

## How to run PCHAIN with MAC

### 1. Mac configuration

Mac: (brew is a tool like ubuntu apt-get, which used to install software on the terminal)

Install Homebrew :

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

```
PCHAINdeMacBook-Air:Desktop pchain$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
==> This script will install:
/usr/local/bin/brew
/usr/local/share/doc/homebrew
/usr/local/share/man/man1/brew.1
/usr/local/share/zsh/site-functions/_brew
/usr/local/etc/bash_completion.d/brew
/usr/local/Homebrew

Press RETURN to continue or any other key to abort
==> Downloading and installing Homebrew...
remote: Enumerating objects: 116774, done.
remote: Total 116774 (delta 0), reused 0 (delta 0), pack-reused 116774
Receiving objects: 100% (116774/116774), 27.47 MiB | 2.79 MiB/s, done.
Resolving deltas: 100% (85247/85247), done.
```

Mac:

```
brew update && brew upgrade
```

```
brew install git
```

```
brew install go
```

```
PCHAINdeMacBook-Air:Desktop pchain$ brew install go
==> Downloading https://homebrew.bintray.com/bottles/go-1.11.4.high_sierra.bottl
##### 100.0%
==> Pouring go-1.11.4.high_sierra.bottle.tar.gz
==> Caveats
A valid GOPATH is required to use the `go get` command.
If $GOPATH is not specified, $HOME/go will be used by default:
  https://golang.org/doc/code.html#GOPATH

You may wish to add the GOROOT-based install location to your PATH:
  export PATH=$PATH:/usr/local/opt/go/libexec/bin
==> Summary
📦 /usr/local/Cellar/go/1.11.4: 9,298 files, 404.3MB
```

### 2、Check git:

```
PCHAINdeMacBook-Air:Desktop pchain$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        <command> [<args>]
```

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)

clone	Clone a repository into a new directory
init	Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)

add	Add file contents to the index
mv	Move or rename a file, a directory, or a symlink
reset	Reset current HEAD to the specified state
rm	Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)

bisect	Use binary search to find the commit that introduced a bug
grep	Print lines matching a pattern
log	Show commit logs
show	Show various types of objects
status	Show the working tree status

## Check go env

```
PCHAINdeMacBook-Air:~ pchain$ go env
GOARCH="amd64"
GOBIN=""
GOCACHE="/Users/pchain/Library/Caches/go-build"
GOEXE=""
GOFLAGS=""
GOHOSTARCH="amd64"
GOHOSTOS="darwin"
GOOS="darwin"
GOPATH="/Users/pchain/go"
GOPROXY=""
GORACE=""
GOROOT="/usr/local/Cellar/go/1.11.4/libexec"
GOTMPDIR=""
GOTOOLDIR="/usr/local/Cellar/go/1.11.4/libexec/pkg/tool/darwin_amd64"
GCCGO="gccgo"
CC="clang"
CXX="clang++"
CGO_ENABLED="1"
GOMOD=""
CGO_CFLAGS="-g -O2"
CGO_CPPFLAGS=""
CGO_CXXFLAGS="-g -O2"
CGO_FFLAGS="-g -O2"
CGO_LDFLAGS="-g -O2"
PKG_CONFIG="pkg-config"
GOGCCFLAGS="-fPIC -m64 -pthread -fno-caret-diagnostics -Qunused-arguments -fmessage-length=0 -fdebug-prefix-map=/var/folders/hz/g_5v9_193cq5rcyc45h5jff2m0000gn/T/go-build960257590=/tmp/go-build -gno-record-gcc-switches -fno-common"
```

## 3、Config go

1) open the terminal, cd ~, then enter home directory

2) Is `--all` check all documents, if you can find document with `.bash_profile`

3) If you cannot find, then create new with `touch .bash_profile`

4) If yes, please edit with `vi .bash_profile`

View the installation directory, add all the following configurations

```
export GOROOT=/usr/local/Cellar/go
```

```
export PATH=$GOROOT/bin:$PATH
```

```
export PATH=$PATH:$GOBIN:$GOROOT/bin
```

5) Save the edit, after run `source ~/.bash_profile` you can finish the configuration

6) Check `go env`

Overview:

**GOPATH:** the common directory which is used frequently. It contains 3 folders `src`、`pkg`、`bin`, `src` is created by ourselves, `pkg` and `bin` are created automatically. Kindly suggest that do not install `GOPATH` in the same directory with `GO`.

