账户操作

cli_wallet 是运行在本地的钱包，与 full node 相连，通过 wallet API 可很容易实现与链的交互；在需要签名的操作中，需要提前导入私钥。

1. 运行 cli_wallet 钱包

```
./cli_wallet -r 0.0.0.0:&{prot1} -s ws://&{server}:&{port} --chain-id &{chain_id} -w empty-wallet.json
```

`注：port1 为 cli_wallet 监听端口，server, port 为全节点 IP 和 port；chain_id 为链的 ID；empty-wallet.json 为空的账户 json 文件，其中 chain_id 必须`

2. websocket 连接

连接 wallet 并解锁:

```
from websocket import create_connection
import json

ws = create_connection("ws://127.0.0.1:&{port1}")
req = {"id":2, "method":"call", "params":[0, "unlock", ["123456"]]}
ws.send(json.dumps(req, sort_keys=True))
```

`cli_wallet 默认解锁密码为 123456`

3. 导入私钥

- 代码：

```
req = {"id":3, "method":"call", "params":[0, "import_key", ["cybex-dextest", "5HyqE3Ru8xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxComE67pPxnjQ"]]}
ws.send(json.dumps(req, sort_keys=True))
print("return:\n" +ws.recv())
```

- 参数：

```
account_name_or_id: 需要签名的账户
wif_key: 该账户的活跃私钥
```

4. 创建账户

- 代码：

```
req = {"id":4, "method":"call", "params":[0, "create_account_with_brain_key", ["testkeyword", "create-acc-cli-test", "1.2.157", "1.2.157", True]]}
ws.send(json.dumps(req, sort_keys=True))
print("return:\n" +ws.recv())
```

- create_account_with_brain_key 参数:

```
brain_key: 根据密码生成私钥
account_name: 新创建的账户名
registrar_account: 注册人，该账户是必须是终身会员
referrer_account: 引荐人，该账户必须是终身会员
broadcast: bool 类型，true 该操作会在链上进行广播
```

- 返回：

```
{
    "id": 4,
    "jsonrpc": "2.0",
    "result": {
        "ref_block_num": 54796,
        "ref_block_prefix": 173327528,
        "expiration": "2018-08-02T06:34:05",
        "operations": [
            [5, {
                "fee": {
                    "amount": 515,
                    "asset_id": "1.3.0"
                },
                "registrar": "1.2.157",
                "referrer": "1.2.157",
                "referrer_percent": 0,
                "name": "create-acc-cli-test",
                "owner": {
                    "weight_threshold": 1,
                    "account_auths": [],
                    "key_auths": [
                        ["CYB6JL6METyT8552s7ZttrgdvqqzRRRdQNupkPHhEBBnGBwzo7Emp", 1]
                    ],
                    "address_auths": []
                },
                "active": {
                    "weight_threshold": 1,
                    "account_auths": [],
                    "key_auths": [
                        ["CYB69tJUZqg6mcvFGVgeABk3nj8w1HTzCuGGEUha2j9bqDshFNdzL", 1]
                    ],
                    "address_auths": []
                },
                "options": {
                    "memo_key": "CYB5V7eVxsH9deU6YEdcbNY5udo2SKFWcPsZvfgXUjDd7cDZ4QUBw",
                    "voting_account": "1.2.5",
                    "num_witness": 0,
                    "num_committee": 0,
                    "votes": [],
                    "extensions": []
                },
                "extensions": {}
            }]
        ],
        "extensions": [],
        "signatures": ["201e6769d4e2b1eafe343cf4f847d11f378dbc5b27e1f705ab4a17b4b53450904b6c7ccd56c49b45fdf7610c4c0bc617e8eb3bfa85be075
ba9d19a60da90a15b84"]
    }
}
```

5. 获取账户操作历史

- 代码:

```
req = {"id":5, "method":"call", "params":[0, "get_account_history", ["cybex-dextest", 2]]}
ws.send(json.dumps(req, sort_keys=True))
print("return:\n" +ws.recv())
```

