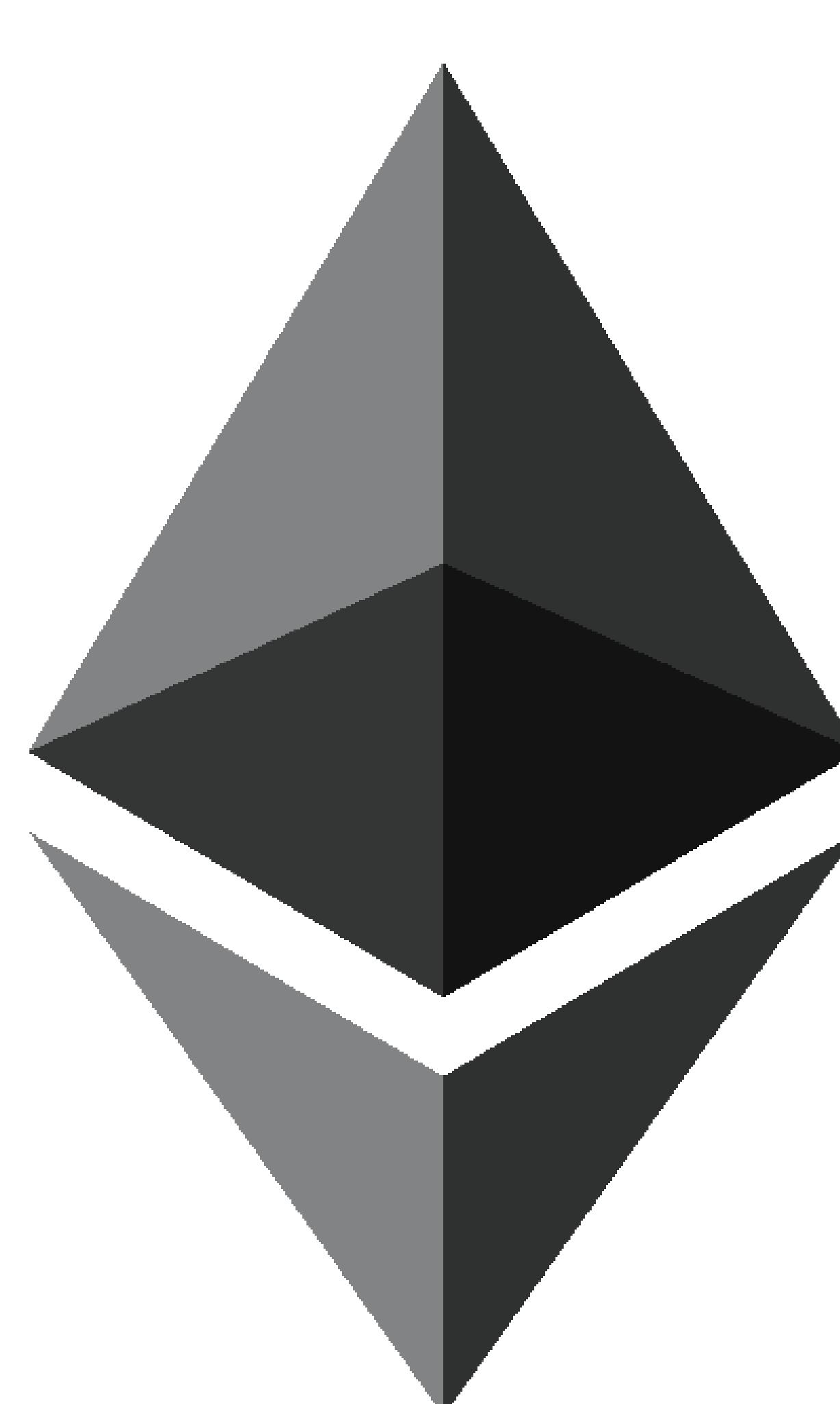




Private Ethereum Network

Dome Charoenyost

DOME CLOUD CO., LTD.



