SpaceChain OS 环境下的 Qtum 测试使用方法

Qtum 命令行工具使用方法

类别	内容
关键词	SpaceChain OS　Qtum　量子链
摘　要	Qtum 命令行工具使用方法

目 录

1. qtumd 命令

1.1 设置选项

命令	描述
-?	打印帮助信息并退出
-version	打印版本信息并退出
-alertnotify=<cmd>	Execute command when a relevant alert is received or we see a really long fork (%s in cmd is replaced by message)
-blocknotify=<cmd>	当最好的货币块改变时执行命令（命令中的 %s 会被替换为货币块哈希值）
-assumevalid=<hex>	If this block is in the chain assume that it and its ancestors are valid and potentially skip their script verification (0 to verify all, default: 00000000000000000013176bf8d7dfeab4e1 db31dc93bc311b436e82ab226b90, testnet: 00000000000128796ee387cf110ccb9 d2f36cffaf7f73079c995377c65ac0dcc)
-conf=<file>	指定配置文件（默认：qtum.conf）
-daemon	以守护进程方式进行运行
-datadir=<dir>	指定数据目录
-dbcache=<n>	设置数据库缓存大小（单位：MB）（有效值 4 至 16384，默认：450）
-loadblock=<file>	在启动阶段，从外部的 blk000**.dat 文件导入区块
-maxorphantx=<n>	在内存中最多保留未连接的交易数量（默认：100）
-maxmempool=<n>	交易池最大存储容量（单位：MB）（默认：300）
-mempoolexpiry=<n>	交易池保存的最大交易时长（单位：小时）（默认：336）
-blockreconstructionextratxn=<n>	Extra transactions to keep in memory for compact block reconstructions (default: 100)
-par=<n>	Set the number of script verification threads (-2 to 16, 0 = auto, <0 = leave that many cores free, default: 0)
-pid=<file>	指定 pid 文件（默认：qtumd.pid）
-prune=<n>	Reduce storage requirements by enabling pruning (deleting) of old blocks. This allows the pruneblockchain RPC to be called to delete specific blocks, and enables automatic pruning of old blocks if a target size in MiB is provided. This mode is incompatible with -txindex and -rescan. Warning: Reverting this setting requires re-downloading the entire blockchain. (default: 0 = disable pruning blocks, 1 = allow manual pruning via RPC, >550 =

	automatically prune block files to stay under the specified target size in MiB)
-record-log-opcodes	Logs all EVM LOG opcode operations to the file vmExecLogs.json
-reindex-chainstate	Rebuild chain state from the currently indexed blocks
-reindex	从磁盘上的 blk*.dat 文件中重新构建链状态和区块索引
-sysperms	Create new files with system default permissions, instead of umask 077 (only effective with disabled wallet functionality)
-txindex	Maintain a full transaction index, used by the getrawtransaction rpc call (default: 0)
-logevents	Maintain a full EVM log index, used by searchlogs and gettransactionreceipt rpc calls (default: 0)

1.2 连接选项

命令	描述
-addnode=<ip>	添加一个节点以供连接，并尝试保持与该节点的连接
-banscore=<n>	与行为异常节点断开连接的临界值（默认：100）
-bantime=<n>	重新允许与异常节点连接所间隔的秒数（默认：86400）
-bind=<addr>	绑定指定的 IP 地址和端口，并且进行监听。对于 IPv6 使用[host]:port 格式
-connect=<ip>	仅连接到指定的节点；-noconnect 或-connect=0 禁用自动连接
-discover	Discover own IP addresses (default: 1 when listening and no -externalip or -proxy)
-dns	允许查询 DNS 并连接，通过-addnode, -seednode 和 -connect（默认：: 1）
-dnsseed	使用 DNS 查找节点（默认：1，忽略 -connect/-noconnect）
-externalip=<ip>	Specify your own public address
-forcednsseed	Always query for peer addresses via DNS lookup (default: 0)
-listen	Accept connections from outside (default: 1 if no -proxy or -connect/-noconnect)
-listenonion	Automatically create Tor hidden service (default: 1)
-maxconnections=<n>	最大允许连接数（默认：125）
-maxreceivebuffer=<n>	最大每次连接的接收缓存，×1000 字节（默认：

