

区块链原理与实践

实验五 以太坊私有链搭建

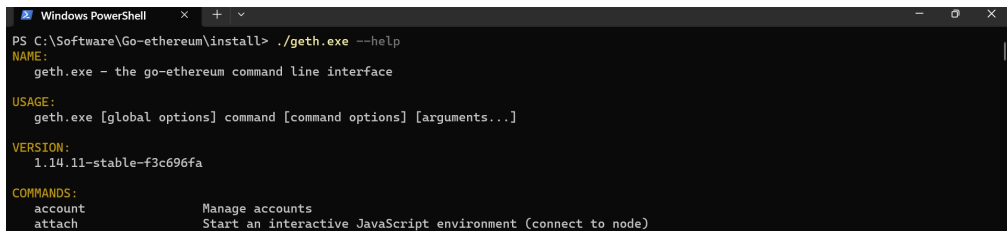
实验目的

使用geth客户端搭建私有链，学习过程中用到的各个命令及选项的含义和作用

实验过程

go-ethereum客户端安装

在[以太坊Geth下载界面](#)下载windows安装程序，进行安装测试



```
Windows PowerShell
PS C:\Software\Go-ethereum\install> ./geth.exe --help
NAME:
  geth.exe - the go-ethereum command line interface

USAGE:
  geth.exe [global options] command [command options] [arguments...]

VERSION:
  1.14.11-stable-f3c696fa

COMMANDS:
  account      Manage accounts
  attach        Start an interactive JavaScript environment (connect to node)
```

准备创世区块配置文件

需执行代码：

```
cd privatechain
C:\Software\Go-ethereum\install\geth.exe --datadir data1 init genesis2.json
```

新建genesis.json配置文件，写入配置如下，**注意**，相较于实验说明，需要于config内添加"eip150Block":0,，否则会产生Fatal: Failed to write genesis block: unsupported fork ordering: eip150Block not enabled, but eip155Block enabled at block 0报错

```
Windows PowerShell
PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian> C:\Software\Go-ethereum\install\geth.exe --datadir data0 init genesis.json
INFO [11-05|16:56:29.822] Maximum peer count ETH=50 total=50
INFO [11-05|16:56:29.864] Set global gas cap cap=50,000,000
INFO [11-05|16:56:29.864] Initializing the KZG library backend=gokzg
INFO [11-05|16:56:29.906] Defaulting to pebble as the backing database
INFO [11-05|16:56:29.906] Allocated cache and file handles database=C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth\chaindata cache=16.00MiB handles=16
INFO [11-05|16:56:29.932] Opened ancient database database=C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth\chaindata\ancient\chaindata readonly=false
ERROR[11-05|16:56:29.933] Head block is not reachable
INFO [11-05|16:56:29.933] State schema set to default schema=path
INFO [11-05|16:56:29.947] Opened ancient database database=C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth\chaindata\ancient\state readonly=false
INFO [11-05|16:56:29.947] Writing custom genesis block
Fatal: Failed to write genesis block: unsupported fork ordering: eip150Block not enabled, but eip155Block enabled at block 0
PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian>
```

```
{
  "config":{
    "chainId": 10,
    "homesteadBlock":0,
    "eip155Block":0,
    "eip150Block":0,
    "eip158Block":0
  },
  "alloc":{},
  "coinbase":"0x0000000000000000000000000000000000000000",
  "difficulty":"0x20000",
  "extraData": "",
  "gasLimit":"0x2fef8",
  "nonce":"0x0000000000000042",
  "mixhash":
    "0x0000000000000000000000000000000000000000000000000000000000000000",
  "parentHash":
    "0x0000000000000000000000000000000000000000000000000000000000000000",
  "timestamp":"0x00"
}
```

成功后，文件结构如下：

```
目录: C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth

Mode                LastWriteTime         Length Name
----                -
d-----          11/5/2024   5:07 PM             chaindata
-a-----          11/5/2024   5:07 PM              0 LOCK
-a-----          11/5/2024   5:07 PM             64 nodekey

PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth> cd chaindata
PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth\chaindata> ls

目录: C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5\privatechian\data0\geth\chaindata

Mode                LastWriteTime         Length Name
----                -
d-----          11/5/2024   5:07 PM             ancient
-a-----          11/5/2024   5:07 PM          1430 000002.log
-a-----          11/5/2024   5:07 PM             16 CURRENT
-a-----          11/5/2024   5:07 PM              0 LOCK
-a-----          11/5/2024   5:07 PM             52 MANIFEST-000001
-a-----          11/5/2024   5:07 PM          2511 OPTIONS-000003
```

