INFINITY
BLOCKCHAIN LABS

Ethereum

for noobs

go-ethereum codebase

Hội anh em blockchain "thiện lành"

Quoc Phu

phu.kq

phukq@blockchainlabs.asia

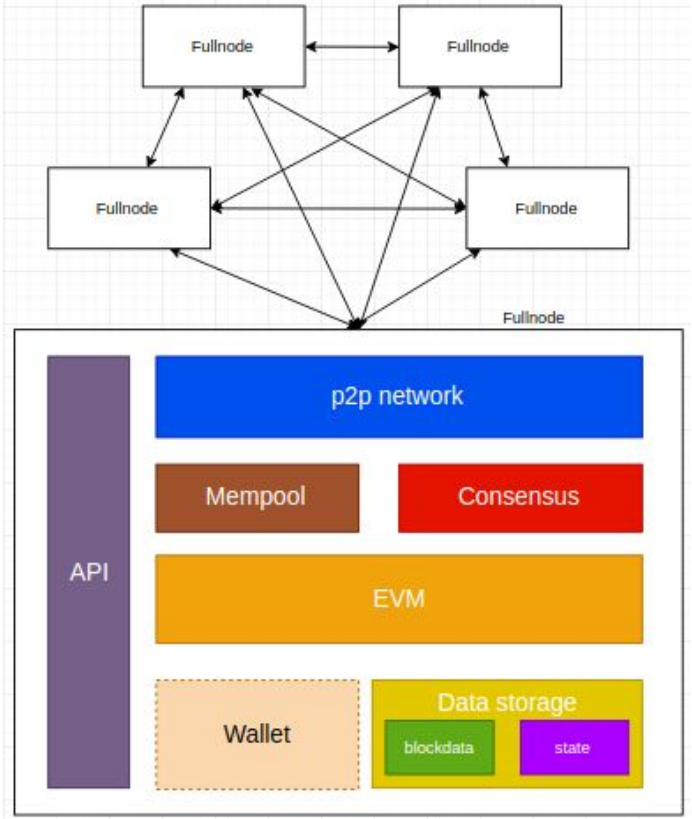
- +7 years experience as software developer.
- +3 years working in blockchain area.
- Enterprise Architecture team at IBL.

Contents

- Project structure.
- Building & modifying source code.
- Running a local network.

Components

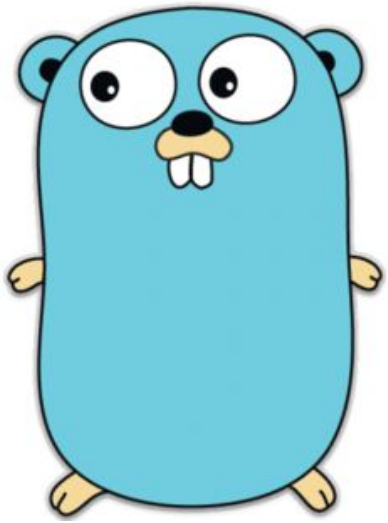
- **p2p network**: Responsible for the propagation of transactions & other messages.
- **Mempool**: Responsible for storing unconfirmed transactions.
- **Consensus**: Running consensus protocol.
- **EVM**: Processing transaction, maintain state.
- **Data storage**: Storing all data of blockchain.
- **Api**: Providing api services: rpc, ipc, ws
- **Wallet**: Responsible for key management.

