

# **Build the Bitcoin core**

on Ubuntu 17.10,16.10,16.04,14.04  
or  
Debian 8(Jessie)

盧瑞山 教授

參照課本第三章 3.1.2節

# Linux 版本差異

	Git	automake	python2.7	python3	pip	pip3
Debian 8	No	No	Yes	Python 3.4.2	No	No
Ubuntu 17.10	Yes	No	Yes	Python3.6	No	No
Ubuntu 17.04	Yes	No	Yes	Python3.5	No	No
Ubuntu 16.10	Yes	No	Yes	Python3.5	No	No
Ubuntu 16.04	Yes	No	Yes	Python3.5	No	No
Ubuntu 14.04	No	No	Yes	Python3.4.3	No	No
Centos 7.0	No	No	Yes	No	No	No

# install git

- on Ubuntu 14.04 you need to install git by yourself
- `sudo apt-get update`
- `sudo apt-get install git`

# install pip or pip3

- `sudo apt-get install python-pip`
- `sudo apt-get install python3-pip`

# download Bitcoin Core 0.14.1

- git clone **-b v0.14.1** <https://github.com/bitcoin/bitcoin.git>
- 或者是
- git clone **-b v0.14.1** <https://github.com/bitcoin/bitcoin.git> bitcoin0.14.1

# Dependencies

These dependencies are required:

Library	Purpose	Description
libssl	Crypto	Random Number Generation, Elliptic Curve Cryptography
libboost	Utility	Library for threading, data structures, etc
libevent	Networking	OS independent asynchronous networking

Optional dependencies:

Library	Purpose	Description
miniupnpc	UPnP Support	Firewall-jumping support
libdb4.8	Berkeley DB	Wallet storage (only needed when wallet enabled)
qt	GUI	GUI toolkit (only needed when GUI enabled)
protobuf	Payments in GUI	Data interchange format used for payment protocol (only needed when GUI enabled)
libqrencode	QR codes in GUI	Optional for generating QR codes (only needed when GUI enabled)
univalue	Utility	JSON parsing and encoding (bundled version will be used unless --with-system-univalue passed to configure)

# install automake

1. **sudo apt-get install autoconf** (will install 4 packages including autoconf automake autotools-dev m4)
2. **sudo apt-get install build-essential libtool autotools-dev automake pkg-config libssl-dev libevent-dev bsdmainutils**
3. **sudo apt-get install libboost-all-dev** 安裝boost函式套件

# BerkeleyDB is required for the wallet

install BerkeleyDB 4.8 for **Ubuntu**

1. `sudo apt-get install software-properties-common`
2. `sudo add-apt-repository ppa:bitcoin/bitcoin`
3. **`sudo apt-get update`**
4. `sudo apt-get install libdb4.8-dev libdb4.8++-dev`

## **For Debian**

```
cd bitcoin
BITCOIN_ROOT=$(pwd)
wget http://download.oracle.com/berkeley-db/db-4.8.30.NC.tar.gz
tar -xvf db-4.8.30.NC.tar.gz
cd db-4.8.30.NC/build_unix
BDB_PREFIX=$(pwd -P)/build
mkdir -p ${BDB_PREFIX}
../dist/configure --disable-shared --enable-cxx --with-pic --
prefix=${BDB_PREFIX}
make install
cd $BITCOIN_ROOT
```



To build with Qt 5 (recommended)  
you need the following:

```
sudo apt-get install libqt5gui5 libqt5core5a libqt5dbus5 qttools5-dev  
qttools5-dev-tools libprotobuf-dev protobuf-compiler
```

# build

- ./autogen.sh
- ./configure or ./configure —disable-wallet **for Ubuntu**
- ./configure LDFLAGS="-L\${BDB\_PREFIX}/lib/"  
CPPFLAGS="-I\${BDB\_PREFIX}/include/" **for Debian**
- make or make -j4
- sudo make install

# run bitcoind

- which bitcoind
- bitcoind -daemon
- bitcoin-cli getinfo
- bitcoin-cli help
- bitcoin-qt

```
[rslu@rslu-virtual-machine:~$ which bitcoind
/usr/local/bin/bitcoind
[rslu@rslu-virtual-machine:~$ bitcoind -daemon
Bitcoin server starting
[rslu@rslu-virtual-machine:~$ which bitcoin-cli
/usr/local/bin/bitcoin-cli
[rslu@rslu-virtual-machine:~$ bitcoin-cli getinfo
{
  "version": 140100,
  "protocolversion": 70015,
  "walletversion": 130000,
  "balance": 0.00000000,
  "blocks": 399,
  "timeoffset": -1,
  "connections": 4,
  "proxy": "",
  "difficulty": 1,
  "testnet": false,
  "keypoololdest": 1497260681,
  "keypoolsize": 100,
  "paytxfee": 0.00000000,
  "relayfee": 0.00001000,
  "errors": ""
}
```

