Lab 5
Ethereum Network and Transactions

You will use Ubuntu VM for this lab exercise.

1. Download *genesis.json* from Moodle.

2. The Ethereum protocol is implemented using Go language. To install Ethereum client, first, you need to install Go.
   a. Download the latest version of Go (for Linux) from
      https://golang.org/dl/
   b. Unpack it to the /usr/local by running and create a directory for *go*
      $ sudo tar -C /usr/local -xzf go1.XX.X.linux-amd64.tar.gz
      $ mkdir -p ~/go
   c. Setup Go folder in *.bashrc*:
      $ echo "export GOPATH=$HOME/go" >> ~/.bashrc
   d. Update your path to Go in *.bashrc*:
      $ echo "export PATH=$PATH:$HOME/go/bin:/usr/local/go/bin" >> ~/.bashrc
   e. Apply the updated setting to the current terminal:
      $ source ~/.bashrc
   f. You can check the version of Go by running.
      $ go version

3. Once *Go* is installed in your Ubuntu, you can build go-ethereum from its source.
   a. Download the go-ethereum source from Github:
      $ sudo apt-get update
      $ sudo apt-get install git
      $ cd ~
      $ git clone https://github.com/ethereum/go-ethereum
   b. Go to the directory that your go-ethereum is downloaded and build it:
      $ cd go-ethereum
      $ make all
   c. Setup the path of execution files and apply it to your current terminal
      $ echo "export PATH=$PATH:$HOME/go-ethereum/build/bin" >> ~/.bashrc
      $ source ~/.bashrc

4. Create Private Network Using Ethereum
   a. You have to make a data folder to save the ledger
      $ mkdir -p ~/cscieth
   b. Then, copy "genesis.json" file to *cscieth* folder and initialize your Ethereum client by running
      $ cd ~/cscieth
      $ geth --datadir ~/cscieth init genesis.json

c. Run Ethereum node by executing the following command:
$ geth --datadir="~/cscieth" --networkid 15 --nodiscover console --http --http.port 8546 --http.corsdomain '*' --http.api "eth,net,web3,miner,debug,personal,rpc"

[In Ethereum console of c.] Create Ethereum account by executing the following command on the terminal. The first created account will get the mining rewards, which is called *coinbase*.

> personal.newAccount()
[Warning!] Do not forget your password.
You can find more information about your account by executing the following:
> personal

d. [In Ethereum console of c.] Start mining by running by executing
> miner.start()
[Note] It takes a few minutes to start mining. "Generating DAG in progress" is not mining. It is just a preparation process for the mining. You will see "⛏ mined potential block" message when the mining actually is started.

e. Check whether *geth.ipc* was created in ~/cscieth folder using another terminal. Then, attaching another console to the current node by running the following in the new terminal:
$ geth --ipcpath ~/cscieth/geth.ipc --datadir ~/cscieth/ attach

5. Install Ethereum Wallet (Metamask) and connect it to your private network.
   a. You can attach one of the famous crypto wallets, Metamask, to your default browser of Ubuntu, which is Firefox.

   b. Visit Metamask webpage (https://metamask.io/), and click the "Download now" button and click "Install MetaMask for Firefox". Add Metamask as an add-on to your browser (Click "+Add to Firefox" button).

   c. You will see the on the upper right corner in the browser and you will be redirected to "Welcome to Metamask" page.

   d. Click "Get Started" button, then "Create a Wallet". Follow the instruction carefully to create your wallet.