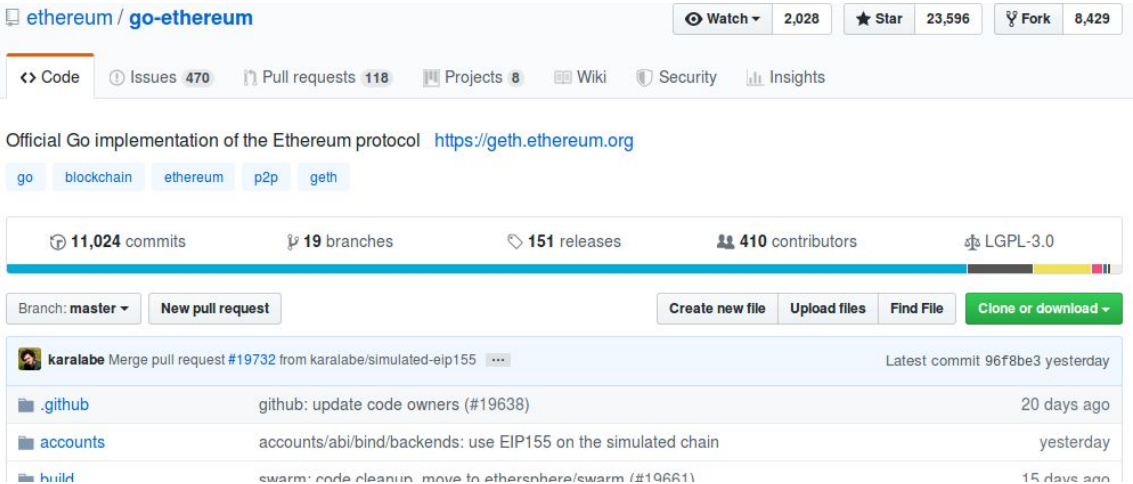


go-ethereum codebase

https://github.com/ethereum/go-ethereum

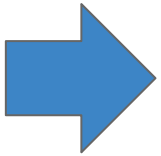
maintained by ETHDEV organisation





Source code structure

```
accounts
appveyor.yml
AUTHORS
build
circle.yml
cmd
common
consensus
console
containers
contracts
COPYING
COPYING.LESSER
core
crypto
dashboard
Dockerfile
Dockerfile.alltools
eth
ethclient
ethdb
ethstats
event
interfaces.go
internal
les
light
log
Makefile
metrics
miner
mobile
node
p2p
params
README.md
rlp
rpc
signer
swarm
tests
trie
vendor
VERSION
whisper
```



```
cmd/
├── abigen
├── bootnode
├── clef
├── ethkey
├── evm
├── faucet
├── geth
├── internal
├── p2psim
├── puppeth
├── rlpdump
├── swarm
├── utils
└── wnode
```

```
./miner/
├── miner.go
├── stress_clique.go
├── stress_ethash.go
├── unconfirmed.go
├── unconfirmed_test.go
├── worker.go
└── worker_test.go
```

```
consensus/
├── clique
├── consensus.go
├── errors.go
├── ethash
└── misc
```

```
eth/
├── api_backend.go
├── api.go
├── api_test.go
├── api_tracer.go
├── backend.go
├── bloombits.go
├── config.go
├── downloader
├── fetcher
├── filters
├── gasprice
├── gen_config.go
├── handler.go
├── handler_test.go
├── helper_test.go
├── metrics.go
├── peer.go
├── protocol.go
├── protocol_test.go
├── sync.go
├── sync_test.go
└── tracers
```

```
core/
├── asm
├── bench_test.go
├── blockchain.go
├── blockchain_insert.go
├── blockchain_test.go
├── blocks.go
├── block_validator.go
├── block_validator_test.go
├── bloombits
├── chain_indexer.go
├── chain_indexer_test.go
├── chain_makers.go
├── chain_makers_test.go
├── dao_test.go
├── error.go
├── events.go
├── evm.go
├── gaspool.go
├── genesis_alloc.go
├── genesis.go
├── genesis_test.go
├── gen_genesis_account.go
├── gen_genesis.go
├── headerchain.go
├── helper_test.go
├── mkalloc.go
├── rawdb
├── state
├── state_processor.go
├── state_transition.go
├── tx_cacher.go
├── tx_journal.go
├── tx_list.go
├── tx_list_test.go
├── tx_pool.go
├── tx_pool_test.go
├── types
├── types.go
└── vm
```

Source code structure

```
Welcome to the Geth JavaScript console!

instance: Geth/v1.8.27-stable/linux-amd64/go1.12.3
coinbase: 0x4678fd8728fdbfda10dc2cdf46f5efe30e7ab9b4
at block: 1018 (Thu, 20 Jun 2019 10:41:20 +07)
 datadir: /home/phu/Documents/Code/go/src/github.com/e
 modules: admin:1.0 clique:1.0 debug:1.0 eth:1.0 miner

> eth.getBalance(eth.accounts[0])
9.046256971665327767466483203803742801036717552003169
> eth.getTransactionCount(eth.accounts[0])
0
> var x = 10;
```

- Javascript runtime & interpreter: github.com/robertkrimen/otto
- Console: github.com/peterh/liner
- Command: github.com/urfave/cli
- Database: github.com/syndtr/goleveI
- ...

How to start?

