

# Лабораторная работа 1.

## Часть 2

### Работа с приватной сетью блокчейна Ethereum с помощью geth

#### Теоретический блок

Для работы с geth необходимо знать основной набор аргументов командной строки.

Ознакомиться с полным набором аргументов можно здесь:

<https://github.com/ethereum/go-ethereum/wiki/Command-Line-Options>

Geth консоль предназначена для разработчиков блокчейна. Она предоставляет следующие пространства имен для работы с API:

- **admin**: управление нодой (узлом) блокчейна
- **debug**: дебаг
- **miner**: майнинг и работа с DAG
- **personal**: работа с аккаунтами
- **txpool**: отслеживание пула транзакций.

Основные и самые популярные команды представлены в таблице ниже:

<a href="#">admin</a>	<a href="#">debug</a>	<a href="#">miner</a>	<a href="#">personal</a>	<a href="#">txpool</a>
<a href="#">addPeer</a>	<a href="#">stacks</a>	<a href="#">setGasPrice</a>	<a href="#">listAccounts</a>	<a href="#">content</a>
<a href="#">datadir</a>	<a href="#">startGoTrace</a>	<a href="#">start</a>	<a href="#">lockAccount</a>	<a href="#">inspect</a>
<a href="#">nodeInfo</a>	<a href="#">stopGoTrace</a>	<a href="#">stop</a>	<a href="#">newAccount</a>	<a href="#">status</a>
<a href="#">peers</a>	<a href="#">traceBlock</a>	<a href="#">setEtherbase</a>	<a href="#">unlockAccount</a>	
	<a href="#">traceBlockByNumber</a>		<a href="#">sendTransaction</a>	
	<a href="#">traceBlockByHash</a>		<a href="#">sign</a>	
	<a href="#">traceTransaction</a>			
	<a href="#">verbosity</a>			

Основные параметры **конфигурационного файла сети** (genesis.json) описаны [здесь](#).

В следующей команде прописываем **enode** первого узла. Его можно узнать, прописав в javascript консоли команду `admin.nodeInfo`.

2.1 Открываем новое окно Командной строки и запускаем первый узел из первой части нашей лабораторной работы.

Инициализация не требуется, поэтому достаточно просто перейти в каталог, где лежит наш конфигурационный файл сети (genesis.json) и ввести команду запуска geth консоли. Она будет иметь примерно следующий вид:

```
geth --datadir node1kurbatova --networkid 9807 console
```

После запуска введите команду

```
admin.nodeInfo
```

для получения информации о состоянии узла. Скопируйте из выведенного текста значение параметра enode.

```
> admin.nodeInfo
{
  enode: "enode://1240e8e3741453eabc48a74d9af57ac831c1d42efee18bdb24b80541fb20cb3614b572bce810877a07c4f6dbbed9920169956f9288af90896c43232062be0d9@37.213.175.145:30303?discport=1024",
  enr: "enr:-J-4Qn8r050z00k107k1CAj0tB170ZAG6Jz1V0tPnaUrT6551rPphnQ90Cwm9Xr8zerCn1r0M01Y0Y5qL6mKvnn0Xy4wgZV0amT0nLkAw0A9mIkgnY0gm1whCXVr5GJc2VjcDI1NmsxoQMSQ0jddBRT6rxIp02a9XrIMcHULv7h19skuAVB-yDLNoN0Y3CCd1-DdWRwggQAhhVkcDaCd18",
  id: "f4134a304e14f654dd987a036b32df22461088cf08fc7de3ad7827c6b40c20d2",
  ip: "37.213.175.145",
  listenAddr: "[::]:30303",
  name: "Geth/v1.9.5-stable-a1c09b93/windows-amd64/go1.13",
  ports: {
    discovery: 1024,
    listener: 30303
  },
  protocols: {
    eth: {
      config: {
        chainId: 1907,
        eip150Hash: "0x0000000000000000000000000000000000000000000000000000000000000000",
        eip155Block: 0,
        eip158Block: 0,
        homesteadBlock: 0
      },
      difficulty: 4861514,
      genesis: "0x357b5eb0b98079e70bdd6cc479ac0b2a749320b2f1dc09b8fcd691db331b20b9",
      head: "0xdd2d073bb3b92bab7390ef9c815f0eed430caa040b3ec0cee7b70268a620213",
      network: 9807
    }
  }
}
```

