Geth 测试链调试记录：

============================================================================

1、初始化创世块:：

**geth --datadir** "./myblockchaindir" **init** genesis.json

注意json初始配置的“chainId”必须与任何其他网链不同，这是唯一标识！

**geth --datadir "myblockchain" --networkid 84348 console --nodiscover 2>output.log**

指定http参数(默认8545端口)才能本地监听合约部署：

geth --datadir "myblockchain" --networkid 84348 console **--http --allow-insecure-unlock** --nodiscover 2>20211228.log

允许不安全的HTTP通道的账户解锁：--allow-insecure-unlock

、允许局域网内的访问（非127默认）：

geth --datadir "myblockchain" --networkid **84348** **--preload** "./mineWhenNeeded.js" console --http --http.addr "0.0.0.0" --http.api admin,debug,web3,eth,txpool,personal,ethash,miner,net --http.corsdomain '\*' --allow-insecure-unlock --nodiscover 2>20220127.log

（加载了有交易才挖矿的js脚本）

注意http.addr配置：“0.0.0.0”或者“节点IP”都可以，就是不能不配置，默认为“127.0.0.1”，无法外部访问！

指定访问端口参数：--http.port 8545

开启后用浏览器访问http://192.168.11.88:8545，可以连接到私链了，如果浏览器钱包连接到此网页，可以看到此私链账户余额（账户已经导入到私链geth内）。

2、启动Miner上已初始化好的的geth，即开启区块链客户端：

**geth --datadir "mychain01" --networkid 84349** console --http --http.addr "0.0.0.0" --http.corsdomain '\*' --allow-insecure-unlock --nodiscover 2>20220107.log

矿机（2022-01-14）：

**geth --datadir "mychain02" --networkid 84350** console --http --http.addr "0.0.0.0" --http.port 8547 --http.corsdomain '\*' --allow-insecure-unlock --nodiscover 2>20220114.log

添加API调试命令支持：

--http.api admin,debug,web3,eth,txpool,personal,ethash,miner,net

、矿机重装后dev模式启动（2022-01-27）：

geth --datadir "myblkchain02" --networkid **84351** console --http --http.addr "0.0.0.0" --http.corsdomain '\*' --allow-insecure-unlock --nodiscover --preload "**./mineWhenNeeded.js**" 2>20220127.log

（？ --http.api admin,debug,web3,eth,txpool,personal,ethash,miner,net ）

（http://192.168.11.**120**:8545/；84351；）

#节点开启后，可以远程连接geth节点操作：

geth attach http://127.0.0.1:8545（8547）

注意用远程连接的方式，一些Console命令执行报错，例如miner.系列命令；

-------------------------------------------------------------------------------------------------------------------------------------

To enable the HTTP server, use the --http flag.

geth --http

By default, geth accepts connections from the loopback interface (127.0.0.1). The default listening port is 8545. You can customize address and port using the **--http.port** and **--http.addr** flags.

geth --http --http.port 3334

JSON-RPC method namespaces must be whitelisted in order to be available through the HTTP server. An RPC error with error code -32602 is generated if you call a namespace that isn’t whitelisted. The default whitelist allows access to the “eth” and “shh” namespaces. To enable access to other APIs like account management (“personal”) and debugging (“debug”), they must be configured via the --http.api flag. We do not recommend enabling such APIs over HTTP, however, since access to these methods increases the attack surface.

**geth --http --http.api personal,eth,net,web3**

Since the HTTP server is reachable from any local application, additional protection is built into the server to prevent misuse of the API from web pages. If you want enable access to the API from a web page, you must configure the server to accept Cross-Origin requests with the --http.corsdomain flag.

Example: if you want to use Remix with geth, allow requests from the remix domain.

**geth --http --http.corsdomain https://remix.ethereum.org**

Use **--http.corsdomain '\*'** to enable access from any origin.

--------------------------------------------------------------------------------------------------------------------------------------

之前版本中geth需要使用--rpc连接本地网络，在新版本中则采用了--http，具体参考**[geth官方文档](https://geth.ethereum.org/docs/)**

含有这种写法的文章比较久，现在连接命令有所更改，这是按照了较新版的命令行但是参考了较老的教程导致的。

更改

geth --datadir data --networkid 1330 --http --http.corsdomain="\*" --http.port 8545 --http.api db,web3,eth,debug,personal,net,miner,admin --allow-insecure-unlock --rpc.allow-unprotected-txs --port 30303 --dev --dev.period 1 console 2>>geth.log

============================================================================

从外部钱包（小狐狸）账户导入到私链geth上账户：

参考：https://blog.csdn.net/weixin\_43988498/article/details/108394012

账户详情->导出私钥->在geth系统下创建一个空白文件把私钥拷贝进去：

**geth account import “私钥文件路径”**

提示输入新账户密码（控制台下使用账户需要的密码）

新账户生成，私钥与账户名都与你原钱包上账户一样，钱包连接到这个私链上时也会认为是同一账户。

**geth account list** 查看所有账户的存储位置，此时需要你手动把这个新账户文件移动到keystore目录下，你就可以在控制台用eth.accounts查看新增的账户了。

在使用此账户挖矿，如果钱包连接到了私链，就能动态看到挖矿的成果了。

------------------------------------------------------------------------------------------------------

、从geth的**keystore内账户导入到**MetaMask**钱包**（反向操作）：

直接把keystore目录下账户对应文件用MM“导入账户”，选择类型：JSON文件（keystore内保存的账户文件其实都是JSON格式），输入账户密码，导入，大约要等等几分钟到十几分钟，就可以查看钱包导入了新账户，连接到http站点对应geth私链上ETH也正确显示了。

因为没找到直接方法从keystore内账户导出私钥（可以用Web3接口调用实现），就直接硬导入JSON了。

============================================================================

**geth控制台，console:**

# 创建账户

personal.newAccount(“your passw”)

# 解锁账户

personal.**unlockAccount**(“……”)

