Blockchain Lab Assignment-3

Setting Up an Ethereum Node

Ethereum node is any device that is running the Ethereum protocol (blockchain). When we connect to the Ethereum protocol we are on the Ethereum blockchain network. By running an Ethereum node we can connect to other nodes in the network, have direct access to the blockchain, and even do things like mine blocks, send transactions, and deploy smart contracts.

Installation:

Visit the Go Ethereum website and install Geth

Visit here: https://geth.ethereum.org/downloads/

Download the latest release of Geth for Windows, make sure you download the 64-bit version.

Establishing Our Own Private Ethereum Network

Create a new folder on your desktop called “Private-chain”

Open command prompt in this folder and create a data directory folder for our chaindata by typing “mkdir chaindata

Next, we need to create and save our genesis.json block in our Private-chain folder, as the genesis block will be used to initialize our private network and store data in the data directory folder “chaindata”

Open up notepad, copy & paste the code below into a new file called “genesis.json” and save this file in our Private-chain folder.

```
{
  "config": {
        "chainId": 4777,
        "homesteadBlock": 0,
```

```
        "eip150Block": 0,
        "eip155Block": 0,
        "eip158Block": 0
    },
  "alloc"      : {},
  "difficulty" : "0x400",
  "extraData"  : "",
  "gasLimit"   : "0x7A1200",
  "parentHash"                                                                    :
"0x0000000000000000000000000000000000000000000000000000000000000000",
  "timestamp"  : "0x00"
}
```


Start the Ethereum peer node (Start the Blockchain)

Run the command:

geth –datadir=./chaindata/init./genesis.json

Now we can start Geth and connect to our own private chain

geth --datadir=./chaindata/

Minimize the terminal and open a new terminal

Type the command in the new terminal: **geth attach ipc:\\.\pipe\geth.ipc**

Type **admin.nodeInfo** to get the information about your current node

Type personal.newAccount to create as many accounts as you need

- ➔ **personal.newAccount()**

- ➔ **personal.newAccount('type your password here')** (Note: Creates an account with the password)

See the created account(s)

Type in the terminal: **eth.accounts**

Mining

- ➔ Type **miner.start()** to start mining
- ➔ Type **miner.stop()** to stop mining

Checking account balances:

- ➔ **web3.fromWei(eth.getBalance(eth.coinbase), "ether")**

Let's send some ethers from one account to another. We will send ethereum from coinbase account (the account that is mining ethers) to any of the other 2 accounts.

- ➔ **eth.sendTransaction({from: eth.coinbase, to: eth.accounts[1], value: web3.toWei(10, "ether")})**

This will return transaction hash.

Now let's get balance of 2nd account.

- ➔ **web3.fromWei(eth.getBalance(eth.accounts[1]), "ether")**

Assignment:

- ➔ **Import one of the Go-Ethereum accounts into MetaMask and perform some transactions.**
- ➔ **Write and compile a contract using Remix IDE and Deploy into the private Blockchain (Geth).**