

The Chinese University of Hong Kong
Department of Information Engineering
FTEC5520 –Applied Blockchain & Cryptocurrency
Lab1 Report guidance

Ethereum Infrastructure Setup & Practice

Please edit this file directly but submitted a PDF version to blackboard finally.

1 Question Answering:

Answers for each question: there will be 3 questions, but you only need to finish the questions before task6.

Q1: What do you need to change in commands if you want to connect to real public world Ethereum network? (1 mark) Hint: puppeth is a private Ethereum network manager, pls consider another approach Please list the main steps and explain why.

Answer:

1) Compared with private network that genesis should be defined, in public network that there are historical blocks, so previous records need to be downloaded.

2) To connect to the mainnet network (real public world Ethereum network),

use *geth*, run the command:

geth --datadir ~/blockchain --networkid 1

Here the "*--datadir ~/blockchain*" means where to store the blockchain,

"*--networkid*" is used to specify the network ID, 1 is the ID of the mainnet network.

Q2: 1) Please describe the process of mining, what are the main steps for finding a new block with Ether? (0.5 mark)

Answer:

the process of mining:

a) use proof of work (PoW) to finish the construction of a new block

for example, hashing calculation with a Nonce, commands like:

miner.start()

b) when finding the Nonce, broadcast to the entire network

c) chain the new block and get reward

2) What's the content of a new mined Ethereum block, and how to identify a block in the blockchain network except using block number? (0.5 mark)

Answer:

contents:

- a) Timestamp – the time when the block mined.**
- b) Block number – the length of the blockchain in blocks.**
- c) Difficulty – the effort required to mine the block.**
- d) mixHash – a unique identifier for that block.**
- e) A parent hash – the unique identifier for previous block.**
- f) Transactions list – the transactions included in the block.**
- g) State root – the entire state of the system: account balances, contract storage, contract code and account nonces are inside.**
- h) Nonce – a hash that, with the mixHash, show the proof of work.**

identify a block:

- mixHash – a unique identifier for that block

Q3: If you want to deploy the voting smart contract on the blockchain you created in tast1~5, what should you do? If you want to deploy your smart contract on other types of Eth networks like main net, Rinkeby, modern test net, etc, what should you do? (1 mark)

Answer:

deploy on the blockchain created in tast1~5:

1) Install nodejs, ganache-cli, Solidity compiler (solc) and web3

2) Write and compile the smart contract

for example, generate bytecode.

set Ether for gas.

3) Deployment script & Access to an Ethereum node

run your own or via an API, use 'deploy / send / then' method.

(sending a transaction containing the code of the compiled smart contract)

deploy on other types of Eth networks:

- 1) install Meta-mask Chrome Extension, then run on the browser**
- 2) Create a wallet at meta-mask**
- 3) Select any one test network, e.g. mainnet/Rinkeby/modern test net, etc**
- 4) Add some dummy Ethers in your wallet**
- 5) Use editor remix to write the smart contract in Solidity**
- 6) Create a .sol extension file**
- 7) Deploy contract by pressing the deploy button in Remix window**

2 Screen Capture of Main Steps:

Your screenshots must include these parts at least and detailed description of each screenshots:

1 Puppeth procedure.

e.g. your description 1

(copy your image1 here)

