CSED 490U Blockchain & Cryptocurrency

Assignment 3

Submitted by- Sajan Maharjan

POVIS id- thesajan@postech.ac.kr

Registration Number- 20182095

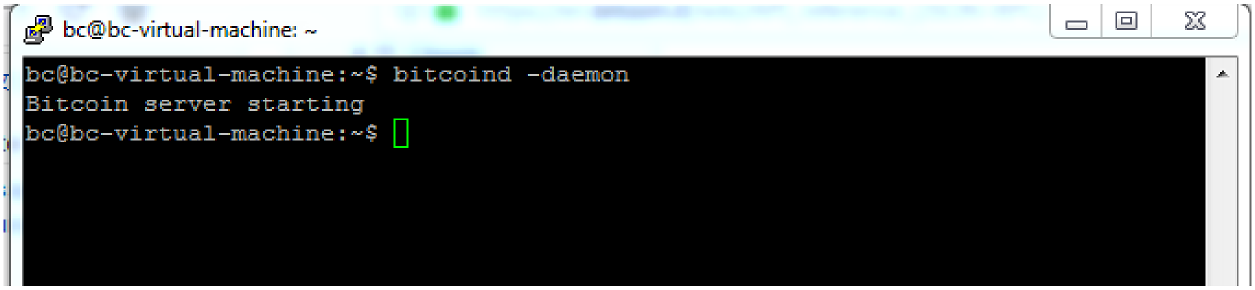
<Step 1> Install Bitcoin Core Client

Bitcoin PPA repository was added to the server, updated and bitcoind was installed using the commands-

```
sudo apt-add-repository ppa:bitcoin/bitcoin
sudo apt-get update
sudo apt-get install bitcoind
```

Starting the bitcoin server

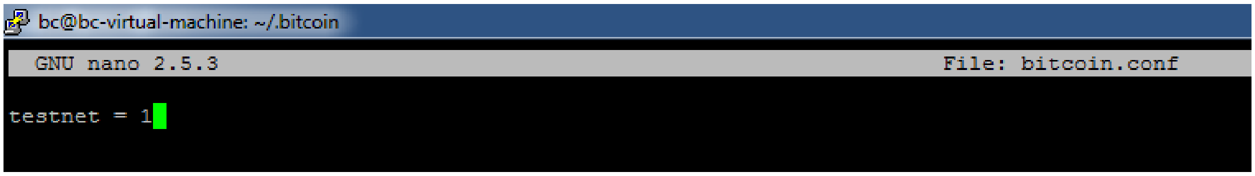
```
bitcoind -daemon
```



bc@bc-virtual-machine: ~

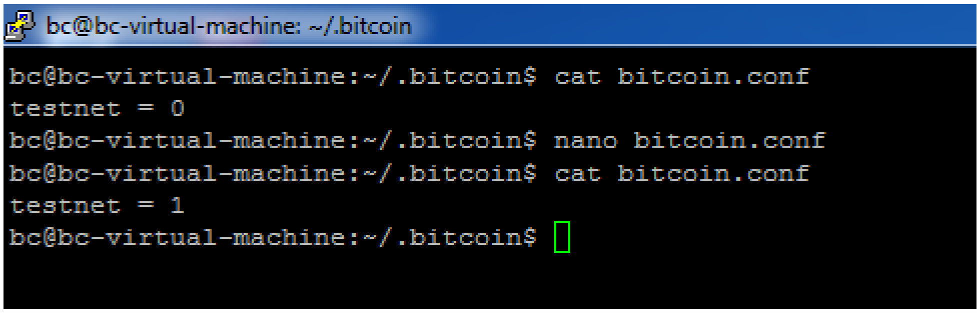
```
bc@bc-virtual-machine:~$ bitcoind -daemon
Bitcoin server starting
bc@bc-virtual-machine:~$
```

Editing the bitcoin.conf file to use the testnet server by setting the value testnet=1



bc@bc-virtual-machine: ~/.bitcoin

```
  GNU nano 2.5.3                                              File: bitcoin.conf

testnet = 1
```



bc@bc-virtual-machine: ~/.bitcoin

```
bc@bc-virtual-machine:~/.bitcoin$ cat bitcoin.conf
testnet = 0
bc@bc-virtual-machine:~/.bitcoin$ nano bitcoin.conf
bc@bc-virtual-machine:~/.bitcoin$ cat bitcoin.conf
testnet = 1
bc@bc-virtual-machine:~/.bitcoin$
```

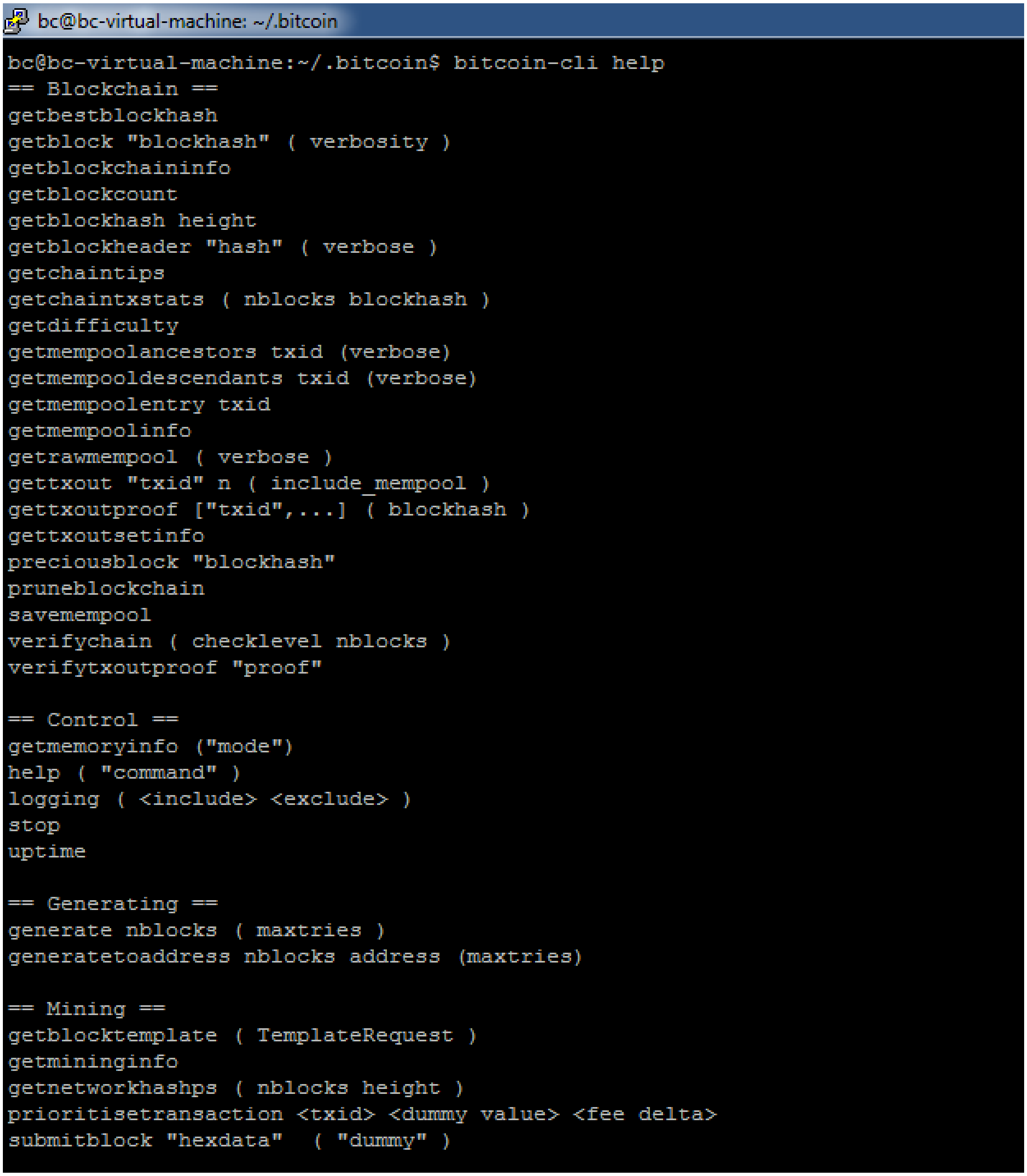
Now, it is possible to execute bitcoin command line without adding the –testnet option.

<Step. 2> Check out the JSON-RPC APIs

The command-

```
bitcoin-cli help
```

lists all the commands that can be executed in the bitcoin server using command line

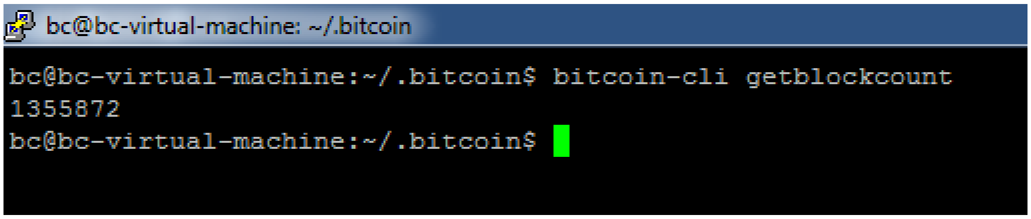


```
bc@bc-virtual-machine:~/.bitcoin$ bitcoin-cli help
== Blockchain ==
getbestblockhash
getblock "blockhash" ( verbosity )
getblockchaininfo
getblockcount
getblockhash height
getblockheader "hash" ( verbose )
getchaintips
getchaintxstats ( nblocks blockhash )
getdifficulty
getmempoolancestors txid (verbose)
getmempooldescendants txid (verbose)
getmempoolentry txid
getmempoolinfo
getrawmempool ( verbose )
gettxout "txid" n ( include_mempool )
gettxoutproof ["txid",...] ( blockhash )
gettxoutsetinfo
preciousblock "blockhash"
pruneblockchain
savemempool
verifychain ( checklevel nblocks )
verifytxoutproof "proof"

== Control ==
getmemoryinfo ("mode")
help ( "command" )
logging ( <include> <exclude> )
stop
uptime

== Generating ==
generate nblocks ( maxtries )
generatetoaddress nblocks address (maxtries)

== Mining ==
getblocktemplate ( TemplateRequest )
getmininginfo
getnetworkhashps ( nblocks height )
prioritisetransaction <txid> <dummy value> <fee delta>
submitblock "hexdata"  ( "dummy" )
```