	5000）
-maxsendbuffer=<n>	最大每次连接的发送缓存，×1000 字节（默认：1000）
-maxtimeadjustment	Maximum allowed median peer time offset adjustment. Local perspective of time may be influenced by peers forward or backward by this amount. (default: 4200 seconds)
-onion=<ip:port>	Use separate SOCKS5 proxy to reach peers via Tor hidden services (default: -proxy)
-onlynet=<net>	设置只允许连接的网络类型（ipv4，ipv6 或 onion）
-permitbaremultisig	Relay non-P2SH multisig (default: 1)
-peerbloomfilters	Support filtering of blocks and transaction with bloom filters (default: 1)
-port=<port>	监听使用的端口（默认为：8333，testnet 时为：18333）
-proxy=<ip:port>	通过 SOCKS5 代理连接
-proxyrandomize	Randomize credentials for every proxy connection. This enables Tor stream isolation (default: 1)
-rpcserialversion	Sets the serialization of raw transaction or block hex returned in non-verbose mode, non-segwit(0) or segwit(1) (default: 1)
-seednode=<ip>	Connect to a node to retrieve peer addresses, and disconnect
-timeout=<n>	设置连接超时（单位：毫秒）（最小值：1，默认：5000）
-torcontrol=<ip>:<port>	Tor control port to use if onion listening enabled (default: 127.0.0.1:9051)
-torpassword=<pass>	Tor control port password (default: empty)
-dgpstorage	Receiving data from DGP via storage (default: -dgpevm)
-dgpevm	Receiving data from DGP via a contract call (default: -dgpevm)
-whitebind=<addr>	Bind to given address and whitelist peers connecting to it. Use [host]:port notation for IPv6
-whitelist=<IP address or network>	Whitelist peers connecting from the given IP address (e.g. 1.2.3.4) or CIDR notated network (e.g. 1.2.3.0/24). Can be specified multiple times. Whitelisted peers cannot be DoS banned and their transactions are always relayed, even if they are already in the mempool, useful e.g. for a gateway
-whitelistrelay	Accept relayed transactions received from whitelisted peers even when not relaying transactions (default: 1)
-whitelistforcerelay	Force relay of transactions from whitelisted peers even if they violate local relay policy (default: 1)

-maxuploadtarget=<n>	Tries to keep outbound traffic under the given target (in MiB per 24h), 0 = no limit (default: 0)

1.3 钱包选项

命令	描述
-disablewallet	不加载钱包，并且禁用钱包的 RPC 调用
-keypool=<n>	设置密匙池的尺寸 （默认：100）
-fallbackfee=<amt>	A fee rate (in BTC/kB) that will be used when fee estimation has insufficient data (default: 0.0002)
-mintxfee=<amt>	Fees (in BTC/kB) smaller than this are considered zero fee for transaction creation (default: 0.00001)
-paytxfee=<amt>	发送的交易每 KB 字节的手续费（单位：BTC/kB）（默认：0.00）
-rescan	在启动阶段，重新扫描区块链中丢失的钱包交易
-salvagewallet	Attempt to recover private keys from a corrupt wallet on startup
-spendzeroconfchange	Spend unconfirmed change when sending transactions (default: 1)
-txconfirmtarget=<n>	If paytxfee is not set, include enough fee so transactions begin confirmation on average within n blocks (default: 6)
-usehd	Use hierarchical deterministic key generation (HD) after BIP32. Only has effect during wallet creation/first start (default: 1)
-walletrbf	Send transactions with full-RBF opt-in enabled (default: 0)
-upgradewallet	在启动阶段升级钱包位最新格式
-wallet=<file>	指定钱包文件（在数据目录中）（默认： wallet.dat）
-walletbroadcast	使能钱包广播交易（默认：1）
-walletnotify=<cmd>	Execute command when a wallet transaction changes (%s in cmd is replaced by TxID)
-zapwallettxes=<mode>	Delete all wallet transactions and only recover those parts of the blockchain through -rescan on startup (1 = keep tx meta data e.g. account owner and payment request information, 2 = drop tx meta data)
-staking =<true/false>	Enables or disables staking (enabled by default)
-stakecache=<true/false>	Enables or disables the staking cache; significantly improves staking performance, but can use a lot of memory (enabled by default)
-rpcmaxgasprice	The max value (in satoshis) for gas price allowed through RPC (default:

	100)

1.4 调试和测试选项

命令	描述
-uacomment=<cmt>	Append comment to the user agent string
-debug=<category>	输出额外的调试信息
-help-debug	显示所有的调试选项（usage: --help -help-debug）
-logips	调试信息中包含 IP 地址（默认：0）
-logtimestamps	调试信息前添加时间戳（默认：1）
-minrelaytxfee=<amt>	Fees (in BTC/kB) smaller than this are considered zero fee for relaying, mining and transaction creation (default: 0.00001)
-maxtxfee=<amt>	Maximum total fees (in BTC) to use in a single wallet transaction or raw transaction; setting this too low may abort large transactions (default: 0.10)
-printtoconsole	发送跟踪/调试信息到控制台而不是 debug.log 文件
-shrinkdebugfile	Shrink debug.log file on client startup (default: 1 when no -debug)

1.5 链选择选项

命令	描述
-testnet	使用 test 链
-regtest	使用回归测试模式，使用了一个会被立刻独立处理的链。用于工具测试和应用开发

1.6 节点延迟选项

命令	描述
-bytespersigop	Equivalent bytes per sigop in transactions for relay and mining (default: 20)
-datacarrier	Relay and mine data carrier transactions (default: 1)
-datacarriersize	Maximum size of data in data carrier transactions we relay and mine (default: 83)
-mempoolreplacement	Enable transaction replacement in the memory pool (default: 1)

1.7 区块创建选项

命令	描述
-blockmaxweight=<n>	设置最大的 BIP141 块重量（默认：3000000）
-blockmaxsize=<n>	设置最大的块大小（单位：字节）（默认：750000）
-blockprioritysize=<n>	设置高优先级/低费用的交易大小（单位：字节）（默认：0）
-blockmintxfee=<amt>	Set lowest fee rate (in BTC/kB) for transactions to be included in block creation. (default: 0.00001)
-staker-min-tx-gas-price=<amt>	Any contract execution with a gas price below this will not be included in a block (defaults to the value specified by the DGP)
-staker-max-tx-gas-limit=<n>	Any contract execution with a gas limit over this amount will not be included in a block (defaults to soft block gas limit)
-staker-soft-block-gas-limit=<n>	After this amount of gas is surpassed in a block, no more contract executions will be added to the block (defaults to consensus-critical maximum block gas limit)

1.8 RPC 服务选项

命令	描述
-server	接受命令行和 JSON-RPC 命令
-rest	接收公共的 REST 请求（默认：0）
-rpcbind=<addr>	绑定到指定的地址监听 JSON-RPC 连接。对于 IPv6 使用[host]:port 的格式。此操作可以指定多次（默认：绑定到所有端口）
-rpccookiefile=<loc>	指定授权 cookie 文件的地址（默认：data dir）
-rpcuser=<user>	JSON-RPC 连接的用户名
-rpcpassword=<pw>	JSON-RPC 连接的密码
-rpcauth=<userpw>	JSON-RPC 连接使用的用户名和哈希密码。<userpw> 使用的格式：<USERNAME>:<SALT>$<HASH>
-rpcport=<port>	JSON-RPC 监听的端口（默认：8332，testnet 为：18332）
-rpcallowip=<ip>	允许指定的 IP 源地址进行 JSON-RPC 连接，有效的<ip>参数必须是单个的 ip（如 1.2.3.4），或带有子网掩码的单个 ip（如 1.2.3.4/255.255.255.0）或带有 CIDR 的 ip，如（1.2.3.4/24）
-rpcthreads=<n>	处理 RPC 调用服务的线程数量（默认：4）