# 檢查bitcoind是否運行正常？

- `ps -A | grep bitcoind`
- `lsof -i:8333`
- 重啟 `bitcoind -daemon -reindex` (重新對交易數據進行索引)

# ./autogen

```
rslu@rslu-virtual-machine:~/bitcoin0.14.1$ ./autogen.sh
configure.ac:28: installing 'build-aux/install-sh'
configure.ac:28: installing 'build-aux/missing'
Makefile.am:8: error: Libtool library used but 'LIBTOOL' is undefined
Makefile.am:8: The usual way to define 'LIBTOOL' is to add 'LT_INIT'
Makefile.am:8: to 'configure.ac' and run 'aclocal' and 'autoconf' again.
Makefile.am:8: If 'LT_INIT' is in 'configure.ac', make sure
Makefile.am:8: its definition is in aclocal's search path.
Makefile.am: installing 'build-aux/depcomp'
/usr/share/automake-1.15/am/depend2.am: error: am__fastdepCXX does not appear in AM_CONDITIONAL
/usr/share/automake-1.15/am/depend2.am: The usual way to define 'am__fastdepCXX' is to add 'AC_PROG_CXX'
/usr/share/automake-1.15/am/depend2.am: to 'configure.ac' and run 'aclocal' and 'autoconf' again
/usr/share/automake-1.15/am/depend2.am: error: AMDEP does not appear in AM_CONDITIONAL
/usr/share/automake-1.15/am/depend2.am: The usual way to define 'AMDEP' is to add one of the compiler tests
/usr/share/automake-1.15/am/depend2.am: AC_PROG_CC, AC_PROG_CXX, AC_PROG_OBJC, AC_PROG_OBJCXX,
/usr/share/automake-1.15/am/depend2.am: AM_PROG_AS, AM_PROG_GCJ, AM_PROG_UPC
/usr/share/automake-1.15/am/depend2.am: to 'configure.ac' and run 'aclocal' and 'autoconf' again
Makefile.am: error: C++ source seen but 'CXX' is undefined
Makefile.am: The usual way to define 'CXX' is to add 'AC_PROG_CXX'
Makefile.am: to 'configure.ac' and run 'autoconf' again.
parallel-tests: installing 'build-aux/test-driver'
autoreconf: automake failed with exit status: 1
```



# bitcoin-cli 指令參數

```
[rs1u@rs1u-virtual-machine:~$ bitcoin-cli help
```

```
== Blockchain ==
getbestblockhash
getblock "blockhash" ( verbose )
getblockchaininfo
getblockcount
getblockhash height
getblockheader "hash" ( verbose )
getchaintips
getdifficulty
getmempoolancestors txid (verbose)
getmempooldescendants txid (verbose)
getmempoolentry txid
getmempoolinfo
getrawmempool ( verbose )
gettxout "txid" n ( include_mempool )
gettxoutproof ["txid",...] ( blockhash )
gettxoutsetinfo
preciousblock "blockhash"
pruneblockchain
verifychain ( checklevel nblocks )
verifytxoutproof "proof"

== Control ==
getinfo
getmemoryinfo
help ( "command" )
stop

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

== Mining ==
getblocktemplate ( TemplateRequest )
getmininginfo
getnetworkhashps ( nblocks height )
prioritisetransaction <txid> <priority delta> <fee delta>
submitblock "hexdata" ( "jsonparametersobject" )

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

== Rawtransactions ==

createrawtransaction [{"txid":"id","vout":n},...] {"address":amount,"data":"hex",...} ( locktime )

decoderawtransaction "hexstring"

decodescript "hexstring"

fundrawtransaction "hexstring" ( options )

getrawtransaction "txid" ( verbose )

sendrawtransaction "hexstring" ( allowhighfees )

signrawtransaction "hexstring" ( [{"txid":"id","vout":n,"scriptPubKey":"hex","redeemScript":"hex"},...] ["privatekey1",...] sighashtype )

== Util ==

createmultisig nrequired ["key",...]

estimatefee nblocks

estimatepriority nblocks

estimatesmartfee nblocks

estimatesmartpriority nblocks

signmessagewithprivkey "privkey" "message"

validateaddress "address"

verifymessage "address" "signature" "message"

== Wallet ==

abandontransaction "txid"

addmultisigaddress nrequired ["key",...] ( "account" )

addwitnessaddress "address"

backupwallet "destination"

bumpfee "txid" ( options )

dumpprivkey "address"

dumpwallet "filename"

encryptwallet "passphrase"

getaccount "address"

getaccountaddress "account"

getaddressesbyaccount "account"

getbalance ( "account" minconf include\_watchonly )