# 解锁账户，指定解锁具体账户

personal.unlockAccount(eth.accounts[0])

# 列出系统中的账户

eth.accounts

# 1、查看账户余额,返回值的单位是 Wei (“ ”里面是自己管理的账户地址)

eth.getBalance(appoint account)

# 2、查看账户余额,返回值的单位是 Wei (“ ”里面是自己管理的账户地址)

eth.getBalance("写上账户地址")

# 3、转换为单位ether,便于阅读

web3.fromWei(eth.getBalance("写上账户地址"),'ether')

# 4、如果是在里面创建的账户可以调用内部函数拿到地址，不用每次都复制地址。

eth.getBalance(eth.accounts[0])

web3.fromWei(eth.getBalance(eth.accounts[0]),'ether')

#发起交易（发起方需要是自己管理的账户，其次需要先解锁账户），from：发起交易的地址；to:接受交易的地址

eth.sendTransaction({from:eth.accounts[0],to:"接受交易的地址",value:100000})

# 列出当前区块高度

eth.blockNumber

# 获取交易信息

eth.getTransaction

# 获取区块信息

eth.getBlock()

#查询合约

eth.contract()

# 开始挖矿

miner.start()

# 表示占用一核挖矿

miner.start(1)

# 查看 当前账户

eth.coinbase

# 停止挖矿

miner.stop()

# 开始挖矿,当挖到一个块时就停止，

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

# Wei 换算成以太币

web3.fromWei()

# 以太币换算成 Wei

web3.toWei()

# 交易池中的状态

txpool.status

#设置当前挖矿账户

miner.**setEtherbase**("0x0301dcfd859d01f8507af8929e41b366425c00ec")

#管理员查看

admin

查看节点信息：admin.nodeInfo

查看节点列表：admin.peers

查看节点个数：net.peerCount

net.listening

#查看合约（参数：交易hash，transactionHash）

eth.**getTransaction**("0x9f8098805e9eaf9e3157cf98333042aef553bac21b88d4c54c01d2603593c2bf")

#发起合约交易（发起者账户需要先解锁）

> eth.**sendTransaction**({from:eth.accounts[0],to:eth.accounts[1],value:web3.toWei(10,'ether')})

"0x9587300dbcc7897368fbf72f4608691cbcc753cd357ed801587678a351447c11"

用truffle console发起交易：

**web3.eth.sendTransaction**({from:"0xced27893eea69b3ff1e57b5ce40f19662e9296f5",to:"0x0301dcfd859d01f8507af8929e41b366425c00ec",value:'100000000000000000000'})

**、创建已部署的合约的实例，用来调用合约文件：**

1、先查看变量是否占用：

> var dnc

undefined

2、创建合约实例(DncToken合约)：

> DncToken.deployed().then((instance)=>{dnc=instance})

undefined

3、查看合约：

> dnc

；会显示DncToken全部合约内容：

4、执行合约中的函数：（前解锁，中挖矿，后完成）

> **dnc.**finishMinting()

--------------------------------------------------------------------------------------------------------------------------------

执行交易：

personal.unlockAccount(eth.coinbase)

//记得先解锁账户

contract.函数名.sendTransaction("参数1","参数2","参数...",{

from:"0xe473d288faf6c2d1812dbb759d7423605c7d1a58"})

//上面是在geth控制台调用transactionn类型（要付gas类的）函数的方法。函数名要替换成自己的，括号内部前部分为函数的参数，分别列出，最后一项大括号内from要写付费的用户的地址（换成自己的）。

contract.函数名.call("参数1","参数2", "参数...")

//调用call类型函数的方法，（不要付gas类的）。

--------------------------------------------------------------------------------------------------------------------------------

-----------------------------------------------------------------------------------------------------------------------------------

账户解锁出错：

> personal.unlockAccount(eth.coinbase)

Unlock account 0xced27893eea69b3ff1e57b5ce40f19662e9296f5

Passphrase:

**GoError: Error:** account unlock with HTTP access is forbidden at web3.js:6357:37(47)

at github.com/ethereum/go-ethereum/internal/jsre.MakeCallback.func1 (native)

at <eval>:1:24(4)

解决方法：添加启动参数--allow-insecure-unlock

原因：新版本geth，出于安全考虑，默认禁止了HTTP通道解锁账户

============================================================================

ethminer设置的资源：

https://www.csdn.net/tags/OtTaMg1sNjk2NTQtYmxvZwO0O0OO0O0O.html

============================================================================

Geth如何连接外测试网（Ropsten/Rinkey）：

https://github.com/inoutcode/ethereum\_book/blob/master/%E7%AC%AC%E5%9B%9B%E7%AB%A0.asciidoc

--------------------------------------------------------------------------------

同步到Ropsten：

**geth --ropsten --syncmode snap --cache=1024 --http --http.api "eth,net,web3,personal" --http.addr <我们的 ETH 测试服务器 ip> --http.port <我们的 ETH 测试服务器 端口> --maxpeers 32 --allow-insecure-unlock**

# --ropsten: 连接到测试网络 ropsten

# --syncmode snap: 同步模式选择 snap

# --cache=1024: 选择

# --http: 开启 http.rpc 服务

# --http.api "eth,net,web3,personal": HTTP-RPC 接口提供的 API

# --http.addr: HTTP-RPC 服务监听地址（内网地址, ubuntu 通过 "ip a" 命令获取）

# --http.port: HTTP-RPC 服务监听端口

# --maxpeers 32: 最大 peer 数量

# --allow-insecure-unlock: Allow insecure account unlocking when account-related RPCs are exposed by http

# --ethstats='<节点名称（随便起，但不能和已有的重复，geth 界面会不停地报登录失败）>:<密码，似乎不能改，维持原来的 'Respect my authoritah!'>@stats.rinkeby.io':

# 仅限 rinkeby（反正别的网络，像 ropsten 啊，我没找到统计的站点）

# 将节点信息上报到 "stats.rinkeby.io"，可到 "https://www.rinkeby.io/#stats" 查看节点信息

sync mode 有几种：

# full

geth --syncmode "full"