2. qtum-cli 本地命令

2.1 设置选项

命令	描述
-?	打印帮助信息
-conf=<file>	指定配置文件（默认：qtum.conf）
-datadir=<dir>	指定数据路径

2.2 链选择选项

命令	描述
-testnet	使用测试链
-regtest	使用回归测试模式，使用了一个会被立刻独立处理的链。用于工具测试和应用开发
-named	以名称替代位置参数（默认：false）
-rpcconnect=<ip>	发送命令给指定 IP 上的节点（默认为：127.0.0.1）
-rpcport=<port>	指定 JSON-RPC 的目标端口（默认：3889，testnet 为：13889）
-rpcwait	等待 RPC 服务启动
-rpcuser=<user>	JSON-RPC 连接的用户名
-rpcpassword=<pw>	JSON-RPC 连接的密码
-rpcclienttimeout=<n>	HTTP 请求的超时（默认：900）
-stdin	从标准输入读取其它参数，每次以 EOF/Ctrl-D 结束为一行（敏感信息建议加密传输）

3. qtum-cli RPC 命令

3.1 区块命令

命令	描述
callcontract "address" "data" (address)	获取信息
getaccountinfo "address"	通过地址获取账户信息
getbestblockhash	返回最长链中的 best 块的哈希值
getblock "blockhash" (verbose)	
getblockcount	获取块数量
getblockhash height	
getblockheader "hash" (verbose)	
getchaintips	
getdifficulty	获取难度

getmempoolancestors txid (verbose)	
getmempooldescendants txid (verbose)	
getmempoolentry txid	
getmempoolinfo	
getrawmempool (verbose)	
getstorage "address"	
gettransactionreceipt "hash"	
gettxout "txid" n (include_mempool)	
gettxoutproof ["txid",...] (blockhash)	
gettxoutsetinfo	
listcontracts (start maxDisplay)	
preciousblock "blockhash"	
pruneblockchain	
searchlogs <fromBlock> <toBlock> (address) (topics)	
verifychain (checklevel nblocks)	
verifytxoutproof "proof"	
waitforlogs (fromBlock) (toBlock) (filter) (minconf)	

3.2 控制命令

命令	描述
getinfo	获取信息
getmemoryinfo	获取内存使用情况的信息
help ("command")	查看某命令使用的帮助信息
stop	停止 qtum server

3.3 生成块命令

命令	描述
generate nblocks (maxtries)	生成块
generatetoaddress nblocks address (maxtries)	

3.4 挖矿命令

命令	描述
getblocktemplate (TemplateRequest)	

getmininginfo	
getnetworkhashps (nblocks height)	
getstakinginfo	
getsubsidy [nTarget]	
prioritisetransaction <txid> <priority delta> <fee delta>	
submitblock "hexdata" ("jsonparametersobject")	

3.5 网络命令

命令	描述
addnode "node" "add|remove|onetry"	
clearbanned	
disconnectnode "node"	
getaddednodeinfo ("node")	
getconnectioncount	
getnettotals	
getnetworkinfo	
getpeerinfo	
listbanned	
ping	
setban "subnet" "add|remove" (bantime) (absolute)	
setnetworkactive true|false	

3.6 交易命令

命令	描述
createrawtransaction [{"txid":"id","vout":n},...] {"address":amount,"data":"hex",...} (locktime)	创建一次交易信息
decoderawtransaction "hexstring"	解密增加签名后的交易信息
sendrawtransaction "hexstring" (allowhighfees)	发送交易
signrawtransaction "hexstring" ([{"txid":"id","vout":n,"scriptPubKey":"hex","redeemScript":"hex"},...] ["privatekey1",...] sighashtype)	对交易信息进行签名
decodescript "hexstring"	
fromhexaddress "hexaddress"	
fundrawtransaction "hexstring"	