2.2 Запускаем сеть на втором узле командой:

!!! **networkid** должен совпадать со значением, которое вы использовали при запуске предыдущей ноды.

```
geth --datadir node2 --networkid 9807 --port 30304 --bootnodes
```

```
"enode://08993401988acce4cd85ef46a8af10d1cacad39652c98a9df4d5785248d1910e51d7f3d330f0a96053001264700c7e94c4ac39d30ed5a5f79758774208adaa1f@127.0.0.1:30303" --
ipcdisable console
```

!!! Запускать второй узел следует в новом окне Командной строки. Следовательно, после запуска у нас будут 2 окна с двумя работающими узлами.

```
Командная строка - geth --datadir node2kurbatova --networkid 9807 --port 30304 --bootnodes enode://1240e8e3741453eabc48a74d9af57ac831c1d42efee...
C:\university\blockchain\lab1>geth --datadir node2kurbatova --networkid 9807 --port 30304 --bootnodes enode://1240e8e3741453eabc48a74d9af57ac831c1d42efee18bdb24b80541fb20cb3614b572bce810877a07c4f6dbbed9920169956f09288af90896c43232062be0d9@127.0.0.1:30303 --ipcdisable console
INFO [09-29|16:52:21.837] Maximum peer count               ln=50 ltn=0 tota=50
INFO [09-29|16:52:21.990] Starting peer-to-peer node       instance=Geth/v1.9.5-stable-a1c09b93/windows-amd64/go1.13
INFO [09-29|16:52:22.000] Allocated trie memory caches     clean=256.00MiB dirty=256.00MiB
INFO [09-29|16:52:22.006] Allocated cache and file handles database=D:\\university\\blockchain\\lab1\\node2kurbatova\\geth\\chaindata
INFO [09-29|16:52:22.006] Opened ancient database          database=D:\\university\\blockchain\\lab1\\node2kurbatova\\geth\\chaindata\\ancient
INFO [09-29|16:52:22.057] Initialised chain configuration  config="{ChainID: 1907 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: <nil> EIP155: 0 EIP158: 0 Byzantium: <nil> Constantinople: <nil> Petersburg: <nil> Istanbul: <nil> Engine: unknown}"
INFO [09-29|16:52:22.079] Disk storage enabled for ethash caches dir=D:\\university\\blockchain\\lab1\\node2kurbatova\\geth\\ethash count=3
INFO [09-29|16:52:22.088] Disk storage enabled for ethash DAGs dir=C:\\Users\\polin\\AppData\\Local\\Ethash count=2
INFO [09-29|16:52:22.100] Initialising Ethereum protocol  versions=[63] network=9807 dbversion=7
INFO [09-29|16:52:22.199] Loaded most recent local header  number=0 hash=357b5e...1b20b9 td=10 age=50y5mo2w
INFO [09-29|16:52:22.207] Loaded most recent local full block number=0 hash=357b5e...1b20b9 td=10 age=50y5mo2w
INFO [09-29|16:52:22.217] Loaded most recent local fast block number=0 hash=357b5e...1b20b9 td=10 age=50y5mo2w
INFO [09-29|16:52:22.225] Loaded local transaction journal transactions=0 dropped=0
INFO [09-29|16:52:22.235] Regenerated local transaction journal transactions=0 account=0
INFO [09-29|16:52:22.257] Allocated fast sync bloom        size=512.00MiB
INFO [09-29|16:52:22.269] Initialized fast sync bloom       items=0 errorrate=0.000 elapsed=0s
INFO [09-29|16:52:22.311] New local node record            seq=5 id=724039934aad7aea ip=127.0.0.1 udp=30304 tcp=30304
INFO [09-29|16:52:22.445] Started P2P networking           self=enode://9d41891c02aea88515a5cbe8105392e1819046bd2b3f5694dfef6c7076282a912a693fc6bd1376df096bb971a47f810cf9efc232950794c26f710263505aa2fa@127.0.0.1:30304
INFO [09-29|16:52:22.543] Ethereum automatically configured address=0x2129ef718FE8b2E5310900386A5b968888FFCE8f
Welcome to the Geth JavaScript console!
```