For example, we can use the command

bitcoin-cli getblockcount

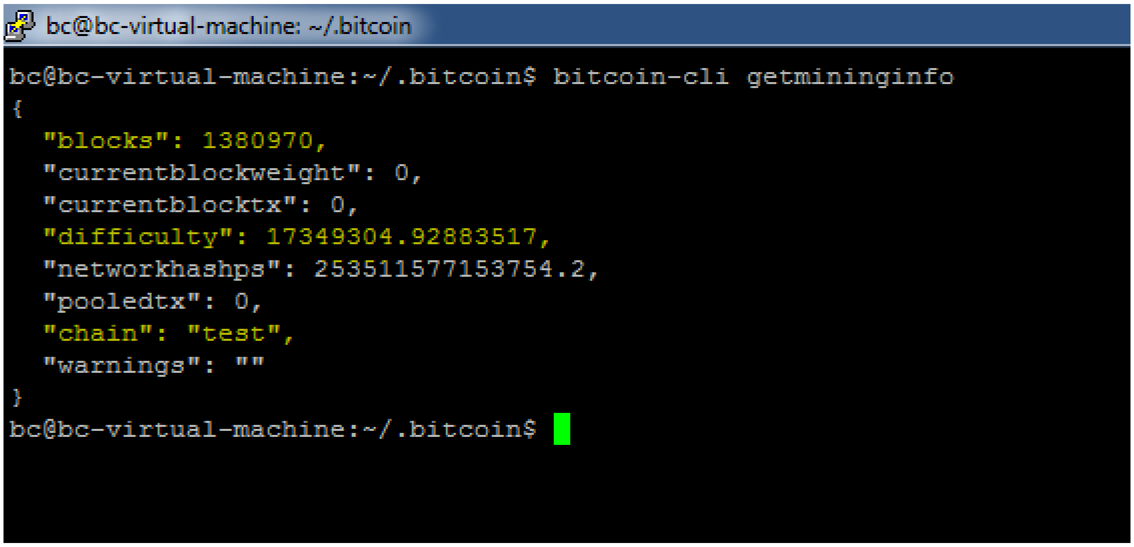
to list the total number of blocks in the bitcoin network (in our case the test network)



```
bc@bc-virtual-machine:~/.bitcoin$ bitcoin-cli getblockcount
1355872
bc@bc-virtual-machine:~/.bitcoin$
```

bitcoin-cli getmininginfo

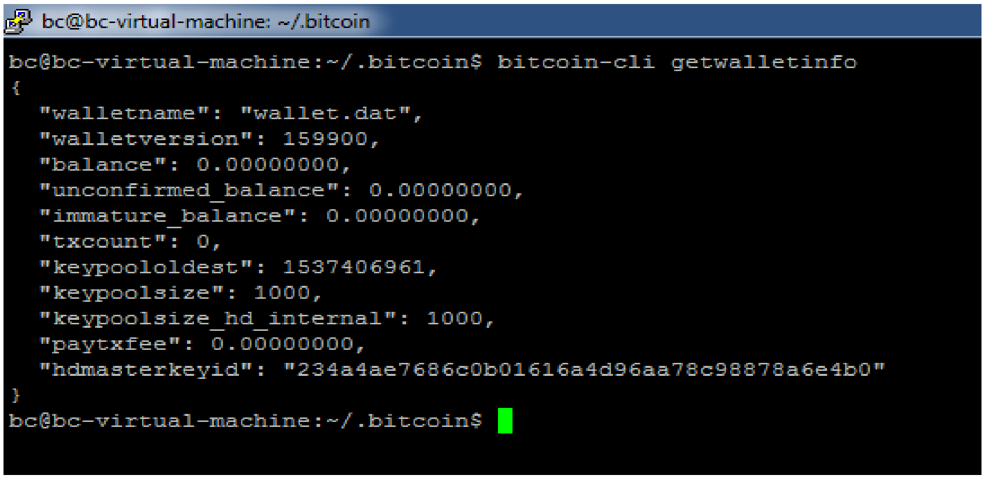
Outputs a JSON object listing the values for different attributes of mining like current no. of blocks, difficulty setting, chain (in our case, testnet)



```
bc@bc-virtual-machine:~/.bitcoin$ bitcoin-cli getmininginfo
{
  "blocks": 1380970,
  "currentblockweight": 0,
  "currentblocktx": 0,
  "difficulty": 17349304.92883517,
  "networkhashps": 253511577153754.2,
  "pooledtx": 0,
  "chain": "test",
  "warnings": ""
}
bc@bc-virtual-machine:~/.bitcoin$
```

Also, the bitcoin core client comes with a pre-installed wallet. We can get the information regarding this wallet using the command-

bitcoin-cli getwalletinfo



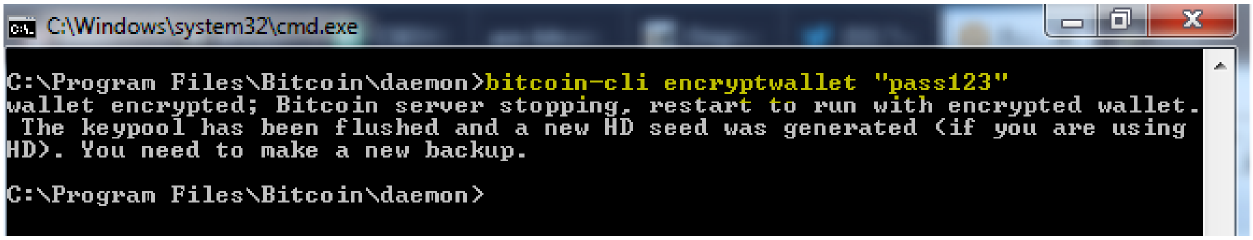
```
bc@bc-virtual-machine:~/.bitcoin$ bitcoin-cli getwalletinfo
{
  "walletname": "wallet.dat",
  "walletversion": 159900,
  "balance": 0.00000000,
  "unconfirmed_balance": 0.00000000,
  "immature_balance": 0.00000000,
  "txcount": 0,
  "keypoololdest": 1537406961,
  "keypoolsize": 1000,
  "keypoolsize_hd_internal": 1000,
  "paytxfee": 0.00000000,
  "hdmasterkeyid": "234a4ae7686c0b01616a4d96aa78c98878a6e4b0"
}
bc@bc-virtual-machine:~/.bitcoin$
```

<Step. 3> Set a password on the wallet / Unlock the password on the wallet.

As seen from the above snapshot, the inbuilt wallet is unencrypted (we know this because the JSON object returned by *getwalletinfo*, doesn’t contain the field- *unlocked_until*).

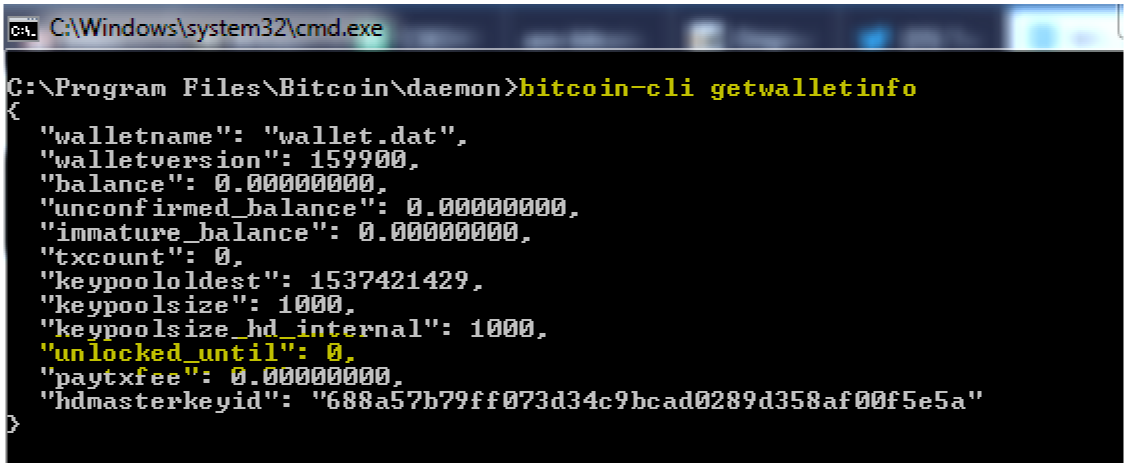
A wallet can be encrypted using the command- encryptwallet followed the the passphrase with which to encrypt the wallet. For example-

```
bitcoin-cli encryptwallet “pass123”
```



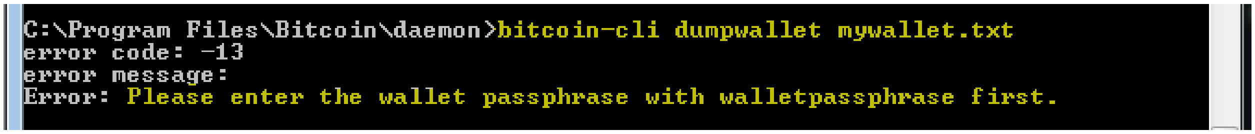
```
C:\Program Files\Bitcoin\daemon>bitcoin-cli encryptwallet "pass123"
wallet encrypted; Bitcoin server stopping, restart to run with encrypted wallet.
 The keypool has been flushed and a new HD seed was generated (if you are using
HD). You need to make a new backup.

C:\Program Files\Bitcoin\daemon>
```

Setting up encryption in your wallet requires a restart of the Bitcoin (testnet) server. After restarting the server, we can see that on executing the *bitcoin-cli getwalletinfo* command, an additional field *unlocked_until* is now added into the JSON object.



```
C:\Program Files\Bitcoin\daemon>bitcoin-cli getwalletinfo
{
  "walletname": "wallet.dat",
  "walletversion": 159900,
  "balance": 0.00000000,
  "unconfirmed_balance": 0.00000000,
  "immature_balance": 0.00000000,
  "txcount": 0,
  "keypoololdest": 1537421429,
  "keypoolsize": 1000,
  "keypoolsize_hd_internal": 1000,
  "unlocked_until": 0,
  "paytxfee": 0.00000000,
  "hdmasterkeyid": "688a57b79ff073d34c9bcad0289d358af00f5e5a"
}
```

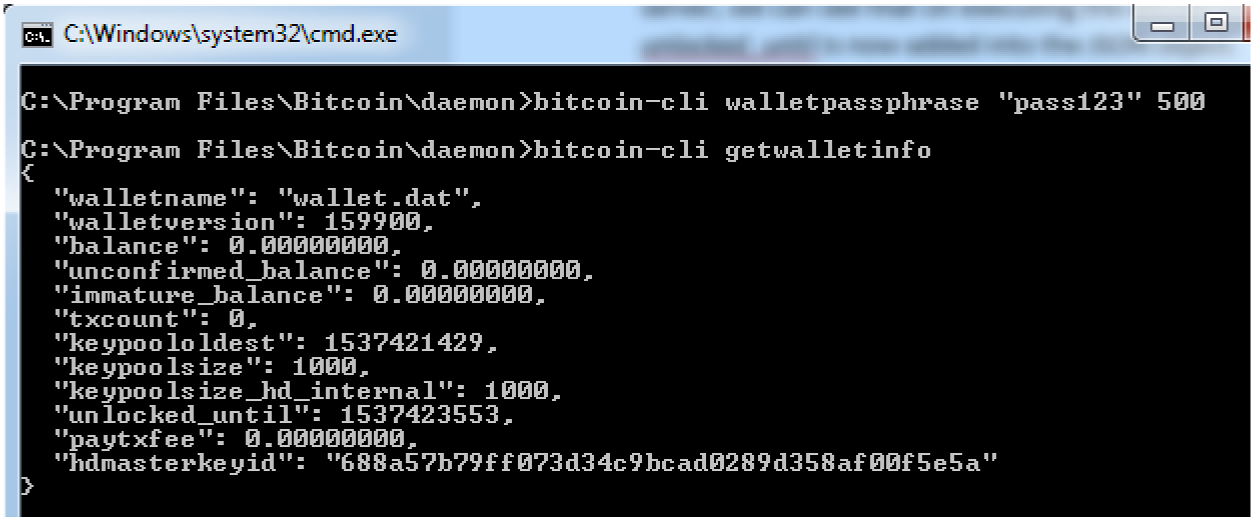
When the wallet is encrypted in this way, you are unable to perform operations concerning wallet such as dump the wallet as text file, dump the private key of the wallet, etc are all disabled as shown in the figure below-



```
C:\Program Files\Bitcoin\daemon>bitcoin-cli dumpwallet mywallet.txt
error code: -13
error message:
Error: Please enter the wallet passphrase with walletpassphrase first.
```