About

Own Private Ethereum Network

Blockchain จะมีบทบาทในองค์กร เมื่อบน No-SQL server เราจะได้เห็น Blockchain Node วางแผนอยู่ข้างๆ RDBMS อย่างแพร่หลายในเร็ว ๆ นี้แน่นอน

Workshop นี้เหมาะสำหรับคนที่อยากเตรียมพร้อมรับมือการมาของ Blockchain ในองค์กร ไม่ว่าจะเป็น Developer ที่เขียน fullstack หรือ ดูแล infrastructure

1. การสร้างเครือข่าย Ethereum ส่วนตัว



1

รายละเอียด System Requirements

OS : Ubuntu 20.04 LTS หรือที่ support docker

Minimum:

- CPU with 2+ cores
- 4GB RAM
- 320GB free storage space to sync the Mainnet
- 8 MBit/sec download Internet service

Recommended:

- Fast CPU with 4+ cores
- 16GB+ RAM
- Fast SSD with at least 500GB free space
- 25+ MBit/sec download Internet service

2

รายละเอียด Software Requirements

docker : 20.10.6 หรือใหม่กว่า

docker-compose : 1.29.1 หรือใหม่กว่า

editor tools : vim nano

3

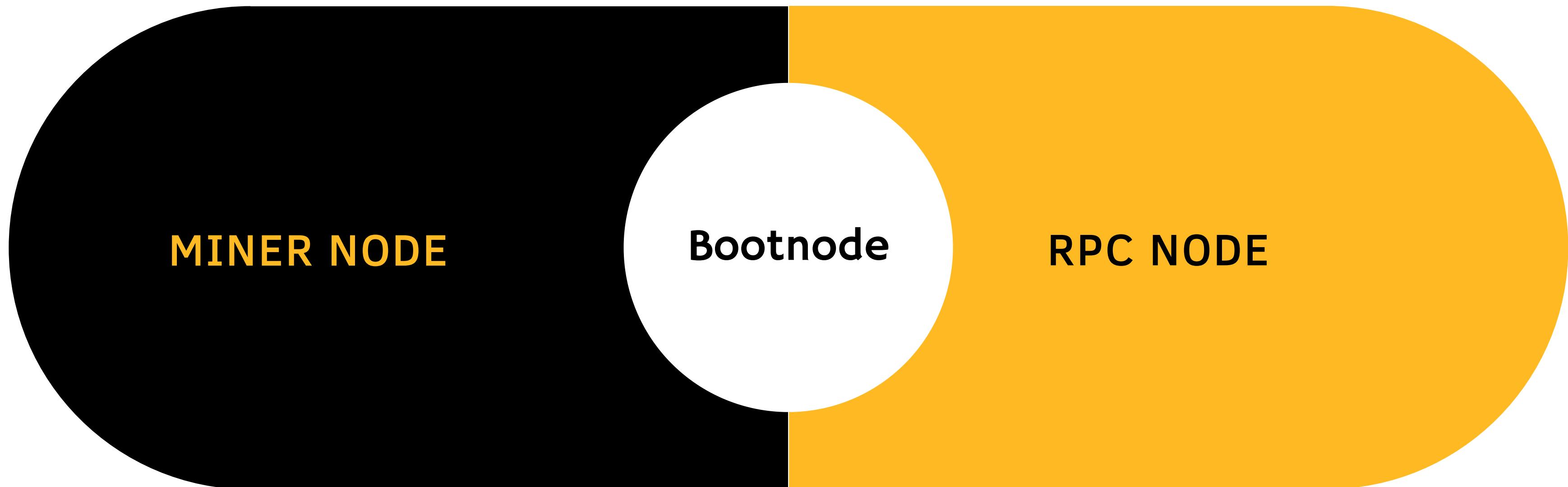
Geth Version

Geth Version docker images

ethereum/client-go:v1.10.12

ethereum/client-go:alltools-v1.10.12

ໂຄສນສ້າງ Private Ethereum Network



2. การ สร้างไฟล์ **Genesis** ตั้งแต่เริ่ม ต้น

สิ่งที่ต้องเตรียมก่อนสร้างไฟล์

- ChainID
- First wallet สำหรับการสร้าง network (extradata)
- wallet สำหรับ pre-mint เหรียญ และจำนวนเหรียญที่ต้องการ pre-mint
- Clique period คือ time block

3. ຄານສູ່ມັດຕົວ [Wá Genesis]

3. ការសិក្សានៃ Genesis Block

4. การติดตั้ง Geth ผ่าน docker และ docker-compose



- 1** กำหนด password สำหรับเครือข่าย Ethereum ส่วนตัว
- 2** สร้างไฟล์ docker-compose.yaml สำหรับ deploy
- 3** เตรียม script สำหรับการสร้าง Address account และ init genesis
- 4** การกำหนด bootnode



4.1 กำหนด password สำหรับเครือข่าย Ethereum ส่วนตัว

สร้างไฟล์ `password.txt` และกำหนดรหัสผ่านในไฟล์ เพื่อใช้สำหรับการ `init` และ `deploy`