Ждём, когда начнётся и завершится синхронизация:

```
> INFO [09-29|16:52:52.447] Block synchronisation started
INFO [09-29|16:52:52.458] Imported new state entries       count=3 elapsed=0s processed=3 pending=0 retry=0 duplicate=0 unexpected=0
INFO [09-29|16:52:53.482] Imported new block headers      count=44 elapsed=1.023s number=44 hash=cdb834...20764d age=10m
INFO [09-29|16:52:53.496] Imported new chain segment      blocks=19 txs=0 mgas=0.000 elapsed=4.986ms mgasps=0.000
INFO [09-29|16:52:53.513] Imported new chain segment      blocks=25 txs=3 mgas=0.063 elapsed=7.979ms mgasps=7.896
INFO [09-29|16:52:53.523] Fast sync complete, auto disabling
```

### 3. Проверяем работу

3.1 В консоли, где развернут первый узел с помощью команды `eth.coinbase` узнаем адрес майнера, после чего выведем его баланс:

```
> eth.coinbase
"0xeb57bd4191b628946317b49b404eee2bad210825"
> eth.getBalance(eth.coinbase)
236000021000000000000000
```

3.2 Во второй консоли проверим баланс этого же адреса с помощью команды

`eth.getBalance("addr")`

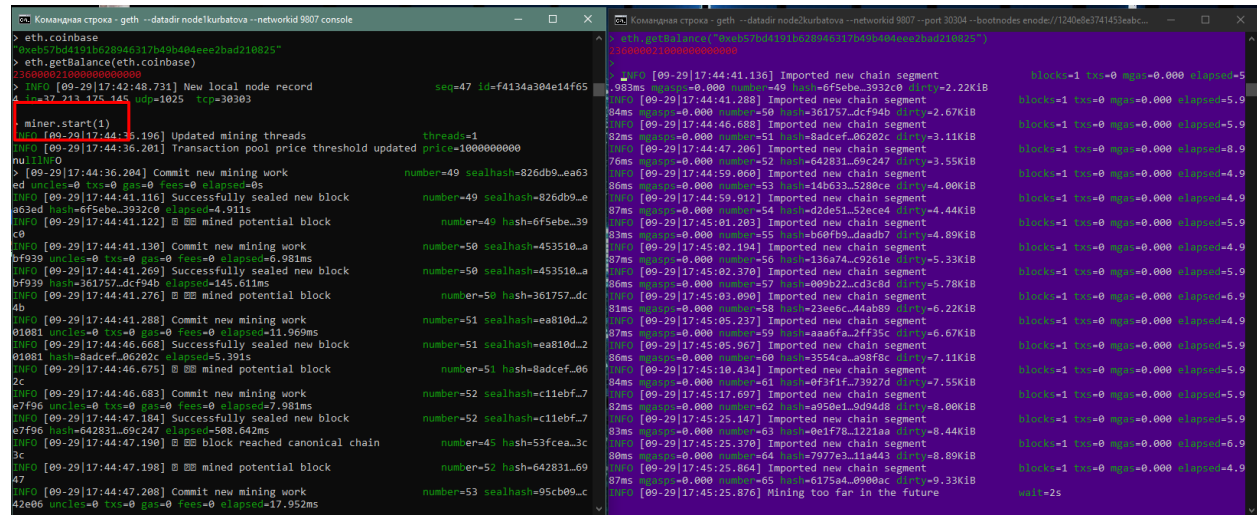
где `addr` – это адрес майнера из первой консоли.

```
> eth.getBalance("0xeb57bd4191b628946317b49b404eee2bad210825")
236000021000000000000000
>
```

Балансы должны совпадать.

3.3 Запустим майнинг на первой консоли и посмотрим, что во второй выводятся логи.