Get the source code from github & put it in $GOPATH dir

```
git clone https://github.com/ethereum/go-ethereum.git
$GOPATH/src/github.com/ethereum/go-ethereum
```

Compile the source code

```
make all
```

```
ls build/bin/
abigen     ethkey     faucet        mimegen   rlpdump      swarm-smoke
bootnode   evm        geth          p2psim    simulations  swarm-snapshot
clef       examples   global-store  puppeth   swarm        wnode
```

How to start?

File: $GOPATH/src/github.com/ethereum/go-ethereum/cmd/geth/main.go

```
255  func main() {
256      fmt.Println("Welcome to 'Hội anh em blockchain thiện lành tại IBL'")
257      if err := app.Run(os.Args); err != nil {
258          fmt.Fprintln(os.Stderr, err)
259          os.Exit(1)
260      }
261  }
```

Build: `make geth`

```
phu@phu-Latitude-E5450:~/Documents/Code/go/src/github.com/ethereum/go-ethereum$ make geth
build/env.sh go run build/ci.go install ./cmd/geth
>>> /usr/local/go/bin/go install -v ./cmd/geth
github.com/ethereum/go-ethereum/cmd/geth
Done building.
```

Run: `./geth`

```
Welcome to 'Hội anh em blockchain thiện lành tại IBL'
INFO [06-20|11:00:21.301] Maximum peer count                       ETH=25 LES=0 total=25
WARN [06-20|11:00:21.303] Found deprecated node list file /home/phu/Documents/Code/go/src/github.com/eth
INFO [06-20|11:00:21.303] Starting peer-to-peer node               instance=Geth/v1.8.27-stable/linux-am
INFO [06-20|11:00:21.303] Allocated cache and file handles         database=/home/phu/Documents/Code/go/
INFO [06-20|11:00:21.341] Initialised chain configuration          config="{ChainID: 2206 Homestead: 1 D
ue}"
```

Create a local network

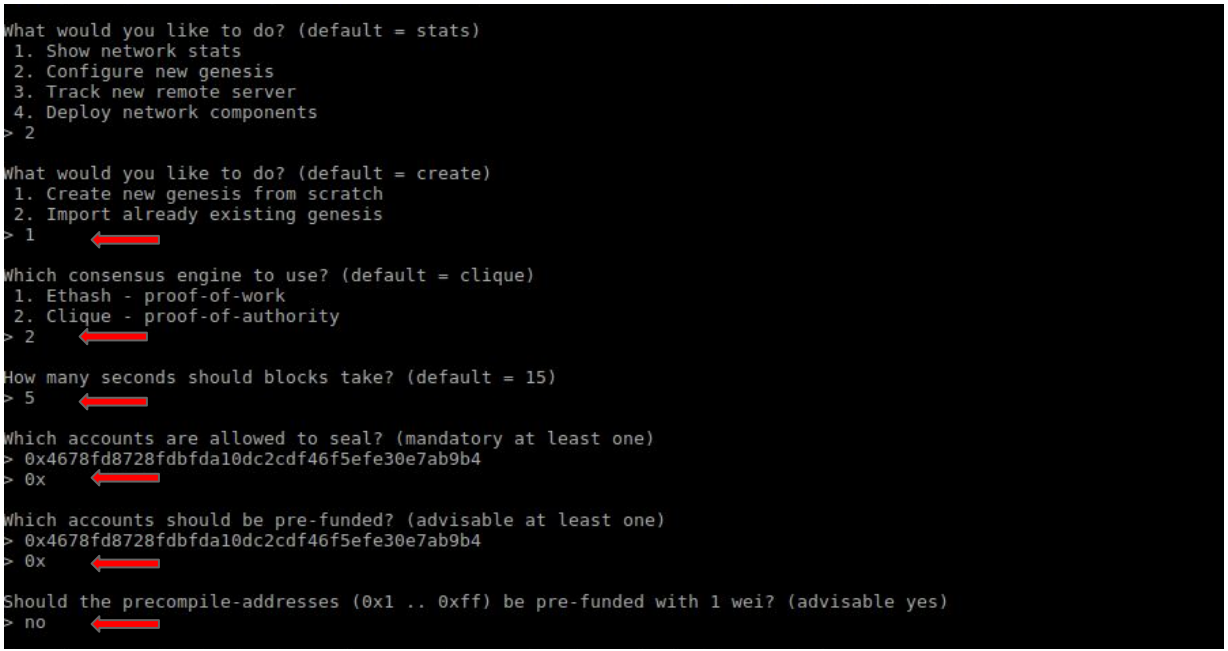
Using puppeth to create config file.

```
./puppeth
```

```
+-----------------------------------------------------------+
| Welcome to puppeth, your Ethereum private network manager |
|                                                           |
| This tool lets you create a new Ethereum network down to  |
| the genesis block, bootnodes, miners and ethstats servers |
| without the hassle that it would normally entail.         |
|                                                           |
| Puppeth uses SSH to dial in to remote servers, and builds |
| its network components out of Docker containers using the |
| docker-compose toolset.                                   |
+-----------------------------------------------------------+

Please specify a network name to administer (no spaces, hyphens or capital letters please)
> ethnoob

Sweet, you can set this via --network=ethnoob next time!

INFO [06-19|17:38:08.897] Administering Ethereum network           name=ethnoob
```

