以太坊私有连搭建及智能合约编写实践

1. 安装go
2. 通过wget下载go-linux-64位源码包，手动wget安装

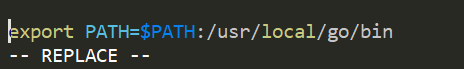


1. 解压二进制文件到 /usr/local 目录



1. 修改环境变量

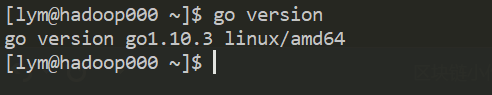
执行 vi ~/.bash\_profile



保存后，执行source ~/.bash\_profile

1. 检验安装情况

执行 go version

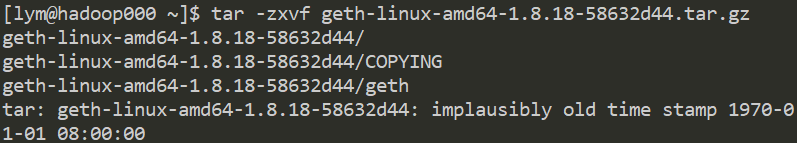


安装成功。

1. 安装go-ethereum
2. 去官网 <https://geth.ethereum.org/downloads/> 下载Geth1.8.18 for Linux



1. 解压到当前文件夹



1. 修改文件夹名称



1. 配置环境变量

vi ~/.bash\_profile

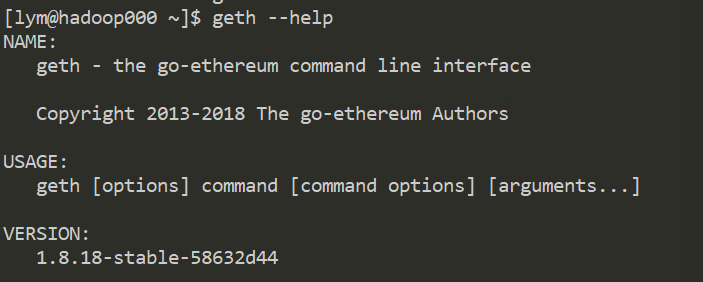


使bash\_profile生效

source ~/.bash\_profile

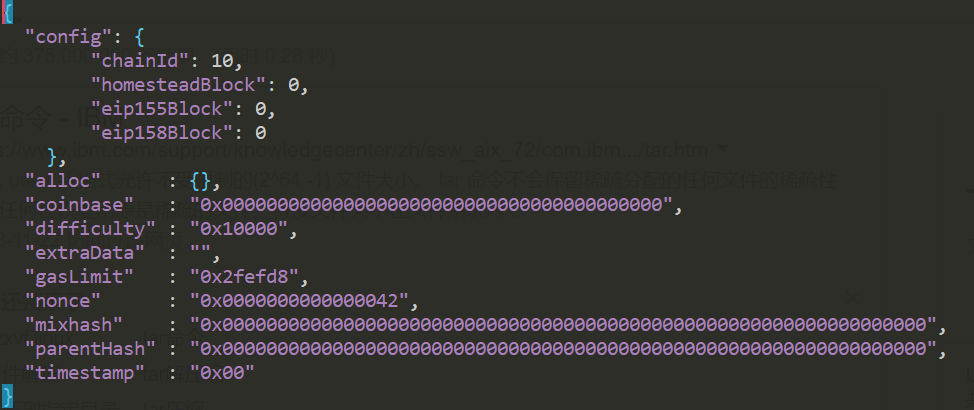
1. 检查是否安装成功

执行geth –help



安装成功