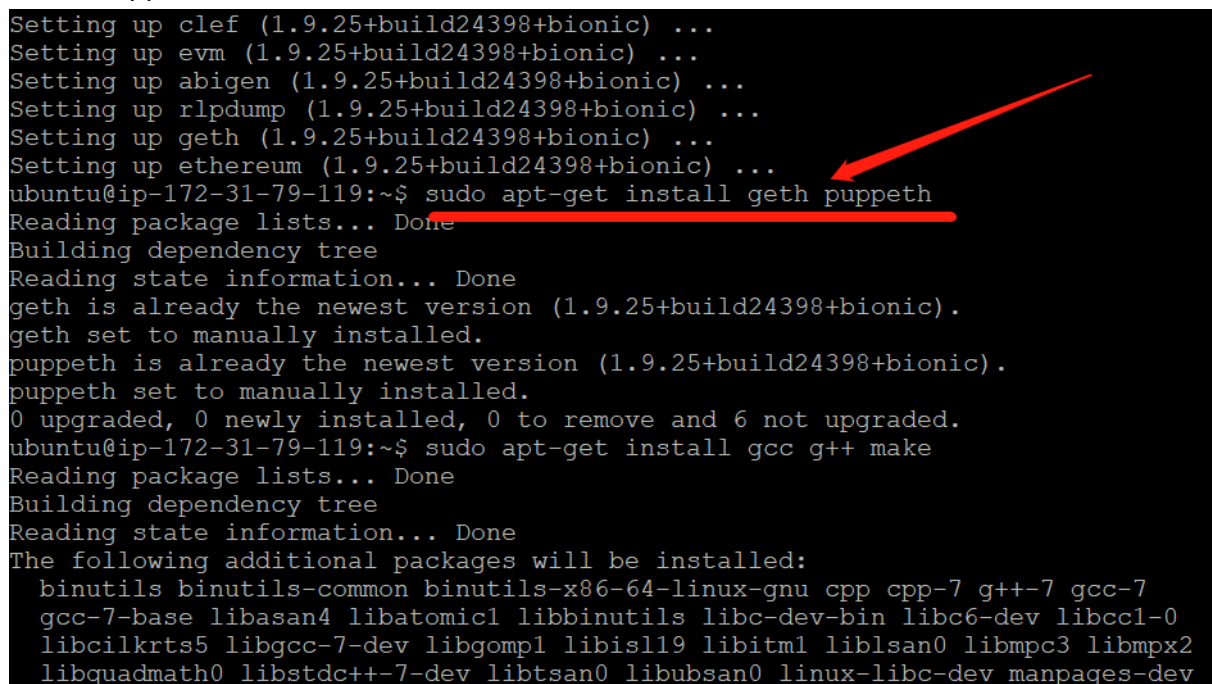
your description 2

(copy your image2 here)

...

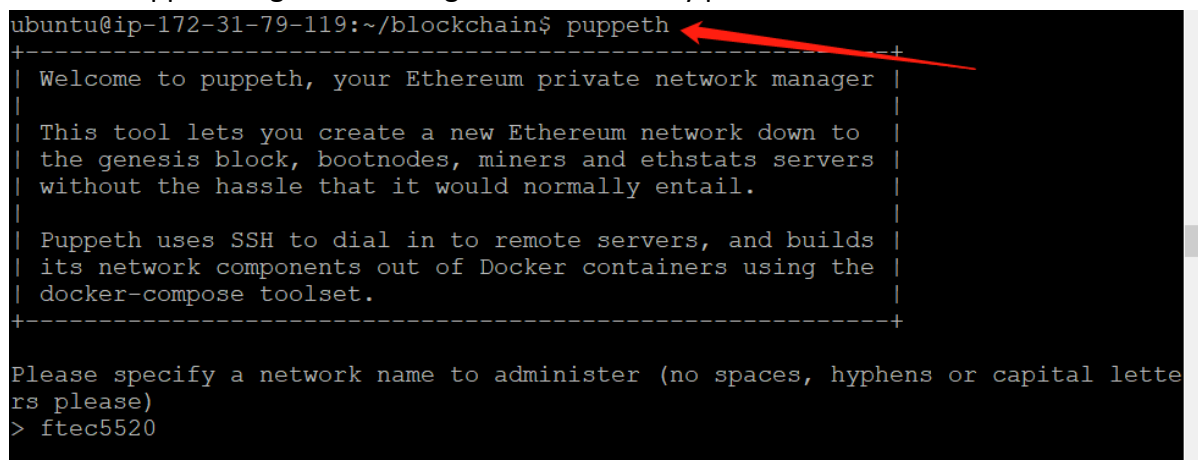
Puppeth is a one-stop shop of blockchain service management tools.

