Wanchain 3.0 Alpha Testing

CLI Wallet - Bitcoin Cross-chain Commands (testnet)

Install environment

1. Fetch code

```
$ git clone https://github.com/wanchain/wanchain-crosschain-walletcli
$ cd wanchain-crosschain-walletcli
$ git checkout -t origin/wanchain3.0
```

2. Install dependencies

```
$ npm install
```

3. Start CLI

```
$ node cli.js
```

4. Get the help text

```
> help
```

General Bitcoin commands

createBtcAddress - Create a new Bitcoin address. It will prompt to enter password for the new address

listBtcAddress - Get list of Bitcoin addresses

getBtcBalance - Get the balance of all Bitcoin addresses

sendBtcToAddress - Send Bitcoin to another BTC address (normal transaction)

General Wanchain commands

listWbtcBalance - List the WBTC balance for all WAN addresses

listWanBalance - List WAN balance for all WAN addresses

Cross-chain Transaction Commands

listTransactions - List all transactions

lockBtc - Lock BTC for cross-chain transaction (BTC to WBTC)

redeemBtc - Finalize a BTC to WBTC transaction

revokeBtc - Cancel BTC to WBTC transaction once HTLC timelock is expired

lockWbtc - Lock WBTC for cross-chain transaction (WBTC to BTC)

redeemWbtc - Finalize a WBTC to BTC transaction

revokeWbtc - Cancel WBTC to BTC transaction once HTLC timelock is expired

listStoremanGroups - List BTC address and WAN address of the Storeman group

Example 1 – Send BTC to a Wanchain Address

First, create a new BTC address

```
> createBtcAddress
```

You will need BTC in your CLI wallet to be able to transact, so before continuing send some testnet BTC to the address from your preferred wallet. Once bitcoin is sent to the wallet address, check that the value has been received.

```
> getBtcBalance
```

Also, make sure that you have a positive balance in your WAN account as well.

```
> listWanBalance
```

If you do not have a positive WAN balance, send some testnet WAN coins to the CLI wallet address.

Now, with balances in both the WAN and BTC addresses, initiate a new BTC to WBTC transaction by locking some bitcoin.

```
> lockBtc
```

Answer the prompts, which ask which storeman group you would like to use, the WAN address you would like to send to, and the amount that you would like to send.

Once completed, you can list the transactions to see that it registered.

```
> listTransactions
```

After a few minutes when the transaction that you just created changes to the status “waitingX” (which means that the BTC locking transaction has confirmed), you can then proceed to finalize the transaction.

```
> redeemBtc
```

Now check your WBTC balance. You should see that you now have WBTC.

```
> listWbtcBalance
```

If instead you waited too long and the status changes to “waitingRevoke” (which means the HTLC timelock has expired), you can then proceed to reclaim your BTC.

```
> revokeBTC
```

Example 2 – Send WBTC to a Bitcoin Address

Note: you must have positive balances of both WAN and WBTC.

Initiate a new BTC to WBTC transaction by locking some WBTC.

```
> lockWbtc
```

Answer the prompts, which ask which storeman group you would like to use, the WAN address you would like to send from, the BTC address you would like to send to, and the amount that you would like to send.

Once completed, you can list the transactions to see that it registered.

```
> listTransactions
```

When the transaction that you just created changes to the status “waitingX” (which means that the WBTC locking transaction has confirmed), you can then proceed to finalize the transaction.

```
> redeemWbtc
```

Now check your BTC balance. After a few minutes, once the BTC transaction confirms, you should see that you received the bitcoin.

```
> getBtcBalance
```

If instead you waited too long and the status changes to “waitingRevoke” (which means the HTLC timelock has expired), you can then proceed to reclaim your WBTC.

```
> revokeWbTC
```