1. 搭建私有链
2. **编写创始区块信息，保存在gensis.json中**



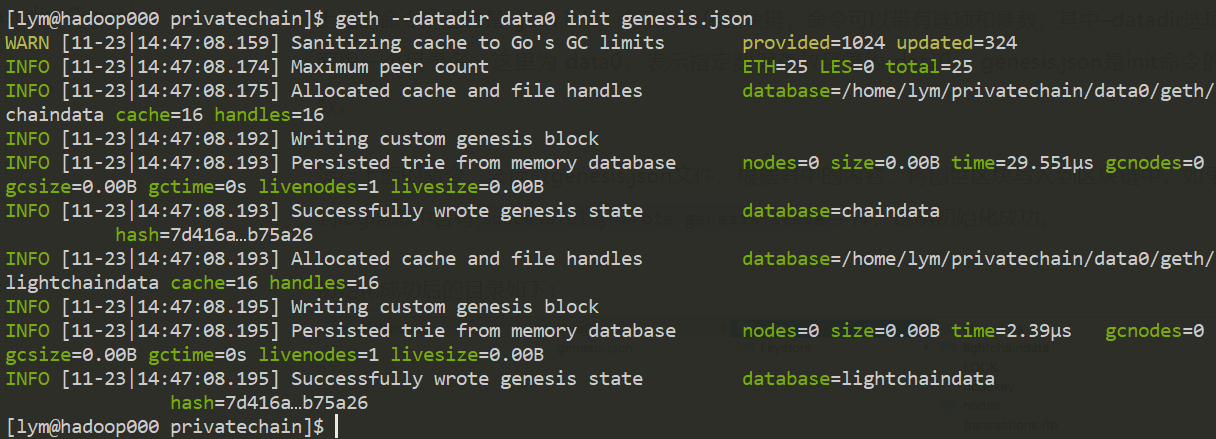
1. **将创世区块信息写入创始区块**

此时目录结构如下:

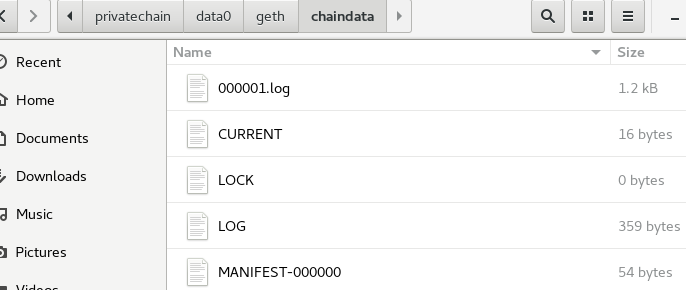


执行命令geth --datadir data0 init genesis.json

输出如下：



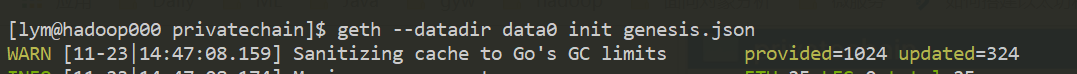
初始化成功，此时data0目录下多出geth和keystore文件夹，geth包含chaindata文件夹，chaindata下有如下文件：



1. **启动私有链节点**

初始化完成后就产生了一条私有链，可以启动该私有链

执行命令：geth --datadir data0 --networkid 999 console

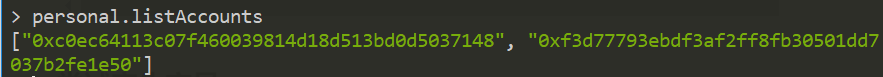




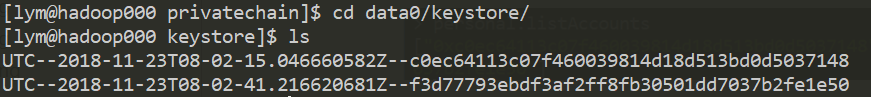
Javascript交互式控制台启动成功

1. **创建账户**

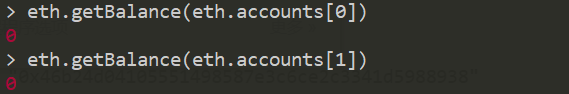
通过personal.newAccount()创建账户，eth.accounts查看账户