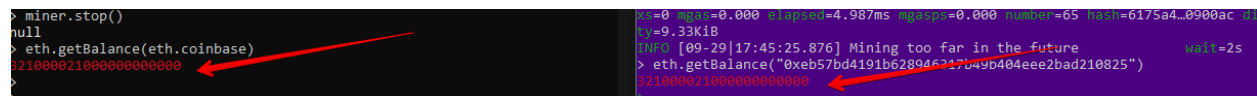
Подождав некоторое время, остановим майнинг.



```
> eth.coinbase
"0xeb57bd4191b628946217b49b404eee2bad210825"
> eth.getBalance(eth.coinbase)
21000021000000000000
INFO [09-29|17:42:48.731] New local node record seq=47 id=f4134a304e14f65
4 10-27-213.175.145 http=1025 tcp=30303

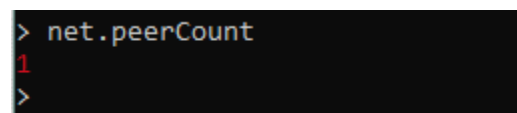
> miner.start(1)
INFO [09-29|17:42:48.196] Updated mining threads threads=1
INFO [09-29|17:44:36.281] Transaction pool price threshold updated price=1000000000
nullINFO
> [09-29|17:44:36.284] Commit new mining work number=49 sealhash=826db9_ea63
ed uncles=0 txs=0 gas=0 fees=0 elapsed=6ms
INFO [09-29|17:44:41.116] Successfully sealed new block number=49 sealhash=826db9_e
a63ed hash=6f5ebe_3932c0 elapsed=4.911s
INFO [09-29|17:44:41.122] mined potential block number=49 hash=6f5ebe_39
32c0
INFO [09-29|17:44:41.130] Commit new mining work number=50 sealhash=453510_a
bf939 uncles=0 txs=0 gas=0 fees=0 elapsed=6.981ms
INFO [09-29|17:44:41.269] Successfully sealed new block number=50 sealhash=453510_a
bf939 hash=361757_dcf94b elapsed=145.611ms
INFO [09-29|17:44:41.276] mined potential block number=50 hash=361757_dc
f94b
INFO [09-29|17:44:41.288] Commit new mining work number=51 sealhash=ea810d_2
01081 uncles=0 txs=0 gas=0 fees=0 elapsed=11.969ms
INFO [09-29|17:44:46.668] Successfully sealed new block number=51 sealhash=ea810d_2
01081 hash=badcef_06292c elapsed=5.301s
INFO [09-29|17:44:46.675] mined potential block number=51 hash=badcef_06
292c
INFO [09-29|17:44:46.683] Commit new mining work number=52 sealhash=c11ebf_7
e7f96 uncles=0 txs=0 gas=0 fees=0 elapsed=7.981ms
INFO [09-29|17:44:47.184] Successfully sealed new block number=52 sealhash=c11ebf_7
e7f96 hash=642831_69c247 elapsed=508.642ms
INFO [09-29|17:44:47.190] block reached canonical chain number=45 hash=53fcea_3c
3c
INFO [09-29|17:44:47.198] mined potential block number=52 hash=642831_69
c247
INFO [09-29|17:44:47.208] Commit new mining work number=53 sealhash=95cb09_c
42e06 uncles=0 txs=0 gas=0 fees=0 elapsed=17.952ms
```

3.4 снова сверим баланс майнера с первой ноды.



```
> miner.stop()
null
> eth.getBalance(eth.coinbase)
21000021000000000000
> eth.getBalance("0xeb57bd4191b628946217b49b404eee2bad210825")
21000021000000000000
```

3.5 Командой net.peerCount проверим, что к первому узлу сейчас подключен только один узел:



```
> net.peerCount
1
>
```

4. Подключение третьего узла

4.1 Создадим папку, новый аккаунт-майнер и инициализируем третий узел по аналогии со вторым. (для этого придётся открыть третье окно Командной строки)