# fast

geth --syncmode "fast"

# light

geth --syncmode "fast"

# snap(新开发的功能，21年刚上线，快照同步模式，同步速度极快)

geth --syncmode "snap"

进入 geth 命令行

geth attach http://<我们的 ETH 测试服务器 ip>:<我们的 ETH 测试服务器 端口>

# 查看节点是否已同步完成

> eth.syncing

False

生成新的地址

创建账户

# 生成1000个地址，打印信息重定向到 tempdata 中

for i in {1..1000}

do

# 账户地址信息均写入 tempdata

geth --ropsten account new --password ./testnet\_account\_password >> ./tempdata

done

# 生成过程中，每个地址生成会匹配两条报错，正常，只要在 key\_store 中能看到密钥生成，即可

将 tempdata 格式化为数据库可导入文件(table\_name.txt)

========================================================================================

Geth Console 是一个交互式的 JavaScript 执行环境，里面内置了一些用来操作以太坊的 JavaScript
对象，我们可以直接调用这些对象来获取区块链上的相关信息。这些对象主要包括：
eth：主要包含对区块链进行访问和交互相关的方法；
net：主要包含查看 p2p 网络状态的方法；
admin：主要包含与管理节点相关的方法；
miner：主要包含挖矿相关的一些方法；
personal：包含账户管理的方法；
txpool：包含查看交易内存池的方法；
web3：包含以上所有对象，还包含一些通用方法。

=========================================================================

The Geth console is an interactive shell for the JavaScript runtime environment

which exposes a node admin interface as well as the Ðapp JavaScript API.

See：

https://geth.ethereum.org/docs/interface/javascript-console

===============================================================================

2022-01-14

转账交易却不被挖矿接受：

INFO [01-14|11:36:17.413] 🔗 block reached canonical chain number=1293 hash=d62183..0a99d5

INFO [01-14|11:36:17.413] 🔨 mined potential block number=1300 hash=eb3e46..627a79

INFO [01-14|11:36:17.413] Commit new mining work number=1301 sealhash=65f4fe..9c24fa uncles=0 txs=0 gas=0 fees=0 elapsed="223.371µs"

WARN [01-14|11:36:27.655] Served eth\_sendRawTransaction conn=192.168.11.96:1331 reqid=4391589708352 duration="106.185µs" err="already known"

WARN [01-14|11:36:27.662] Served eth\_sendRawTransaction conn=192.168.11.96:1331 reqid=4391589708353 duration="401.917µs" err="replacement transaction underpriced"

WARN [01-14|11:36:35.277] **Caller gas above allowance, capping requested=1,204,803,264 cap=50,000,000**

WARN [01-14|11:36:40.264] Caller gas above allowance, capping requested=1,204,803,264 cap=50,000,000

WARN [01-14|11:36:43.467] Caller gas above allowance, capping requested=1,204,803,264 cap=50,000,000

INFO [01-14|11:36:56.090] Submitted transaction hash=0x5b1bae1ff076c5d55cd53272d9c379290db7b3f4767d1896a8a73c78249c2813 from=0xCED27893eeA69b3fF1E57b5Ce40F19662e9296F5 **nonce=72** recipient=0xF57c31F925624DBc935c11172DEb47375d8409Be value=50,000,000,000,000,000,000

INFO [01-14|11:37:12.143] Updated mining threads threads=1

INFO [01-14|11:37:12.143] Transaction pool price threshold updated price=1,000,000,000

INFO [01-14|11:37:12.144] Commit new mining work number=1301 sealhash=65f4fe..9c24fa uncles=0 txs=0 gas=0 fees=0 elapsed="276.168µs"

、这里明显看出来Gas要求12亿，上限却只有5千万！

因为是新起的创世链块，那么nonce就应该不对！

因为是新创世，那么肯定nonce就没有记录在区块上，应该是在钱包本地？

INFO [01-14|11:37:28.174] Commit new mining work number=1310 sealhash=77c242..43e1e5 uncles=0 txs=0 gas=0 fees=0 elapsed="229.613µs"

WARN [01-14|11:37:47.668] Served **eth\_sendRawTransaction** conn=192.168.11.96:25249 reqid=4391589708358 duration="113.786µs" err="**already known**"

WARN [01-14|11:37:47.675] Served **eth\_sendRawTransaction** conn=192.168.11.96:25249 reqid=4391589708359 duration="410.592µs" err="**replacement transaction underpriced**"

======================================================================================

可以源码级配置自己geth私链的资源：

原文章链接：https://blog.csdn.net/tianbaochao/article/details/88179341

5 修改源码

参考https://hackernoon.com/how-to-reduce-block-difficulty-in-ethereum-private-testnet-2ad505609e82?gi=1ca92726aa1f

修改go-ethereum-1.8.12/consensus/ethash目录下consensus.go文件中代码，将

return CalcDifficulty(chain.Config(), time, parent)

替换成

return big.NewInt(1)

6 交易才挖矿

参考：

<https://ethereum.stackexchange.com/questions/3151/how-to-make-miner-to-mine-only-when-there-are-pending-transactions>

编写minebytrans.js，代码如下：

var mining\_threads = 1

var txBlock = 0

function checkWork() {

if (eth.getBlock("pending").transactions.length > 0) {

txBlock = eth.getBlock("pending").number

if (eth.mining) return;

console.log(" Transactions pending. Mining...");

miner.start(mining\_threads)

while (eth.getBlock("latest").number < txBlock + 12) {

if (eth.getBlock("pending").transactions.length > 0) txBlock = eth.getBlock("pending").number;

}

console.log(" 12 confirmations achieved; mining stopped.");

miner.stop()

}

else {

miner.stop()

}

}

eth.filter("latest", function(err, block) { checkWork(); });

eth.filter("pending", function(err, block) { checkWork(); });

checkWork();

启动以太坊私链时，通过js加载上述js文件，限制区块不间断挖矿。

geth --preload "/path/to/mineWhenNeeded.js"

、geth dev Failed to unlock developer account

geth --dev creates a preconfigured private network, so --unlock and --password don't work.