```
getnewaddress ( "account" )
getrawchangeaddress
getreceivedbyaccount "account" ( minconf )
getreceivedbyaddress "address" ( minconf )
gettransaction "txid" ( include_watchonly )
getunconfirmedbalance
getwalletinfo
importaddress "address" ( "label" rescan p2sh )
importmulti "requests" "options"
importprivkey "bitcoinprivkey" ( "label" ) ( rescan )
importprunedfunds
importpubkey "pubkey" ( "label" rescan )
importwallet "filename"
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)
listunspent ( minconf maxconf ["addresses",...] [include_unsafe] )
lockunspent unlock ([{"txid":"txid","vout":n},...])
move "fromaccount" "toaccount" amount ( minconf "comment" )
removeprunedfunds "txid"
sendfrom "fromaccount" "toaddress" amount ( minconf "comment" "comment_to" )
sendmany "fromaccount" {"address":amount,...} ( minconf "comment" ["address",...] )
sendtoaddress "address" amount ( "comment" "comment_to" subtractfeefromamount )
setaccount "address" "account"
settxfee amount
signmessage "address" "message"
```



# 期中考

編譯出20170801從Bitcoin Core分裂出的另一套版本的軟體-  
BitcoinABC

自行編譯

BitcoinABC

# download BitcoinABC 0.14.6

- `git clone -b v0.14.6 https://github.com/Bitcoin-ABC/bitcoin-abc`
- 或者是
- `git clone -b v0.14.6 https://github.com/Bitcoin-ABC/bitcoin-abc bitcoinABC0.14.6`

# install automake

1. **sudo apt-get install autoconf** (will install 4 packages including autoconf automake autotools-dev m4)
2. **sudo apt-get install build-essential libtool autotools-dev automake pkg-config libssl-dev libevent-dev bsdmainutils**
3. **sudo apt-get install libboost-all-dev** 安裝boost函式套件

# BerkeleyDB is required for the wallet

install BerkeleyDB 4.8 for **Ubuntu**

1. `sudo apt-get install software-properties-common`
2. `sudo add-apt-repository ppa:bitcoin/bitcoin`
3. **`sudo apt-get update`**
4. `sudo apt-get install libdb4.8-dev libdb4.8++-dev`

## **For Debian**

```
cd bitcoin
BITCOIN_ROOT=$(pwd)
wget http://download.oracle.com/berkeley-db/db-4.8.30.NC.tar.gz
tar -xvf db-4.8.30.NC.tar.gz
cd db-4.8.30.NC/build_unix
BDB_PREFIX=$(pwd -P)/build
mkdir -p ${BDB_PREFIX}
../dist/configure --disable-shared --enable-cxx --with-pic --
prefix=${BDB_PREFIX}
make install
cd $BITCOIN_ROOT
```

To build with Qt 5 (recommended)  
you need the following:

```
sudo apt-get install libqt5gui5 libqt5core5a libqt5dbus5 qttools5-dev  
qttools5-dev-tools libprotobuf-dev protobuf-compiler
```

# build

- ./autogen.sh
- ./configure or ./configure —disable-wallet **for Ubuntu**
- ./configure LDFLAGS="-L\${BDB\_PREFIX}/lib/"  
CPPFLAGS="-I\${BDB\_PREFIX}/include/" **for Debian**
- make or make -j4
- sudo make install

# run bitcoind

- which bitcoind
- bitcoind -daemon
- bitcoin-cli getinfo
- bitcoin-cli help
- bitcoin-qt