(options)	
gethexaddress "address"	
getrawtransaction "txid" (verbose)	

3.7 工具命令

命令	描述
createmultisig nrequired ["key",...]	创建一次交易信息
estimatefee nblocks	解密增加签名后的交易信息
estimatepriority nblocks	发送交易
estimatesmartpriority nblocks	对交易信息进行签名
estimatesmartfee nblocks	
signmessagewithprivkey "privkey" "message"	
validateaddress "address"	
verifymessage "address" "signature" "message"	

3.8 钱包命令

命令	描述
encryptwallet "passphrase"	加密钱包，“passphrase”为密码
walletpassphrase "passphrase" timeout	解密钱包，“passphrase”为密码，timeout 为多久钱包会再次被自动锁定的秒数数字（计时器）
walletpassphrasechange "oldpassphrase" "newpassphrase"	修改密码，“oldpassphrase”为原密码，“newpassphrase”为新密码
backupwallet "destination"	创建钱包的备份文件，“destination”为备份的文件名
importwallet "filename"	加载钱包的备份文件，“filename”为备份的文件名
dumpwallet "filename"	将钱包转变为人类可读的文件，“filename”为文件名
getnewaddress	比特币客户端维护了一个地址池，地址池的大小可以用 getinfo 命令 keypoolsize 参数获取。这些地址是自动生成的，可以被用作公开接收地址或零钱地址。使用 getnewaddress 命令可以获得其中的

	一个地址
getreceivedbyaddress “address” minconf	可以询问对应地址的客户端已经接收到的比特币数额，以及指定该数额要被加到余额中所需要的确认数。在从另一个钱包发送比特币数秒之后，另一个钱包将有反应。“address”为指定的地址，minconf 为设置的确认数
getaddressesbyaccount "account"	列出整个钱包的所有地址，“account”为帐户，可以为空字符
getbalance "account" minconf include_watchonly	显示所有经过至少 minconf 个确认的交易加和后的余额
listunspent (minconf maxconf ["addresses",...] [include_unsafe])	查看钱包未花费的金额、交易 ID 等信息
getnewaddress ("account")	从地址池中获取新地址
getwalletinfo	获取钱包信息
getaccount "address"	获取指定地址的账户信息
abandontransaction "txid"	
addmultisigaddress nrequired ["key",...] ("account")	
addwitnessaddress "address"	
bumpfee "txid" (options)	
createcontract "bytecode" (gaslimit gasprice "senderaddress" broadcast)	
dumpprivkey "address"	
getaccountaddress "account"	
getrawchangeaddress	
getreceivedbyaccount "account" (minconf)	
getreceivedbyaddress "address" (minconf)	
gettransaction "txid" (include_watchonly) (waitconf)	
getunconfirmedbalance	
importaddress "address" ("label" rescan p2sh)	
importmulti "requests" "options"	
importprivkey "qtum" ("label") (rescan)	
importprunedfunds importpubkey "pubkey" ("label" rescan)	
keypoolrefill (newsize)	
listaccounts (minconf	

include_watchonly)	
listaddressgroupings	
listlockunspent	
listreceivedbyaccount (minconf include_empty include_watchonly)	
listreceivedbyaddress (minconf include_empty include_watchonly)	
listsinceblock ("blockhash" target_confirmations include_watchonly)	
listtransactions ("account" count skip include_watchonly)	
lockunspent unlock ([{"txid":"txid","vout":n},...])	
move "fromaccount" "toaccount" amount (minconf "comment")	
removeprunedfunds "txid"	
reservebalance [<reserve> [amount]]	
sendfrom "fromaccount" "toaddress" amount (minconf "comment" "comment_to")	
sendmany "fromaccount" {"address":amount,...} (minconf "comment" ["address",...])	
sendmanywithdupes "fromaccount" {"address":amount,...} (minconf "comment" ["address",...])	
sendtoaddress "address" amount ("comment" "comment_to" subtractfeefromamount)	
sendtocontract "contractaddress" "data" (amount gaslimit gasprice senderaddress broadcast)	
setaccount "address" "account"	
settxfee amount	
signmessage "address" "message"	
walletlock	