ในเอกสารนี้จะใช้คำสั่งเพื่อ random text มาเป็น password ดังนี้

```
# head -1 /dev/urandom | base64 | md5sum
```

3781765b223b882cf89b08a7f1be6cab -

จากนั้น copy 3781765b223b882cf89b08a7f1be6cab ไปใส่ใน `password.txt`

4.2 สร้างไฟล์ docker-compose.yaml สำหรับ deploy

```
1 version: '3.8'
2 services:
3   poa:
4     container_name: validator
5     image: ethereum/client-go:v1.10.12
6     restart: always
7     working_dir: /poa
8     hostname: validator
9     volumes:
10      - $PWD:/poa
11     ports:
12      - 30333:30333
13     command: |
14       --networkid 35
15       --syncmode "full"
16       --mine
17       --miner.gasprice 10000000000
18       --miner.gastarget 20000000
19       --miner.gaslimit 20000000
20       --nat extip:128.199.102.108
21       --datadir "node"
22       --nousb
23       --password password.txt
24       --unlock 0x0001112222333344445555666677778888999999
25       --port 30333
26       --bootnodes ""
```

➤ 4.3 script สำหรับการสร้าง Address account และ init genesis

คำสั่งสำหรับ gen new account

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir /poa/node --password password.txt account new
```

หลังจากได้ public address มาแล้วให้ปรับแก้ genesis ให้เรียบร้อยแล้วจากนั้นทำการ init genesis ด้วยคำสั่งนี้

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 -datadir /poa/node --nousb --password password.txt init genesis.json
```

5. การสร้างโหนดระบบเครือข่าย Ethereum ส่วนตัว

- 5.1 สร้าง Address account สำหรับ first validator
- 5.2 การ init genesis เพื่อเริ่มต้นการใช้งาน geth
- 5.3 ปรับค่า docker-compose.yaml
- 5.4 deploy โหนดระบบเครือข่าย Ethereum ส่วนตัว
- 5.5 การกำหนด bootnode

➤ 5.1 สร้าง Address account สำหรับ first validator

เริ่มจากสร้าง password.txt ตามหัวข้อ 4.1

```
# head -l /dev/urandom | base64 | md5sum  
3781765b223b882cf89b08a7f1be6cab -  
# echo "3781765b223b882cf89b08a7f1be6cab" > password.txt
```

คำสั่งสำหรับ gen new account

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --  
datadir /poa/node --password password.txt account new
```

```
INFO [11-24|04:08:10.094] Maximum peer count          ETH=50 LES=0 total=50
INFO [11-24|04:08:10.094] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"

Your new key was generated

Public address of the key: 0x834656C1b6D0C43823D546CBC1BB547fE50Dc57d
Path of the secret key file: /poa/node/keystore/UTC--2021-11-24T04-08-10.094747511Z--834656c1b6d0c43823d546cbc1bb547fe50dc57d

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!
```

5.2 การ init genesis เพื่อเริ่มต้นการใช้งาน geth

หลังจากได้ Public Address มาแล้วให้นำไปใส่ที่ไฟล์ genesis.json ที่ extradata โดยตัด 0x ออก เช่น 0x834656C1b6D0C43823D546CBC1BB547fE50Dc57d จะใช้แค่ 834656C1b6D0C43823D546CBC1BB547fE50Dc57d ไปใส่ใน genesis.json

หลังจากได้ public address มาแล้วให้ปรับแก้ genesis ให้เรียบร้อยแล้วจากนั้นทำการ init genesis ด้วยคำสั่งนี้

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --  
datadir /poa/node --nousb --password password.txt init genesis.json
```

```
root@node-1:/data/workshop/validator# vim genesis.json
root@node-1:/data/workshop/validator# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir /poa/node --nousb --password password.txt init genesis.json
INFO [11-24|04:24:48.225] Maximum peer count                                     ETH=50 LES=0 total=50
INFO [11-24|04:24:48.226] Smartcard socket not found, disabling      err="stat /run/pcscd/pcscd.comm: no such file or directory"
WARN [11-24|04:24:48.226] Option nousb is deprecated and USB is deactivated by default. Use --usb to enable
INFO [11-24|04:24:48.229] Set global gas cap                                    cap=50,000,000
INFO [11-24|04:24:48.229] Allocated cache and file handles                   database=/poa/node/geth/chaindata cache=16.00MiB handles=16
INFO [11-24|04:24:48.234] Writing custom genesis block
INFO [11-24|04:24:48.236] Persisted trie from memory database                 nodes=3 size=417.00B time="781.855µs" gcnodes=0 gcsize=0.00B g
ctime=0s livenodes=1 livesize=0.00B
INFO [11-24|04:24:48.237] Successfully wrote genesis state                      database=chaindata hash=7c3f71..7474eb
INFO [11-24|04:24:48.237] Allocated cache and file handles                   database=/poa/node/geth/lightchaindata cache=16.00MiB handles=
16
INFO [11-24|04:24:48.243] Writing custom genesis block
INFO [11-24|04:24:48.244] Persisted trie from memory database                 nodes=3 size=417.00B time="656.892µs" gcnodes=0 gcsize=0.00B g
ctime=0s livenodes=1 livesize=0.00B
INFO [11-24|04:24:48.244] Successfully wrote genesis state                      database=lightchaindata hash=7c3f71..7474eb
root@node-1:/data/workshop/validator# |
```