If you want them to work, create a new private network and geth init it with a custom genesis.json.

I would try adding : --password './password.txt'.

=================================================================================

**以太坊源码研读0x01GenesisBlock(创世区块)**

// DefaultGenesisBlock returns the Ethereum main net genesis block.

// DefaultGenesisBlock返回以太坊主网络创世块

func DefaultGenesisBlock() \*Genesis {

return &Genesis{

Config: params.MainnetChainConfig,

Nonce: 66,

ExtraData: hexutil.MustDecode("0x11bbe8db4e347b4e8c937c1c8370e4b5ed33adb3db69cbdb7a38e1e50b1b82fa"),

GasLimit: 5000,

Difficulty: big.NewInt(17179869184),///0x040000000

Alloc: decodePrealloc(mainnetAllocData),

}

}

// DefaultTestnetGenesisBlock returns the Ropsten network genesis block.

// DefaultTestnetGenesisBlock返回Ropsten网络创世块。

func DefaultTestnetGenesisBlock() \*Genesis {

return &Genesis{

Config: params.TestnetChainConfig,

Nonce: 66,

ExtraData: hexutil.MustDecode("0x3535353535353535353535353535353535353535353535353535353535353535"),

GasLimit: 16777216,

Difficulty: big.NewInt(1048576),///0x0100000

Alloc: decodePrealloc(testnetAllocData),

}

}

// DefaultRinkebyGenesisBlock returns the Rinkeby network genesis block.

// DefaultRinkebyGenesisBlock返回Rinkeby网络创世块。

func DefaultRinkebyGenesisBlock() \*Genesis {

return &Genesis{

Config: params.RinkebyChainConfig,

Timestamp: 1492009146,

ExtraData: hexutil.MustDecode("0x52657370656374206d7920617574686f7269746168207e452e436172746d616e42eb768f2244c8811c63729a21a3569731535f067ffc57839b00206d1ad20c69a1981b489f772031b279182d99e65703f0076e4812653aab85fca0f00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000"),

GasLimit: 4700000,

Difficulty: big.NewInt(1),

Alloc: decodePrealloc(rinkebyAllocData),

}

}

// DeveloperGenesisBlock returns the 'geth --dev' genesis block. Note, this must

// be seeded with the

// eveloperGenesisBlock返回'geth --dev'创世块。

func DeveloperGenesisBlock(period uint64, faucet common.Address) \*Genesis {

// Override the default period to the user requested one

config := \*params.AllCliqueProtocolChanges

config.Clique.Period = period

// Assemble and return the genesis with the precompiles and faucet pre-funded

return &Genesis{

Config: &config,

ExtraData: append(append(make([]byte, 32), faucet[:]...), make([]byte, 65)...),

GasLimit: 6283185,

Difficulty: big.NewInt(1),

Alloc: map[common.Address]GenesisAccount{

common.BytesToAddress([]byte{1}): {Balance: big.NewInt(1)}, // ECRecover

common.BytesToAddress([]byte{2}): {Balance: big.NewInt(1)}, // SHA256

common.BytesToAddress([]byte{3}): {Balance: big.NewInt(1)}, // RIPEMD

common.BytesToAddress([]byte{4}): {Balance: big.NewInt(1)}, // Identity

common.BytesToAddress([]byte{5}): {Balance: big.NewInt(1)}, // ModExp

common.BytesToAddress([]byte{6}): {Balance: big.NewInt(1)}, // ECAdd

common.BytesToAddress([]byte{7}): {Balance: big.NewInt(1)}, // ECScalarMul

common.BytesToAddress([]byte{8}): {Balance: big.NewInt(1)}, // ECPairing

faucet: {Balance: new(big.Int).Sub(new(big.Int).Lsh(big.NewInt(1), 256), big.NewInt(9))},

},

}

}

====================================================================================

leon@ubuntu:~/ethmchain/consortium\_blockchain$ geth --datadir "myblockchain" --networkid 84348 console --http --httpaddr "192.168.11.88" --allow-insecure-unlock --nodiscover 2>20220105.log

Incorrect Usage: flag provided but not defined: -rpc

console [command options]

The Geth console is an interactive shell for the JavaScript runtime environment

which exposes a node admin interface as well as the Ðapp JavaScript API.

See https://geth.ethereum.org/docs/interface/javascript-console.

ACCOUNT OPTIONS:

--unlock value Comma separated list of accounts to unlock

--password value Password file to use for non-interactive password input

--signer value External signer (url or path to ipc file)

--allow-insecure-unlock Allow insecure account unlocking when account-related RPCs are exposed by http

ETHASH OPTIONS:

--ethash.cachedir value Directory to store the ethash verification caches (default = inside the datadir)

--ethash.cachesinmem value Number of recent ethash caches to keep in memory (16MB each) (default: 2)

--ethash.cachesondisk value Number of recent ethash caches to keep on disk (16MB each) (default: 3)

--ethash.cacheslockmmap Lock memory maps of recent ethash caches

--ethash.dagdir value Directory to store the ethash mining DAGs (default: "/home/leon/.ethash")

--ethash.dagsinmem value Number of recent ethash mining DAGs to keep in memory (1+GB each) (default: 1)

--ethash.dagsondisk value Number of recent ethash mining DAGs to keep on disk (1+GB each) (default: 2)

--ethash.dagslockmmap Lock memory maps for recent ethash mining DAGs

NETWORKING OPTIONS:

--bootnodes value Comma separated enode URLs for P2P discovery bootstrap

--port value Network listening port (default: 30303)

--maxpeers value Maximum number of network peers (network disabled if set to 0) (default: 50)

--maxpendpeers value Maximum number of pending connection attempts (defaults used if set to 0) (default: 0)

--nat value NAT port mapping mechanism (any|none|upnp|pmp|extip:<IP>) (default: "any")