4. 交易测试

4.1 进行一次本地交易

1. 创建测试币

需要创建满 500 个 block，在账户中会产生余额（也可以分多次创建，一次创建 500 个 block 耗时会更久）。

```
./qtum-cli generate -regtest 500
```

或者 （先设置/root/.qtum/qtum.conf）

```
./qtum-cli -rpcuser=test -rpcpassword=test -regtest generate 500
```

2. 检查账户余额

```
./qtum-cli getbalance
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest getbalance
1019999.99990000
```

```
./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest getbalance
```

3. 获取可用的 txid

```
./qtum-cli listunspent
```

```
[
  {
    "txid": "4e564cff32e204fec55305c20950400a595333f5d70d53d7c0b22a840e9ba5f5",
    "vout": 0,
    "address": "qPtnNtX36uhpSnqz59VYHZRuYoS3PwqSTq",
    "scriptPubKey": "2103d9b1f8714e6c3556c9d1836dea4faa8ed2a24be60bdf4f06fac03e4404b7ef6bac",
    "amount": 20000.00000000,
    "confirmations": 503,
    "spendable": true,
    "solvable": true
  },
  {
    "txid": "eb00e4d56c0d99d2fdc5871091f28e98f9a5225fa10553d35266ce6075f17ffb",
    "vout": 0,
    "address": "qPtnNtX36uhpSnqz59VYHZRuYoS3PwqSTq",
    "scriptPubKey": "2103d9b1f8714e6c3556c9d1836dea4faa8ed2a24be60bdf4f06fac03e4404b7ef6bac",
    "amount": 20000.00000000,
    "confirmations": 549,
    "spendable": true,
    "solvable": true
  }
```

4. 获取新地址

```
./qtum-cli getnewaddress
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest getnewaddress
qHxXXv5Kz1XqewPNGYTXN5jbYuyCipabqz
```

5. 创建交易

通过 createrawtransaction 命令创建一个交易，返回一串加密后的 16 进制字符串表示该交易的信息。

```
./qtum-cli createrawtransaction
"[{\"txid\":\"eb00e4d56c0d99d2fdc5871091f28e98f9a5225fa10553d35266ce6075f17f
fb\",\"vout\":0}]" "{\"qHxXXv5Kz1XqewPNGYTXN5jbYuyCipabqz\":19999.99990}"
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest createrawtransaction "[{\"txid\":\"eb00e4d56c
0d99d2fdc5871091f28e98f9a5225fa10553d35266ce6075f17ffb\",\"vout\":0}]" "{\"qHxXXv5Kz1XqewPNGYTXN5jbYuyCipabqz\":19999.9999}"
020000001fb7ff17560ce6652d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb000000000ffffffff01f0f849a9d10100001976a91404608f66645ea7
a90046641e9cc48c17590f775088ac00000000
```

6. 签名交易信息

通过 signrawtransaction 可以对已加密的 16 进制交易字符串进行签名，得到签名后的字符串。

```
./qtum-cli signrawtransaction
0200000001fb7ff17560ce6652d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb00
00000000ffffffff01f0f849a9d10100001976a91404608f66645ea7a90046641e9cc48c1759
0f775088ac00000000
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest signrawtransaction 0200000001fb7ff17560ce6652
d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb000000000ffffffff01f0f849a9d10100001976a91404608f66645ea7a90046641e9cc48c17590f7750
88ac00000000
{
  "hex": "0200000001fb7ff17560ce6652d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb000000004847304402207 78a131af1a595985eac9cc772d1
e4234115f904a0c96ad51a36c53c2eb6bc3702200f8f81f7f5785c33cc10083a915dc36a13959d1ceb00bc621a52cb3a7c12229401ffffffff01f0f849a9d1010000
1976a91404608f66645ea7a90046641e9cc48c17590f775088ac00000000",
  "complete": true
}
```

