**PRACTICAL: 3**

**AIM:** Configure Geth (Go-Ethereum) over Windows/Linux/Mac operating system.

Perform the following tasks:

* Build Your Own Ethereum Private Blockchain with two peer nodes locally.
* Build Your Own Ethereum Private Blockchain with two peer nodes as geographically distributed.
* Configure Ethereum testnet such as Sepolia, Goerli and Layer 2.

**Part 1: Local Private Blockchain Setup**

**Step 1: Creating Genesis Block**

First, we created a genesis.json file with the following configuration:

{

"config": { "chainId": 1337,

"homesteadBlock": 0,

"eip150Block": 0,

"eip155Block": 0,

"eip158Block": 0,

"byzantiumBlock": 0,

"constantinopleBlock": 0,

"petersburgBlock": 0,

"istanbulBlock": 0

},

"difficulty": "1",

"gasLimit": "8000000", "alloc": {},

"coinbase": "0x0000000000000000000000000000000000000000",

"extraData": "",

"nonce": "0x0000000000000042",

"mixHash":

"0x0000000000000000000000000000000000000000000000000000000000000000",

"parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",

"timestamp": "0x00"

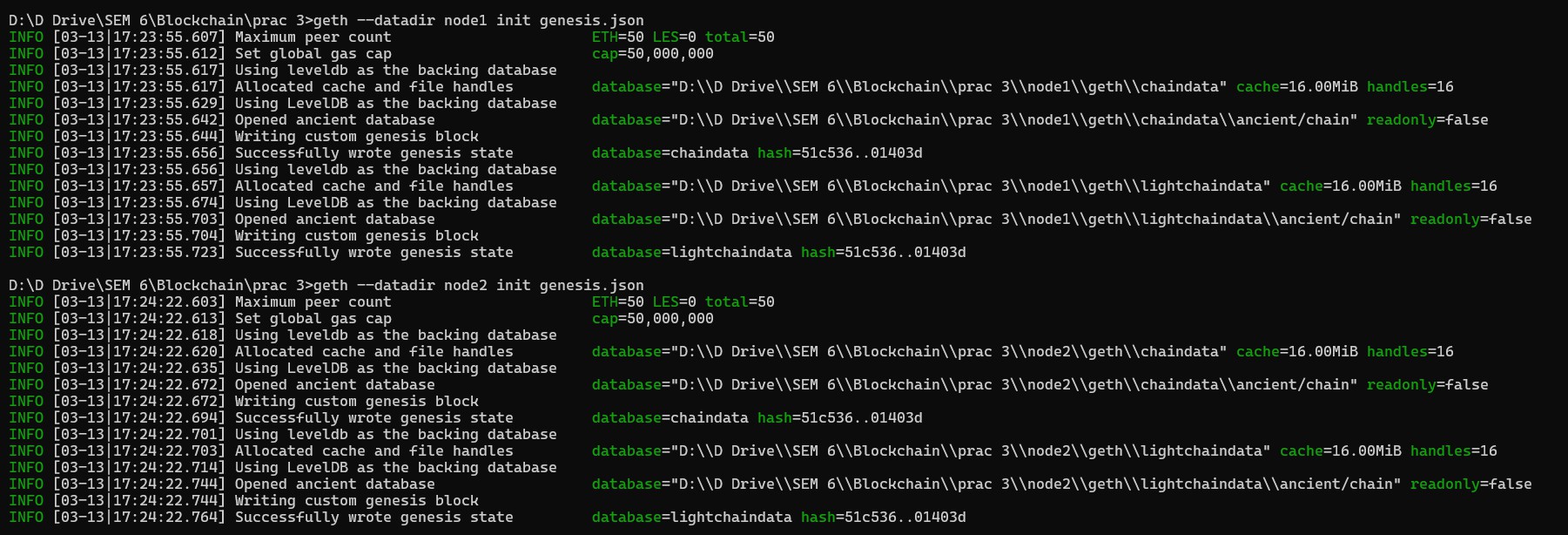
}

[Screenshot 1: Genesis block configuration]

**Step 2: Initializing Nodes**

We initialized two nodes using the following commands:

geth --datadir node1 init genesis.json geth --datadir node2 init genesis.json



**Step 3: Starting Nodes**

We started both nodes with different ports:

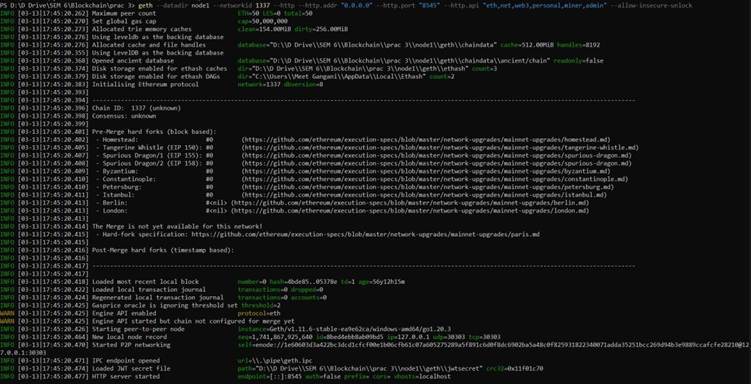
# Node 1

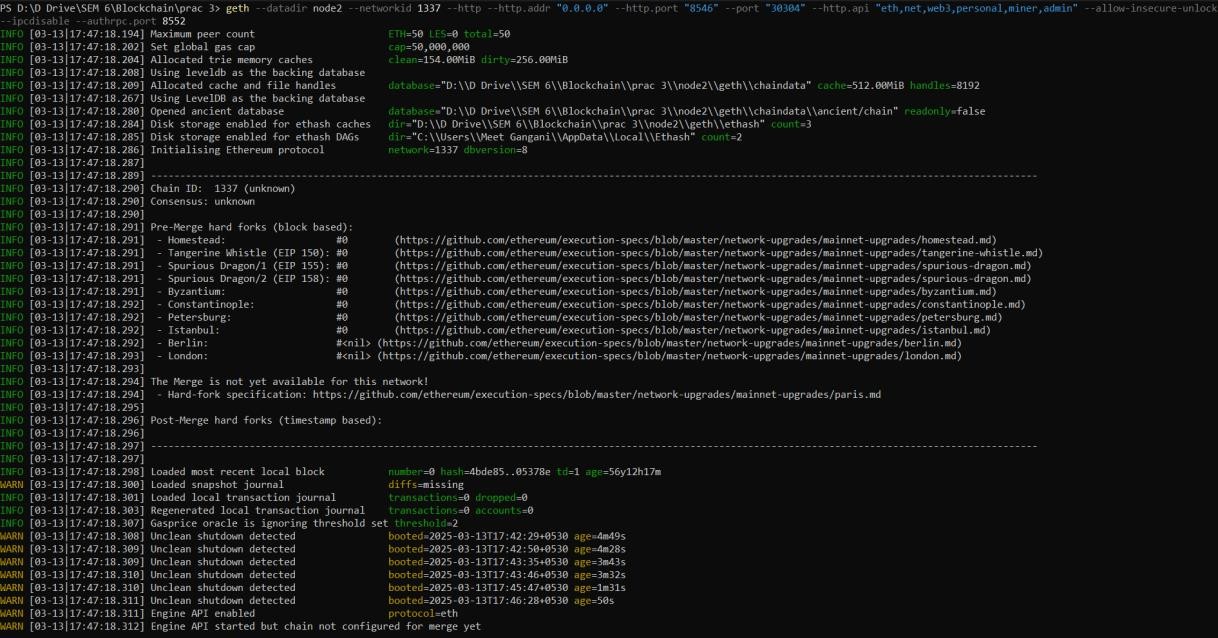
geth --datadir node1 --networkid 1337 --http --http.addr "0.0.0.0" -- http.port "8545" --http.api "eth,net,web3,personal,miner,admin" --allow- insecure-unlock

# Node 2

geth --datadir node2 --networkid 1337 --http --http.addr "0.0.0.0" -- http.port "8546" --port "30304" --http.api "eth,net,web3,personal,miner,admin" --allow-insecure-unlock --ipcdisable -- authrpc.port 8552

|  |
| --- |
|  |
|  |  |





**Step 4: Creating Account on Node 1**

Connected to Node 1 and created an account:

geth attach [http://localhost:8545](http://localhost:8545/)

Then in the console:

personal.newAccount("abc1234") eth.accounts

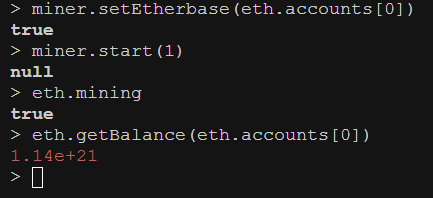
|  |
| --- |
|  |
|  |  |

## **Step 5: Setting up Mining on Node 1**

Configured and started mining on Node 1:

miner.setEtherbase(eth.accounts[0]) miner.start(1)

eth.mining eth.getBalance(eth.accounts[0])