需执行指令：

```
C:\Software\Go-ethereum\install\geth.exe --datadir data1 --networkid 1108 console
```

```
PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian> C:\Software\Go-ethereum\install\geth.exe --datadir data0 --networkid 1108 console
INFO [11-05]20:35:32.656] Maximum peer count ETH=50 total=50
INFO [11-05]20:35:32.665] Set global gas cap cap=50,000,000
INFO [11-05]20:35:32.666] Initializing the KZG library backend=gokzg
INFO [11-05]20:35:32.688] Allocated trie memory caches clean=154.00MiB dirty=256.00MiB
INFO [11-05]20:35:32.689] Using pebble as the backing database
INFO [11-05]20:35:32.689] Allocated cache and file handles database=C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian\data0\geth\chaindata cache=512.00MiB handles=8192
INFO [11-05]20:35:32.706] Opened ancient database database=C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian\data0\geth\chaindata\ancient\chain readonly=false
INFO [11-05]20:35:32.706] State schema set to default scheme=path
Fatal: Failed to register the Ethereum service: only PoS networks are supported, please transition old ones with Geth v1.13.x
PS C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian>
```

但正如图示，我们会发现由于以太坊Geth早已更新，不再支持PoW证明，仅支持PoS证明，因此我们需要使用新的配置文件与编译指令。具体而言，在配置文件的config中，我们增加了"chainId": 1108,, 并且使用了如下的编译指令C:\Software\Go-ethereum\install\geth.exe --datadir data1 init genesis2.json, 并且通过这条指令C:\Software\Go-ethereum\install\geth.exe --datadir data1 --http --http.addr "localhost" --http.port "8545" --http.api "web3,eth,personal" --rpc.enabled deprecatedpersonal --allow-insecure-unlock console启动命令行。

```
INFO [11-05]23:01:18.724] WebSocket enabled url=ws://127.0.0.1:8551
INFO [11-05]23:01:18.725] HTTP server started endpoint=127.0.0.1:8551 auth=true prefix= cors=localhost vhosts=localhost
Welcome to the Geth JavaScript console!

instance: Geth/v1.14.11-stable-f3c696fa/windows-amd64/go1.23.1
at block: 0 (Thu Jan 01 1970 08:00:00 GMT+0800 (CST))
datadir: C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian\data1
modules: admin: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
>
```

Javascript Console 创建账户

执行命令eth.accounts进行账户验证，若不存在账户，使用personal.newAccount()进行创建。

```
Windows PowerShell
ckchainTech\5_privatechian\data1\geth\jwtsecret crc32=0x1b28e50b
INFO [11-06]12:26:28.138] HTTP server started endpoint=127.0.0.1:8545 auth=false prefix= cors= vhos
ts=localhost
INFO [11-06]12:26:28.139] WebSocket enabled url=ws://127.0.0.1:8551
INFO [11-06]12:26:28.139] HTTP server started endpoint=127.0.0.1:8551 auth=true prefix= cors=localhost
host vhosts=localhost
WARN [11-06]12:26:28.181] Enabling deprecated personal namespace reqid=3 duration=0s err="etherbase must be explicitly
WARN [11-06]12:26:28.182] Served eth_coinbase specified"
Welcome to the Geth JavaScript console!

instance: Geth/v1.11.4-stable-7e3b149b/windows-amd64/go1.20.2
at block: 0 (Thu Jan 01 1970 08:00:00 GMT+0800 (CST))
datadir: C:\Users\w1098\Documents\Code\Local\Others\5_BlockchainTech\5_privatechian\data1
modules: admin:1.0 debug:1.0 engine:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

To exit, press ctrl-d or type exit
> INFO [11-06]12:26:32.959] New local node record seq=1,730,867,146,137 id=a7f037db92a39354 ip=36.112
.3.176 udp=48766 tcp=30303
> personal.newAccount()
Passphrase: INFO [11-06]12:26:38.479] Looking for peers peercount=0 tried=18 static=0
Repeat passphrase: INFO [11-06]12:26:48.483] Looking for peers peercount=0 tried=32 static=0

INFO [11-06]12:26:51.344] Your new key was generated address=0x7405DC559d0aefbf77226C8F9ecA3a6B3Ee47b2c
WARN [11-06]12:26:51.344] Please backup your key file! path=C:\Users\w1098\Documents\Code\Local\Others\5_Blo
ckchainTech\5_privatechian\data1\keystore\UTC--2024-11-06T04-26-49.7627346002--7405dc559d0aefbf77226c8f9eca3a6b3ee47b2c
WARN [11-06]12:26:51.344] Please remember your password!
"0x7405dc559d0aefbf77226c8f9eca3a6b3ee47b2c"
> 12345678
```

记录地址及代码如下：

```
"0x7405dc559d0aefbf77226c8f9eca3a6b3ee47b2c"  
"0x671cc879f99c6cfb50d2c79b7dae1ae973d90b37"  
12345678
```

