



AUSTRALIAN INSTITUTE
OF HIGHER EDUCATION

Unit Outline

Unit code: MBIS5019

Unit name: Blockchain and Cryptocurrency

Session 4 – Week 2
6 Marks

Lab 3: Create a Private Network- 6 Marks

Purpose:

This lab explains how to build your own private network in Ethereum. Ethereum is a cryptocurrency blockchain system. In this lab, you will create a minimum of two nodes on your own virtual machine. You also can create other nodes in participation with your class colleagues. Your lecturer may decide to create this lab in a group of two students or individual.

Materials:

You will need:

- Internet Connectivity.
- Virtual Machine of Ubuntu LINUX

Instructions:

Go Ethereum:

As explained in the lecture 3, the slides (12 – 35) explain the steps of creating a private network with a minimum of two nodes.

The commands of GO will be continued in the next session. This session you need to focus on setting up the network and ensure it's running properly, let us start.

Create Private Network

1. Start your Ubuntu VM and open one terminal
2. Under your own profile create a directory, the steps are:
 - a. `sudo cd ~`
 - b. `sudo mkdir goethereum`
 - c. `sudo chmod -R 777 goethereum`
 - d. `cd goethereum`
3. open another terminal and install git
 - a. `sudo apt-get update`
 - b. `sudo apt-get install git`
4. go back to the first terminal and continue
 - a. Download and install Geth, you may use this link:
<https://geth.ethereum.org/downloads/>



- b. Choose Geth for Linux
- c. To install geth on Ubuntu:
`sudo apt-get install ethereum`
- d. Create two directories “node0” and “node1” inside the directory goethereum
`sudo mkdir node0`
`sudo mkdir node1`
`sudo chmod -R 777 node0`
`sudo chmod -R 777 node1`

- e. We need to run the following command to create the pair of keys for Proof of Authority (PoA):

f. THIS IS FOR NODE 0

```
geth account new --datadir node0  
password:  
repeat password:
```

Your new key was generated

Public address of the key 0x7E58A6fc8358e63dB1d251c8A8c7E4a2Ea13EdA1
Path of the secret key file: node0\keystore\UTC--2022-12-19T02-42-14.128871400Z--7e58a6fc8358e63db1d251c8a8c7e4a2ea13eda1

g. THIS IS FOR NODE 1

```
geth account new --datadir node1  
password:  
repeat password:
```

Your new key was generated

Public address of the key 0xB4748eDbF5f4724B5a8549Cfe6F5987d8e7979B1
Path of the secret key file: node1\keystore\UTC--2022-12-19T02-42-54.487781200Z--b4748edbf5f4724b5a8549cfe6f5987d8e7979b1

- h. To create genesis.json file, run the following command

```

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
> 2

How many seconds should blocks take? (default = 15)
> 15

Which accounts are allowed to seal? (mandatory at least one)
> 0x09F54B44b33db45D1581aAf0495ee3315CCDc919
> 0x8B26EE9d9fb100b966F84F138Ba96a6a7352629a
> 0x

Which accounts should be pre-funded? (advisable at least one)
> 0x09F54B44b33db45D1581aAf0495ee3315CCDc919
> 0x8B26EE9d9fb100b966F84F138Ba96a6a7352629a
> 0x

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

Specify your chain/network ID if you want an explicit one (default = random)
> 11202
INFO [10-02|20:16:28.802] 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

```

Puppet

Then choose the following options:

The network name: **aih**

What would you like to do? (default = stats): **2. Configure new genesis**

What would you like to do? (default = create): **1. Create new genesis from scratch**

Which consensus engine to use? (default = clique): **2. Clique - proof-of-authority**

How many seconds should blocks take? (default = 15): **15**

Which accounts are allowed to seal? (mandatory at least one)