1. `src`: Source code
2. `bin`: Store the executable file generated after compilation, you can execute it yourself
3. `pkg`: Compiled file (`.a` file) (files other than main function are generated after go install)

**GOBIN:** the directory of `bin` under `GOPATH`

**PATH:** You need to add the `go-bin` directory to the path, than you can run the executable file directly.

4. Pchain download and edit

Create folders:

```
mkdir pchain
```

```
mkdir pchain/bin
```

```
mkdir pchain/src
```

```
mkdir pchain/src/github.com
```

```
cd pchain/src/github.com
```

Source Code:

**Notice:** source has to be under current directory.

```
cd pchain/src/github.com/
```

```
git clone https://github.com/pchain-org/pchain.git
```

```
PCHAINdeMacBook-Air:Desktop pchain$ git clone https://github.com/pchain-org/pchain.git
Cloning into 'pchain'...
remote: Enumerating objects: 561, done.
remote: Counting objects: 100% (561/561), done.
remote: Compressing objects: 100% (389/389), done.
remote: Total 22773 (delta 299), reused 311 (delta 163), pack-reused 22212
Receiving objects: 100% (22773/22773), 56.29 MiB | 284.00 KiB/s, done.
Resolving deltas: 100% (13561/13561), done.
Checking out files: 100% (5267/5267), done.
```

Copy the build.sh script file to the pchain directory:

```
cp pchain/src/github.com/pchain/restore/build.sh pchain/
```

```
cd pchain
```

build the pchain:

```
./build.sh
```

```
[PCHAINdeMacBook-Air:pchain_dev pchain$ ./build.sh
/Users/pchain/Desktop/pchain_dev
/Users/pchain/Desktop/pchain_dev
/Users/pchain/Desktop/pchain_dev/src/github.com/pchain
go build -o /Users/pchain/Desktop/pchain_dev/bin/pchain ./cmd/
```

After compile, you will find the binary executable file pchain under pchain/bin/

## 5. Run Pchain

```
cd pchain/bin/
```

```
chmod +x pchain
```

1) Init the local account:

```
./pchain --datadir .pchain init_eth_genesis "{10000000000000000000,100}"
```

```
[PCHAINdeMacBook-Air:bin pchain$ ls
pchain
[PCHAINdeMacBook-Air:bin pchain$ ./pchain --datadir .pchain init_eth_genesis "{10000000000000000000,100}"
INFO [01-09|20:06:14.420] this is init_eth_genesis
INFO [01-09|20:06:15.429] createPriValidators          account:=0x9da538D8A08B09991fEE6445487C4cBFCc9f
944B cwd:=pchain
```

Check

```
[PCHAINdeMacBook-Air:bin pchain$ ls -all
total 84432
drwxrwxrwx  5 pchain  staff      160  1  9 20:06 .
drwxr-xr-x  6 pchain  staff      192  1  9 20:04 ..
drwx-----  4 pchain  staff      128  1  9 20:06 .pchain
drwx-----  3 pchain  staff        96  1  9 20:06 log
-rwxr-xr-x  1 pchain  staff 42727728  1  9 20:05 pchain
```

2) 1) Init the local pchain:

```
./pchain --datadir .pchain init .pchain/pchain/eth_genesis.json
```

```
PCHAINdeMacBook-Air:bin pchain$ ./pchain --datadir .pchain init .pchain/pchain/eth_genesis.json
[INFO [01-09|20:08:04.810] init_eth_blockchain 0 with dbPath: .pchain/pchain/geth/chaindata
[INFO [01-09|20:08:04.810] Allocated cache and file handles          database=.pchain/pchain/geth/chaindata cache=16
handles=16
[INFO [01-09|20:08:04.813] init_eth_blockchain 1
[INFO [01-09|20:08:04.813] init_eth_blockchain 2
[INFO [01-09|20:08:04.813] Writing custom genesis block
[INFO [01-09|20:08:04.814] Persisted trie from memory database      nodes=1 size=271.00B time=94.672µs gcnodes=0 gc
size=0.00B gctime=0s livenodes=1 livesize=0.00B
[INFO [01-09|20:08:04.814] init_eth_blockchain end
[INFO [01-09|20:08:04.814] successfully wrote genesis block and/or chain rule set: b27ffcf18db20cf4333219e9f1cef1cb
09da049bc86493d833b63dac96528251
[INFO [01-09|20:08:04.814] Database closed                  database=.pchain/pchain/geth/chaindata
[INFO [01-09|20:08:04.815] checkAccount(), coinbase is 9da538d8a08b09991fee6445487c4cbfcc9f944b
[INFO [01-09|20:08:04.815] checkAccount(), address is 9da538d8a08b09991fee6445487c4cbfcc9f944b, balance is 100000000
0000000000000000, amount is 100
```