```
[rslu@rslu-virtual-machine:~$ which bitcoind
/usr/local/bin/bitcoind
[rslu@rslu-virtual-machine:~$ bitcoind -daemon
Bitcoin server starting
[rslu@rslu-virtual-machine:~$ which bitcoin-cli
/usr/local/bin/bitcoin-cli
[rslu@rslu-virtual-machine:~$ bitcoin-cli getinfo
{
  "version": 140100,
  "protocolversion": 70015,
  "walletversion": 130000,
  "balance": 0.00000000,
  "blocks": 399,
  "timeoffset": -1,
  "connections": 4,
  "proxy": "",
  "difficulty": 1,
  "testnet": false,
  "keypoololdest": 1497260681,
  "keypoolsize": 100,
  "paytxfee": 0.00000000,
  "relayfee": 0.00001000,
  "errors": ""
}
```



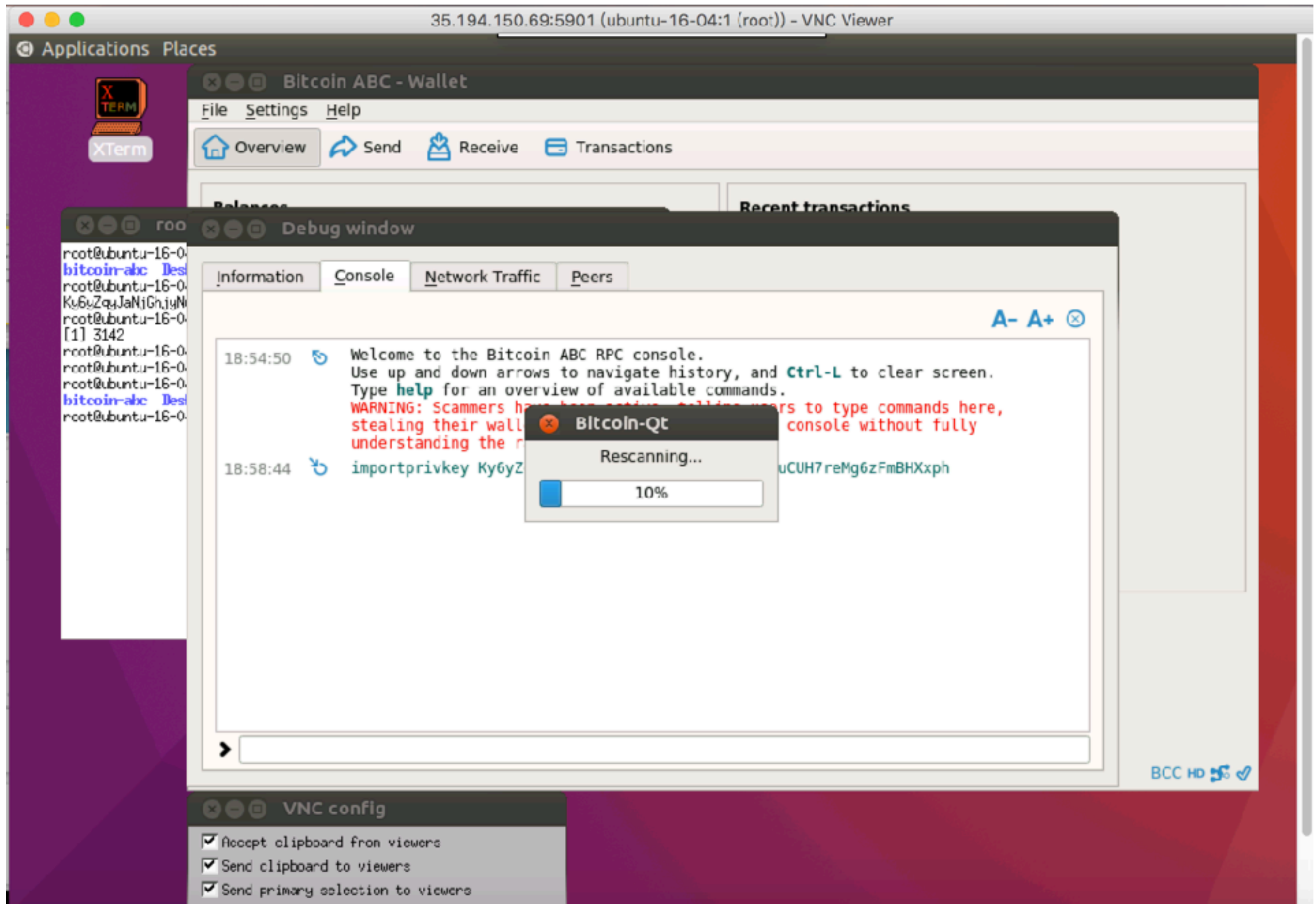
# 檢查bitcoind是否運行正常？

- `ps -A | grep bitcoind`
- `lsof -i:8333`
- 重啟 `bitcoind -daemon -reindex` (重新對交易數據進行索引)

將原本在BTC上的錢包  
地址，匯入Bitcoin ABC

1APtYTnTxVG1HEt9b7V4pyA4JDgwtDqjvV

# bitcoinABC import privatekey





XTerm

## Bitcoin ABC - Wallet

File Settings Help

Overview Send Receive Transactions

**Balances**Available: **0.02000000 BCC**Pending: **0.00000000 BCC**Total: **0.02000000 BCC****Recent transactions**

7/10/17 00:51

+0.02000000 BCC

(1APtYTnTxVG1HEt9b7V4pyA4JDgwtDqjvV)



5/15/17 05:27

**-0.04300000 BCC**

(1A45Y3T9zEMSiCNuy4vhwWqUKNwVc6DnnW)



12/19/16 06:49

+0.00300000 BCC

(1APtYTnTxVG1HEt9b7V4pyA4JDgwtDqjvV)



12/19/16 06:49

+0.00300000 BCC

(1APtYTnTxVG1HEt9b7V4pyA4JDgwtDqjvV)



12/19/16 06:49

+0.00300000 BCC

(1APtYTnTxVG1HEt9b7V4pyA4JDgwtDqjvV)



UXTerm

## Bitcoin Core - 錢包

檔案 設定 說明



總覽



付款



收款



交易

## 餘額

可用金額: **0.02000000 BTC**未定金額: **0.00000000 BTC**總金額: **0.02000000 BTC**

## 最近的交易



7/10/17 00:51

+0.02000000 BTC

(1APtYTnTxVG1HEt9b7V4pyA4jDgwtDqjvV)