(1) Install Puppeth



```
Setting up clef (1.9.25+build24398+bionic) ...
Setting up evm (1.9.25+build24398+bionic) ...
Setting up abigen (1.9.25+build24398+bionic) ...
Setting up rlpdump (1.9.25+build24398+bionic) ...
Setting up geth (1.9.25+build24398+bionic) ...
Setting up ethereum (1.9.25+build24398+bionic) ...
ubuntu@ip-172-31-79-119:~$ sudo apt-get install geth puppeth
Reading package lists... Done
Building dependency tree
Reading state information... Done
geth is already the newest version (1.9.25+build24398+bionic).
geth set to manually installed.
puppeth is already the newest version (1.9.25+build24398+bionic).
puppeth set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 6 not upgraded.
ubuntu@ip-172-31-79-119:~$ sudo apt-get install gcc g++ make
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-7 g++-7 gcc-7
  gcc-7-base libasan4 libatomic1 libbinutils libc-dev-bin libc6-dev libccl-0
  libcilkrts5 libgcc-7-dev libgomp1 libisl19 libitm1 liblsan0 libmpc3 libmpx2
  libquadmath0 libstdc++-7-dev libtsan0 libubsan0 linux-libc-dev manpages-dev
```

(2) Use the Puppeth to generate the genesis block of my private blockchains.



```
ubuntu@ip-172-31-79-119:~/blockchain$ 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)
> ftec5520
```

(3) Give a network name for administration.

```

+-----+
Please specify a network name to administer (no spaces, hyphens or capital letters please)
> ftec5520

Sweet, you can set this via --network=ftec5520 next time!

INFO [03-04|05:36:28.095] Administering Ethereum network      name=ftec5520
WARN [03-04|05:36:28.095] No previous configurations found      path=/home/ubuntu/.puppeth/ftec5520

```

(4) Configure and create new genesis, then choose POW, others default.

```

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

What would you like to do? (default = create)
 1. Create new genesis from scratch
 2. Import already existing genesis
> 1

Which consensus engine to use? (default = clique)
 1. Ethash - proof-of-work
 2. Clique - proof-of-authority
> 1

Which accounts should be pre-funded? (advisable at least one)
> 0x

Should the precompile-addresses (0x1 .. 0xff) be pre-funded with 1 wei? (answer yes)
>

```

(5) After creating genesis, manage and export it by saving as 4 files:

ftec5520.json, ftec5520-aleth.json, ftec5520-harmony.json, ftec5520-parity.json

```

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 configurations
 2. Export genesis configurations
 3. Remove genesis configuration
> 2

Which folder to save the genesis specs into? (default = current)
Will create ftec5520.json, ftec5520-aleth.json, ftec5520-harmony.json, ftec5520-parity.json
>
INFO [03-04|05:38:00.402] Saved native genesis chain spec      path=ftec5520.json
INFO [03-04|05:38:00.403] Saved genesis chain spec              client=aleth path=ftec5520-aleth.json
INFO [03-04|05:38:00.404] Saved genesis chain spec              client=parity path=ftec5520-parity.json
INFO [03-04|05:38:00.406] Saved genesis chain spec              client=harmony path=ftec5520-harmony.json

```

- 2 The content of your ftec5520.json after modified the difficulty.

(1) Set the "chainId" as 5520, and "difficulty" 10.

```
{
  "config": {
    "chainId": 5520,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "eip155Block": 0,
    "eip158Block": 0,
    "byzantiumBlock": 0,
    "constantinopleBlock": 0,
    "petersburgBlock": 0,
    "istanbulBlock": 0,
    "ethash": {}
  },
  "nonce": "0x0",
  "timestamp": "0x604071e5",
  "extraData": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "gasLimit": "0x47b760",
  "difficulty": "10",
  "mixHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "alloc": {
    "0000000000000000000000000000000000000000000000000000000000000000": {
      "balance": "0x1"
    },
    "0000000000000000000000000000000000000000000000000000000000000001": {
      "balance": "0x1"
    }
  }
}
"ftec5520.json" 795L, 21289C 14,4 Top
```

- 3 The creation step of Ethereum account with returned HASH value or account address (either using "geth account new" in CLI or "personal.newAccount("ftec5520" in Geth console)