!!! Папка должна называться node3\*, где \* - это фамилия студента

```

D:\university\blockchain\lab1>geth --datadir node3kurbatova account new
INFO [09-29|18:00:21.333] Maximum peer count          FD=50 FS=0 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: 0xbAb8a2269C5Dd9B990774df10FBaAB0FD9EB46D1
Path of the secret key file: node3kurbatova\keystore\UTC--2019-09-29T15-00-27.221683700Z--bab8a2269c5dd9b990774df10fbaab0fd9eb46d1

- 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!

D:\university\blockchain\lab1>geth --datadir node3kurbatova init genesis.json
INFO [09-29|18:00:47.902] Maximum peer count          FD=50 FS=0 total=50
INFO [09-29|18:00:47.985] Allocated cache and file handles      database=D:\university\blockchain\lab1\node3kurbatova\geth\chaindata cache=16.00MiB handles=16
INFO [09-29|18:00:48.002] Writing custom genesis block
INFO [09-29|18:00:48.005] Persisted trie from memory database    nodes=0 size=0.00B time=0s gcnodes=0 gcsz=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [09-29|18:00:48.013] Successfully wrote genesis state      database=chaindata hash=357b5e1b20b9
INFO [09-29|18:00:48.019] Allocated cache and file handles      database=D:\university\blockchain\lab1\node3kurbatova\geth\chaindata cache=16.00MiB handles=16
INFO [09-29|18:00:48.037] Writing custom genesis block
INFO [09-29|18:00:48.040] Persisted trie from memory database    nodes=0 size=0.00B time=0s gcnodes=0 gcsz=0.00B gctime=0s livenodes=1 livesize=0.00B

```

4.2 Запустим третий узел для начала подключив его только ко второму.

Команда будет выглядеть примерно следующим образом:

```

geth --datadir node3 --networkid 9807 --port 30305 --bootnodes
enode://08993401988acce4cd85ef46a8af10d1cacad39652c98a9df4d5785248d1910e51d7f3d3
30f0a96053001264700c7e94c4ac39d30ed5a5f79758774208adaa1f@127.0.0.1:30303 --
ipcdisable console

```

```

INFO [09-29|18:42:57.737] Persisted trie from memory database    nodes=0 size=0.00B time=0s gcnodes=0 gcsz=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [09-29|18:42:57.747] Successfully wrote genesis state      database=lightchaindata hash=357b5e1b20b9

D:\university\blockchain\lab1>geth --datadir node3kurbatova --networkid 9807 --port 30305 --bootnodes enode://1240e8e3741453eabc48a74d9af57ac831c1d42efe11
b2db24b80541fb20cb3614b572bce810877a07c4f6d8bed9920169956f09288af90896c43232062be0d9@127.0.0.1:30303 --ipcdisable console
INFO [09-29|18:43:06.354] Maximum peer count          FD=50 FS=0 total=50
INFO [09-29|18:43:06.455] Starting peer-to-peer node          instance=Geth/v1.9.5-stable-a1c09b93/windows-amd64/go1.13
INFO [09-29|18:43:06.460] Allocated trie memory caches        clean=256.00MiB dirty=256.00MiB
INFO [09-29|18:43:06.466] Allocated cache and file handles    database=D:\university\blockchain\lab1\node3kurbatova\geth\chaindata cache=512.00MiB handles=8192
INFO [09-29|18:43:06.500] Opened ancient database            database=D:\university\blockchain\lab1\node3kurbatova\geth\chaindata\ancient
INFO [09-29|18:43:06.508] Initialized chain configuration      config="{ChainID: 1907 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: <nil> EIP155: 0 EIP158: 0 Byzantium: <nil> Constantinople: <nil> Petersburg: <nil> Istanbul: <nil> Engine: unknown}"
INFO [09-29|18:43:06.522] Disk storage enabled for ethash caches dir=D:\university\blockchain\lab1\node3kurbatova\geth\ethash count=3
INFO [09-29|18:43:06.536] Disk storage enabled for ethash DAGs dir=C:\Users\polin\AppData\Local\Ethash count=2
INFO [09-29|18:43:06.548] Initialising Ethereum protocol      version=[63] network=9807 dbversion=<nil>
WARN [09-29|18:43:06.554] Upgrade blockchain database version from=<nil> to=7
INFO [09-29|18:43:06.651] Loaded most recent local header      number=0 hash=357b5e1b20b9 tx=10 age=50y5mo2w
INFO [09-29|18:43:06.657] Loaded most recent local full block  number=0 hash=357b5e1b20b9 tx=10 age=50y5mo2w
INFO [09-29|18:43:06.663] Loaded most recent local fast block  number=0 hash=357b5e1b20b9 tx=10 age=50y5mo2w
INFO [09-29|18:43:06.673] Loaded local transaction journal     transactions=0 dropped=0
INFO [09-29|18:43:06.678] Regenerated local transaction journal transactions=0 accounts=0
INFO [09-29|18:43:06.694] Allocated fast sync bloom           size=512.00MiB
INFO [09-29|18:43:06.697] Initialized fast sync bloom          time=0 appropriate=0.000 elapsed=0s
INFO [09-29|18:43:06.737] New local node record               seq=19 tx=0fdb6fee1cbd8237 tx=127.0.0.1 node=30305 port=30305
INFO [09-29|18:43:06.744] Started P2P networking              enode://11c9e94c61791c42f227a635f5bd04ad530b2d6482e92735f8bc2ce0dc85b38b46f21fa9a2
INFO [09-29|18:43:06.847] Ethereum automatically configured    address=0xbAb8a2269C5Dd9B990774df10FBaAB0FD9EB46D1
Welcome to the Geth JavaScript console!

instance: Geth/v1.9.5-stable-a1c09b93/windows-amd64/go1.13
coinbase: 0xbAb8a2269c5dd9b990774df10fbaab0fd9eb46d1
at block: 0 (Thu, 01 Jan 1970 03:00:00 +03)
datadir: D:\university\blockchain\lab1\node3kurbatova