5.3 ปรับค่า docker-compose.yaml

การปรับแต่งค่าใน docker-compose.yaml มีอยู่ 5 ตำแหน่ง

1. **container_name**
2. **--networkid** คือ ChainID
3. **--nat extip:128.199.102.108** ให้ใส่ public ip ลงไปหรือถ้ากำใน local ก็ private ip ลงไปแทนได้
4. **--unlock** ให้ใส่ Public Address ที่ได้จากการ gen account จากข้อ 5.1
5. **--bootnode** จะได้หลังจาก start node และนำมาใส่ทีหลัง
เมื่อแก้ไขครบแล้วให้รัน `# docker-compose up -d`

```
1 version: '3.8'
2 services:
3   poa:
4     container_name: validator ←①
5     image: ethereum/client-go:v1.10.12
6     restart: always
7     working_dir: /poa
8     hostname: validator
9     volumes:
10       - $PWD:/poa
11     ports:
12       - 30333:30333
13     command: |
14       --networkid 39 ←②
15       --syncmode "full"
16       --mine
17       --miner.gasprice 10000000000
18       --miner.gastarget 20000000
19       --miner.gaslimit 20000000
20       --nat extip:128.199.102.108 ←③
21       --datadir "node"
22       --nousb
23       --password password.txt
24       --unlock 0x0001111222233334445555666677778888999999 ←④
25       --port 30333
26       --bootnodes "" ←⑤
27
```



5.4 คำสั่งเช็ค bootnode

หลังจากรัน # docker-compose up -d และให้รันคำสั่งนี้เพื่ออุด nodeInfo เราจะใช้ enode จาก nodeInfo เพื่อนำไปใช้เป็น bootnode ใน docker-compose.yaml

```
# docker exec -it validator geth --datadir /poa/node attach --exec  
'admin.nodeInfo'
```

```
root@node-1:/data/workshop/validator# docker exec -it validator geth --datadir /poa/node attach --exec 'admin.nodeInfo'  
{  
  enode: "enode://4ee9c6bad27ac8db1e8bd044378bf0197a5b4c0a95358e4982ba47fd368f46a0ece0c45e358b67b81823cf4c7ecab134315ce  
89fc0465709908f9e119869b39e@128.199.102.108:30333",
```

```
root@node-1:/data/workshop/validator# docker exec -it validator geth --datadir /poa/node attach --exec 'admin.nodeInfo'
{
  enode: "enode://4ee9c6bad27ac8db1e8bd044378bf0197a5b4c0a95358e4982ba47fd368f46a0ece0c45e358b67b81823cf4c7ecab134315ce
89fc0465709908f9e119869b39e@128.199.102.108:30333",
  enr: "enr:-KO4QKPrX_L8DxUwdBKC_RSkRoH2VD2wZCwLBv7fjGyPMJxfZ7nVyvriMkxPerf7K6-jaoXnpGwW9obU2YpU_YZPBaWGAX1QVPV6g2V0aMf
GhD54GPeAgmlkgnY0gmlwhIDHZmyJc2VjcDI1NmsxoQJ06ca60nrI2x6L0EQ3i_AZeltMCpU1jkmCukf9No9GoIRzbmFwwIN0Y3CCdn2DdWRwgnZ9",
  id: "48ae08c737e3ae063bec5b89d181cd2457ca288d06bb6fcfa8736341835e2f4c",
  ip: "128.199.102.108",
  listenAddr: "[::]:30333",
  name: "Geth/v1.10.12-stable-6c4dc6c3/linux-amd64/gol.17.3",
  ports: {
    discovery: 30333,
    listener: 30333
  },
  protocols: {
    eth: {
      config: {
        berlinBlock: 0,
        byzantiumBlock: 0,
        chainId: 39,
        clique: {...},
        constantinopleBlock: 0,
        eip150Block: 0,
        eip150Hash: "0x000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000",
        eip155Block: 0,
        eip158Block: 0,
        homesteadBlock: 0,
        istanbulBlock: 0,
        muirGlacierBlock: 0,
        petersburgBlock: 0
      },
      difficulty: 69,
      genesis: "0x7c3f7171314b605099a36550b27e476017c2c151a4c1f6db3d14b4e9c77474eb",
      head: "0xf30f39ea329aaca305aea5a3cb1e4af3e82d69c600fc55ce50e0c626252abcff",
      network: 39
    },
    snap: {}
  }
}
```

