

以太坊客户端 Geth 命令用法-参数详解

链客区块链技术问答社区

Geth 在以太坊智能合约开发中最常用的工具 (必备开发工具) , 一个多用途的命令行工具。

熟悉 Geth 可以让我们有更好的效率, 大家可收藏起来作为 Geth 命令用法手册。 本文主要是对 `geth help` 的翻译, 基于最新的 `geth 1.7.3-stable` 版本。

以下开始正文。

命令用法

```
geth [选项] 命令 [命令选项] [参数...]
```

版本:

```
1.7.3-stable
```

命令:

account 管理账户

attach 启动交互式 JavaScript 环境 (连接到节点)

bug 上报 bug Issues

console	启动交互式 JavaScript 环境
copydb	从文件夹创建本地链
dump	Dump（分析）一个特定的块存储
dumpconfig	显示配置值
export	导出区块链到文件
import	导入一个区块链文件
init	启动并初始化一个新的创世纪块
js	执行指定的 JavaScript 文件(多个)
license	显示许可信息
makecache	生成 ethash 验证缓存(用于测试)
makedag	生成 ethash 挖矿 DAG(用于测试)
monitor	监控和可视化节点指标
removedb	删除区块链和状态数据库
version	打印版本号
wallet	管理 Ethereum 预售钱包

help,h 显示一个命令或帮助一个命令列表

ETHEREUM 选项:

--config value TOML 配置文件

--datadir "xxx" 数据库和 keystore 密钥的数据目录

--keystore keystore 存放目录 (默认在 datadir 内)

--nousb 禁用监控和管理 USB 硬件钱包

--networkid value 网络标识符 (整型, 1=Frontier, 2=Morden (弃用), 3=Ropsten, 4=Rinkeby) (默认: 1)

--testnet Ropsten 网络: 预先配置的 POW(proof-of-work)测试网络

--rinkeby Rinkeby 网络: 预先配置的 POA(proof-of-authority)测试网络

--syncmode "fast" 同步模式 ("fast", "full", or "light")

```
--ethash.cachedir          ethash 验证缓存目录(默认 = datadir 目录内)
```

--ethash.cachesinmem value 在内存保存的最近的 ethash 缓存个数 (每个缓存 16MB) (默认: 2)

--ethash.cachesondisk value 在磁盘保存的最近的 ethash 缓存个数 (每个缓存 16MB) (默认: 3)

```
--ethash.dagdir ""          存 ethash
```

DAGs 目录 (默认 = 用户 hom 目录)

--ethash.dagsinmem value 在内存保存的最近的 ethash DAGs 个数 (每个 1GB 以上) (默认: 1)

--ethash.dagsondisk value 在磁盘保存的最近的 ethash DAGs 个数 (每个 1GB 以上) (默认: 2)

交易池选项:

--txpool.nolocals	为本地提交交易禁用价格豁免
--------------------------	----------------------

--txpool.journal value 本地交易的磁盘日志：用于节点重启（默认: "transactions.rlp"）

--txpool.rejournal value	重新生成本地交易日志的时间间隔（默认：1 小时）
---------------------------------	---------------------------------

--txpool.pricelimit value 加入交易池的最小的 gas 价格限制(默认: 1)

--txpool.pricebump value 价格波动百分比（相对之前已有交易）（默认: 10）

--txpool.accountslots value 每个帐户保证可执行的最少交易槽数量（默认: 16）

--txpool.globalslots value 所有帐户可执行的最大交易槽数量（默认: 4096）

--txpool.accountqueue value 每个帐户允许的最多非可执行交易槽数量（默认: 64）

--txpool.globalqueue value 所有帐户非可执行交易最大槽数量（默认: 1024）

--txpool.lifetime value 非可执行交易最大入队时间(默认: 3 小时)

性能调优的选项:

--cache value 分配给内部缓存的内存 MB 数量，缓存值(最低 16 mb /数据库强制要求)(默认:128)

--trie-cache-gens value 保持在内存中产生的 trie node 数量(默认:120)

帐户选项:

--unlock value	需解锁账户用逗号分隔
--password value	用于非交互式密码输入的密码
文件	

API 和控制台选项:

--rpc	启用 HTTP-RPC 服务器
--rpcaddr value (默认值: "localhost")	HTTP-RPC 服务器接口地址
--rpcport value (默认值:8545)	HTTP-RPC 服务器监听端口
--rpcapi value API	基于 HTTP-RPC 接口提供的
--ws	启用 WS-RPC 服务器
--wsaddr value 址(默认值: "localhost")	WS-RPC 服务器监听接口地
--wsport value 认值:8546)	WS-RPC 服务器监听端口(默

--wsapi value	基于 WS-RPC 的接口提供的 API
--wsorigins value	websockets 请求允许的源
--ipcdisable	禁用 IPC-RPC 服务器
--ipcpath	包含在 datadir 里的 IPC socket/pipe 文件名(转义过的显式路径)
--rpccorsdomain value	允许跨域请求的域名列表(逗号分隔)(浏览器强制)
--jspath loadScript	JavaScript 加载脚本的根路径(默认值: ".")
--exec value	执行 JavaScript 语句(只能结合 console/attach 使用)
--preload value	预加载到控制台的 JavaScript 文件列表(逗号分隔)

网络选项:

--bootnodes value	用于 P2P 发现引导的 enode urls(逗号分隔)(对于 light servers 用 v4+v5 代替)
--------------------------	---

--bootnodesv4 value 用于 P2P v4 发现引导的 enode
urls(逗号分隔) (light server, 全节点)

--bootnodesv5 value 用于 P2P v5 发现引导的 enode
urls(逗号分隔) (light server, 轻节点)

--port value 网卡监听端口(默认值:30303)

--maxpeers value 最大的网络节点数量(如果设置为 0,
网络将被禁用)(默认值:25)

--maxpendpeers value 最大尝试连接的数量(如果设置为
0, 则将使用默认值)(默认值:0)

--nat value NAT 端 口 映 射 机 制
(any|none|upnp|pmp|extip:<IP>) (默认: "any")

--nodiscover 禁用节点发现机制(手动添加节点)

--v5disc 启用实验性的 RLPx V5(Topic 发现)
机制

--nodekey value P2P 节点密钥文件

--nodekeyhex value 十六进制的 P2P 节点密钥(用于测
试)

矿工选项:

--mine	打开挖矿
--minerthreads value	挖矿使用的 CPU 线程数量(默认值:8)
--etherbase value	挖矿奖励地址(默认=第一个创建的帐户)(默认值: "0")
--targetgaslimit value	目标 gas 限制: 设置最低 gas 限制 (低于这个不会被挖?) (默认值: "4712388")
--gasprice value	挖矿接受交易的最低 gas 价格
--extradata value	矿工设置的额外块数据 (默认=client version)

GAS 价格选项:

--gpoblocks value	用于检查 gas 价格的最近块的个数 (默认: 10)
--gpopercentile value	建议 gas 价参考最近交易的 gas 价的百分位数, (默认: 50)

虚拟机的选项:

--vmdebug 记录 VM 及合约调试信息

日志和调试选项:

--metrics 启用 metrics 收集和报告

--fakepow 禁用 proof-of-work 验证

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

**--vmodule value 每个模块详细度:以
<pattern>=<level> 的逗号分隔列表 (比如
eth/=6,p2p=5)**

**--backtrace value 请求特定日志记录堆栈跟踪 (比如
"block.go:271")**

**--debug 突出显示调用位置日志(文件
名及行号)**

--pprof 启用 pprof HTTP 服务器

**--pprofaddr value pprof HTTP 服务器监听接口
(默认值:127.0.0.1)**

--pprofport value (默认值:6060)	pprof HTTP 服务器监听端口
--memprofilerate value profiling (默认:524288)	按指定频率打开 memory
--blockprofilerate value profiling (默认值:0)	按指定频率打开 block
--cpuprofile value	将 CPU profile 写入指定文件
--trace value	将 execution trace 写入指定文件

WHISPER 实验选项:

--shh	启用 Whisper
--shh.maxmessagesize value (默认值: 1048576)	可接受的最大的消息大小
--shh.pow value 值: 0.2)	可接受的最小的 POW (默认

弃用选项:

--fast	开启快速同步
---------------	---------------

--light 启用轻客户端模式

其他选项:

-help, -h 显示帮助

我是【链客】六级算力等级《守护平井一夫》 为各位解答区块链技术问题，欢迎加入。

链客区块链技术问答社区，有问必答！！

国内域名：www.liankexing.com 复制网址至浏览器即可进入社区

国际域名：www.lk.wiki

QQ 群: 725414372