**Part 2: Geographically Distributed Nodes**

**Step 1: Getting Node 1's Enode URL**

Retrieved the enode URL from Node 1:

admin.nodeInfo.enode

|  |
| --- |
|  |
|  |  |

**Step 2: Finding Public IP Address**

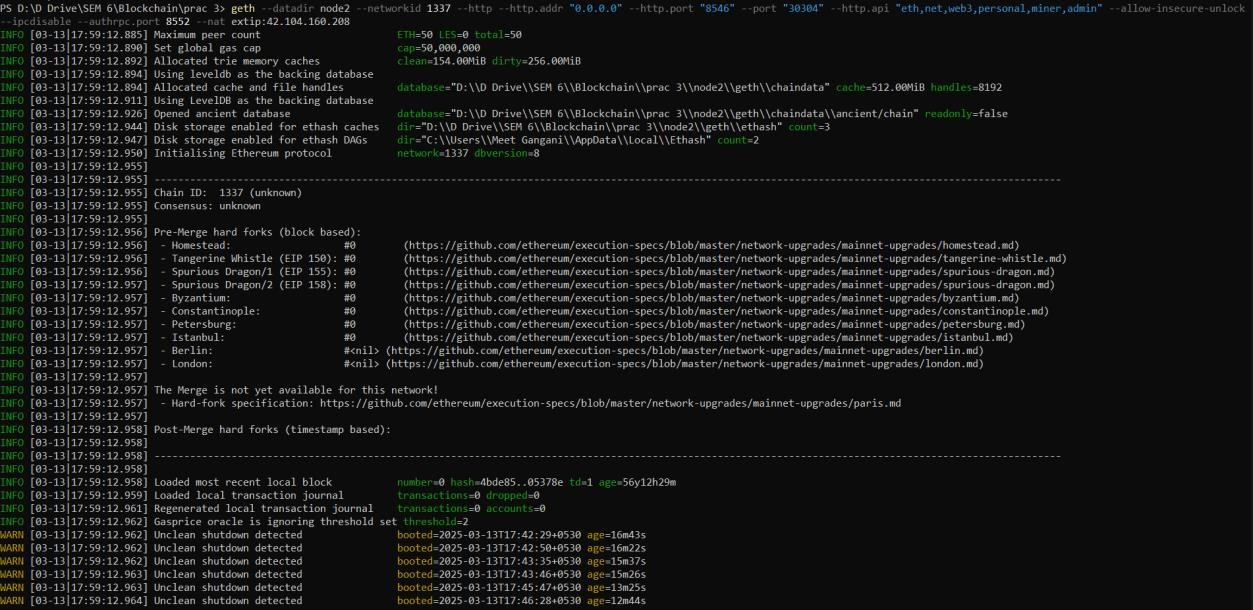
Found the public IP address using:

|  |
| --- |
|  |
|  |  |

**Step 3: Starting Node 2 with External Access**

Started Node 2 with modified parameters for external access:

geth --datadir node2 --networkid 1337 --http --http.addr "0.0.0.0" -- http.port "8546" --port "30304" --http.api "eth,net,web3,personal,miner,admin" --allow-insecure-unlock --ipcdisable -- authrpc.port 8552 --nat extip:42.104.160.208



**Step 4: Creating Account on Node 2**

Connected to Node 2 and created a new account: geth attach [http://localhost:8546](http://localhost:8546/)

Then in the console:

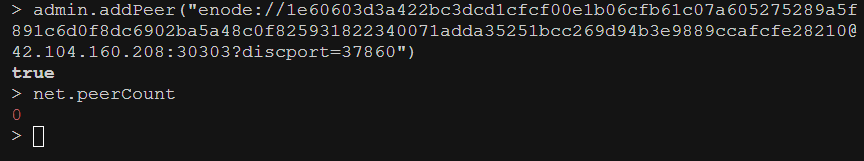
personal.newAccount("your\_password") eth.accounts

|  |
| --- |
|  |
|  |  |

**Step 5: Connecting Nodes**

Connected Node 2 to Node 1 using the enode URL:

admin.addPeer("PASTE\_NODE1\_ENODE\_URL\_HERE") net.peerCount



**Part 3: Testnet Configuration**

**Step 1: Sepolia Testnet Setup**

Initialized and started Sepolia testnet node:

geth --datadir sepolia init genesis.json

geth --datadir sepolia --networkid 11155111 --http --http.addr "0.0.0.0" -- http.port "8545" --http.api "eth,net,web3,personal,miner,admin" --allow- insecure-unlock