```

Необходимо подождать, пока завершится синхронизация.



Если вы увидели ошибку «**Synchronisation failed, retrying**», попробуйте включить майнинг на первой ноде, пока синхронизация не будет завершена.

Чтобы проверить, что синхронизация завершилась успешно, используйте команду `eth.syncing`

Результат команды должен быть: `false`.

```
5  udp=1026  tcp=30303  
> eth.syncing  
false  
>
```

4.3 Командой `net.peerCount` проверим, что к первому узлу сейчас подключено 2 узла:

```
5  udp=1026  tcp=30303  
> net.peerCount  
2
```

4.4 Командой `net.peerCount` проверим, что ко второму узлу всё ещё подключен 1 узел:

```
> net.peerCount  
1  
>
```

5. Завершите работу всех узлов командой `exit`.

## Контрольные вопросы

1. Назовите команду `geth` консоли для получения баланса аккаунта.
2. Опишите назначения аргументов командной строки для следующей `geth` команды:  
`geth --datadir node2 --networkid 9876 --port 30303 --rpc -rpcport 8545 --etherbase "0x0060672fce7ae3d14379dbfd4e45c9631710a527" console --rpcapi "db,eth,net,web3,personal,admin" --rpccorsdomain "192.168.1.0" --rpcaddr "0.0.0.0"`
3. Опишите назначение параметров **nonce**, **difficulty** и **gasLimit** в конфигурационном файле `genesis.json`.
4. Почему значение параметра `difficulty` в тестовых сетях предпочитают сохранять низким?
5. Объясните простыми словами, что такое майнинг.

### **Задание**

- 1) Изучить теоретический материал
- 2) Выполнить практическую часть лабораторной работы
- 3) Ответить на контрольные вопросы
- 4) Оформить отчёт

**Защита лабораторной работы 1 (часть 2):** отчёт должен быть отправлен в google classroom. Пункт 4 должен быть продемонстрирован преподавателю на паре.

### **Общие требования к отчёту**

Отчёт должен содержать следующие элементы:

- 1) номер и название лабораторной работы,
- 2) ФИО студента, группу и курс
- 3) выполненные задания практического блока: номер задания, скриншот рабочего окна в соответствии с требованиями.
- 4) краткие ответы на контрольные вопросы.

В случае обнаружения плагиата (в том числе и в ответах на контрольные вопросы) отчёт не будет принят преподавателем!