--nodiscover Disables the peer discovery mechanism (manual peer addition)

--v5disc Enables the experimental RLPx V5 (Topic Discovery) mechanism

--netrestrict value Restricts network communication to the given IP networks (CIDR masks)

--nodekey value P2P node key file

--nodekeyhex value P2P node key as hex (for testing)

--discovery.dns value Sets DNS discovery entry points (use "" to disable DNS)

API AND CONSOLE OPTIONS:

--http Enable the HTTP-RPC server

--http.addr value HTTP-RPC server listening interface (default: "localhost")

--http.port value HTTP-RPC server listening port (default: 8545)

--http.corsdomain value Comma separated list of domains from which to accept cross origin requests (browser enforced)

--http.vhosts value Comma separated list of virtual hostnames from which to accept requests (server enforced). Accepts '\*' wildcard. (default: "localhost")

--graphql Enable GraphQL on the HTTP-RPC server. Note that GraphQL can only be started if an HTTP server is started as well.

--graphql.corsdomain value Comma separated list of domains from which to accept cross origin requests (browser enforced)

--graphql.vhosts value Comma separated list of virtual hostnames from which to accept requests (server enforced). Accepts '\*' wildcard. (default: "localhost")

--http.api value API's offered over the HTTP-RPC interface

--http.rpcprefix value HTTP path path prefix on which JSON-RPC is served. Use '/' to serve on all paths.

--ws Enable the WS-RPC server

--ws.addr value WS-RPC server listening interface (default: "localhost")

--ws.port value WS-RPC server listening port (default: 8546)

--ws.api value API's offered over the WS-RPC interface

--ws.origins value Origins from which to accept websockets requests

--ws.rpcprefix value HTTP path prefix on which JSON-RPC is served. Use '/' to serve on all paths.

--ipcdisable Disable the IPC-RPC server

--ipcpath value Filename for IPC socket/pipe within the datadir (explicit paths escape it)

--rpc.gascap value Sets a cap on gas that can be used in eth\_call/estimateGas (0=infinite) (default: 50000000)

--rpc.evmtimeout value Sets a timeout used for eth\_call (0=infinite) (default: 5s)

--rpc.txfeecap value Sets a cap on transaction fee (in ether) that can be sent via the RPC APIs (0 = no cap) (default: 1)

--rpc.allow-unprotected-txs Allow for unprotected (non EIP155 signed) transactions to be submitted via RPC

--jspath loadScript JavaScript root path for loadScript (default: ".")

--exec value Execute JavaScript statement

--preload value Comma separated list of JavaScript files to preload into the console

ALIASED (deprecated) OPTIONS:

--nousb Disables monitoring for and managing USB hardware wallets (deprecated)

VIRTUAL MACHINE OPTIONS:

--vmdebug Record information useful for VM and contract debugging

LOGGING AND DEBUGGING OPTIONS:

--fakepow Disables proof-of-work verification

--nocompaction Disables db compaction after import

ETHEREUM OPTIONS:

--identity value Custom node name

--datadir value Data directory for the databases and keystore (default: "/home/leon/.ethereum")

--datadir.ancient value Data directory for ancient chain segments (default = inside chaindata)

--datadir.minfreedisk value Minimum free disk space in MB, once reached triggers auto shut down (default = --cache.gc converted to MB, 0 = disabled)

--keystore value Directory for the keystore (default = inside the datadir)

--usb Enable monitoring and management of USB hardware wallets

--pcscdpath value Path to the smartcard daemon (pcscd) socket file (default: "/run/pcscd/pcscd.comm")

--syncmode value Blockchain sync mode ("snap", "full" or "light") (default: snap)

--exitwhensynced Exits after block synchronisation completes

--gcmode value Blockchain garbage collection mode ("full", "archive") (default: "full")

--txlookuplimit value Number of recent blocks to maintain transactions index for (default = about one year, 0 = entire chain) (default: 2350000)

--lightkdf Reduce key-derivation RAM & CPU usage at some expense of KDF strength

--whitelist value Comma separated block number-to-hash mappings to enforce (<number>=<hash>)

--mainnet Ethereum mainnet

--ropsten Ropsten network: pre-configured proof-of-work test network

--sepolia Sepolia network: pre-configured proof-of-work test network

--rinkeby Rinkeby network: pre-configured proof-of-authority test network

--goerli Görli network: pre-configured proof-of-authority test network

--networkid value Explicitly set network id (integer)(For testnets: use --ropsten, --rinkeby, --goerli instead) (default: 1)

--ethstats value Reporting URL of a ethstats service (nodename:secret@host:port)

--config value TOML configuration file

PERFORMANCE TUNING OPTIONS:

--cache value Megabytes of memory allocated to internal caching (default = 4096 mainnet full node, 128 light mode) (default: 1024)

--cache.database value Percentage of cache memory allowance to use for database io (default: 50)

--cache.trie value Percentage of cache memory allowance to use for trie caching (default = 15% full mode, 30% archive mode) (default: 15)

--cache.trie.journal value Disk journal directory for trie cache to survive node restarts (default: "triecache")

--cache.trie.rejournal value Time interval to regenerate the trie cache journal (default: 1h0m0s)

--cache.gc value Percentage of cache memory allowance to use for trie pruning (default = 25% full mode, 0% archive mode) (default: 25)

--cache.snapshot value Percentage of cache memory allowance to use for snapshot caching (default = 10% full mode, 20% archive mode) (default: 10)

--cache.noprefetch Disable heuristic state prefetch during block import (less CPU and disk IO, more time waiting for data)

--cache.preimages Enable recording the SHA3/keccak preimages of trie keys

TRANSACTION POOL OPTIONS:

--txpool.locals value Comma separated accounts to treat as locals (no flush, priority inclusion)

--txpool.nolocals Disables price exemptions for locally submitted transactions

--txpool.journal value Disk journal for local transaction to survive node restarts (default: "transactions.rlp")

--txpool.rejournal value Time interval to regenerate the local transaction journal (default: 1h0m0s)