Now, in order to continue operations on the wallet, you should unlock the wallet using the command- *bitcoin-cli walletpassphrase* followed by two parameters- first is the passphrase that was used to encrypt the wallet and the second parameter is the no. of seconds to unlock the wallet before it gets locked again.

bitcoin-cli walletpassphrase "pass123" 500



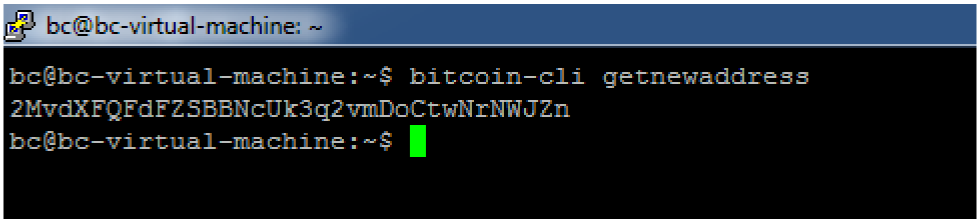
```
C:\Program Files\Bitcoin\daemon>bitcoin-cli walletpassphrase "pass123" 500

C:\Program Files\Bitcoin\daemon>bitcoin-cli getwalletinfo
{
  "walletname": "wallet.dat",
  "walletversion": 159900,
  "balance": 0.00000000,
  "unconfirmed_balance": 0.00000000,
  "immature_balance": 0.00000000,
  "txcount": 0,
  "keypoololdest": 1537421429,
  "keypoolsize": 1000,
  "keypoolsize_hd_internal": 1000,
  "unlocked_until": 1537423553,
  "paytxfee": 0.00000000,
  "hdmasterkeyid": "688a57b79ff073d34c9bcad0289d358af00f5e5a"
}
```

The above snapshot shows that the wallet was successfully unlocked using the command *walletpassphrase.* We can also see additional info by executing the command *getwalletinfo* that the wallet is now unlocked until a specified UNIX Epoch timestamp (no. of seconds since 1st January 1970).

<Step. 4> Generate an address / Get a Bitcoin / Check the Balance

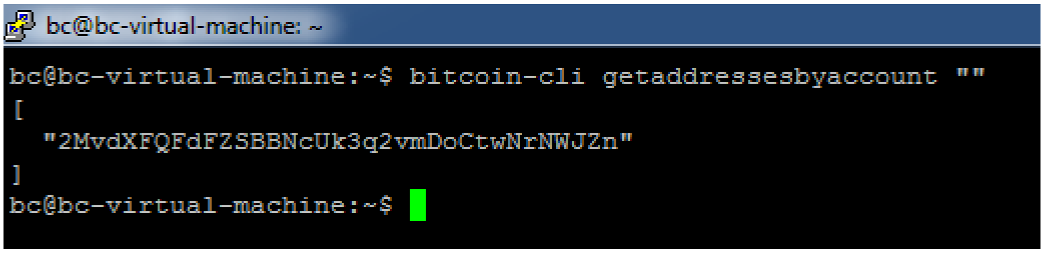
Addresses in bitcoin enable users to send/receive bitcoins. Addresses can be created in bitcoin using the command- *getnewaddress*

bitcoin-cli getnewaddress



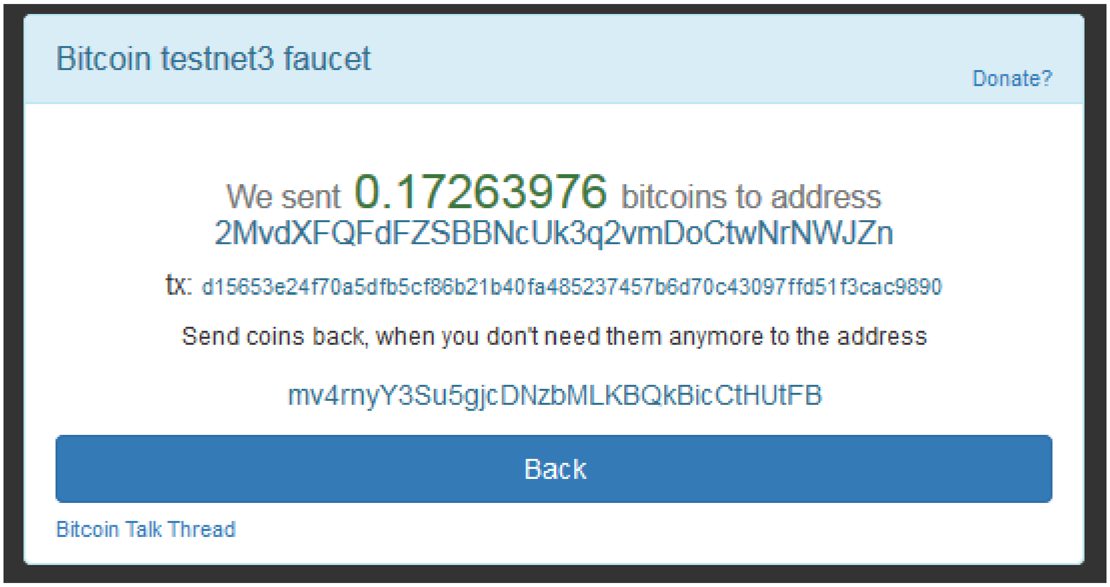
```
bc@bc-virtual-machine:~$ bitcoin-cli getnewaddress
2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn
bc@bc-virtual-machine:~$ 
```

We can confirm that the address has been created for the wallet using the command- *getaddressesbyaccount* which lists all the addresses contained in the wallet i.e.



```
bc@bc-virtual-machine:~$ bitcoin-cli getaddressesbyaccount ""
[
  "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn"
]
bc@bc-virtual-machine:~$ 
```

The above generated address- **2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn** was generated within the test network of bitcoin and we can get dummy bitcoins for such testnet address. The URL https://coinfaucet.eu/en/btc-testnet/ was used to get some bitcoins for the above address as shown below



Bitcoin testnet3 faucet

We sent 0.17263976 bitcoins to address 2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn

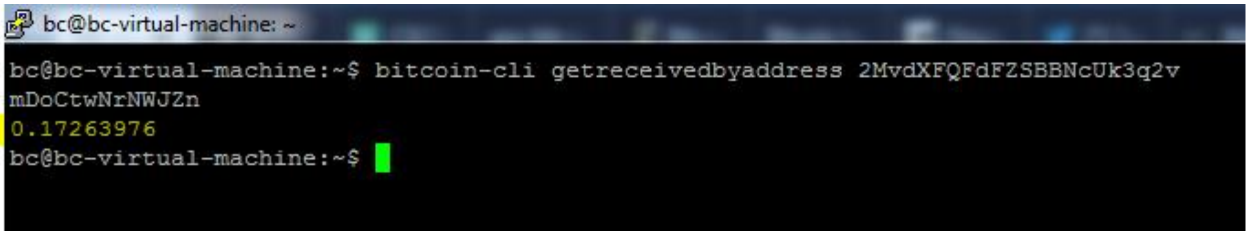
tx: d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890

Send coins back, when you don't need them anymore to the address

mv4rnyY3Su5gjcDNzbMLKBQkBicCtHUtFB

We can check the bitcoin balance available in a bitcoin address using the command- *getreceivedbyaddress* followed the address we want to check the balance for.

```
bitcoin-cli getreceivedbyaddress 2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn
```



```
bc@bc-virtual-machine:~$ bitcoin-cli getreceivedbyaddress 2MvdXFQFdFZSBBNcUk3q2v
mDoCtwNrNWJZn
0.17263976
bc@bc-virtual-machine:~$ 
```

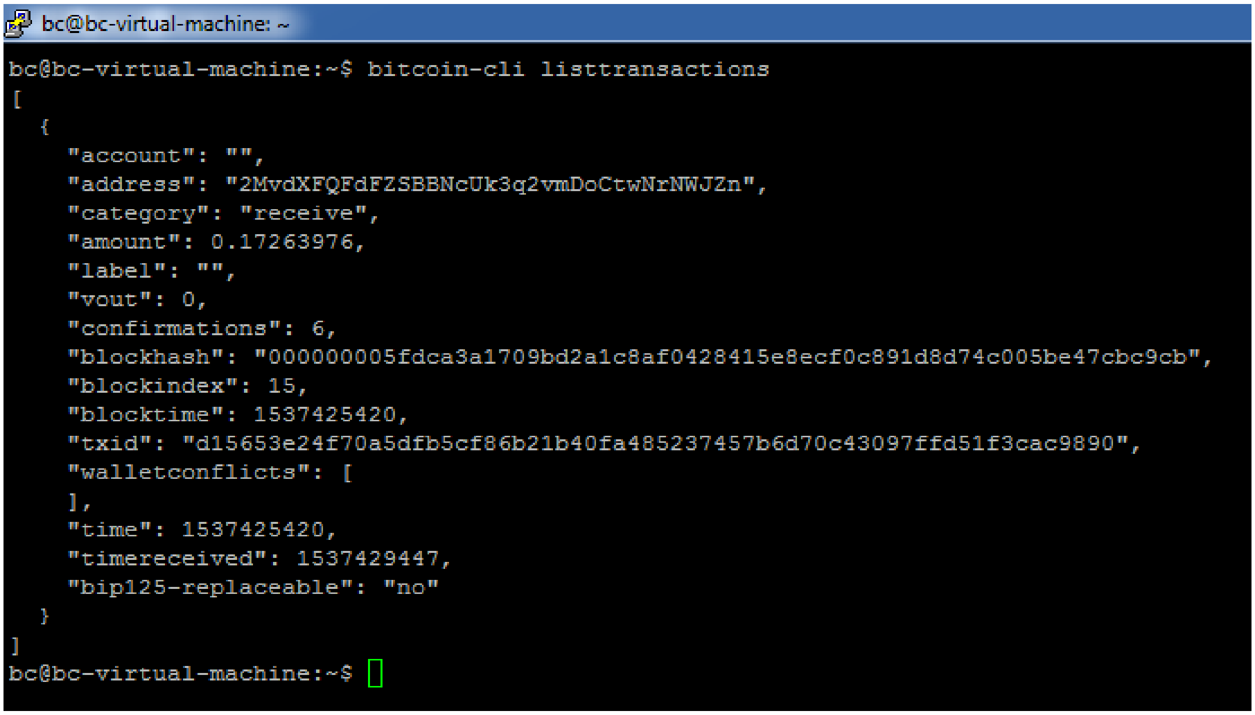
In this way, we can get bitcoins inside the test network and then check the balance in a given address.

<Step. 5> Check the reception of BTC, examine transaction contents / investigate more detailed transaction contents

5-1) Verify the transaction related to BTC reception and then check the balance. You can get information such as txid, amount and recipient's address.

