

DA51 Lab Session 2: The Geth client

Description:

This session teaches us how to set up a private Ethereum network with Geth, focusing on the Clique Proof of Authority (PoA) consensus algorithm. It covers creating accounts, a genesis block, and signer account keys for the network. The session also explains the structure of the extradata field in the genesis block for Clique Proof of Authority.

A private Ethereum blockchain is created, initialized, and two nodes are established. A bootnode is configured to facilitate peer-to-peer connections, and the nodes are launched with unique IDs and ports. Ether is transferred between accounts on the nodes, demonstrating transaction functionality.

Transfer ether between addresses using Wei units.

Question 5:

```
PS C:\dev\lab-session-2> geth --datadir node1 account new
INFO [09-18|14:07:15.461] Maximum peer count          ETH=50 total=50
Your new account is locked with a password. Please give a password. Do not forget this password.
Password:
Repeat password:

Your new key was generated

Public address of the key:   0x547a58C011f6121F913C378F7338B95A43526353
Path of the secret key file: node1\keystore\UTC--2024-09-18T12-07-19.616153200Z--547a58c011f6121f913c378f7338b95a43526353

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!
```

The command creates a keyfile that is stored in the keystore path

Question 10:

The command initializes the database. To connect the two nodes later, this initialization must be done in both directories.

```
PS C:\dev\lab-session-2> geth init --datadir node1 genesis.json
INFO [09-18|16:15:57.383] Maximum peer count          ETH=50 total=50
WARN [09-18|16:15:57.391] Lowering memory allowance on 32bit arch available=8826 addressable=2048
WARN [09-18|16:15:57.391] Sanitizing cache to Go's GC limits provided=1024 updated=682
INFO [09-18|16:15:57.393] Set global gas cap          cap=50,000,000
INFO [09-18|16:15:57.393] Initializing the KZG Library backend=gokzg
INFO [09-18|16:15:57.750] Defaulting to pebble as the backing database
INFO [09-18|16:15:57.750] Allocated cache and file handles database=C:\dev\lab-session-2\node1\geth\chaindata cache=16.00MiB handles=16
INFO [09-18|16:15:57.800] Opened ancient database      database=C:\dev\lab-session-2\node1\geth\chaindata\ancient\chain readonly=false
INFO [09-18|16:15:57.801] State schema set to default scheme=hash
INFO [09-18|16:15:57.802] Writing custom genesis block
INFO [09-18|16:15:57.806] Persisted trie from memory database nodes=3 size=405.00B time=2.1806ms gcnodes=0 gcsizes=0.00B gctime=0s livenodes=0 livenessize=0.00B
INFO [09-18|16:15:57.826] Successfully wrote genesis state database=chaindata hash=166a0e..3ccd7f
INFO [09-18|16:15:57.826] Defaulting to pebble as the backing database
INFO [09-18|16:15:57.826] Allocated cache and file handles database=C:\dev\lab-session-2\node1\geth\lightchaindata cache=16.00MiB handles=16
INFO [09-18|16:15:57.889] Opened ancient database      database=C:\dev\lab-session-2\node1\geth\lightchaindata\ancient\chain readonly=false
INFO [09-18|16:15:57.890] State schema set to default scheme=hash
INFO [09-18|16:15:57.890] Writing custom genesis block
INFO [09-18|16:15:57.893] Persisted trie from memory database nodes=3 size=405.00B time=2.6261ms gcnodes=0 gcsizes=0.00B gctime=0s livenodes=0 livenessize=0.00B
INFO [09-18|16:15:57.909] Successfully wrote genesis state database=lightchaindata hash=166a0e..3ccd7f
```

Question 12:

```
PS C:\dev\lab-session-2> bootnode -nodekey boot.key -addr :30305
enode://ec572066a2059f2f330b7f2a9bc53e3cdc7b12614c90fd94c155103be7a2
da15247f96f62b42bae60029b862dd3c1c35159d878134d433b73b1483ede2b904c5
@127.0.0.1:0?discport=30305
Note: you're using cmd/bootnode, a developer tool.
We recommend using a regular node as bootstrap node for production d
eployments.
INFO [09-18|14:21:26.532] New local node record                      s
eq=1,726,662,086,529 id=10c5af03f5e031bf ip=<nil> udp=0 tcp=0
```