--txpool.pricelimit value Minimum gas price limit to enforce for acceptance into the pool (default: 1)

--txpool.pricebump value Price bump percentage to replace an already existing transaction (default: 10)

--txpool.accountslots value Minimum number of executable transaction slots guaranteed per account (default: 16)

--txpool.globalslots value Maximum number of executable transaction slots for all accounts (default: 5120)

--txpool.accountqueue value Maximum number of non-executable transaction slots permitted per account (default: 64)

--txpool.globalqueue value Maximum number of non-executable transaction slots for all accounts (default: 1024)

--txpool.lifetime value Maximum amount of time non-executable transaction are queued (default: 3h0m0s)

MISC OPTIONS:

--override.arrowglacier value Manually specify Arrow Glacier fork-block, overriding the bundled setting (default: 0)

--override.terminaltotaldifficulty value Manually specify TerminalTotalDifficulty, overriding the bundled setting (default: 0)

--snapshot Enables snapshot-database mode (default = enable)

--bloomfilter.size value Megabytes of memory allocated to bloom-filter for pruning (default: 2048)

--miner.gastarget value Target gas floor for mined blocks (deprecated) (default: 0)

--catalyst Catalyst mode (eth2 integration testing)

LIGHT CLIENT OPTIONS:

--light.serve value Maximum percentage of time allowed for serving LES requests (multi-threaded processing allows values over 100) (default: 0)

--light.ingress value Incoming bandwidth limit for serving light clients (kilobytes/sec, 0 = unlimited) (default: 0)

--light.egress value Outgoing bandwidth limit for serving light clients (kilobytes/sec, 0 = unlimited) (default: 0)

--light.maxpeers value Maximum number of light clients to serve, or light servers to attach to (default: 100)

--light.nopruning Disable ancient light chain data pruning

--ulc.servers value List of trusted ultra-light servers

--ulc.fraction value Minimum % of trusted ultra-light servers required to announce a new head (default: 75)

--ulc.onlyannounce Ultra light server sends announcements only

--light.nosyncserve Enables serving light clients before syncing

MINER OPTIONS:

--mine Enable mining

--miner.threads value Number of CPU threads to use for mining (default: 0)

--miner.notify value Comma separated HTTP URL list to notify of new work packages

--miner.gaslimit value Target gas ceiling for mined blocks (default: 8000000)

--miner.gasprice value Minimum gas price for mining a transaction (default: 1000000000)

--miner.etherbase value Public address for block mining rewards (default = first account) (default: "0")

--miner.extradata value Block extra data set by the miner (default = client version)

--miner.recommit value Time interval to recreate the block being mined (default: 3s)

--miner.noverify Disable remote sealing verification

--miner.notify.full Notify with pending block headers instead of work packages

DEVELOPER CHAIN OPTIONS:

--dev Ephemeral proof-of-authority network with a pre-funded developer account, mining enabled

--dev.period value Block period to use in developer mode (0 = mine only if transaction pending) (default: 0)

--dev.gaslimit value Initial block gas limit (default: 11500000)

GAS PRICE ORACLE OPTIONS:

--gpo.blocks value Number of recent blocks to check for gas prices (default: 20)

--gpo.percentile value Suggested gas price is the given percentile of a set of recent transaction gas prices (default: 60)

--gpo.maxprice value Maximum transaction priority fee (or gasprice before London fork) to be recommended by gpo (default: 500000000000)

--gpo.ignoreprice value Gas price below which gpo will ignore transactions (default: 2)

---------------------------------------------------------------------------------------------------------------------------------------------

leon@ubuntu:~/ethmchain/consortium\_blockchain$ **geth help**

NAME:

geth - the go-ethereum command line interface

Copyright 2013-2021 The go-ethereum Authors

USAGE:

geth [options] [command] [command options] [arguments...]

VERSION:

1.10.13-stable-7a0c19f8

COMMANDS:

account Manage accounts

attach Start an interactive JavaScript environment (connect to node)

console Start an interactive JavaScript environment

db Low level database operations

dump Dump a specific block from storage

dumpconfig Show configuration values

dumpgenesis Dumps genesis block JSON configuration to stdout

export Export blockchain into file

export-preimages Export the preimage database into an RLP stream

import Import a blockchain file

import-preimages Import the preimage database from an RLP stream

init Bootstrap and initialize a new genesis block

js Execute the specified JavaScript files

license Display license information

makecache Generate ethash verification cache (for testing)

makedag Generate ethash mining DAG (for testing)

removedb Remove blockchain and state databases

show-deprecated-flags Show flags that have been deprecated

snapshot A set of commands based on the snapshot

version Print version numbers

version-check Checks (online) whether the current version suffers from any known security vulnerabilities

wallet Manage Ethereum presale wallets

help, h Shows a list of commands or help for one command

ETHEREUM OPTIONS:

--config value TOML configuration file

--datadir value Data directory for the databases and keystore (default: "/home/leon/.ethereum")

--datadir.ancient value Data directory for ancient chain segments (default = inside chaindata)

--datadir.minfreedisk value Minimum free disk space in MB, once reached triggers auto shut down (default = --cache.gc converted to MB, 0 = disabled)

--keystore value Directory for the keystore (default = inside the datadir)

--usb Enable monitoring and management of USB hardware wallets

--pcscdpath value Path to the smartcard daemon (pcscd) socket file (default: "/run/pcscd/pcscd.comm")

--networkid value Explicitly set network id (integer)(For testnets: use --ropsten, --rinkeby, --goerli instead) (default: 1)

--mainnet Ethereum mainnet

--goerli Görli network: pre-configured proof-of-authority test network

--rinkeby Rinkeby network: pre-configured proof-of-authority test network

--ropsten Ropsten network: pre-configured proof-of-work test network

--sepolia Sepolia network: pre-configured proof-of-work test network

--syncmode value Blockchain sync mode ("fast", "full", "snap" or "light") (default: snap)