In order to view transaction and its details, we can use the commands- *listtransactions* and *gettransaction*. The *listtransactions* will output an array of JSON object for each transaction. i.e.

```
bitcoin-cli listtransactions
```



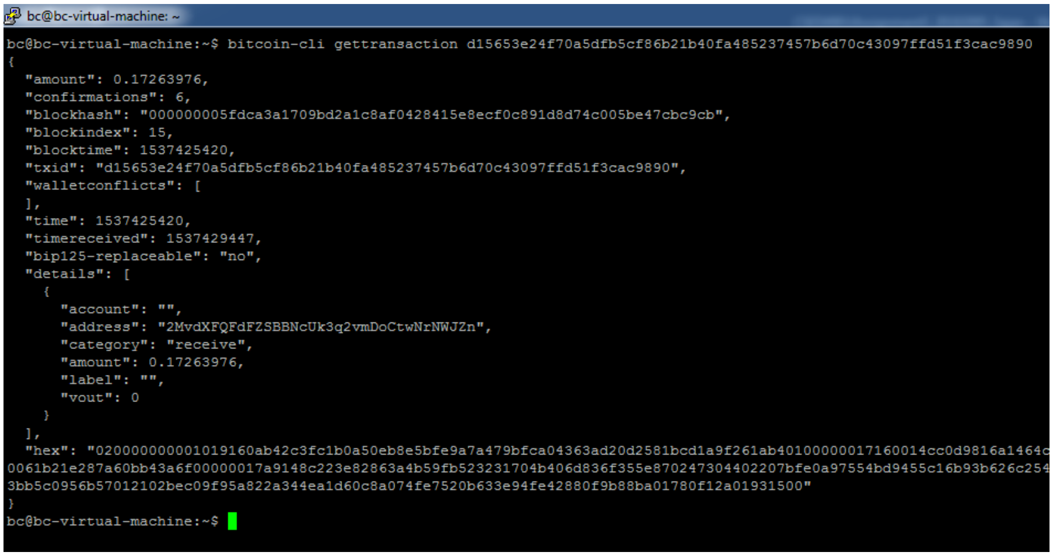
```
bc@bc-virtual-machine:~$ bitcoin-cli listtransactions
[
  {
    "account": "",
    "address": "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn",
    "category": "receive",
    "amount": 0.17263976,
    "label": "",
    "vout": 0,
    "confirmations": 6,
    "blockhash": "000000005fdca3a1709bd2a1c8af0428415e8ecf0c891d8d74c005be47cbc9cb",
    "blockindex": 15,
    "blocktime": 1537425420,
    "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
    "walletconflicts": [
    ],
    "time": 1537425420,
    "timereceived": 1537429447,
    "bip125-replaceable": "no"
  }
]
bc@bc-virtual-machine:~$
```

We can see from the snapshot above that the bitcoin sent to us in the previous step is being shown here with the correct address and amount. This command will list all the transactions applicable to all the addresses in the wallet.

5-2) Examine the transaction found in 5-1 using transaction id

One can also view a single transaction by using the command *gettransaction* followed by the transaction id in hex code

bitcoin-cli gettransaction d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890

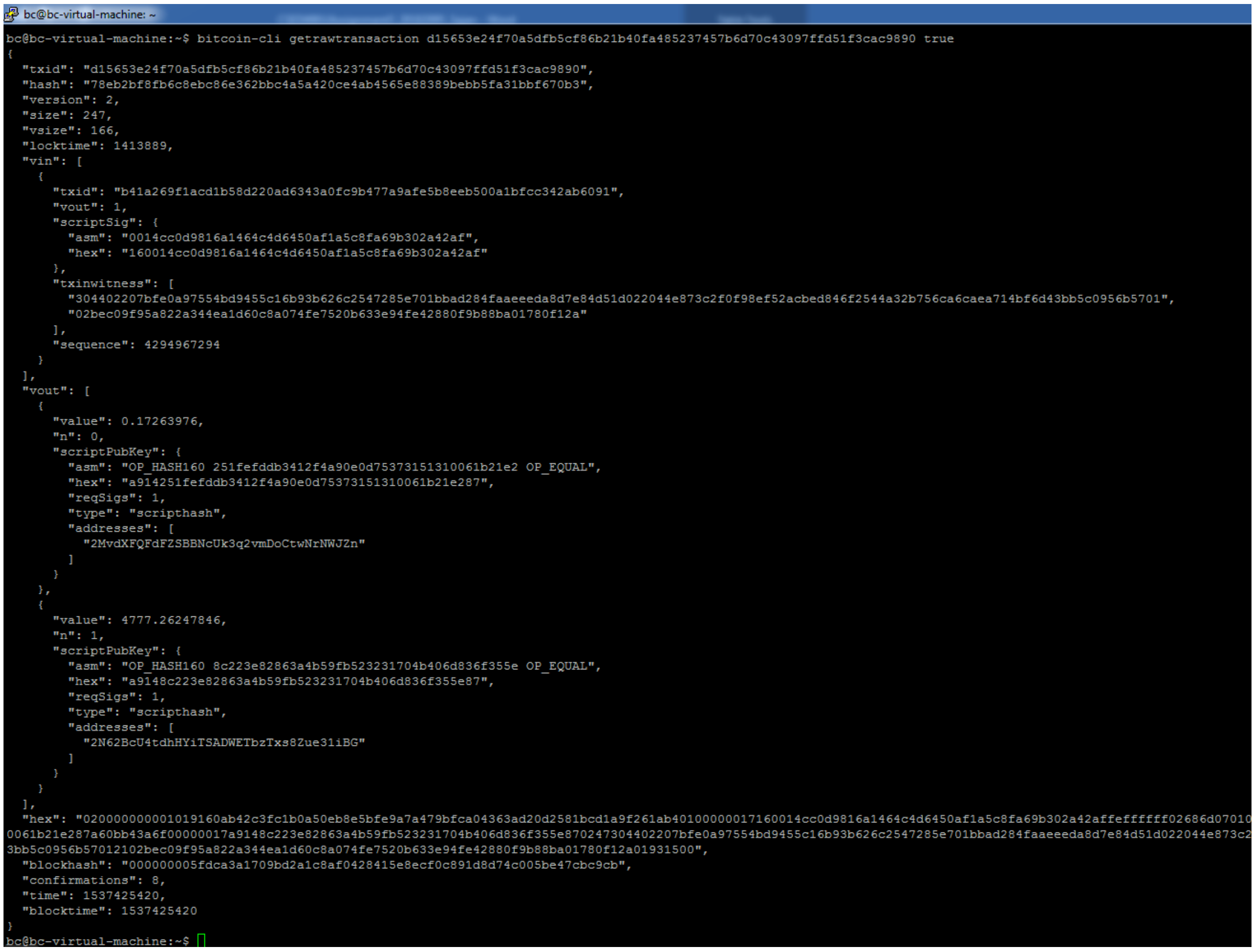


```
bc@bc-virtual-machine:~$ bitcoin-cli gettransaction d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890
{
  "amount": 0.17263976,
  "confirmations": 6,
  "blockhash": "000000005fdca3a1709bd2a1c8af0428415e8ecf0c891d8d74c005be47cbc9cb",
  "blockindex": 15,
  "blocktime": 1537425420,
  "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
  "walletconflicts": [
  ],
  "time": 1537425420,
  "timereceived": 1537429447,
  "bip125-replaceable": "no",
  "details": [
    {
      "account": "",
      "address": "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn",
      "category": "receive",
      "amount": 0.17263976,
      "label": "",
      "vout": 0
    }
  ],
  "hex": "020000000001019160ab42c3fc1b0a50eb8e5bfe9a7a479bfca04363ad20d2581bcd1a9f261ab401000000017160014cc0d9816a1464c
0061b21e287a60bb43a6f00000017a9148c223e82863a4b59fb523231704b406d836f355e870247304402207bfe0a97554bd9455c16b93b626c254
3bb5c0956b57012102bec09f95a822a344ea1d60c8a074fe7520b633e94fe42880f9b88ba01780f12a01931500"
}
bc@bc-virtual-machine:~$
```

5-3) The results of 5-2 are simple. Print out the actual details of the transactions

If we want to investigate into further details of the transaction, we can do so using the commands- *getrawtransaction* followed by the value true i.e.

```
bitcoin-cli getrawtransaction
d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890 true
```



```
bc@bc-virtual-machine:~$ bitcoin-cli getrawtransaction d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890 true
{
  "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
  "hash": "78eb2bf8fb6c8ebc86e362bbc4a5a420ce4ab4565e88389bebb5fa31bbf670b3",
  "version": 2,
  "size": 247,
  "vsize": 166,
  "locktime": 1413889,
  "vin": [
    {
      "txid": "b41a269f1acd1b58d220ad6343a0fc9b477a9afe5b8eeb500a1bfcc342ab6091",
      "vout": 1,
      "scriptSig": {
        "asm": "0014cc0d9816a1464c4d6450af1a5c8fa69b302a42af",
        "hex": "160014cc0d9816a1464c4d6450af1a5c8fa69b302a42af"
      },
      "txinwitness": [
        "304402207bfe0a97554bd9455c16b93b626c2547285e701bbad284faaeeeda8d7e84d51d022044e873c2f0f98ef52acbed846f2544a32b756ca6caea714bf6d43bb5c0956b5701",
        "02bec09f95a822a344ea1d60c8a074fe7520b633e94fe42880f9b88ba01780f12a"
      ],
      "sequence": 4294967294
    }
  ],
  "vout": [
    {
      "value": 0.17263976,
      "n": 0,
      "scriptPubKey": {
        "asm": "OP_HASH160 251fefddb3412f4a90e0d75373151310061b21e2 OP_EQUAL",
        "hex": "a914251fefddb3412f4a90e0d75373151310061b21e287",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn"
        ]
      }
    },
    {
      "value": 4777.26247846,
      "n": 1,
      "scriptPubKey": {
        "asm": "OP_HASH160 8c223e82863a4b59fb523231704b406d836f355e OP_EQUAL",
        "hex": "a9148c223e82863a4b59fb523231704b406d836f355e87",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2N62BcU4tdhHYiTSADWETbzTxs8Zue31iBG"
        ]
      }
    }
  ],
  "hex": "020000000001019160ab42c3fc1b0a50eb8e5bfe9a7a479bfca04363ad20d2581bcd1a9f261ab40100000017160014cc0d9816a1464c4d6450af1a5c8fa69b302a42affeffffff02686d07010
0061b21e287a60bb43a6f00000017a9148c223e82863a4b59fb523231704b406d836f355e870247304402207bfe0a97554bd9455c16b93b626c2547285e701bbad284faaeeeda8d7e84d51d022044e873c2
3bb5c0956b57012102bec09f95a822a344ea1d60c8a074fe7520b633e94fe42880f9b88ba01780f12a01931500",
  "blockhash": "000000005fdca3a1709bd2a1c8af0428415e8ecf0c891d8d74c005be47cbc9cb",
  "confirmations": 8,
  "time": 1537425420,
  "blocktime": 1537425420
}
bc@bc-virtual-machine:~$
```