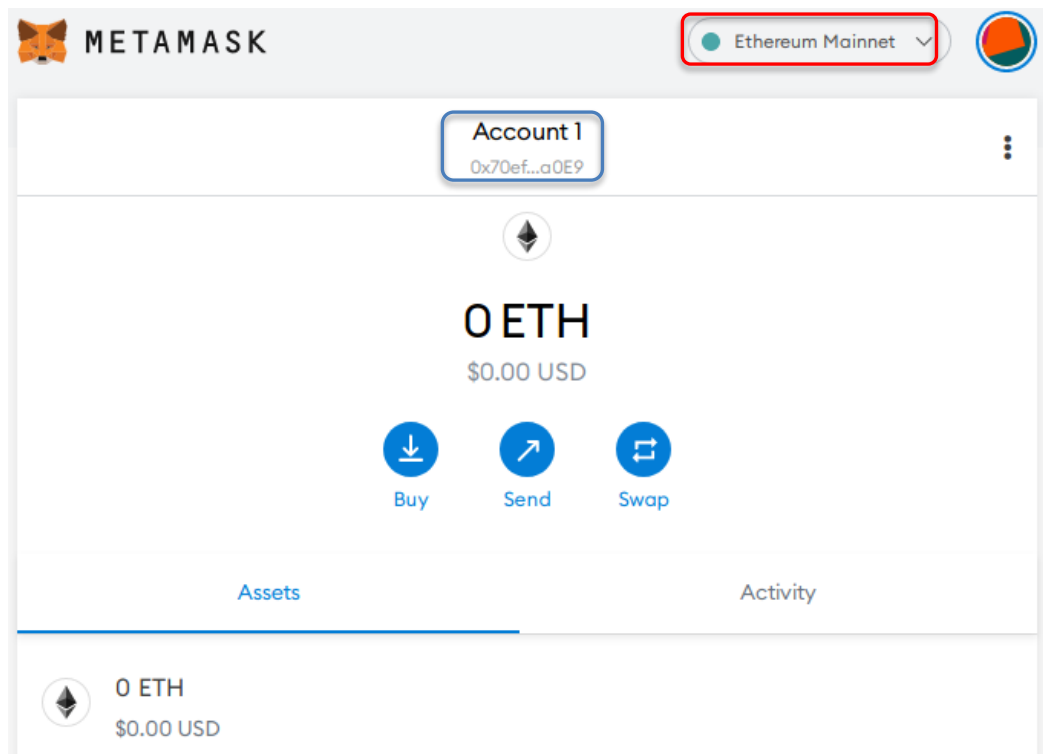e. If you successfully create your wallet, you may see the page as follows:

Figure 1.Metamask intial page

Note that the blue and red boxes are used in Figure 1 to explain some activities which will explained later.

f. You are currently connected to "Main Ethereum Network". Click network list button highlighted by red box in Figure 1. Click (Add Network) button. Use the following information to register your local Ethereum network.

- Network Name: PrivETH (This can be any name)
- New RPC URL: http://localhost:8546
- Chain ID: 15

Click the (Save) button to connect to your own Ethereum network.

6. (Make a transaction between accounts) So far, you have two accounts. One is used as a coinbase in your Ethereum network and the other you just created using your Metamask wallet. The following activities can be performed at the geth console created in 4.e.

a. Check your coinbase account, and check your balance on coinbase account to see how much ethereum coins have been mined.
> eth.coinbase
> web3.fromWei(eth.getBalance(eth.coinbase),"ether")

b. Make a transaction of 1.23 eth from your coinbase account to the account in Metamask Wallet.

3

```
> var tx = {from: [Your coinbase account number], to: [Your account
number of Metamask Wallet], value: web3.toWei(1.23,"ether")}
> personal.sendTransaction(tx, [passphrase of coinbase account] )
```

For example, the following are example of actual commands:
```
> var tx = {from:
"0x4dca99e3e5b85772d5693c30d191ad19387a32d2", to:
"0xF2A03c816F4e1bcaB1D4b1271a4e478CD3407B89", value:
web3.toWei(1.23,"ether")}
> personal.sendTransaction(tx,"********")
```
where
1) your coin base account number is
0x4dca99e3e5b85772d5693c30d191ad19387a32d2,
2) your account number in the wallet is
0xF2A03c816F4e1bcaB1D4b1271a4e478CD3407B89, and
3) "********" is the passphrase, which you inputted in 4.c.

[Tip] You can copy the account number of your Metamask wallet by clicking the account number highlighted by blue box in Figure 1.

c. Check if the transaction was made by checking the balances of your account in Metamask and coinbase account are changed.

d. **[Task]** Make a transaction to send 1 eth back to the coinbase account from your Metamask Wallet.

7. Audit blocks and transaction records in the attached console 4.e.
   a. Check the transaction information in the console and the receipt of a transaction using the following commands:
   ```
   > eth.getTransaction("[Transaction hash]")
   > eth.getTransactionReceipt("[Transaction hash]")
   ```
   [Note] You can get the transaction hash in 6.b. after you execute the transaction. Check which information you can see in the transaction.
   b. Check the information of the entire block which contains the your transaction in the console:
   ```
   > eth.getBlock([block number])
   ```
   (e.g., eth.getBlock(1) if your transaction is recorded in the first block.)