6. การเพิ่มโหนดในเครือข่าย Ethereum ส่วนตัว

- 6.1 สร้าง Address account สำหรับ validator โหนดที่จะเชื่อมต่อ
- 6.2 การ init genesis เพื่อเริ่มต้นการใช้งาน geth
- 6.3 ปรับค่า docker-compose.yaml
- 6.4 การโหวตเพื่อเพิ่ม validator บนเครือข่าย Ethereum ส่วนตัว
- 6.5 การกำหนด bootnode

> 6.1 สร้าง Address account สำหรับ validator ในดีที่จะเชื่อมต่อ

เริ่มจากสร้าง password.txt ตามหัวข้อ 4.1

```
# head -1 /dev/urandom | base64 | md5sum  
3781765b223b882cf89b08a7f1be6cab -  
# echo "3781765b223b882cf89b08a7f1be6cab" > password.txt
```

คำสั่งสำหรับ gen new account

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir  
/poa/node --password password.txt account new
```

```
INFO [11-24|05:58:57.565] Maximum peer count          ETH=50 LES=0 total=50
INFO [11-24|05:58:57.565] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
```

Your new key was generated

Public address of the key: 0x048F519b032bAfa19Cf28D0cbf717a5fd119fa7A

Path of the secret key file: /poa/node/keystore/UTC--2021-11-24T05-58-57.567072726Z--048f519b032bafa19cf28d0cbf717a5fd119fa7a

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!



6.2 การ init genesis เพื่อเริ่มต้นการใช้งาน geth

การ init genesis สำหรับ node อื่น ๆ เพื่อเชื่อมต่อกับ node แรก ที่ทำไปที่ข้อ 5 เราจะไม่ปรับแต่ง genesis.json อีก เราจะใช้ genesis.json จาก node แรกมา init ได้เลย

ทำการ init genesis ด้วยคำสั่งนี้

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir /poa/node --nousb --password password.txt init genesis.json
```

```
root@node-2:/data/workshop/validator# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir /poa/node --nousb --password password.txt init genesis.json
INFO [11-24|06:07:47.887] Maximum peer count
INFO [11-24|06:07:47.887] Smartcard socket not found, disabling
WARN [11-24|06:07:47.887] Option nousb is deprecated and USB is deactivated by default. Use --usb to enable
INFO [11-24|06:07:47.888] Set global gas cap
INFO [11-24|06:07:47.888] Allocated cache and file handles
INFO [11-24|06:07:47.893] Writing custom genesis block
INFO [11-24|06:07:47.893] Persisted trie from memory database
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:07:47.894] Successfully wrote genesis state
b
INFO [11-24|06:07:47.894] Allocated cache and file handles
B handles=16
INFO [11-24|06:07:47.897] Writing custom genesis block
INFO [11-24|06:07:47.897] Persisted trie from memory database
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:07:47.898] Successfully wrote genesis state
7474eb
root@node-2:/data/workshop/validator# |
```

INFO [11-24|06:07:47.887] Maximum peer count
INFO [11-24|06:07:47.887] Smartcard socket not found, disabling
WARN [11-24|06:07:47.887] Option nousb is deprecated and USB is deactivated by default. Use --usb to enable
INFO [11-24|06:07:47.888] Set global gas cap
INFO [11-24|06:07:47.888] Allocated cache and file handles
INFO [11-24|06:07:47.893] Writing custom genesis block
INFO [11-24|06:07:47.893] Persisted trie from memory database
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:07:47.894] Successfully wrote genesis state
b
INFO [11-24|06:07:47.894] Allocated cache and file handles
B handles=16
INFO [11-24|06:07:47.897] Writing custom genesis block
INFO [11-24|06:07:47.897] Persisted trie from memory database
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:07:47.898] Successfully wrote genesis state
7474eb