--exitwhensynced Exits after block synchronisation completes

--gcmode value Blockchain garbage collection mode ("full", "archive") (default: "full")

--txlookuplimit value Number of recent blocks to maintain transactions index for (default = about one year, 0 = entire chain) (default: 2350000)

--ethstats value Reporting URL of a ethstats service (nodename:secret@host:port)

--identity value Custom node name

--lightkdf Reduce key-derivation RAM & CPU usage at some expense of KDF strength

--whitelist value Comma separated block number-to-hash mappings to enforce (<number>=<hash>)

LIGHT CLIENT OPTIONS:

--light.serve value Maximum percentage of time allowed for serving LES requests (multi-threaded processing allows values over 100) (default: 0)

--light.ingress value Incoming bandwidth limit for serving light clients (kilobytes/sec, 0 = unlimited) (default: 0)

--light.egress value Outgoing bandwidth limit for serving light clients (kilobytes/sec, 0 = unlimited) (default: 0)

--light.maxpeers value Maximum number of light clients to serve, or light servers to attach to (default: 100)

--ulc.servers value List of trusted ultra-light servers

--ulc.fraction value Minimum % of trusted ultra-light servers required to announce a new head (default: 75)

--ulc.onlyannounce Ultra light server sends announcements only

--light.nopruning Disable ancient light chain data pruning

--light.nosyncserve Enables serving light clients before syncing

DEVELOPER CHAIN OPTIONS:

--dev Ephemeral proof-of-authority network with a pre-funded developer account, mining enabled

--dev.period value Block period to use in developer mode (0 = mine only if transaction pending) (default: 0)

--dev.gaslimit value Initial block gas limit (default: 11500000)

ETHASH OPTIONS:

--ethash.cachedir value Directory to store the ethash verification caches (default = inside the datadir)

--ethash.cachesinmem value Number of recent ethash caches to keep in memory (16MB each) (default: 2)

--ethash.cachesondisk value Number of recent ethash caches to keep on disk (16MB each) (default: 3)

--ethash.cacheslockmmap Lock memory maps of recent ethash caches

--ethash.dagdir value Directory to store the ethash mining DAGs (default: "/home/leon/.ethash")

--ethash.dagsinmem value Number of recent ethash mining DAGs to keep in memory (1+GB each) (default: 1)

--ethash.dagsondisk value Number of recent ethash mining DAGs to keep on disk (1+GB each) (default: 2)

--ethash.dagslockmmap Lock memory maps for recent ethash mining DAGs

TRANSACTION POOL OPTIONS:

--txpool.locals value Comma separated accounts to treat as locals (no flush, priority inclusion)

--txpool.nolocals Disables price exemptions for locally submitted transactions

--txpool.journal value Disk journal for local transaction to survive node restarts (default: "transactions.rlp")

--txpool.rejournal value Time interval to regenerate the local transaction journal (default: 1h0m0s)

--txpool.pricelimit value Minimum gas price limit to enforce for acceptance into the pool (default: 1)

--txpool.pricebump value Price bump percentage to replace an already existing transaction (default: 10)

--txpool.accountslots value Minimum number of executable transaction slots guaranteed per account (default: 16)

--txpool.globalslots value Maximum number of executable transaction slots for all accounts (default: 5120)

--txpool.accountqueue value Maximum number of non-executable transaction slots permitted per account (default: 64)

--txpool.globalqueue value Maximum number of non-executable transaction slots for all accounts (default: 1024)

--txpool.lifetime value Maximum amount of time non-executable transaction are queued (default: 3h0m0s)

PERFORMANCE TUNING OPTIONS:

--cache value Megabytes of memory allocated to internal caching (default = 4096 mainnet full node, 128 light mode) (default: 1024)

--cache.database value Percentage of cache memory allowance to use for database io (default: 50)

--cache.trie value Percentage of cache memory allowance to use for trie caching (default = 15% full mode, 30% archive mode) (default: 15)

--cache.trie.journal value Disk journal directory for trie cache to survive node restarts (default: "triecache")

--cache.trie.rejournal value Time interval to regenerate the trie cache journal (default: 1h0m0s)

--cache.gc value Percentage of cache memory allowance to use for trie pruning (default = 25% full mode, 0% archive mode) (default: 25)

--cache.snapshot value Percentage of cache memory allowance to use for snapshot caching (default = 10% full mode, 20% archive mode) (default: 10)

--cache.noprefetch Disable heuristic state prefetch during block import (less CPU and disk IO, more time waiting for data)

--cache.preimages Enable recording the SHA3/keccak preimages of trie keys

ACCOUNT OPTIONS:

--unlock value Comma separated list of accounts to unlock

--password value Password file to use for non-interactive password input

--signer value External signer (url or path to ipc file)

--allow-insecure-unlock Allow insecure account unlocking when account-related RPCs are exposed by http

API AND CONSOLE OPTIONS:

--ipcdisable Disable the IPC-RPC server

--ipcpath value Filename for IPC socket/pipe within the datadir (explicit paths escape it)

--http Enable the HTTP-RPC server

--http.addr value HTTP-RPC server listening interface (default: "localhost")

--http.port value HTTP-RPC server listening port (default: 8545)

--http.api value API's offered over the HTTP-RPC interface

--http.rpcprefix value HTTP path path prefix on which JSON-RPC is served. Use '/' to serve on all paths.

--http.corsdomain value Comma separated list of domains from which to accept cross origin requests (browser enforced)

--http.vhosts value Comma separated list of virtual hostnames from which to accept requests (server enforced). Accepts '\*' wildcard. (default: "localhost")

--ws Enable the WS-RPC server

--ws.addr value WS-RPC server listening interface (default: "localhost")

--ws.port value WS-RPC server listening port (default: 8546)

--ws.api value API's offered over the WS-RPC interface

--ws.rpcprefix value HTTP path prefix on which JSON-RPC is served. Use '/' to serve on all paths.

--ws.origins value Origins from which to accept websockets requests

--graphql Enable GraphQL on the HTTP-RPC server. Note that GraphQL can only be started if an HTTP server is started as well.