### 3) Run pchain

Copy run.sh under pchain directory then run ./run.sh

Mining mode: (Default add --mine)

```
./pchain --datadir .pchain --networkid=10000 --port=30201 --syncmode "full" --mine
--rpc --rpcport 6969 --rpcaddr=0.0.0.0 --rpccorsdomain=* --rpcapi
"eth,web3,personal,net,admin,miner,tdm" --verbosity=4 > pchain.log 2>&1 &
```

### Non-mining mode:

```
./pchain --datadir .pchain --networkid=10000 --port=30201 --syncmode "full" --rpc --
rpcport 6969 --rpcaddr=0.0.0.0 --rpccorsdomain=* --rpcapi
"eth,web3,personal,net,admin,miner,tdm" --verbosity=4 > pchain.log 2>&1 &
```

Check log

```
tail -f pchain.log
```

```
PCHAINdeMacBook-Air:bin pchain$ ./pchain --datadir .pchain --networkid=10000 --port=30201 --rpc --rpcport=6969 --r
pcaddr=0.0.0.0 --rpccorsdomain=* --rpcapi=db,eth,net,web3,admin,tdm,miner,person,chain,txpool --targetgaslimit=971
23880000 --syncmode "full" --gcmode=archive --verbosity=4 > pchain.log 2>&1 &
INFO [01-09|20:14:05.006] Starting PChain...
INFO [01-09|20:14:05.006] PChain supports large scale block-chain applications with multi-chain
INFO [01-09|20:14:05.007] Create peer-to-peer node instance=pchain/darwin-amd64/go1.11.4
INFO [01-09|20:14:05.007] ethereum.MakeSystemNode
INFO [01-09|20:14:05.021] Allocated cache and file handles database=/Users/pchain/Desktop/pchain_dev/bin/.
pchain/pchain/gets/chaindata cache=768 handles=128
WARN [01-09|20:14:05.037] Upgrading database to use lookup entries
INFO [01-09|20:14:05.037] Database deduplication successful deduped=0
INFO [01-09|20:14:05.037] Initialised chain configuration config="{PChainId: pchain ChainID: 241608434543
25667600331855523506733810605584168331177014437733538279768116753 Homestead: 0 DAO: <nil> DAOsupport: false EIP150
: 0 EIP155: 0 EIP158: 0 Byzantium: 0 Constantinople: <nil> Engine: tendermint}"
INFO [01-09|20:14:05.042] Tendermint (backend) Protocol, add logic here
INFO [01-09|20:14:05.043] Initialising Ethereum protocol versions=[64] network=10000
INFO [01-09|20:14:05.043] Loaded most recent local header number=0 hash=b27ffc...528251 td=1024
INFO [01-09|20:14:05.043] Loaded most recent local full block number=0 hash=b27ffc...528251 td=1024
INFO [01-09|20:14:05.043] Loaded most recent local fast block number=0 hash=b27ffc...528251 td=1024
INFO [01-09|20:14:05.044] Regenerated local transaction journal transactions=0 accounts=0
INFO [01-09|20:14:05.044] Tendermint (backend) SetBroadcaster: 0xc0001a0000
INFO [01-09|20:14:05.044] Tendermint (backend) Protocol, add logic here
INFO [01-09|20:14:05.047] Allocated cache and file handles database=.pchain/tx3cache
cache=16 handles=16
INFO [01-09|20:14:05.050] Starting P2P networking
```