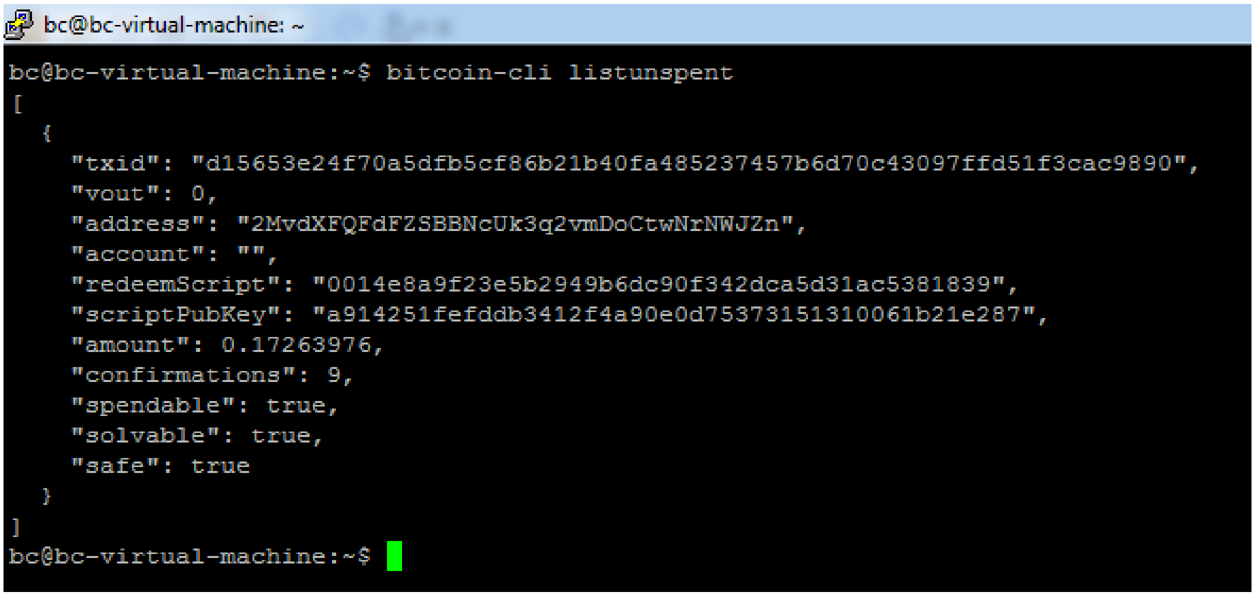
Using this command, we view the details about transactions such as- input UTXO consumed, no. of inputs and outputs, previous transaction referenced, change value created, locking script encumbered to the transaction output as such.

<Step. 6> Create a transaction using UTXO / Sign the transaction / Send the transaction

6-1) Determine unspent and confirmed outputs to use as an input to a transaction

The confirmed and unspent outputs of the transactions can be obtained using the command- *listunspent*

```
bitcoin-cli listunspent
```

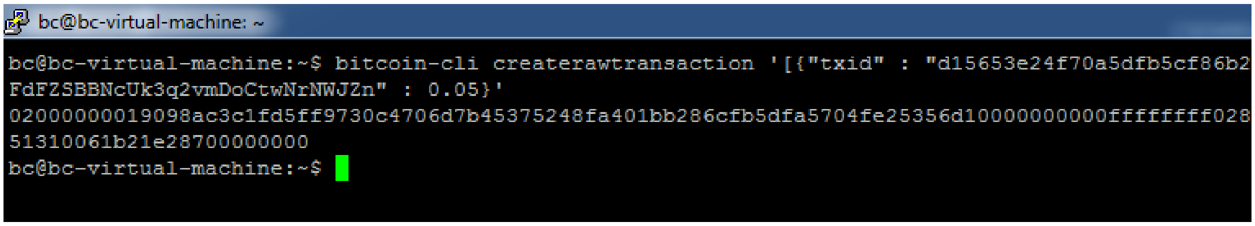


```
bc@bc-virtual-machine:~$ bitcoin-cli listunspent
[
  {
    "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
    "vout": 0,
    "address": "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn",
    "account": "",
    "redeemScript": "0014e8a9f23e5b2949b6dc90f342dca5d31ac5381839",
    "scriptPubKey": "a914251fefddb3412f4a90e0d75373151310061b21e287",
    "amount": 0.17263976,
    "confirmations": 9,
    "spendable": true,
    "solvable": true,
    "safe": true
  }
]
bc@bc-virtual-machine:~$
```

6-2) Choose one of UTXOs gained from 6-1 and then create a transaction.

Transactions in bitcoin are created using the *createrawtransaction* command which takes the unspent transaction output address as its input and two transaction outputs- bitcoin sent to the new address and change sent to the previous address i.e.

```
bitcoin-cli createrawtransaction ‘[{“txid” :
“d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890”, “vout” : 0}]’
‘{“2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX” : 0.1,
“2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn” : 0.05}’
```

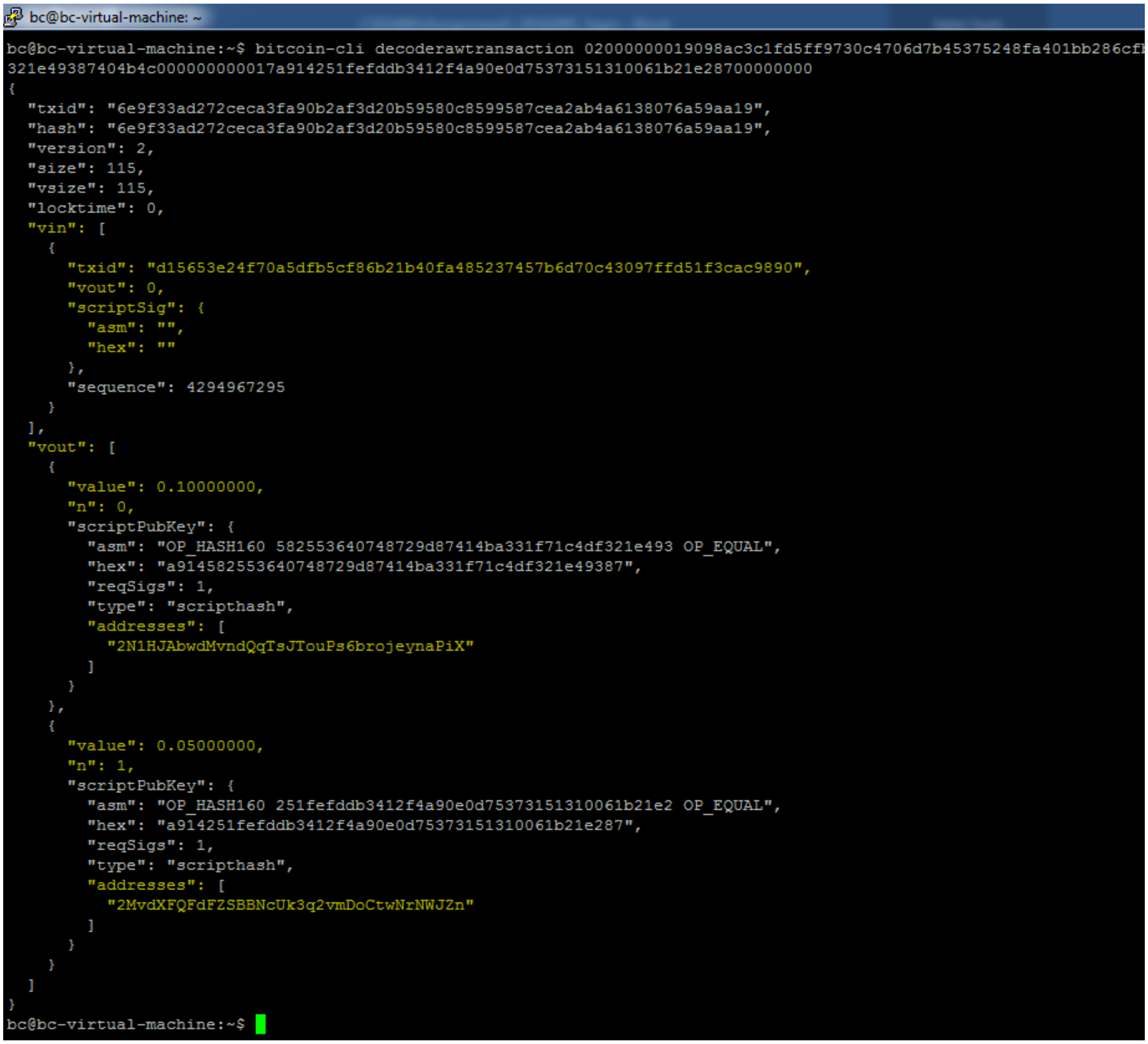


```
bc@bc-virtual-machine:~$ bitcoin-cli createrawtransaction '[{"txid" : "d15653e24f70a5dfb5cf86b2
FdFZSBBNcUk3q2vmDoCtwNrNWJZn" : 0.05}'
02000000019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cfb5dfa5704fe25356d10000000000ffffffff028
51310061b21e28700000000
bc@bc-virtual-machine:~$
```

The first argument to the createrawtransaction is the ith output produced by the previous transaction id and the second argument is the list of addresses with the values of bitcoin to send. The output of the transaction is the raw hex string of the the transaction created.

6-3) Decode rawtransaction with hex string generated in 6-2. You can see that the 'scriptsig' field is empty. You can see that the scriptSig field is empty

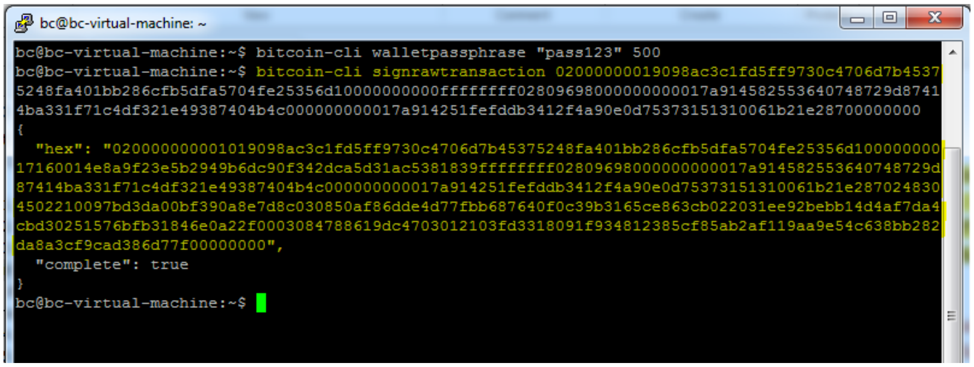
```
bitcoin-cli decoderawtransaction
02000000019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cfb5dfa5704fe25356d100000
00000ffffffff0280969800000000000017a9145825536407487 29d87414ba331f71c4df321e493874
04b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e28700000000
```



```
bc@bc-virtual-machine:~$ bitcoin-cli decoderawtransaction 02000000019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cf
321e49387404b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e28700000000
{
  "txid": "6e9f33ad272ceca3fa90b2af3d20b59580c8599587cea2ab4a6138076a59aa19",
  "hash": "6e9f33ad272ceca3fa90b2af3d20b59580c8599587cea2ab4a6138076a59aa19",
  "version": 2,
  "size": 115,
  "vsize": 115,
  "locktime": 0,
  "vin": [
    {
      "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
      "vout": 0,
      "scriptSig": {
        "asm": "",
        "hex": ""
      },
      "sequence": 4294967295
    }
  ],
  "vout": [
    {
      "value": 0.10000000,
      "n": 0,
      "scriptPubKey": {
        "asm": "OP_HASH160 582553640748729d87414ba331f71c4df321e493 OP_EQUAL",
        "hex": "a914582553640748729d87414ba331f71c4df321e49387",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX"
        ]
      }
    },
    {
      "value": 0.05000000,
      "n": 1,
      "scriptPubKey": {
        "asm": "OP_HASH160 251fefddb3412f4a90e0d75373151310061b21e2 OP_EQUAL",
        "hex": "a914251fefddb3412f4a90e0d75373151310061b21e287",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn"
        ]
      }
    }
  ]
}
bc@bc-virtual-machine:~$
```