查看账户余额

执行指令`eth.getBalance(eth.accounts[0])`

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

启动&停止挖矿

执行指令绑定挖矿地址，在此以第一个用户地址为例。

`miner.setEtherbase(eth.accounts[0])`

然后可以确认地址`eth.coinbase`

可以使用`miner.start(2)`进行双线程挖矿，并使用`miner.stop()`进行挖矿中止。

值得注意的是，可以使用

```
miner.start(1);admin.sleepBlocks(1);miner.stop();
```

在挖到一个区块后自动停止。

```
INFO [11-06|12:42:52.000] Generating DAG in progress  
INFO [11-06|12:42:53.115] Generating DAG in progress  
INFO [11-06|12:42:53.558] Generating DAG in progress  
WARN [11-06|12:42:53.762] Dropping unsynced node during sync  
0303 type="Geth/ITGOLD Test Nod..."  
INFO [11-06|12:42:54.015] Generating DAG in progress  
INFO [11-06|12:42:54.485] Generating DAG in progress  
INFO [11-06|12:42:55.032] Generating DAG in progress  
INFO [11-06|12:42:55.782] Generating DAG in progress  
INFO [11-06|12:42:55.782] Generated ethash verification cache  
INFO [11-06|12:42:57.841] Looking for peers  
> miner.stop()  
null  
epoch=1 percentage=93 elapsed=45.338s  
epoch=1 percentage=94 elapsed=45.847s  
epoch=1 percentage=95 elapsed=46.289s  
id=ac860a60a75fe021 conn=dyndial addr=78.140.242.88:3  
epoch=1 percentage=96 elapsed=46.746s  
epoch=1 percentage=97 elapsed=47.217s  
epoch=1 percentage=98 elapsed=47.764s  
epoch=1 percentage=99 elapsed=48.514s  
epoch=1 elapsed=48.514s  
peercount=0 tried=44 static=0
```

发送交易

查看余额后可以进行交易，需要先解锁账户：

`personal.unlockAccount(eth.accounts[0])`

```
INFO [11-06|13:05:13.901] Looking for peers  
> personal.unlockAccount(eth.accounts[0])  
Unlock account 0x7405dc559d0aefbf77226c8f9eca3a6b3ee47b2c  
Passphrase:  
true
```

然后进行交易：`amount = web3.toWei(5, 'ether')`

但目前该交易还没有被记录，可通过以下指令查看：`txpool.status`

为使交易被记录，需要再挖一个区块。

查看交易和区块

可以使用如下指令查看区块量：`eth.blockNumber`

```
> eth.blockNumber
0
```

使用如下指令查看交易：`eth.getTransaction(<交易哈希>)`

可能由于geth版本过低，目前的挖矿结果并不能得到正常返回，会提示内存访问错误。正在搜索资料处理中，相关报错如下：

```
Windows PowerShell
> INFO [11-06|13:11:01.595] Looking for peers           peercount=0 tried=32 static=0
> miner.stoINFO [11-06|13:11:11.729] Looking for peers   peercount=0 tried=39 static=0
> miner.stop();
null
> INFO [11-06|13:11:21.913] Looking for peers           peercount=0 tried=31 static=0
INFO [11-06|13:11:32.039] Looking for peers           peercount=0 tried=45 static=0
unexpected fault address 0x80026
fatal error: fault
[signal 0xc0000005 code=0x1 addr=0x80026 pc=0x7ff6f89e335c]

goroutine 1794 [running]:
runtime.throw(0x7ff6f97713f7?, 0xc001a50a00?)
    runtime/panic.go:1047 +0x65 fp=0xc00287b730 sp=0xc00287b700 pc=0x7ff6f806d025
runtime.sigpanic()
    runtime/signal_windows.go:270 +0xd0 fp=0xc00287b778 sp=0xc00287b730 pc=0x7ff6f8081e70
runtime: g 1794: unexpected return pc for github.com/ethereum/go-ethereum/p2p/rlpx.(*readBuffer).read called from 0x8000
100000000
stack: frame={sp:0xc00287b778, fp:0xc00287b7e8} stack=[0xc002878000,0xc00287c000)
0x000000c00287b678: 0x000000c00287b6b0 0x00007ff6f806ead1 <runtime.gwrite+0x00000000000000f1>
0x000000c00287b688: 0x00007ff6f9ac3de8 0x0000000000000001
0x000000c00287b698: 0x0000000000000001 0x000000c00287b719
0x000000c00287b6a8: 0x0000000000000007 0x00007ff6f809fd03 <runtime.systemstack+0x0000000000000033>
0x000000c00287b6b8: 0x00007ff6f806d36c <runtime.fatalthrow+0x000000000000006c> 0x000000c00287b6d0
0x000000c00287b6c8: 0x000000c001570d00 0x00007ff6f806d3a0 <runtime.fatalthrow.func1+0x0000000000000000>
0x000000c00287b6d8: 0x000000c001570d00 0x00007ff6f806d025 <runtime.throw+0x0000000000000065>
0x000000c00287b6e8: 0x000000c00287b700 0x000000c00287b720
0x000000c00287b6f8: 0x00007ff6f806d025 <runtime.throw+0x0000000000000065> 0x000000c00287b708
0x000000c00287b708: 0x00007ff6f806d040 <runtime.throw.func1+0x0000000000000000> 0x00007ff6f97713f7
0x000000c00287b718: 0x0000000000000005 0x000000c00287b768
0x000000c00287b728: 0x00007ff6f8081e70 <runtime.sigpanic+0x00000000000000d0> 0x00007ff6f97713f7
```