Create a local network

Using puppeth to create config file.



Create a local network

Using puppeth to create config file.

```
Specify your chain/network ID if you want an explicit one (default = random)
> 2206
INFO [06-19|17:39:00.079] Configured new genesis block

What would you like to do? (default = stats)
 1. Show network stats
 2. Manage existing genesis
 3. Track new remote server
 4. Deploy network components
> 2

 1. Modify existing fork rules
 2. Export genesis configurations
 3. Remove genesis configuration
> 2

Which folder to save the genesis specs into? (default = current)
  Will create ethnoob.json, ethnoob-aleth.json, ethnoob-harmony.json, ethnoob-parity.json
>
INFO [06-19|17:39:16.997] Saved native genesis chain spec          path=ethnoob.json
ERROR[06-19|17:39:16.997] Failed to create Aleth chain spec        err="unsupported consensus engine"
ERROR[06-19|17:39:16.997] Failed to create Parity chain spec       err="unsupported consensus engine"
INFO [06-19|17:39:16.997] Saved genesis chain spec                 client=harmony path=ethnoob-harmony.json
```

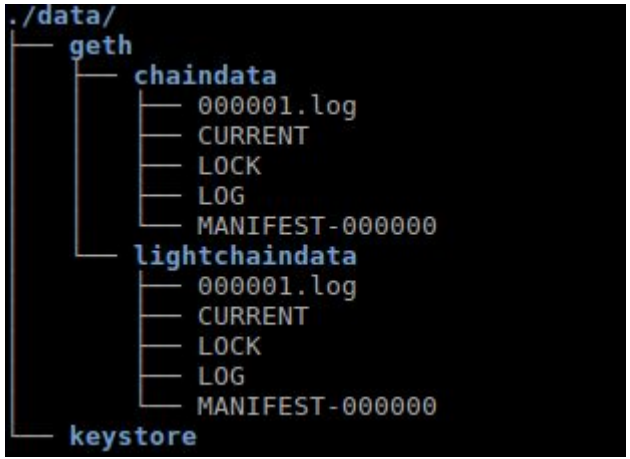
Create a local network

Using puppeth to create config file.


```
{
  "config": {
    "chainId": 2206,
    "homesteadBlock": 1,
    "eip150Block": 2,
    "eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "eip155Block": 3,
    "eip158Block": 3,
    "byzantiumBlock": 4,
    "constantinopleBlock": 5,
    "clique": {
      "period": 5,
      "epoch": 30000
    }
  },
  "nonce": "0x0",
  "timestamp": "0x5d0a1097",
  "extraData": "0x0000000000000000000000000000000000000000000000000000000000000000004678fd8728fdbfda10dc2cdf46f5efe30e7ab9b400000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000",
  "gasLimit": "0x47b760",
  "difficulty": "0x1",
  "mixHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase": "0x0000000000000000000000000000000000000000",
  "alloc": {
    "4678fd8728fdbfda10dc2cdf46f5efe30e7ab9b4": {
      "balance": "0x200000000000000000000000000000000000000000000000000000000000000000"
    }
  },
  "number": "0x0",
  "gasUsed": "0x0",
  "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000"
}
```

Create a local network

Init data: ./geth init --datadir ./data ./ethnoob.json



Create a local network

- Create a file acc.txt that contents a private key:
  "cc6d94090069d54bf9169a40312792e7ce5f09672bd640e5c2e3ef565771a97d"
- Import account: ./geth --datadir ./data account import ./acc.txt

```
INFO [06-19|18:03:36.549] Maximum peer count                       ETH=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this password.
Passphrase:
Repeat passphrase:
Address: {4678fd8728fdbfda10dc2cdf46f5efe30e7ab9b4}
```

