Starting the Blocks

mkdir ethereum

vi $ethereum\_home/genesis.json

genesis.json:

{ "nonce": "0x0000000000000042", "timestamp": "0x0", "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000", "extraData": "0x0", "gasLimit": "0x8000000", "difficulty": "0x400", "mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000", "coinbase": "0x3333333333333333333333333333333333333333", "alloc": { } }

1. initialise the block geth --datadir "$ethereum\_home/youtube1" init "$ethereum\_home/genesis.json"

2. start the console geth --datadir "$ethereum\_home/chain5" console 2 &gt console.log

3.create a 2nd node geth --datadir "$ethereum\_home/youtube1-a" init "$ethereum\_home/genesis.json"

4. Start on a different port and specify networkid geth --datadir "$ethereum\_home/youtube1-a" --port 30304 --nodiscover --networkid 1234 console 2 &gt console.log

5. get the admin.nodeInfo enode from the second instance and copy it into admin.addPeer in the first node

6. Restart specifying IPC path geth --datadir "$ethereum\_home/youtube1-a" --port 30304 --nodiscover --ipcpath "$ethereum\_home/youtube1-a/geth.ipc" --networkid 1234 console 2 &gt console.log

7 run get attach to attach another window to the node using the pic file geth attach ipc:$ethereum\_home/youtube1-a/geth.ipc

8 Key commands from the javascript console Personal.newAccount() (add password)

$ "0x3a1bfc0027351a8d65a7bfc375881ae27a87dfde"

Miner.start(1) - start mining eth.blockNumber - current block height eth.getBlock(number).miner - miner of block at that number eth.getBalance(account address) - current balance of that account

Geth Account {8c155f771ad0bcc46b760a0091a4851a68457aa7}

Create path variable:

echo 'export ethereum\_home= C:\Users\biswas\ethereum’ >> ~/.bash\_profile

To do

geth --datadir %ethereum\_home%/block1 init %ethereum\_home%/genesis.json

Genesis Block: config added

<https://github.com/anirudh16b/Ethereum/tree/Basic-Setup>

{

"nonce": "0x0000000000000042",

"timestamp": "0x00",

"parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",

"extraData": "0x00",

"gasLimit": "0x8000000",

"difficulty": "0x400",

"mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",

"coinbase": "0x3333333333333333333333333333333333333333",

"config": {

"chainId": 15,

"homesteadBlock": 0,

"eip155Block": 0,

"eip158Block": 0

},

"alloc": {

}

}

Mixhash: 64 bit—can be an arbitrary number

Nonce: 16 bit—can be an arbitrary number

Nonce and mixhash verify the PoW (Proof of Work)

Parenhash: it’s the reference to the previous block in the blockchain. Its always set to zero (0).

Extradata: can hold 64 character value that is built into the genesis block.

Gaslimit: Maximum amount of gas a single block can have.

Difficulty: how many transactions a miner will have to make to mine a successful block into the blockchain.

Coinbase: the address to which the ether would be sent on successful mining. Basically the miner’s account. On the genesis block is set to some random value, 64 character.

Alloc: specific to genesis block. No other block in the blockchain would have this. Key value pairs of the account address and the initial balance (pre-mine)

Genesis Block Initiation:

C:\Users\biswas\ethereum>geth --datadir "%ethereum\_home%/block1" init "%ethereum\_home%/genesis.json"

Start the console:

geth --datadir "%ethereum\_home%/block1" console 2>console.log

Console 1:

In windows, to pl ace the geth.ipc file in the specific folder

geth --datadir %ethereum\_home%/block1 --port 30303 --nodiscover --ipcpath %ethereum\_home%/geth.ipc --networkid 1234 console

Console 2:

Connect to block1 node-

geth --datadir %ethereum\_home%/block2 --port 30304 --nodiscover --networkid 1234 **--ipcdisable** console 2>console.log

From the geth Console window (after starting):

INFO [09-06|19:41:21.480] IPC endpoint opened url=\\\\.\\pipe\\C:\\Users\\biswas\\ethereum/geth.ipc