> 6.3 ปรับค่า docker-compose.yaml

การปรับแต่งค่าใน docker-compose.yaml มีอยู่ 5 ตำแหน่ง

1. **container_name**
2. **--networkid** คือ ChainID
3. **--nat extip:128.199.102.108** ให้ใส่ public ip ลงไปหรือถ้ากำใน local ก็ private ip ลงไปแทนได้
4. **--unlock** ให้ใส่ Public Address ที่ได้จากการ gen account จากข้อ 6.1
5. **--bootnode** ที่ได้จาก enode ของ node แรก
เมื่อแก้ไขครบแล้วให้รัน `# docker-compose up -d`

```
1 version: '3.8'  
2 services:  
3   poa:  
4     container_name: validator ← 1  
5     image: ethereum/client-go:v1.10.12  
6     restart: always  
7     working_dir: /poa  
8     hostname: validator  
9     volumes:  
10       - $PWD:/poa  
11     ports:  
12       - 30333:30333  
13     command: |  
14       --networkid 39 ← 2  
15       --syncmode "full"  
16       --mine  
17       --miner.gasprice 10000000000  
18       --miner.gastarget 20000000  
19       --miner.gaslimit 20000000  
20       --nat extip:139.59.103.239 ← 3  
21       --datadir "node"  
22       --nousb  
23       --password password.txt  
24       --unlock 0x048F519b032bAfa19Cf28D0cbf717a5fd119fA7A ← 4  
25       --port 30333  
26       --bootnodes "enode://4ee9c6bad27ac8db1e8bd044378bf0197a5b4c0a95358e4982ba47fd368f46a0ece0c45e358b67b81823cf4c  
27         7ecab134315ce89fc0465709908f9e119869b39e@128.199.102.108:30333" ← 5  
28
```



6.4 การໂຫວຕເພື່ອເພີ່ມ validator ແລະ ຄຣູອຂ່າຍ Ethereum ສ່ວນຕົວ

ຮັບຈຳກັດ # docker-compose up -d log ຂອງ node ທີ່ 2 ທີ່ join ເຂົ້າກັບ node ທີ່ 1
ຈະມີ err="unauthorized signer" ໃຫ້ກຳການເພີ່ມ validator ຈາກ node ແລກ

```
# docker exec -it validator geth --datadir /poa/node attach --exec  
'clique.propose("0x048F519b032bAfa19Cf28D0cbf717a5fd119fA7A",  
true)'
```

```
# docker exec -it validator geth --datadir /poa/node attach --exec  
'clique.getSigners()'
```

```
root@node-1:/data/workshop/validator# docker exec -it validator geth --datadir /poa/node --exec 'cliqe.propose("0x048F519b032bAfa19Cf28D0cbf717a5fd119fa7A", true)' attach
null
root@node-1:/data/workshop/validator# docker exec -it validator geth --datadir /poa/node attach --exec 'cliqe.getSigners()'
["0x048f519b032bafa19cf28d0cbf717a5fd119fa7a", "0x834656c1b6d0c43823d546cbc1bb547fe50dc57d"]
root@node-1:/data/workshop/validator# |
```

7. การเปิดตัวเครือข่าย Ethereum ส่วนตัว (RPC)

- 7.1 การ init genesis เพื่อเริ่มต้นการใช้งาน geth
- 7.2 ปรับค่า docker-compose.yaml



7.2 การ init genesis เพื่อเริ่มต้นการใช้งาน geth

การ init genesis สำหรับ node rpc เพื่อเชื่อมต่อกับ Private Chain ที่ทำไป 2 node เราจะไม่ปรับแต่ง genesis.json อีก เราจะใช้ genesis.json จาก node แรกมา init ได้เลย

ทำการ init genesis ด้วยคำสั่งนี้

```
# docker run --rm -it -v $PWD:/poa -w /poa ethereum/client-go:v1.10.12 --datadir /poa/node --nousb init genesis.json
```

```
INFO [11-24|06:31:33.090] Maximum peer count          ETH=50 LES=0 total=50
INFO [11-24|06:31:33.090] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
WARN [11-24|06:31:33.090] Option nousb is deprecated and USB is deactivated by default. Use --usb to enable
INFO [11-24|06:31:33.091] Set global gas cap           cap=50,000,000
INFO [11-24|06:31:33.091] Allocated cache and file handles   database=/poa/node/geth/chaindata cache=16.00MiB handles=16
INFO [11-24|06:31:33.095] Writing custom genesis block
INFO [11-24|06:31:33.095] Persisted trie from memory database   nodes=3 size=417.00B time="144.49µs" gcnodes=0 gcsiz
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:31:33.096] Successfully wrote genesis state   database=chaindata                                     hash=7c3f71..7474eb
INFO [11-24|06:31:33.096] Allocated cache and file handles   database=/poa/node/geth/lightchaindata cache=16.00MiB handles=16
INFO [11-24|06:31:33.098] Writing custom genesis block
INFO [11-24|06:31:33.099] Persisted trie from memory database   nodes=3 size=417.00B time="71.717µs" gcnodes=0 gcsiz
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [11-24|06:31:33.099] Successfully wrote genesis state   database=lightchaindata                                hash=7c3f71..7474eb
root@node-3:/data/workshop/rpc# |
```