(1) Use "personal.newAccount("ftec5520")" command line to create account.

```
Welcome to the Geth JavaScript console!

instance: Geth/v1.9.25-stable-e7872729/linux-amd64/go1.15.6
coinbase: 0x0891cd92bdf15368a18cbbabe430d39957d016f9
at block: 0 (Thu Mar 04 2021 05:36:37 GMT+0000 (UTC))
datadir: /home/ubuntu/blockchain
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0
rpc:1.0 txpool:1.0 web3:1.0

To exit, press ctrl-d
> web3.eth.accounts
["0x0891cd92bdf15368a18cbbabe430d39957d016f9"]
> personal.newAccount("ftec5520")
"0x2b191e48fda55e0ee714c07657fd056ef70bfebd"
```

- 4 Use "web3.eth.accounts" to show the accounts addresses (hash values in []) you created in CLI or Geth console

(1) There are now 2 accounts, show them by "web3.eth.accounts".


```

ubuntu@ip-172-31-79-119:~/blockchain$ tail -f ftec5520-lab1.log
INFO [03-04|05:42:43.898] Generating DAG in progress          epoch=0 perce
ntage=68 elapsed=1m21.514s
INFO [03-04|05:42:45.005] Generating DAG in progress          epoch=0 perce
ntage=69 elapsed=1m22.621s
INFO [03-04|05:42:46.127] Generating DAG in progress          epoch=0 perce
ntage=70 elapsed=1m23.744s
INFO [03-04|05:42:47.212] Generating DAG in progress          epoch=0 perce
ntage=71 elapsed=1m24.828s
INFO [03-04|05:42:47.222] Looking for peers                    peercount=0 t
ried=46 static=0
INFO [03-04|05:42:48.340] Generating DAG in progress          epoch=0 perce
ntage=72 elapsed=1m25.956s
INFO [03-04|05:42:49.504] Generating DAG in progress          epoch=0 perce
ntage=73 elapsed=1m27.120s
INFO [03-04|05:42:50.616] Generating DAG in progress          epoch=0 perce
ntage=74 elapsed=1m28.232s
INFO [03-04|05:42:51.690] Generating DAG in progress          epoch=0 perce
ntage=75 elapsed=1m29.306s
INFO [03-04|05:42:52.831] Generating DAG in progress          epoch=0 perce

```

(2) Mined block successfully:

```

sh="5538ee...ec0954"
INFO [03-04|05:47:13.567] Commit new mining work              number=97 sea
lhash="72205f...5be5d8" uncles=0 txs=0 gas=0 fees=0 elapsed="180.091µs"
INFO [03-04|05:47:13.740] Successfully sealed new block ← number=97 sea
lhash="72205f...5be5d8" hash="700cb3...37d292" elapsed=172.697ms
INFO [03-04|05:47:13.740] ⚙ block reached canonical chain    number=90 ha
sh="6fb82c...37d6ab"
INFO [03-04|05:47:13.740] ⚙ mined potential block            number=97 ha
sh="700cb3...37d292"
INFO [03-04|05:47:13.740] Mining too far in the future        wait=2s
INFO [03-04|05:47:13.983] Generating DAG in progress          epoch=1 perce
ntage=39 elapsed=3m51.394s
INFO [03-04|05:47:15.115] Generating DAG in progress          epoch=1 perce
ntage=40 elapsed=3m52.526s
INFO [03-04|05:47:15.751] Commit new mining work              number=98 sea
lhash="led379...033b64" uncles=0 txs=0 gas=0 fees=0 elapsed=2.010s
INFO [03-04|05:47:16.403] Successfully sealed new block ← number=98 sea
lhash="led379...033b64" hash="2bfc75...e7e18b" elapsed=651.797ms
INFO [03-04|05:47:16.403] ⚙ block reached canonical chain    number=91 ha
sh="2fdf15...96e882"
INFO [03-04|05:47:16.403] ⚙ mined potential block            number=98 ha
sh="2bfc75...e7e18b"
INFO [03-04|05:47:16.421] Commit new mining work              number=99 sea
lhash="f10e27...3643b6" uncles=0 txs=0 gas=0 fees=0 elapsed="175.69µs"
INFO [03-04|05:47:17.215] Successfully sealed new block ← number=99 sea
lhash="f10e27...3643b6" hash="c5bec2...730e94" elapsed=794.369ms
INFO [03-04|05:47:17.215] ⚙ block reached canonical chain    number=92 ha
sh="582f31...c23a5e"

```