7. 解密交易信息

通过 decoderawtransaction 可以对已加密的 16 进制交易信息进行解密，获取交易信息详情。

```
./qtum-cli decoderawtransaction
0200000001fb7ff17560ce6652d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb00
000000484730440220778a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2e
b6bc3702200f8f81f7f5785c33cc10083a915dc36a13959d1ceb00bc621a52cb3a7c12229401
ffffffff01f0f849a9d10100001976a91404608f66645ea7a90046641e9cc48c17590f775088
ac00000000
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest decoderawtransaction 0200000001fb7ff17560ce66
52d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb000000004847304402207 78a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2eb6bc
3702200f8f81f7f5785c33cc10083a915dc36a13959d1ceb00bc621a52cb3a7c12229401ffffffff01f0f849a9d10100001976a91404608f66645ea7a90046641e9c
c48c17590f775088ac00000000
{
  "txid": "f56013c0c70ca9a68df97b12deced7f1b6b6f1b556f66edb7e7582cfe9ec64d8",
  "hash": "f56013c0c70ca9a68df97b12deced7f1b6b6f1b556f66edb7e7582cfe9ec64d8",
  "size": 157,
  "vsize": 157,
  "version": 2,
  "locktime": 0,
  "vin": [
    {
      "txid": "eb00e4d56c0d99d2fdc5871091f28e98f9a5225fa10553d35266ce6075f17ffb",
      "vout": 0,
      "scriptSig": {
        "asm": "30440220778a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2eb6bc3702200f8f81f7f5785c33cc10083a915dc36a13959d1c
eb00bc621a52cb3a7c122294[ALL]",
        "hex": "4730440220778a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2eb6bc3702200f8f81f7f5785c33cc10083a915dc36a13959d
1ceb00bc621a52cb3a7c12229401"
      },
      "sequence": 4294967295
    }
  ],
  "vout": [
    {
      "value": 19999.99990000,
      "n": 0,
      "scriptPubKey": {
        "asm": "OP_DUP OP_HASH160 04608f66645ea7a90046641e9cc48c17590f7750 OP_EQUALVERIFY OP_CHECKSIG",
        "hex": "76a91404608f66645ea7a90046641e9cc48c17590f775088ac",
        "reqSigs": 1,
        "type": "pubkeyhash",
        "addresses": [
          "qHxXXv5Kz1XqewPNGYTXN5jbYuyCipabqz"
        ]
      }
    }
  ]
}
```

8. 发送交易信息

```
./qtum-cli sendrawtransaction
0200000001fb7ff17560ce6652d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb00
000000484730440220778a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2e
b6bc3702200f8f81f7f5785c33cc10083a915dc36a13959d1ceb00bc621a52cb3a7c12229401
ffffffff01f0f849a9d10100001976a91404608f66645ea7a90046641e9cc48c17590f775088
ac00000000
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest sendrawtransaction 0200000001fb7ff17560ce6652
d35305a15f22a5f9988ef2911087c5fdd2990d6cd5e400eb000000004847304402207 78a131af1a595985eac9cc772d1e4234115f904a0c96ad51a36c53c2eb6bc37
02200f8f81f7f5785c33cc10083a915dc36a13959d1ceb00bc621a52cb3a7c1222940 1ffffffff01f0f849a9d1010001976a91404608f66645ea7a90046641e9cc4
8c17590f775088ac00000000
f56013c0c70ca9a68df97b12deced7f1b6b6f1b556f66edb7e7582cfe9ec64d8
```

9. 再次检查账户余额

```
./qtum-cli getbalance
```

```
[root@sylixos:/apps/qtum-cli]# ./qtum-cli -rpcuser=test -rpcpassword=test1234 -regtest getbalance
1019999.99980000
```

4.2 进行一次跨板级交易

1. 启动各自的 qtumd

```
./qtumd -regtest &
```

2. 获取接收方的地址

```
./qtum-cli getnewaddress
```

3. 进行交易

同4.1。

4. 创建块进行打包

```
./qtum-cli generate 1
```