Question 13:

First command: `geth --datadir node1 --port 30307 --bootnodes`

```
enode://7f851530fc477c9f183d02719e4066e4252b6bd2572d29d4f3d57e785701b6a19a8
7c12c499327d5cbcd5dac5afbd437ab57cc69c02c5619f65f327d9da40580@127.0.0.1:0?d
iscport=30305 --networkid 1234567890 --unlock
B6E9D58c3A76f9E5640e6920cA03dF9d30FcD331 --password node1/password.txt --
authrpc.port 8551 --ipcpath node1 --miner.etherbase
0xB6E9D58c3A76f9E5640e6920cA03dF9d30FcD331
```

Second command: `geth --datadir node2 --port 30308 --bootnodes`

```
enode://7f851530fc477c9f183d02719e4066e4252b6bd2572d29d4f3d57e785701b6a19a8
7c12c499327d5cbcd5dac5afbd437ab57cc69c02c5619f65f327d9da40580@127.0.0.1:0?d
iscport=30305 --networkid 1234567890 --unlock
c2024d10C9F18176A0Eb290F9e35DAd6F10BeeF4 --password node2/password.txt --
authrpc.port 8552 --ipcpath node2
```

I needed to add `--ipcpath` because of a forbidden access of the start of the two commands at the same time. In the same way, I needed to add `--miner.etherbase` to sepecific the etherbase in order to mine, requirement for something later in the TP.

Lab Session 2

Question 15:

As I specified in the previous question the `-ipcpath` the command look like : `geth attach` `\\.\pipe\node1` for the node 1 and `geth attach` `\\.\pipe\node2` for the node 2.

```
PS C:\dev\lab-session-2> geth attach \\.\pipe\node1
Welcome to the Geth JavaScript console!

instance: Geth/v1.13.15-stable-c5ba367e/windows-386/go1.21.6
coinbase: 0xb6e9d58c3a76f9e5640e6920ca03df9d30fcd331
at block: 2 (Wed Sep 18 2024 16:26:05 GMT+0200 (CEST))
datadir: C:\dev\lab-session-2\node1
modules: admin:1.0 clique:1.0 debug:1.0 engine:1.0 eth:1.0 miner:1.0 net:1.0 rpc:1.0 txpool:1.0 web3:1.0

To exit, press ctrl-d or type exit
```

```
PS C:\Users\jules> geth attach \\.\pipe\node2
Welcome to the Geth JavaScript console!

instance: Geth/v1.13.15-stable-c5ba367e/windows-386/go1.21.6
at block: 0 (Thu Jan 01 1970 01:00:00 GMT+0100 (CET))
datadir: C:\dev\lab-session-2\node2
modules: admin:1.0 clique:1.0 debug:1.0 engine:1.0 eth:1.0 miner:1.0 net:1.0 rpc:1.0 txpool:1.0 web3:1.0

To exit, press ctrl-d or type exit
```

Question 16:

```
> net.peerCount
1
```

Question 17:

```
> admin.peers
[
  {
    caps: ["eth/68", "snap/1"],
    enode: "enode://baea567b4a411d5052fa5b83de8bc0ef1ba0020a4670bb0c21f884feab7b2449a3bb460ef552146f52d9dbdbf1b2b982b2dfa68a6655c4b31ff8c5c9bcb6b982@127.0.0.1:58004",
    id: "4b566c159451beea618364c4f8e7ac3d1408d4c505529101d7241f82ea69173f",
    name: "Geth/v1.13.15-stable-c5ba367e/windows-386/go1.21.6",
    network: {
      inbound: true,
      localAddress: "127.0.0.1:58007",
      remoteAddress: "127.0.0.1:58004",
      static: false,
      trusted: false
    },
    protocols: {
      eth: {
        version: 68
      },
      snap: {
        version: 1
      }
    }
  }
]
```