Restart specifying IPC path geth --datadir ethereum\_home/block2 --port 30304 --nodiscover --ipcpath %ethereum\_home%/block2/geth.ipc --networkid 1234 console 2>console.log

run get attach to attach another window to the node using the ipc file

geth attach ipc:%ethereum\_home%/block2/geth.ipc

Admin.nodeInfo

{

enode: "enode://9a49765e042542aed66a41dc944b5833446e119f3f09807bc31b38cb236723fd279cf0bc96a51ea530285e1c02f67219439a8d7a4947d64cd80c8f95e4bcae2b@172.16.41.166:30303",

id: "9a49765e042542aed66a41dc944b5833446e119f3f09807bc31b38cb236723fd279cf0bc96a51ea530285e1c02f67219439a8d7a4947d64cd80c8f95e4bcae2b",

ip: "172.16.41.166",

listenAddr: "[::]:30303",

name: "Geth/v1.8.14-stable-316fc7ec/windows-amd64/go1.10.3",

ports: {

discovery: 30303,

listener: 30303

},

protocols: {

eth: {

config: {

chainId: 15,

eip150Hash: "0x0000000000000000000000000000000000000000000000000000000000000000",

eip155Block: 0,

eip158Block: 0,

homesteadBlock: 0

},

difficulty: 1024,

genesis: "0x6650a0ac6c5e805475e7ca48eae5df0e32a2147a154bb2222731c770ddb5c158",

head: "0x6650a0ac6c5e805475e7ca48eae5df0e32a2147a154bb2222731c770ddb5c158",

network: 1

}

}

}

Personal.newAccount()

Passphrase:hrhkhrhk10

Address: "0xd1e0f712e4d4d6a98948c914020bd054a18c9236"

Personal.listAccounts

Eth.blocknumber

Second Terminal Node:

> admin.nodeInfo

{

enode: "enode://8d63ce3bd3afcadc2faf678cdbe33d03fcbace093c6c37807a3a04dd560d459e5d8ea17ae781c11288c0574ed42fc5ccc1a114dbdf45d006d20250a08b901d94@[::]:30304?discport=0",

id: "8d63ce3bd3afcadc2faf678cdbe33d03fcbace093c6c37807a3a04dd560d459e5d8ea17ae781c11288c0574ed42fc5ccc1a114dbdf45d006d20250a08b901d94",

ip: "::",

listenAddr: "[::]:30304",

name: "Geth/v1.8.14-stable-316fc7ec/windows-amd64/go1.10.3",

ports: {

discovery: 0,

listener: 30304

},

protocols: {

eth: {

config: {

chainId: 15,

eip150Hash: "0x0000000000000000000000000000000000000000000000000000000000000000",

eip155Block: 0,

eip158Block: 0,

homesteadBlock: 0

},

difficulty: 1024,

genesis: "0x6650a0ac6c5e805475e7ca48eae5df0e32a2147a154bb2222731c770ddb5c158",

head: "0x6650a0ac6c5e805475e7ca48eae5df0e32a2147a154bb2222731c770ddb5c158",

network: 1234

}

}

}

Miner.start(1)

miner.stop()

Block Address: "0xd1e0f712e4d4d6a98948c914020bd054a18c9236"

Solidity

pragma solidity ^0.4.0;

contract greeter {

string greeting;

function greet (string name) public {

greeting = name;

}

function getGreet() **constant** returns (string) {

return greeting;

}

}

Solidity

ABI: [

{

"constant": true,

"inputs": [],

"name": "getGreet",

"outputs": [

{

"name": "",

"type": "string"

}

],

"payable": false,

"stateMutability": "view",

"type": "function"

},

{

"constant": false,

"inputs": [

{

"name": "name",

"type": "string"

}

],

"name": "greet",

"outputs": [],

"payable": false,

"stateMutability": "nonpayable",

"type": "function"

}

]

Web3 Provider Endpoint: <http://localhost:8545>

Deploy a local blockchain network where we can mine our blockchain:

Make sure geth is started: geth --datadir %ethereum\_home%/block1 init %ethereum\_home%/genesis.json

geth --datadir %ethereum\_home%/block1 --nodiscover --rpc --rpcport "8545" --rpccorsdomain "\*" console 2>console.log

Then change the endpoint to Web3 provider in the solidity html.

Unlock your account on the geth client

Personal.unlockAccount

What is Ethereum?

State

* Account
  + Contracts
  + External
* Address
  + Balance
    - Ether