6-4) Sign the transaction so that the ‘scriptSig’ field can be filled. As a result of signing, you will obtain a hex string

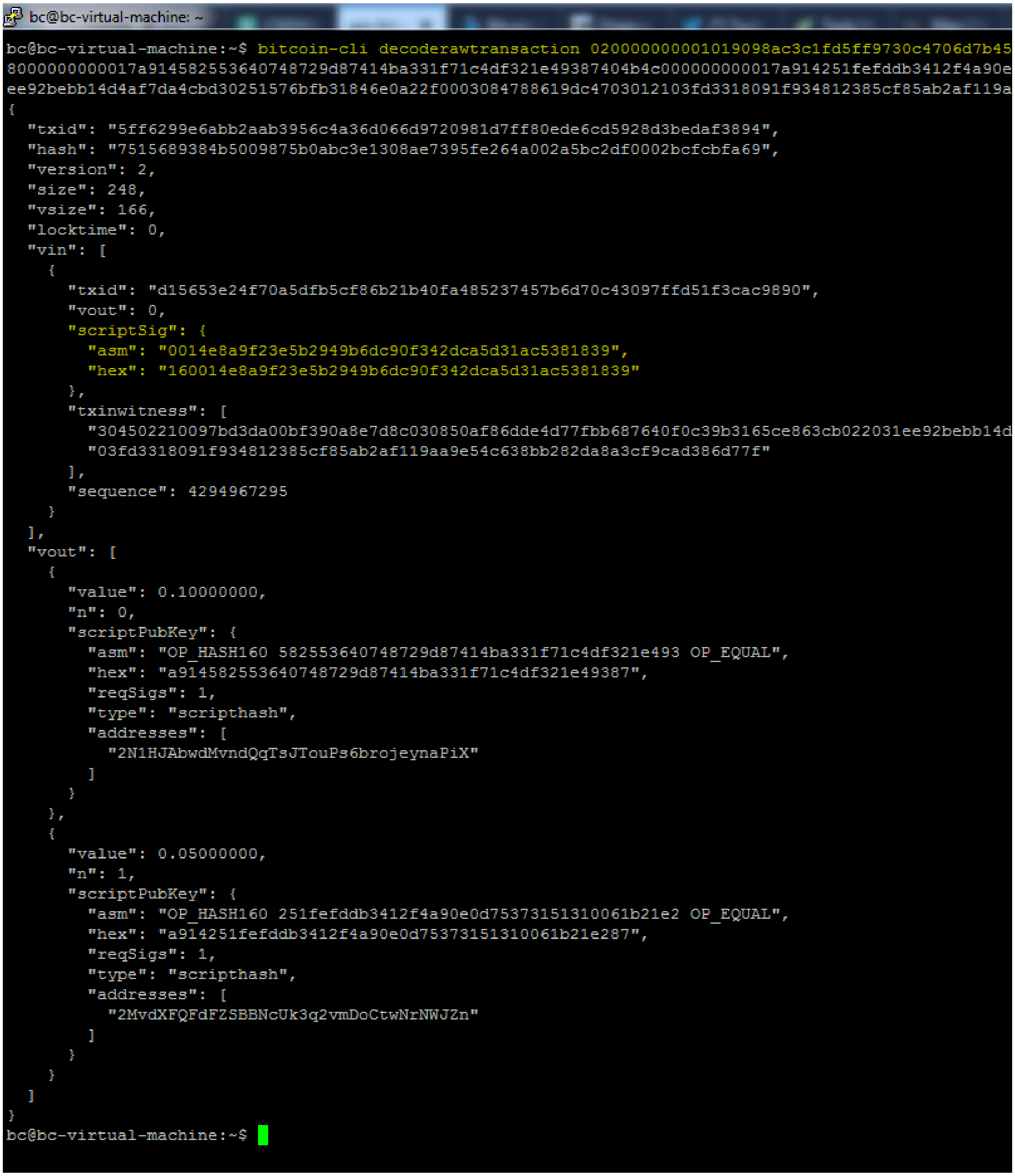
bitcoin-cli signrawtransaction **HEX_STRING**



```
bc@bc-virtual-machine:~$ bitcoin-cli walletpassphrase "pass123" 500
bc@bc-virtual-machine:~$ bitcoin-cli signrawtransaction 02000000019098ac3c1fd5ff9730c4706d7b4537
5248fa401bb286cfb5dfa5704fe25356d10000000000ffffffff0280969800000000000017a914582553640748729d8741
4ba331f71c4df321e49387404b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e28700000000
{
  "hex": "0200000000001019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cfb5dfa5704fe25356d100000000
17160014e8a9f23e5b2949b6dc90f342dca5d31ac5381839ffffffff0280969800000000000017a914582553640748729d
87414ba331f71c4df321e49387404b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e287024830
4502210097bd3da00bf390a8e7d8c030850af86dde4d77fbb687640f0c39b3165ce863cb022031ee92bebb14d4af7da4
cbd30251576bfb31846e0a22f0003084788619dc4703012103fd3318091f934812385cf85ab2af119aa9e54c638bb282
da8a3cf9cad386d77f00000000",
  "complete": true
}
bc@bc-virtual-machine:~$
```

6-5) Check if ‘scriptsig’ field of the transaction is filled or not. If you have performed everything correctly, you will be able to see the filled ‘scriptsig’ field after decoding the hex string

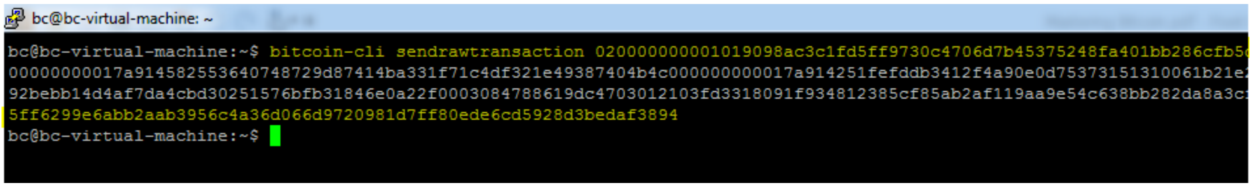
bitcoin-cli decoderawtransaction **HEX_CODE_OBTAINED_FROM_6-4**



```
bc@bc-virtual-machine:~$ bitcoin-cli decoderawtransaction 020000000001019098ac3c1fd5ff9730c4706d7b45
8000000000017a914582553640748729d87414ba331f71c4df321e49387404b4c000000000017a914251fefddb3412f4a90e
ee92bebb14d4af7da4cbd30251576bfb31846e0a22f0003084788619dc4703012103fd3318091f934812385cf85ab2af119a
{
  "txid": "5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894",
  "hash": "7515689384b5009875b0abc3e1308ae7395fe264a002a5bc2df0002bcfcbfa69",
  "version": 2,
  "size": 248,
  "vsize": 166,
  "locktime": 0,
  "vin": [
    {
      "txid": "d15653e24f70a5dfb5cf86b21b40fa485237457b6d70c43097ffd51f3cac9890",
      "vout": 0,
      "scriptSig": {
        "asm": "0014e8a9f23e5b2949b6dc90f342dca5d31ac5381839",
        "hex": "160014e8a9f23e5b2949b6dc90f342dca5d31ac5381839"
      },
      "txinwitness": [
        "304502210097bd3da00bf390a8e7d8c030850af86dde4d77fbb687640f0c39b3165ce863cb022031ee92bebb14d
        "03fd3318091f934812385cf85ab2af119aa9e54c638bb282da8a3cf9cad386d77f"
      ],
      "sequence": 4294967295
    }
  ],
  "vout": [
    {
      "value": 0.10000000,
      "n": 0,
      "scriptPubKey": {
        "asm": "OP_HASH160 582553640748729d87414ba331f71c4df321e493 OP_EQUAL",
        "hex": "a914582553640748729d87414ba331f71c4df321e49387",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX"
        ]
      }
    },
    {
      "value": 0.05000000,
      "n": 1,
      "scriptPubKey": {
        "asm": "OP_HASH160 251fefddb3412f4a90e0d75373151310061b21e2 OP_EQUAL",
        "hex": "a914251fefddb3412f4a90e0d75373151310061b21e287",
        "reqSigs": 1,
        "type": "scripthash",
        "addresses": [
          "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn"
        ]
      }
    }
  ]
}
bc@bc-virtual-machine:~$
```

6-6) Send the transaction to the bitcoin network

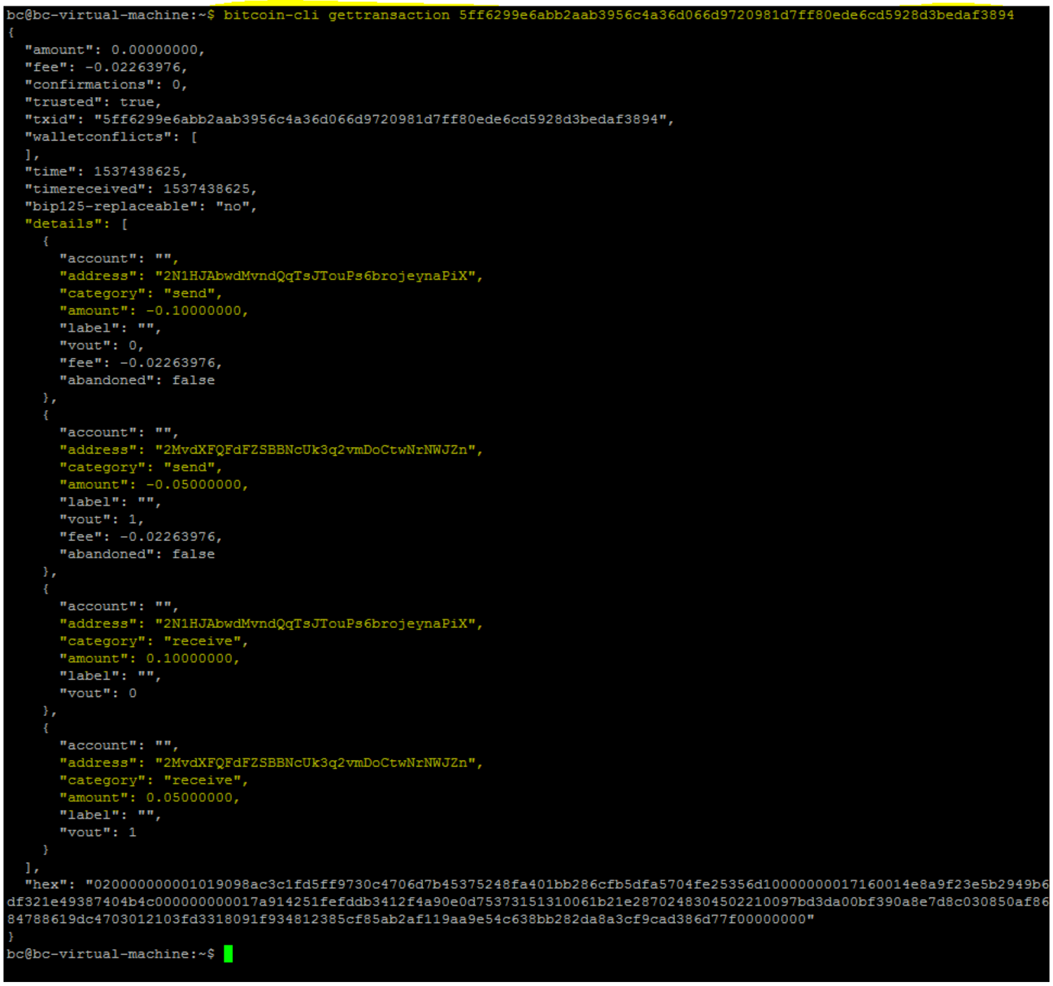
bitcoin-cli sendrawtransaction **HEX_STRING**



```
bc@bc-virtual-machine:~$ bitcoin-cli sendrawtransaction 020000000001019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cfb5
00000000017a914582553640748729d87414ba331f71c4df321e49387404b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e
92bebb14d4af7da4cbd30251576bfb31846e0a22f0003084788619dc4703012103fd3318091f934812385cf85ab2af119aa9e54c638bb282da8a3c
5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894
bc@bc-virtual-machine:~$
```