UXTerm

Bitcoin Core - 錢包

檔案 設定 說明



總覽



付款



收款



交易

除錯視窗

資訊

主控台

網路流量

節點

**普通**

客戶端軟體版本	v0.14.1
使用者代理	/Satoshi:0.14.1/
使用 BerkeleyDB 版本	Berkeley DB 4.8.30: (April 9, 2010)
資料目錄	/home/rsly/.bitcoin
啟動時間	Fri Aug 11 10:08:45 2017

**網路**

名稱	main
連線數	8 (來: 0 / 去: 8)

**區塊鏈**

目前區塊數	480079
最近區塊時間	Fri Aug 11 12:20:31 2017

**記憶體暫存池**

目前交易數目	1633
記憶體使用量	3.21 MB

除錯紀錄檔

開啟

Applications Places



UXTerm

Bitcoin Core - 钱包

檔案 設定 說明

除錯視窗

資訊

主控台

網路流量

節點

**普通**

客戶端軟體版本	v0.14.1
使用者代理	/Satoshi:0.14.1/
使用 BerkeleyDB 版本	Berkeley DB 4.8.30: (April 9, 2010)
資料目錄	/home/rslu/.bitcoin
啟動時間	Fri Aug 11 10:08:45 2017

**網路**

名稱	main
連線數	8 (來: 0 / 去: 8)

**區塊鏈**

目前區塊數	480130
最近區塊時間	Fri Aug 11 20:50:53 2017

**記憶體暫存池**

目前交易數目	13046
記憶體使用量	29.10 MB

除錯紀錄檔

開啟

+0.02  
tDqjvV





XTerm

## Bitcoin ABC - Wallet

File Settings Help

[Overview](#) [Send](#) [Receive](#) [Transactions](#)

## Debug window

[Information](#) [Console](#) [Network Traffic](#) [Peers](#)**General**

Client version v0.14.6.0-a46e63c  
User Agent /Bitcoin ABC:0.14.6(EB8.0)/  
Using BerkeleyDB version Berkeley DB 4.8.30: (April 9, 2010)  
Datadir /home/ta01/.bitcoin  
Startup time Thu Aug 10 23:21:36 2017

**Network**

Name main  
Number of connections 8 (In: 0 / Out: 8)

**Block chain**

Current number of blocks 479065  
Last block time Fri Aug 11 21:57:27 2017

**Memory Pool**

Current number of transactions 87  
Memory usage 128.19 KB

Debug log file

[Open](#)

Amount  
Amount (BCC) ▲  
2000000  
4300000  
0300000  
0300000  
0300000  
0300000  
0300000  
0300000  
2800000  
1280000  
0010000  
1290000  
0976000  
0000063

[Export](#)

BCC HD