--graphql.corsdomain value Comma separated list of domains from which to accept cross origin requests (browser enforced)

--graphql.vhosts value Comma separated list of virtual hostnames from which to accept requests (server enforced). Accepts '\*' wildcard. (default: "localhost")

--rpc.gascap value Sets a cap on gas that can be used in eth\_call/estimateGas (0=infinite) (default: 50000000)

--rpc.evmtimeout value Sets a timeout used for eth\_call (0=infinite) (default: 5s)

--rpc.txfeecap value Sets a cap on transaction fee (in ether) that can be sent via the RPC APIs (0 = no cap) (default: 1)

--rpc.allow-unprotected-txs Allow for unprotected (non EIP155 signed) transactions to be submitted via RPC

--jspath loadScript JavaScript root path for loadScript (default: ".")

--exec value Execute JavaScript statement

--preload value Comma separated list of JavaScript files to preload into the console

NETWORKING OPTIONS:

--bootnodes value Comma separated enode URLs for P2P discovery bootstrap

--discovery.dns value Sets DNS discovery entry points (use "" to disable DNS)

**--port** value IPv6 Network listening port (default: 30303)

--maxpeers value Maximum number of network peers (network disabled if set to 0) (default: 50)

--maxpendpeers value Maximum number of pending connection attempts (defaults used if set to 0) (default: 0)

--nat value NAT port mapping mechanism (any|none|upnp|pmp|extip:<IP>) (default: "any")

--nodiscover Disables the peer discovery mechanism (manual peer addition)

--v5disc Enables the experimental RLPx V5 (Topic Discovery) mechanism

--netrestrict value Restricts network communication to the given IP networks (CIDR masks)

--nodekey value P2P node key file

--nodekeyhex value P2P node key as hex (for testing)

MINER OPTIONS:

--mine Enable mining

--miner.threads value Number of CPU threads to use for mining (default: 0)

--miner.notify value Comma separated HTTP URL list to notify of new work packages

--miner.notify.full Notify with pending block headers instead of work packages

--miner.gasprice value Minimum gas price for mining a transaction (default: 1000000000)

--miner.gaslimit value Target gas ceiling for mined blocks (default: 8000000)

--miner.etherbase value Public address for block mining rewards (default = first account) (default: "0")

--miner.extradata value Block extra data set by the miner (default = client version)

--miner.recommit value Time interval to recreate the block being mined (default: 3s)

--miner.noverify Disable remote sealing verification

GAS PRICE ORACLE OPTIONS:

--gpo.blocks value Number of recent blocks to check for gas prices (default: 20)

--gpo.percentile value Suggested gas price is the given percentile of a set of recent transaction gas prices (default: 60)

--gpo.maxprice value Maximum transaction priority fee (or gasprice before London fork) to be recommended by gpo (default: 500000000000)

--gpo.ignoreprice value Gas price below which gpo will ignore transactions (default: 2)

VIRTUAL MACHINE OPTIONS:

--vmdebug Record information useful for VM and contract debugging

LOGGING AND DEBUGGING OPTIONS:

--fakepow Disables proof-of-work verification

--nocompaction Disables db compaction after import

--verbosity value Logging verbosity: 0=silent, 1=error, 2=warn, 3=info, 4=debug, 5=detail (default: 3)

--vmodule value Per-module verbosity: comma-separated list of <pattern>=<level> (e.g. eth/\*=5,p2p=4)

--log.json Format logs with JSON

--log.backtrace value Request a stack trace at a specific logging statement (e.g. "block.go:271")

--log.debug Prepends log messages with call-site location (file and line number)

--pprof Enable the pprof HTTP server

--pprof.addr value pprof HTTP server listening interface (default: "127.0.0.1")

--pprof.port value pprof HTTP server listening port (default: 6060)

--pprof.memprofilerate value Turn on memory profiling with the given rate (default: 524288)

--pprof.blockprofilerate value Turn on block profiling with the given rate (default: 0)

--pprof.cpuprofile value Write CPU profile to the given file

--trace value Write execution trace to the given file

METRICS AND STATS OPTIONS:

--metrics Enable metrics collection and reporting

--metrics.expensive Enable expensive metrics collection and reporting

--metrics.addr value Enable stand-alone metrics HTTP server listening interface (default: "127.0.0.1")

--metrics.port value Metrics HTTP server listening port (default: 6060)

--metrics.influxdb Enable metrics export/push to an external InfluxDB database

--metrics.influxdb.endpoint value InfluxDB API endpoint to report metrics to (default: "http://localhost:8086")

--metrics.influxdb.database value InfluxDB database name to push reported metrics to (default: "geth")

--metrics.influxdb.username value Username to authorize access to the database (default: "test")

--metrics.influxdb.password value Password to authorize access to the database (default: "test")

--metrics.influxdb.tags value Comma-separated InfluxDB tags (key/values) attached to all measurements (default: "host=localhost")

--metrics.influxdbv2 Enable metrics export/push to an external InfluxDB v2 database

--metrics.influxdb.token value Token to authorize access to the database (v2 only) (default: "test")

--metrics.influxdb.bucket value InfluxDB bucket name to push reported metrics to (v2 only) (default: "geth")

--metrics.influxdb.organization value InfluxDB organization name (v2 only) (default: "geth")

ALIASED (deprecated) OPTIONS:

--nousb Disables monitoring for and managing USB hardware wallets (deprecated)

MISC OPTIONS:

--snapshot Enables snapshot-database mode (default = enable)

--bloomfilter.size value Megabytes of memory allocated to bloom-filter for pruning (default: 2048)

--help, -h show help

--catalyst Catalyst mode (eth2 integration testing)

--override.arrowglacier value Manually specify Arrow Glacier fork-block, overriding the bundled setting (default: 0)

COPYRIGHT:

Copyright 2013-2021 The go-ethereum Authors

leon@ubuntu:~/ethmchain/consortium\_blockchain$