6-7) Examine the contents of the transaction using tx_id obtained from 6-6

bitcoin-cli gettransaction **TXN_ID_FROM_6-6**



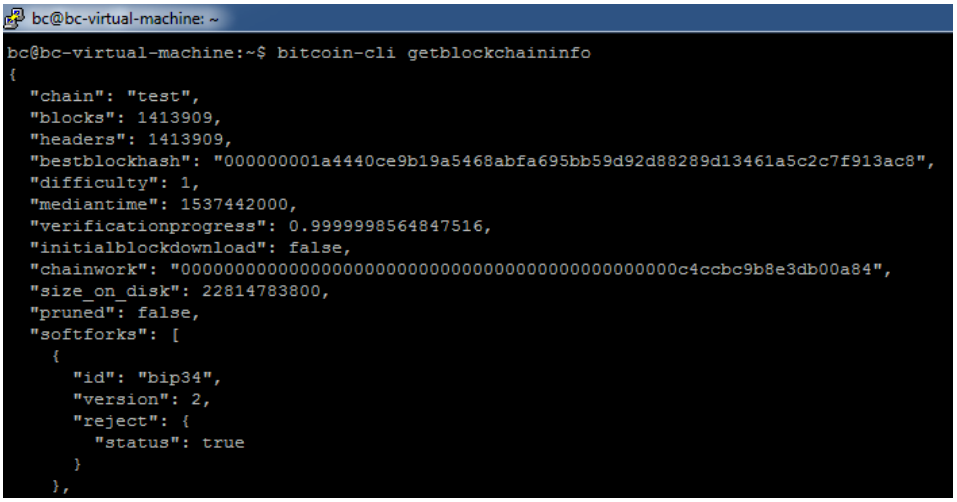
```
bc@bc-virtual-machine:~$ bitcoin-cli gettransaction 5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894
{
  "amount": 0.00000000,
  "fee": -0.02263976,
  "confirmations": 0,
  "trusted": true,
  "txid": "5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894",
  "walletconflicts": [
  ],
  "time": 1537438625,
  "timereceived": 1537438625,
  "bip125-replaceable": "no",
  "details": [
    {
      "account": "",
      "address": "2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX",
      "category": "send",
      "amount": -0.10000000,
      "label": "",
      "vout": 0,
      "fee": -0.02263976,
      "abandoned": false
    },
    {
      "account": "",
      "address": "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn",
      "category": "send",
      "amount": -0.05000000,
      "label": "",
      "vout": 1,
      "fee": -0.02263976,
      "abandoned": false
    },
    {
      "account": "",
      "address": "2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX",
      "category": "receive",
      "amount": 0.10000000,
      "label": "",
      "vout": 0
    },
    {
      "account": "",
      "address": "2MvdXFQFdFZSBBNcUk3q2vmDoCtwNrNWJZn",
      "category": "receive",
      "amount": 0.05000000,
      "label": "",
      "vout": 1
    }
  ],
  "hex": "020000000001019098ac3c1fd5ff9730c4706d7b45375248fa401bb286cfb5dfa5704fe25356d10000000017160014e8a9f23e5b2949b6
df321e49387404b4c000000000017a914251fefddb3412f4a90e0d75373151310061b21e28702483045022100097bd3da00bf390a8e7d8c030850af86
84788619dc4703012103fd3318091f934812385cf85ab2af119aa9e54c638bb282da8a3cf9cad386d77f00000000"
}
bc@bc-virtual-machine:~$
```

<Step. 7> Search for blocks

7-1) Search for blocks connected to main chain

The general summary of blocks connected to the blockchain can be obtained using the command- *getblockchaininfo* i.e.

bitcoin-cli getblockchaininfo

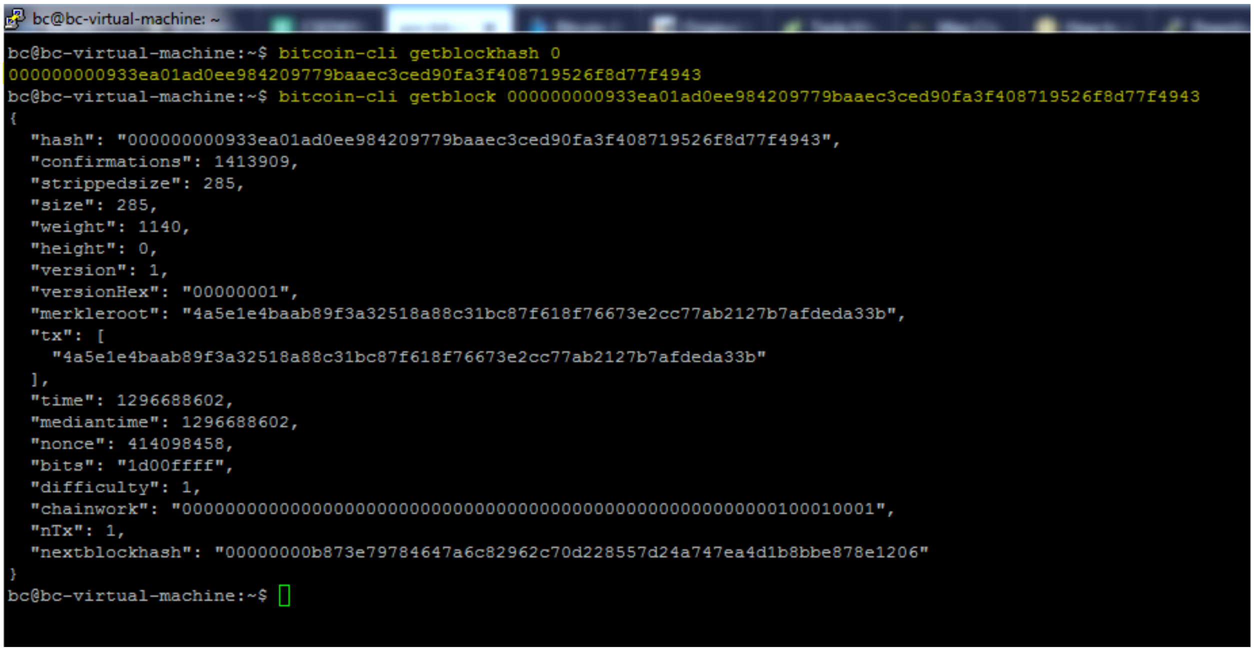


```
bc@bc-virtual-machine:~$ bitcoin-cli getblockchaininfo
{
  "chain": "test",
  "blocks": 1413909,
  "headers": 1413909,
  "bestblockhash": "000000001a4440ce9b19a5468abfa695bb59d92d88289d13461a5c2c7f913ac8",
  "difficulty": 1,
  "mediantime": 1537442000,
  "verificationprogress": 0.9999998564847516,
  "initialblockdownload": false,
  "chainwork": "0000000000000000000000000000000000000000000000c4ccbc9b8e3db00a84",
  "size_on_disk": 22814783800,
  "pruned": false,
  "softforks": [
    {
      "id": "bip34",
      "version": 2,
      "reject": {
        "status": true
      }
    },
```

7-2) The block of height 0

bitcoin-cli getblockhash 0

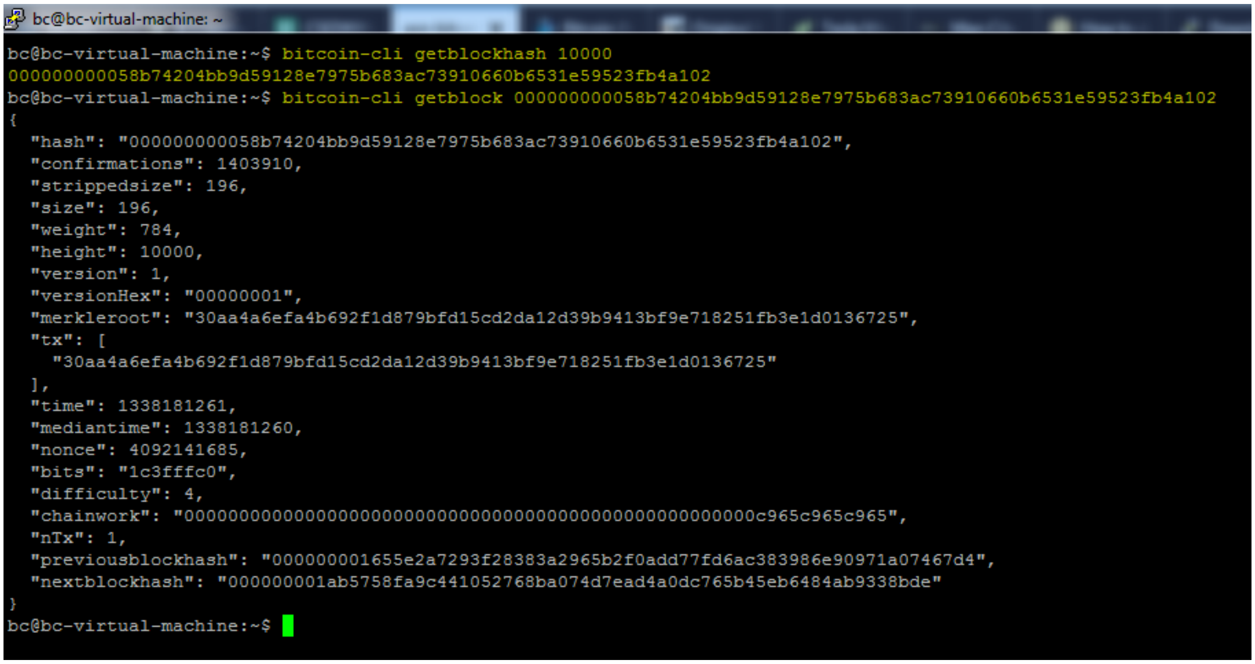
bitcoin-cli getblock **HASH_RETURNED_BY_BLOCK_0**



```
bc@bc-virtual-machine:~$ bitcoin-cli getblockhash 0
000000000933ea01ad0ee984209779baaec3ced90fa3f408719526f8d77f4943
bc@bc-virtual-machine:~$ bitcoin-cli getblock 000000000933ea01ad0ee984209779baaec3ced90fa3f408719526f8d77f4943
{
  "hash": "000000000933ea01ad0ee984209779baaec3ced90fa3f408719526f8d77f4943",
  "confirmations": 1413909,
  "strippedsize": 285,
  "size": 285,
  "weight": 1140,
  "height": 0,
  "version": 1,
  "versionHex": "00000001",
  "merkleroot": "4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b",
  "tx": [
    "4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b"
  ],
  "time": 1296688602,
  "mediantime": 1296688602,
  "nonce": 414098458,
  "bits": "1d00ffff",
  "difficulty": 1,
  "chainwork": "0000000000000000000000000000000000000000000000000000000100010001",
  "nTx": 1,
  "nextblockhash": "00000000b873e79784647a6c82962c70d228557d24a747ea4d1b8bbe878e1206"
}
bc@bc-virtual-machine:~$ 
```