- get_account_history 参数:

```
account_name: 要查询的账户名
limit: 返回操作个数
```

- 返回:

```
{
    "id": 5,
    "jsonrpc": "2.0",
    "result": [{
        "memo": "another transaction",
        "description": "Transfer 10 CYB from hxy-dextest to cybex-dextest -- Memo: another transaction   (Fee: 0.02105 CYB)",
        "op": {
            "id": "1.11.51717",
            "op": [0, {
                "fee": {
                    "amount": 2105,
                    "asset_id": "1.3.0"
                },
                "from": "1.2.157",
                "to": "1.2.2766",
                "amount": {
                    "amount": 1000000,
                    "asset_id": "1.3.0"
                },
                "memo": {
                    "from": "CYB6H7LSpTTyFo5AiWAebeE3TdsAf2KvNiC3tTPkWMkkLb1aVm3ad",
                    "to": "CYB5ask3B3o1roG6irZLkJzGfBWe8UmAuXsCPf87chtqQ2FMKAABJ",
                    "nonce": "11242517862925630434",
                    "message": "16ee8c50374e3cf8b197f959d8e99723d0a5b50dc7bee09041afa4f42235f06a"
                },
                "extensions": []
            }],
            "result": [0, {}],
            "block_num": 1627879,
            "trx_in_block": 0,
            "op_in_trx": 0,
            "virtual_op": 3487
        }
    }, {
        "memo": "for milk",
        "description": "Transfer 3 CYB from cybex-dextest to create-acc-cli-test -- Memo: for milk   (Fee: 0.02089 CYB)",
        "op": {
            "id": "1.11.51715",
            "op": [0, {
                "fee": {
                    "amount": 2089,
                    "asset_id": "1.3.0"
                },
                "from": "1.2.2766",
                "to": "1.2.2781",
                "amount": {
                    "amount": 300000,
                    "asset_id": "1.3.0"
                },
                "memo": {
                    "from": "CYB5ask3B3o1roG6irZLkJzGfBWe8UmAuXsCPf87chtqQ2FMKAABJ",
                    "to": "CYB5V7eVxsH9deU6YEdcbNY5udo2SKFWcPsZvfgXUjDd7cDZ4QUBw",
                    "nonce": "3532355300571614764",
                    "message": "d16eb2897407f1fb3503a8860347e165"
                },
                "extensions": []
            }],
            "result": [0, {}],
            "block_num": 1627832,
            "trx_in_block": 0,
            "op_in_trx": 0,
            "virtual_op": 3481
        }
    }]
}
```

6. 转账

- 代码:

```
req = {"id":6, "method":"call", "params":[0, "transfer", ["cybex-dextest", "create-acc-cli-test", 7, "CYB", "transaction", True]]}
ws.send(json.dumps(req, sort_keys=True))
print("return:\n" +ws.recv())
```

- transfer 参数:

```
from：转出账户
to：转入账户
amount：资产数量
asset_symbol：资产代号
memo：操作说明
broadcast：bool 类型，true 该操作会在链上广播
```

- 返回：

```
{
    "id": 6,
    "jsonrpc": "2.0",
    "result": {
        "ref_block_num": 55093,
        "ref_block_prefix": 3928671869,
        "expiration": "2018-08-02T07:05:40",
        "operations": [
            [0, {
                "fee": {
                    "amount": 2089,
                    "asset_id": "1.3.0"
                },
                "from": "1.2.2766",
                "to": "1.2.2781",
                "amount": {
                    "amount": 700000,
                    "asset_id": "1.3.0"
                },
                "memo": {
                    "from": "CYB5ask3B3o1roG6irZLkJzGfBWe8UmAuXsCPf87chtqQ2FMKAABJ",
                    "to": "CYB5V7eVxsH9deU6YEdcbNY5udo2SKFWcPsZvfgXUjDd7cDZ4QUBw",
                    "nonce": "145648381586179527",
                    "message": "ff3bf015e55249077718263a88940ddb"
                },
                "extensions": []
            }]
        ],
        "extensions": [],
        "signatures": ["2018970c9eed7d5f933657193bf123dede36b356d69dfb099551ca594cc4d426d03ec5f7a94722390b68d907b40b8bca5a4e6f6113130a4
2e9a9864fb4920d2709"]
    }
}
```

`转出账户需要签名，执行转账操作前需要导入转出账户的私钥`

`* 所有的调用都是 json 格式，返回的也是 json 格式`