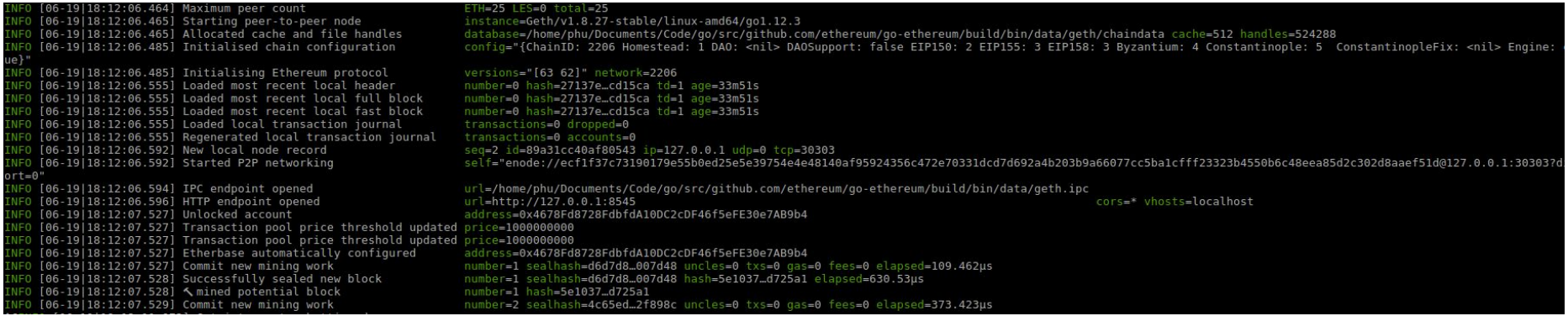
- Create a file pwd.txt that contents password which we've recently input.

```
├── acc.txt
├── data
│   ├── geth
│   └── keystore
│       └── UTC--2019-06-19T11-03-43.326781423Z--46
├── geth
└── pwd.txt
```

Create a local network

Start miner:

```
./geth --networkid 2206 --datadir "./data" --nodiscover --rpc --rpccorsdomain "*"
--rpcapi eth,net,web3,personal,clique,miner,admin,debug,txpool --unlock
'0x4678fd8728fdbfda10dc2cdf46f5efe30e7ab9b4' --password ./pwd.txt --mine
```

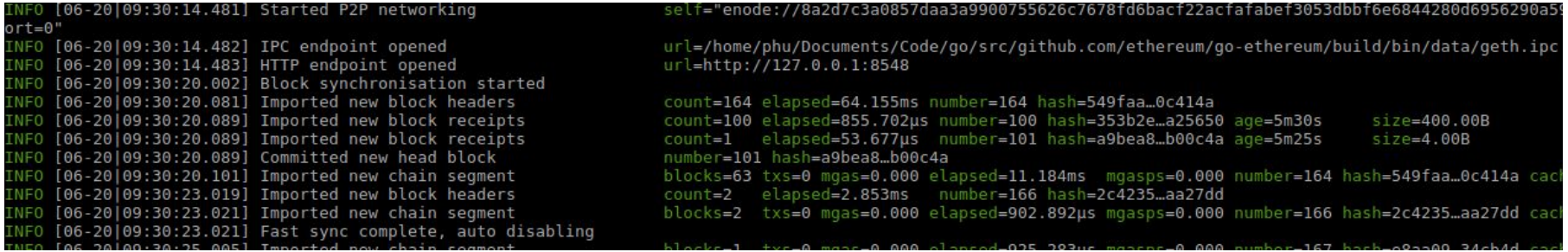


enode://ecf1f37c73190179e55b0ed25e5e39754e4e48140af95924356c472e70331dcd7d692a4b203b9a66077cc5ba1cfff23323b4550b6c48eea85d2c302d8aaef51d@127.0.0.1:30303

Create a local network

Connect to the local network

- Init data: `./geth init --datadir ./data  ./ethnoob.json`
- Create a file: **./data/geth/static-nodes.json** that contents id of the miner
  enode://ecf1f37c73190179e55b0ed25e5e39754e4e48140af95924356c472e70331dcd7d692a4b203b9a66077cc5ba1cfff23323b4550b6c48eea85d2c302d8aaef51d@127.0.0.1:30303
- Start new node:
  ```
  ./geth --networkid 2206 --datadir "./data" --nodiscover --rpc --rpccorsdomain "*" --rpcport 8548 --port 30304 --rpcapi "eth,net,web3,personal,clique,miner,admin,debug,txpool"
  ```



Thank you.