Question 18:

```
> eth.getBalance(eth.accounts[0])
9.99999999934101971356999e+22
```

Question 19:

```
eth.sendTransaction({to: 'c2024d10C9F18176A0Eb290F9e35DAd6F10BeeF4', from:  
eth.accounts[0], value: 25000});
```

```
> eth.sendTransaction({to: 'c2024d10C9F18176A0Eb290F9e35DAd6F10BeeF4', from: eth.accounts[0], value: 25000});  
"0x7f459fe9b10a7ebb012be79e2b629bbdb964f550d81aac81207780817d8f524"
```

Question 20:

In order to get the value, we need to start the miner by miner.start()

```
> miner.start()  
null
```

```
> eth.getBalance('c2024d10C9F18176A0Eb290F9e35DAd6F10BeeF4');  
125000
```

Question 21:

```
PS C:\Users\jules> geth attach \\.pipe\node2  
Welcome to the Geth JavaScript console!  
  
instance: Geth/v1.13.15-stable-c5ba367e/windows-386/go1.21.6  
at block: 0 (Thu Jan 01 1970 01:00:00 GMT+0100 (CET))  
datadir: C:\dev\lab-session-2\node2  
modules: admin:1.0 clique:1.0 debug:1.0 engine:1.0 eth:1.0 miner:1.0 net:1.0 rpc:1.0 txpool:1.0 web3:1.0  
To exit, press ctrl-d or type exit
```

Question 22:

```
> eth.getBalance(eth.accounts[0])  
125000
```

Question 23:

Yes, it matches

Question 24:

Balance in Ethereum is represented in Wei by default.

Question 25:

1 ether is equal to 10^{18} Wei

Lab Session 2

Question 26:

List of command in this order:

- mkdir node3
- geth --datadir node3 account new

```
PS E:\Documents\UTBM\Cours\DA51\Lab session 2> geth --datadir node3 account new
INFO [10-08|16:48:08.822] Maximum peer count          ETH=50 total=50
Your new account is locked with a password. Please give a password. Do not forget this password.
Password:
Repeat password:

Your new key was generated

Public address of the key: 0xe56E032e6430d1D3A0D0221e5869103fAA4d934F
Path of the secret key file: node3\keystore\UTC--2024-10-08T14-48-12.925717600Z--e56e032e6430d1d3a0d0221e5869103faa4d934f

- You can share your public address with anyone. Others need it to interact with you.
- You must NEVER share the secret key with anyone! The key controls access to your funds!
- You must BACKUP your key file! Without the key, it's impossible to access account funds!
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!
```

- geth init --datadir node3 genesis.json

```
PS E:\Documents\UTBM\Cours\DA51\Lab session 2> geth init --datadir node3 genesis.json
INFO [10-08|16:45:15.327] Maximum peer count          ETH=50 total=50
WARN [10-08|16:45:15.331] Lowering memory allowance on 32bit arch available=32581 addressable=2848
WARN [10-08|16:45:15.331] Sanitizing cache to Go's GC limits provided=1024 updated=682
INFO [10-08|16:45:15.331] Set global gas cap          cap=50,000,000
INFO [10-08|16:45:15.331] Initializing the KZG library backend=gokzg
INFO [10-08|16:45:15.432] Defaulting to pebble as the backing database
INFO [10-08|16:45:15.432] Allocated cache and file handles database="E:\Documents\UTBM\Cours\DA51\Lab session 2\node3\geth\chaindata" cache=16.00MiB handles=16
INFO [10-08|16:45:15.457] Opened ancient database      database="E:\Documents\UTBM\Cours\DA51\Lab session 2\node3\geth\chaindata\ancient\chain" readonly=false
INFO [10-08|16:45:15.457] State schema set to default scheme=hash
INFO [10-08|16:45:15.457] Writing custom genesis block
INFO [10-08|16:45:15.459] Persisted trie from memory database nodes=3 size=405.00B time=1.5574ms gcnodes=0 gcsz=0.00B gctime=0s livenodes=0 livesize=0.00B
INFO [10-08|16:45:15.470] Successfully wrote genesis state database=chaindata hash=fca880..c53498
INFO [10-08|16:45:15.470] Defaulting to pebble as the backing database
INFO [10-08|16:45:15.470] Allocated cache and file handles database="E:\Documents\UTBM\Cours\DA51\Lab session 2\node3\geth\lightchaindata" cache=16.00MiB handles=16
INFO [10-08|16:45:15.498] Opened ancient database      database="E:\Documents\UTBM\Cours\DA51\Lab session 2\node3\geth\lightchaindata\ancient\chain" readonly=false
INFO [10-08|16:45:15.498] State schema set to default scheme=hash
INFO [10-08|16:45:15.498] Writing custom genesis block
INFO [10-08|16:45:15.500] Persisted trie from memory database nodes=3 size=405.00B time=1.5357ms gcnodes=0 gcsz=0.00B gctime=0s livenodes=0 livesize=0.00B
INFO [10-08|16:45:15.511] Successfully wrote genesis state database=lightchaindata hash=fca880..c53498
```