> 7.2 ปรับค่า docker-compose.yaml

การปรับแต่งค่าใน docker-compose.yaml มีอยู่ 5 ตำแหน่ง

1. **container_name**
2. **ports**
3. **--networkid** คือ ChainID
4. **--nat extip:128.199.102.108** ให้ใส่ public ip ลงไปหรือถ้ากำใน local ก็ private ip ลงไปแทนได้
5. **--bootnode** ที่ได้จาก enode ของ node แรก

เมื่อแก้ไขครบแล้วให้รัน # docker-compose up -d

```
1 version: '3.8'
2 services:
3   poa:
4     container_name: rpc ← ①
5     image: ethereum/client-go:v1.10.12
6     restart: always
7     working_dir: /poa
8     hostname: rpc
9     volumes:
10       - $PWD:/poa
11     ports:
12       - 30333:30333
13       - 8545:8545 ← ②
14       - 8546:8546 ← ③
15     command: |
16       --networkid 39 ← ④
17       --syncmode "full"
18       --gcmode "archive"
19       --miner.gasprice 10000000000
20       --miner.gastarget 20000000
21       --miner.gaslimit 20000000
22       --nat extip:139.59.126.43 ← ⑤
23       --datadir "node"
24       --nousb
25       --http --http.addr "0.0.0.0" --http.port 8545 --http.corsdomain "*" --http.vhosts "*" --http.api "debug,eth,net,web3,txpool"
26       --ws --ws.addr "0.0.0.0" --ws.port 8546 --ws.origins "*" --ws.api "net,web3,eth"
27       --port 30333
28       --bootnodes "enode://4ee9c6bad27ac8db1e8bd044378bf0197a5b4c0a95358e4982ba47fd368f46a0ece0c45e358b67b81823cf4c
29         7ecab134315ce89fc0465709908f9e119869b39e@128.199.102.108:30333" ← ⑥
```

8. การทดสอบเครือข่าย Ethereum ส่วนตัว

8.1 การทดสอบเครือข่าย Ethereum ส่วนตัวผ่าน Metamask

Metamask เป็น extension ของ browser สามารถใช้งานได้กับ Chrome และ Firefox

หลังจากได้ Metamask แล้ว ให้ custom network ดังนี้

Settings



General

Advanced

Contacts

Security & Privacy

Alerts

Networks

Experimental

About

Networks > Add a network



A malicious network provider can lie about the state of the blockchain and record your network activity. Only add custom networks you trust.

Network Name

POA Testnet

New RPC URL

http://139.59.126.43:8545

Chain ID ⓘ

39

Currency Symbol (Optional)

POA

Block Explorer URL (Optional)

Cancel

Save



METAMASK

POA Testnet ▾



Account 1

0x506...2B4F ▾

⋮



1000000000000 POA



Buy



Send



Swap

Assets

Activity



1000000000000 POA

>

Don't see your token?

[Import tokens](#)

9. การใช้งานเครือข่าย Ethereum ส่วนตัว

9.1 การใช้งานเครือข่าย Ethereum ส่วนตัวผ่าน Metamask

หลังจากได้ custom network Metamask เรียบร้อยแล้ว ให้ลองทดสอบโอน coin ไปยัง wallet address อื่นที่อยู่ใน network ของ blockchain เดียวกัน

 METAMASK

Account 1
0x506...2B4F □

1000000000000 POA

Buy Send Swap

Assets Activity

1000000000000 POA >

Don't see your token?
[Import tokens](#)

 METAMASK

POA Testnet ▾

Send

Account 3
0x5f6f72099ec9661e02Ae5126A0008F55bE1eEd9b X

Asset: POA Balance: 1000000000000 POA

Amount: 1000000 POA Max ↑
No Conversion Rate Available

Gas Price (GWEI) 10 Gas Limit 21000

Cancel Next



POA Testnet

Account 1 → Account 3

SENDING POA

1000000

Estimated gas fee 0.00021 POA

Max fee: 0.00021 POA

Total 1000000.00021 **1000000.00021 POA**

Amount + gas fee Max amount: 1000000.00021 POA

EDIT

Reject **Confirm**

This screenshot shows the MetaMask transaction confirmation dialog. It displays a transfer from Account 1 to Account 3 for 1,000,000 units of the POA token. The estimated gas fee is 0.00021 POA. The total amount including the fee is 1,000,000.00021 POA. The user has the option to edit the transaction or confirm it by clicking the "Confirm" button.