7-3) The block of height 10000

bitcoin-cli getblockhash 10000
bitcoin-cli getblock **HASH_RETURNED_BY_BLOCK_10000**

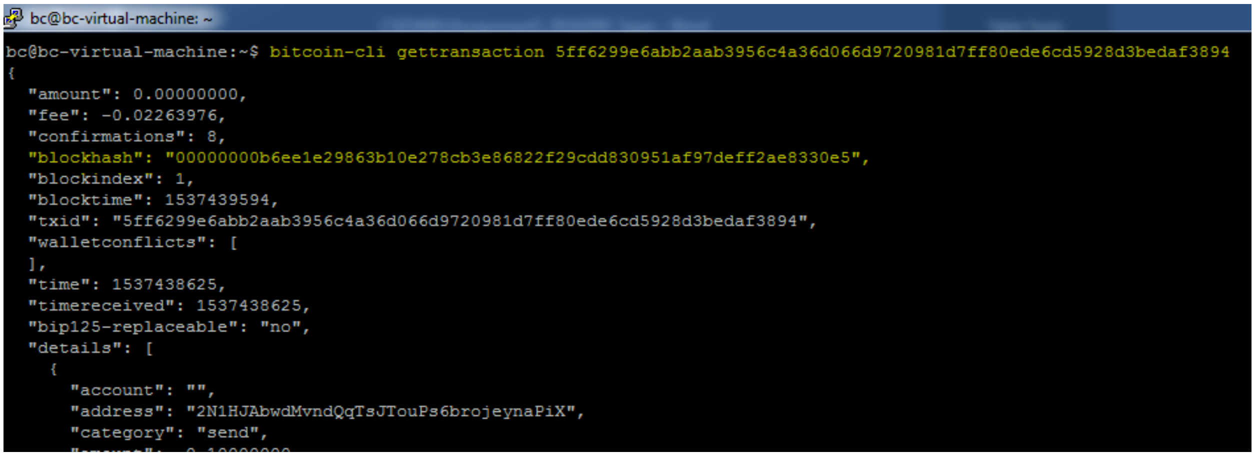


```
bc@bc-virtual-machine:~$ bitcoin-cli getblockhash 10000
0000000000058b74204bb9d59128e7975b683ac73910660b6531e59523fb4a102
bc@bc-virtual-machine:~$ bitcoin-cli getblock 0000000000058b74204bb9d59128e7975b683ac73910660b6531e59523fb4a102
{
  "hash": "0000000000058b74204bb9d59128e7975b683ac73910660b6531e59523fb4a102",
  "confirmations": 1403910,
  "strippedsize": 196,
  "size": 196,
  "weight": 784,
  "height": 10000,
  "version": 1,
  "versionHex": "00000001",
  "merkleroot": "30aa4a6efa4b692f1d879bfd15cd2da12d39b9413bf9e718251fb3e1d0136725",
  "tx": [
    "30aa4a6efa4b692f1d879bfd15cd2da12d39b9413bf9e718251fb3e1d0136725"
  ],
  "time": 1338181261,
  "mediantime": 1338181260,
  "nonce": 4092141685,
  "bits": "1c3fffc0",
  "difficulty": 4,
  "chainwork": "0000000000000000000000000000000000000000000000000000c965c965c965",
  "nTx": 1,
  "previousblockhash": "000000001655e2a7293f28383a2965b2f0add77fd6ac383986e90971a07467d4",
  "nextblockhash": "000000001ab5758fa9c441052768ba074d7ead4a0dc765b45eb6484ab9338bde"
}
bc@bc-virtual-machine:~$
```

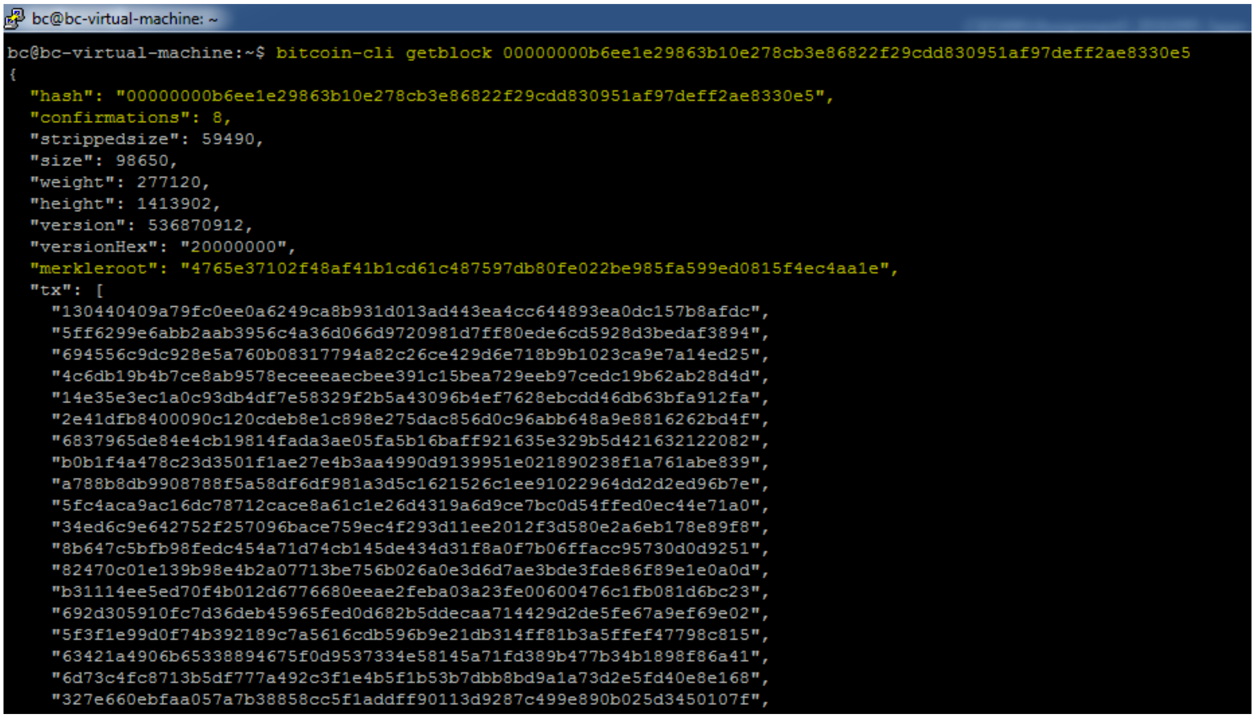
7-4) Block including the transaction in which you transferred the BTC

In order to find the block containing the transaction we incurred, we have to first execute the *gettransaction* command followed by the transaction id as its argument, which also contains the *blockchainhash* field. Using this value of *blockchainhash* field as argument into the *getblock* command, we can find the block that contains the transaction we used for transferring bitcoin i.e.

bitcoin-cli gettransaction 5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894
bitcoin-cli getblock **BLOCKCHAINHASH_FIELD_VALUE**



```
bc@bc-virtual-machine:~$ bitcoin-cli gettransaction 5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894
{
  "amount": 0.00000000,
  "fee": -0.02263976,
  "confirmations": 8,
  "blockhash": "00000000b6ee1e29863b10e278cb3e86822f29cdd830951af97deff2ae8330e5",
  "blockindex": 1,
  "blocktime": 1537439594,
  "txid": "5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894",
  "walletconflicts": [
  ],
  "time": 1537438625,
  "timereceived": 1537438625,
  "bip125-replaceable": "no",
  "details": [
    {
      "account": "",
      "address": "2N1HJAbwdMvndQqTsJTouPs6brojeynaPiX",
      "category": "send",
      "amount": -0.10000000
```



```
bc@bc-virtual-machine:~$ bitcoin-cli getblock 00000000b6ee1e29863b10e278cb3e86822f29cdd830951af97deff2ae8330e5
{
  "hash": "00000000b6ee1e29863b10e278cb3e86822f29cdd830951af97deff2ae8330e5",
  "confirmations": 8,
  "strippedsize": 59490,
  "size": 98650,
  "weight": 277120,
  "height": 1413902,
  "version": 536870912,
  "versionHex": "20000000",
  "merkleroot": "4765e37102f48af41b1cd61c487597db80fe022be985fa599ed0815f4ec4aa1e",
  "tx": [
    "130440409a79fc0ee0a6249ca8b931d013ad443ea4cc644893ea0dc157b8afdc",
    "5ff6299e6abb2aab3956c4a36d066d9720981d7ff80ede6cd5928d3bedaf3894",
    "694556c9dc928e5a760b08317794a82c26ce429d6e718b9b1023ca9e7a14ed25",
    "4c6db19b4b7ce8ab9578eceeeaecbee391c15bea729eeb97cedc19b62ab28d4d",
    "14e35e3ec1a0c93db4df7e58329f2b5a43096b4ef7628ebcdd46db63bfa912fa",
    "2e41dfb8400090c120cdeb8e1c898e275dac856d0c96abb648a9e8816262bd4f",
    "6837965de84e4cb19814fada3ae05fa5b16baff921635e329b5d421632122082",
    "b0b1f4a478c23d3501f1ae27e4b3aa4990d9139951e021890238f1a761abe839",
    "a788b8db9908788f5a58df6df981a3d5c1621526c1ee91022964dd2d2ed96b7e",
    "5fc4aca9ac16dc78712cace8a61c1e26d4319a6d9ce7bc0d54ffed0ec44e71a0",
    "34ed6c9e642752f257096bace759ec4f293d11ee2012f3d580e2a6eb178e89f8",
    "8b647c5bfb98fedc454a71d74cb145de434d31f8a0f7b06ffacc95730d0d9251",
    "82470c01e139b98e4b2a07713be756b026a0e3d6d7ae3bde3fde86f89e1e0a0d",
    "b31114ee5ed70f4b012d6776680eeae2feba03a23fe00600476c1fb081d6bc23",
    "692d305910fc7d36deb45965fed0d682b5ddecaa714429d2de5fe67a9ef69e02",
    "5f3f1e99d0f74b392189c7a5616cdb596b9e21db314ff81b3a5ffef47798c815",
    "63421a4906b65338894675f0d9537334e58145a71fd389b477b34b1898f86a41",
    "6d73c4fc8713b5df777a492c3f1e4b5f1b53b7dbb8bd9a1a73d2e5fd40e8e168",
    "327e660ebfaa057a7b38858cc5f1addff90113d9287c499e890b025d3450107f",
```