Lab Session 2

- `geth --datadir node3 --port 30309 --bootnodes enode://13327bbcc754edccb30b6aaf6383238f49629259899cf5c84d8bba8df51b8a427190a3c80a7b068f4b24f0a651f8f94180d514141ea30589f24356e14ee70f98@127.0.0.1:0?discport=30305 --networkid 1234567890 --unlock e56E032e6430d1D3A0D0221e5869103fAA4d934F --password password.txt --authrpc.port 8553 --ipcpath node3`

```
PS E:\Documents\UTBM\Cours\DA51\Lab session 2> geth --datadir node3 --port 30309 --bootnodes enode://13327bbcc754edccb30b6aaf6383238f49629259899cf5c84d8bba8df51b8a427190a3c80a7b068f4b24f0a651f8f94180d514141ea30589f24356e14ee70f98@127.0.0.1:0?discport=30305 --networkid 1234567890 --unlock e56E032e6430d1D3A0D0221e5869103fAA4d934F --password password.txt --authrpc.port 8553 --ipcpath node3
INFO [10-08]16:49:50.277] Maximum peer count               <nil>=50  <nil>=50
WARN [10-08]16:49:50.279] Lowering memory allowance on 32bit arch  available=32581  addressable=2848
INFO [10-08]16:49:50.279] Sanitizing cache to Go's GC limits      provided=1024  updated=682
INFO [10-08]16:49:50.280] Set global gas cap                       cap=50,000,000
INFO [10-08]16:49:50.280] Initializing the KZG library             backend=gokzg
INFO [10-08]16:49:50.388] Allocated trie memory caches            clean=102.00MiB  dirty=170.00MiB
INFO [10-08]16:49:50.388] Using pebble as the backing database
INFO [10-08]16:49:50.388] Allocated cache and file handles
INFO [10-08]16:49:50.407] Opened ancient database
INFO [10-08]16:49:50.411] State scheme set to already existing
INFO [10-08]16:49:50.412] Initialising Ethereum protocol
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414] Chain ID: 1234567890 (unknown)
INFO [10-08]16:49:50.414] Consensus: Clique (proof-of-authority)
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414] Pre-Merge hard forks (block based):
INFO [10-08]16:49:50.414] - Homestead: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/homestead.md)
INFO [10-08]16:49:50.414] - Tangerine Whistle (EIP 150): #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/tangerine-whistle.md)
INFO [10-08]16:49:50.414] - Spurious Dragon/1 (EIP 155): #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/spurious-dragon.md)
INFO [10-08]16:49:50.414] - Spurious Dragon/2 (EIP 158): #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/spurious-dragon.md)
INFO [10-08]16:49:50.414] - Byzantium: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/byzantium.md)
INFO [10-08]16:49:50.414] - Constantinople: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/constantinople.md)
INFO [10-08]16:49:50.414] - Petersburg: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/petersburg.md)
INFO [10-08]16:49:50.414] - Istanbul: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/istanbul.md)
INFO [10-08]16:49:50.414] - Muir Glacier: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/muir-glacier.md)
INFO [10-08]16:49:50.414] - Berlin: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/berlin.md)
INFO [10-08]16:49:50.414] - London: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/london.md)