> **0x7E58A6fc8358e63dB1d251c8A8c7E4a2Ea13EdA1**

> **0xB4748eDbF5f4724B5a8549Cfe6F5987d8e7979B1**

> **0x**

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

> **0x7E58A6fc8358e63dB1d251c8A8c7E4a2Ea13EdA1**

> 0xB4748eDbF5f4724B5a8549Cfe6F5987d8e7979B1

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

(advisable yes): Yes

Specify your chain/network ID if you want an explicit one (default = random):

13516

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

1. Modify existing configurations
2. Export genesis configurations
3. Remove genesis configuration

Which folder to save the genesis spec into? (default = current)

Will create aih.json: current

Finish configuration by pressing ctrl + c + d

- i. aih.json is now successfully created. Now we need to initialize the nodes, as shown in Slide 16 – Lecture 3
sudo geth --datadir node0 init aih.json

sudo geth --datadir node1 init aih.json

5. Create a boot node

- a. bootnode --genkey=boot.key
- b. bootnode --nodekey=boot.key

Results are:

enode://7717d4d8b62ec8c809c95bdd224f3bfba417ca39f11e450936fdaf5633d183f4cab1596580bbc97856a70a3aa2dafade2fe975e82d6a145ad018ddb61f5cdd4@127.0.0.1:0?discport=30301

Note: you're using cmd/bootnode, a developer tool.

We recommend using a regular node as bootstrap node for production deployments.

INFO [12-19|15:43:20.677] New local node record

seq=1,671,425,000,677 id=fb3f791e624b95ef ip=<nil> udp=0 tcp=0

6. Create a file inside node0 and node1. The file name password.txt, add the password that you chose in (4.g) to the file password.txt
7. Start the node: Open another terminal to start the nodes. The bootnode, in step 5, must stay running.

- a. For NODE 0: **do not forget to use the results appear in step 5. The results start with 'encode://771.....'. It is shown below with the blue colour.**

```
sudo geth --datadir node0 --authrpc.port 8550 --syncmode "full" --verbosity 6 --
port 30302 --http --http.addr 'localhost' --http.port 8545 --http.api
'personal,db,eth,net,web3,txpool,miner' --bootnodes
'encode://7717d4d8b62ec8c809c95bdd224f3bfba417ca39f11e450936daf5633d1
83f4cab1596580bbc97856a70a3aa2dafade2fe975e82d6a145ad018ddbf61f5cdd
4@127.0.0.1:0?discport=30301' --networkid 13516--unlock
"0xcda7557c4d7f05f8e2560d819b8f79732d2a358e" --password
node0/password.txt --mine console --allow-insecure-unlock console
2>console.txt
```

- b. Open another terminal and start NODE 1:

```
sudo geth --datadir node1 --authrpc.port 8552 --syncmode "full" --verbosity 6 --
port 30303 --http --http.addr 'localhost' --http.port 8546 --http.api
'personal,db,eth,net,web3,txpool,miner' --bootnodes
'encode://7717d4d8b62ec8c809c95bdd224f3bfba417ca39f11e450936daf5633d1
83f4cab1596580bbc97856a70a3aa2dafade2fe975e82d6a145ad018ddbf61f5cdd
4@127.0.0.1:0?discport=30301' --networkid 13516 --unlock
"0x81c2d7750d06ae14a88c04eac829652771e67d7" --password
node1/password.txt --mine console --allow-insecure-unlock console
2>console.txt
```

8. Now you have three terminals, one for the bootnode, one for the node0, and the third one for node1
9. A mark will be given to you based on the three terminals. Marks will be deducted if were unable to run the three terminals
10. Next lab will show you how to use these consoles to check the blockchain system

Assessment number & type	Assessment description	Week assessed and date due	Weighting	ULOs to be assessed
A1: Portfolio (Individual)	In-class assessment techniques (In class Activity /Quiz/Practical submission) systematize the process of getting useful and timely	session 1-7	30%	ULO1, ULO2,

	feedback on student learning.			
--	-------------------------------	--	--	--

ULO1: Explain how blockchain, distributed ledger, smart contracts, finance, privacy and security technologies work.

ULO2: Critically analyze and assess the key issues, challenges and priorities in blockchain technology