_block_prefix": "4159312339", "expiration": "2017-09-25T06:28:49", "scope":
[{"inita", "initc"}, {"actions": [{"code": "currency", "type": "transfer", "recipients": [{"inita", "initc"}], "authorization":
[{"account": "inita", "permission": "active"}], "data": "00000000008040934b00000000008041934be803000000000000"}, {"signatures":
[{"authorizations": []}]}'
```

get_required_keys

Get required keys to sign a transaction from list of your keys.

Example get_required_keys Usage

```
curl http://localhost:8888/v1/chain/get_required_keys -X POST -d '{"transaction":
{ "ref_block_num": "100", "ref_block_prefix": "137469861", "expiration": "2017-09-25T06:28:49", "scope": [{"initb", "initc"}], "actions":
[{"code": "currency", "type": "transfer", "recipients": [{"initb", "initc"}], "authorization":
[{"account": "initb", "permission": "active"}], "data": "000000000041934b00000000008041934be803000000000000"}, {"signatures":
[{"authorizations": []}], "available_keys": ["E054toFS3YXEQckuwlqDLrtHim86Gz9u3hBdcBw5KNPZcursVHq", "E057d9A3uLe6As66jzN8j44TXJUqj
```

Example get_required_keys Result

```
{
  "required_keys": [
    "E056MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV"
  ]
}
```

Wallet API

wallet_create

Create a new wallet with the given name

Example wallet_create Usage

```
$ curl http://localhost:8889/v1/wallet/create -X POST -d '{"default"'
```

Example wallet_create Result

This command will return the password that can be used to unlock the wallet in the future

```
PW5KFWYKqvt63d4iNvedfDEPVZL227D3RQ1zpVFzuUwhMAJmRAYyX
```

wallet_open

Open an existing wallet of the given name

Example wallet_open Usage

```
$ curl http://localhost:8889/v1/wallet/open -X POST -d '{"default"'
```

Example wallet_open Result

```
{}
```

wallet_lock

Lock a wallet of the given name

Example wallet_lock Usage

```
$ curl http://localhost:8889/v1/wallet/lock -X POST -d '{"default"'
```

Example wallet_lock Result

```
{}
```

wallet_lock_all

Lock all wallets

Example wallet_lock_all Usage

```
$ curl http://localhost:8889/v1/wallet/lock_all
```

Example wallet_lock_all Result

```
{}
```

wallet_unlock

Unlock a wallet with the given name and password

Example wallet_unlock Usage

```
$ curl http://localhost:8889/v1/wallet/unlock -X POST -d '{"default", "Pw5KFWYKqvt63d4iNvedfDEPVZL227D3RQ1zpVFzuUwhMAJmRAYyX"}'
```

Example wallet_unlock Result

```
{}
```

wallet_import_key

Import a private key to the wallet of the given name

Example wallet_import_key Usage

```
$ curl http://localhost:8889/v1/wallet/import_key -X POST -d '{"default", "5KQwrPbwdL6PhXujxW37FSSQZ1JiwsST4cqQzDeyXtP79zkvFD3"}'
```

Example wallet_import_key Result

```
{}
```

wallet_list

List all wallets

Example wallet_list Usage

```
$ curl http://localhost:8889/v1/wallet/list_wallets
```

Example wallet_list Result

```
["default *"]
```

wallet_list_keys

List all key pairs across all wallets

Example wallet_list_keys Usage

```
$ curl http://localhost:8889/v1/wallet/list_keys
```

Example wallet_list_keys Result

```
[["E0S6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV", "5KQwrPbwdL6PhXujxW37FSSQZ1JiwsST4cqQzDeyXtP79zkvFD3"]]
```

wallet_get_public_keys

List all public keys across all wallets

Example wallet_get_public_keys Usage

```
$ curl http://localhost:8889/v1/wallet/get_public_keys
```

Example wallet_get_public_keys Result

```
["E0S6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV"]
```

wallet_set_timeout

Set wallet auto lock timeout (in seconds)

Example wallet_set_timeout Usage

```
$ curl http://localhost:8889/v1/wallet/set_timeout -X POST -d '10'
```

Example wallet_set_timeout Result

```
{}
```

wallet_sign_trx

Sign transaction given an array of transaction, require public keys, and chain id

Example wallet_sign_trx Usage

```
$ curl http://localhost:8889/v1/wallet/sign_transaction -X POST -d '{
  "ref_block_num": 21453,
  "ref_block_prefix": 3165644999,
  "expiration": "2017-12-08T10:28:49",
  "scope": [
    {
      "initb": "initb",
      "initc": "initc",
      "read_scope": [
        {
          "code": "currency",
          "type": "transfer",
          "authorization": [
            {
              "account": "initb",
              "permission": "active"
            }
          ],
          "data": "000000008093dd740000000000094dd74e803000000000000"
        }
      ],
      "signatures": [
        {
          "signature": "E0S6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV"
        }
      ]
    }
  ]
}'
```

Example wallet_sign_trx Result

```
{
  "ref_block_num": 21453,
  "ref_block_prefix": 3165644999,
  "expiration": "2017-12-08T10:28:49",
  "scope": [
    {
      "initb": "initb",
      "initc": "initc",
      "read_scope": [
        {
          "code": "currency",
          "type": "transfer",
          "authorization": [
            {
              "account": "initb",
              "permission": "active"
            }
          ],
          "data": "000000008093dd740000000000094dd74e803000000000000"
        }
      ],
      "signatures": [
        {
          "signature": "E0S6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV"
        }
      ]
    }
  ]
}
```

```
{
  "scope": [
    "initb",
    "initc"
  ],
  "read_scope": [],
  "messages": [
    {
      "code": "currency",
      "type": "transfer",
      "authorization": [
        {
          "account": "initb",
          "permission": "active"
        }
      ]
    },
    {
      "data": "000000008093dd740000000000094dd74e803000000000000"
    }
  ],
  "signatures": [
    "1f393cc5ce6a6951fb53b11812345bcf14ffd978b07be386fd639eaf440bca7dca16b14833ec661ca0703d15e55a2a599a36d55ce78c4539433f6ce8bcee0158c3'"
  ]
}
```