INFO [10-08]16:49:50.414] - Arrow Glacier: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/arrow-glacier.md)
INFO [10-08]16:49:50.414] - Gray Glacier: #0 (https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/gray-glacier.md)
INFO [10-08]16:49:50.414] The Merge is not yet available for this network!
INFO [10-08]16:49:50.414] - Hard-fork specification: https://github.com/ethereum/execution-specs/blob/master/network-upgrades/mainnet-upgrades/paris.md
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414] Post-Merge hard forks (timestamp based):
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414]
INFO [10-08]16:49:50.414] Loaded most recent local block          number=0  hash=fca880..c53498  td=1  age=55y6mo3w
WARN [10-08]16:49:50.416] Failed to load snapshot                 err="missing or corrupted snapshot"
INFO [10-08]16:49:50.419] Rebuilding state snapshot
INFO [10-08]16:49:50.421] Initialized transaction indexer         range="Last 2350000 blocks"
INFO [10-08]16:49:50.421] Resuming state snapshot generation      root=042841..01ba01  accounts=0  slots=0  storage=0.008  dangling=0  elapsed=2.515ms
INFO [10-08]16:49:50.424] Generated state snapshot                accounts=2  slots=0  storage=88.008  dangling=0  elapsed=5.170ms
INFO [10-08]16:49:50.424] Enabled snap sync                       head=0  hash=fca880..c53498
INFO [10-08]16:49:50.491] Gasprice oracle is ignoring threshold set threshold=2
INFO [10-08]16:49:50.493] Stored checkpoint snapshot to disk       number=0  hash=fca880..c53498
WARN [10-08]16:49:50.495] Engine API enabled                     protocol=meth
WARN [10-08]16:49:50.495] Engine API started but chain not configured for merge yet
INFO [10-08]16:49:50.496] Starting peer-to-peer node              instance=Geth/v1.13.15-stable-c5ba367e/windows-386/go1.21.6
INFO [10-08]16:49:50.505] New local node record                   seq=1,728,398,990,504  id=c8caf62813c94e64  ip=127.0.0.1  udp=30309  tcp=30309
INFO [10-08]16:49:50.505] Started P2P networking                  self=enode://bf497c4bd8221686df87228a0b649edd52188847b858509b26dfbd89aa17ec4b99519efd40882d099b22b1f3a1e59419af5f6cb82d31f778c7e6ba937e6098f8
127.0.0.1:30309
INFO [10-08]16:49:50.506] IPC endpoint opened                     url=\\.\pipe\node3
INFO [10-08]16:49:50.506] Generated JMT secret                    path="E:\Documents\UTBM\Cours\DA51\Lab session 2\node3\geth\jmtsecret"
INFO [10-08]16:49:50.511] WebSocket enabled                       url=ws://127.0.0.1:8553
```

To send money to node3 from node1:

```
eth.sendTransaction({ to: "e56E032e6430d1D3A0D0221e5869103fAA4d934F", from: eth.accounts[0], value: 25000});
```

```
> eth.sendTransaction({ to: "e56E032e6430d1D3A0D0221e5869103fAA4d934F", from: eth.accounts[0], value: 25000});
"0xa647c24a5fcd7f2f5b5c2d396f72f5e068adefea3cea949e65d6700f111e0bfcd"
```

Conclusion:

Geth commands are used to initialize Ethereum nodes, specifying datadir, port, bootnodes, networkid, unlock accounts, and other parameters. Transactions are sent between nodes using `eth.sendTransaction`, and miners are started with `miner.start()`. Balance is displayed in Wei, with 1 ether equal to 10^{18} Wei.