POA Testnet

Account 3
0x5f6...Ed9b

1000000 POA

Buy **Send** **Swap**

Assets **Activity**

You have no transactions

This screenshot shows the MetaMask wallet interface for Account 3. The account balance is 1,000,000 POA. There are three main buttons: "Buy", "Send", and "Swap". Below these buttons are tabs for "Assets" and "Activity". A message indicates that there are no transactions.



IO. การทดสอบและ deploy Smart Contract กับเครือข่าย Ethereum PoA ที่สร้างขึ้น

- IO.1 RPC และ Network ID
- IO.2 Gas Price และ Gas limit
- IO.3 ทดสอบ deploy smartcontract ด้วย Remix

II. การ Deploy blockscout explorer

- II.1 ปรับแต่งค่า coin ของ blockscout
- II.2 ปรับแต่ง Theme ของ blockscout
- II.3 การ build docker images blockscout
- II.4 เตรียมไฟล์ docker-compose.yaml
- II.5 deploy blockscout



II.I ปรับแต่งค่า coin ของ blockscout

I. download blockscout

```
# wget https://github.com/blockscout/blockscout/archive/refs/tags/v3.7.3-beta.zip  
# unzip v3.7.3-beta.zip  
# cd blockscout-3.7.3-beta
```

2. เปลี่ยน Coin name

```
# vim apps/block_scout_web/priv/gettext/en/LC_MESSAGES/default.po  
ค้นหา ETH กับ Ether และเปลี่ยนเป็น coin ใหม่  
msgid "ETH"  
msgstr "POA"
```

```
msgid "Ether"  
msgstr "POA"
```



II.2 ปรับแต่ง Theme ของ blockscout

vim apps/block_scout_web/assets/css/theme/_variables.scss

สามารถเปลี่ยน theme ได้ตามที่มี template ไว้ได้เลย โดยการ uncomment

```
1 @import "base_variables";
2 @import "neutral_variables";
3 // @import "xusdt_variables";
4 // @import "dai_variables";
5 // @import "ethereum_classic_variables";
6 // @import "ethereum_variables";
7 // @import "etherl_variables";
8 // @import "expanse_variables";
9 // @import "gochain_variables";
10 // @import "goerli_variables";
11 // @import "kovan_variables";
12 // @import "lukso_variables";
13 // @import "musicoin_variables";
14 // @import "pirl_variables";
15 // @import "poa_variables";
16 // @import "rinkeby_variables";
17 // @import "ropsten_variables";
18 // @import "social_variables";
19 // @import "sokol_variables";
20 // @import "tobalaba_variables";
21 // @import "tomochain_variables";
22 // @import "rsk_variables";
23 // @import "ethercore_variables";
```



II.3 การ build docker images blockscout

```
# cd docker/  
# docker build --build-arg COIN="POA" -f ./Dockerfile -t blockscout_poa ..
```

ใช้เวลาโดยประมาณ 5 ถึง 10 นาทีในการ build แล้วแต่ spec ของเครื่องที่นำมาใช้งาน



II.4 เตรียมไฟล์ docker-compose.yaml

```
# git clone https://github.com/thanachaiTP/blockscout.git  
# cd blockscout
```

ด้านในจะมีไฟล์ docker-compose.yaml ให้ปรับแต่งค่า postgres

- POSTGRES_PASSWORD=1234567890
- migrations && blockscout
 - DATABASE_URL=postgresql://postgres:1234567890@postgres:5432/explorer?ssl=false
 - NETWORK=POA
 - SUBNETWORK=Mainnet
 - ETHEREUM_JSONRPC_HTTP_URL=https://rpc.xyz.io
 - ETHEREUM_JSONRPC_WS_URL=ws://ws.xyz.io
 - COIN=POA
 - BLOCKSCOUT_HOST=exp.xyz.io



II.5 deploy blockscout

- Deploy PostgreSQL

```
# docker-compose up -d postgres
```

- Migrations blockscout

```
# docker-compose up migrations
```

- Deploy blockscout

```
# docker-compose up -d blockscout
```