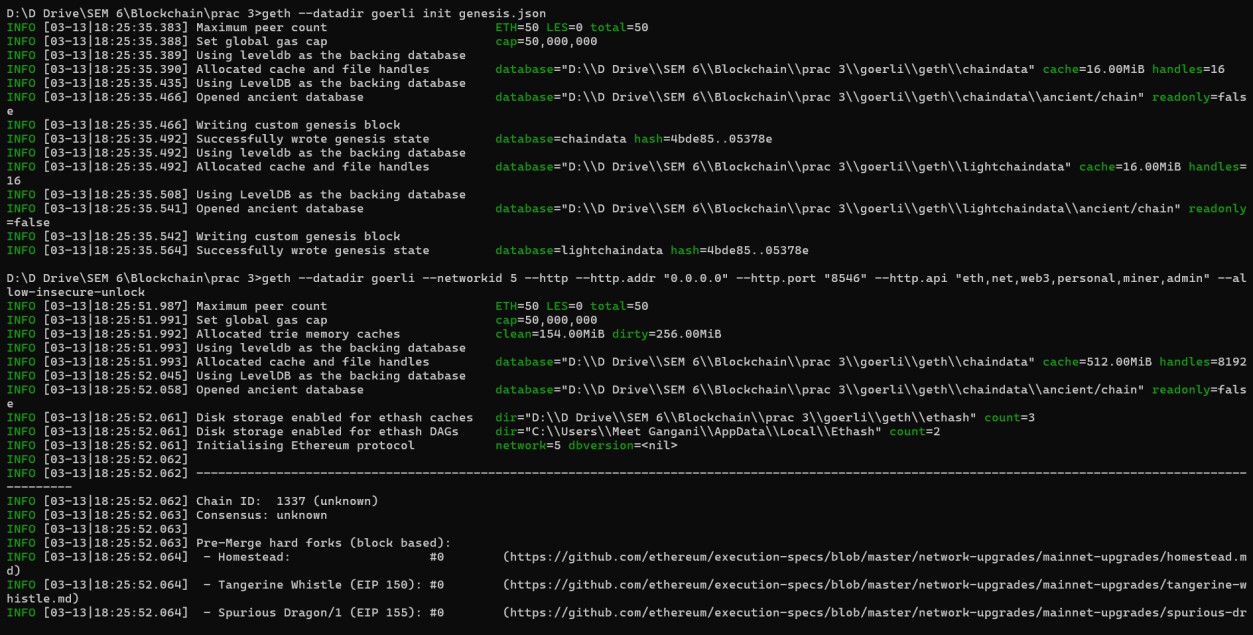
|  |
| --- |
|  |
|  |  |

**Step 2: Goerli Testnet Setup**

Initialized and started Goerli testnet node:

geth --datadir goerli init genesis.json

geth --datadir goerli --networkid 5 --http --http.addr "0.0.0.0" -- http.port "8545" --http.api "eth,net,web3,personal,miner,admin" --allow- insecure-unlock



**Results and Observations**

1.     **Local Private Blockchain:**

* Successfully created and initialized two nodes
* Established mining on Node 1
* Created accounts on both nodes
* Connected nodes successfully

2.     **Geographically Distributed Setup:**

* Configured Node 2 for external access
* Established connection between geographically distributed nodes
* Monitored network latency and connectivity

3.     **Testnet Configuration:**

* Successfully connected to Sepolia testnet
* Successfully connected to Goerli testnet
* Verified network synchronization

**LEARNING OUTCOME:**

To set up an Ethereum Private Blockchain using Geth, first, install and configure it on Windows, Linux, or Mac. After setup, initialize a genesis block and create a private network with two peer nodes, both locally and geographically distributed. This enables decentralized communication and secure transaction validation. Additionally, explore Ethereum testnets like Sepolia, Goerli, and Layer 2 solutions (e.g., Polygon) to test smart contracts and DApps in a real blockchain-like environment without using real ETH.