在~/privatechain/data0/keystore中查看，发现多了两个json文件



1. **查看账户余额**



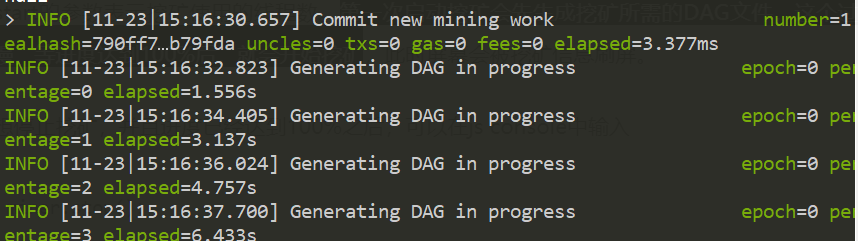
目前账户余额都是0，需要从其他账户转入或者挖矿获取

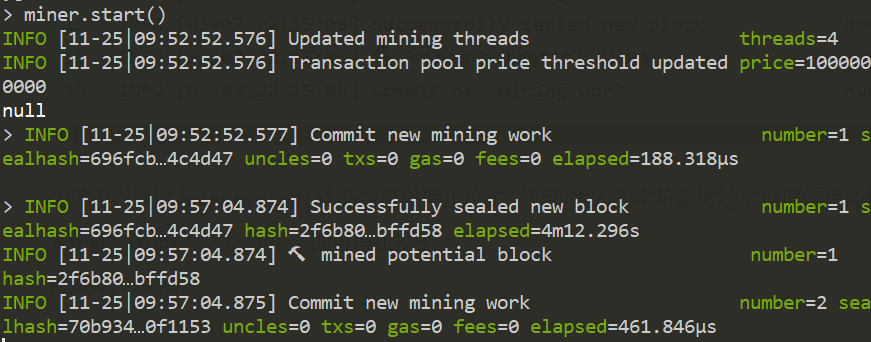
1. **开始挖矿**

默认使用第一个账户作为矿工账户

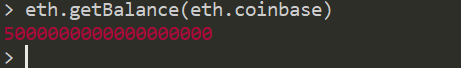
执行miner.start()

第一次启动挖矿会生成挖矿所需的DAG文件，这个过程有点慢，等进度达到100%后，就会开始挖矿。下图为生成DAG：



 DAG生成完成后开始正式挖矿，试了几次才挖到矿

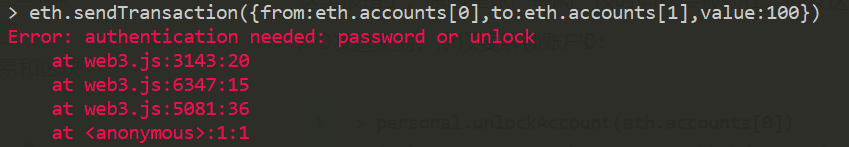
查看账户余额：



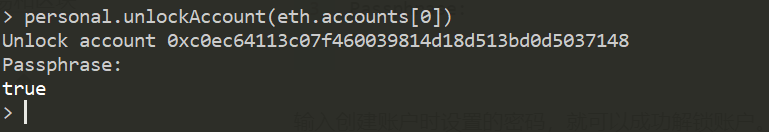
此时coinbase账户里已经有钱了

1. **转账**

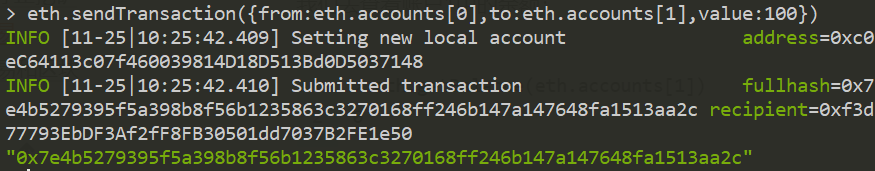
尝试将basecoin账户的钱转到另外一个账户



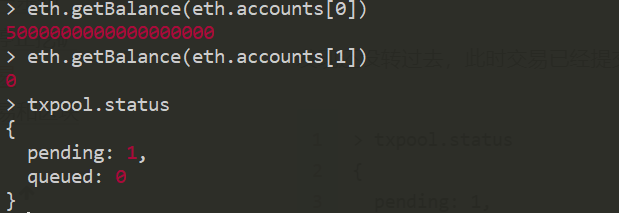
报错，因为每隔一段时间账户会自动锁定，需输入密码解锁



再次尝试交易



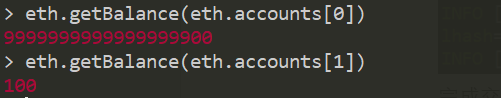
查看两个账户的余额



发现账户余额并未发生改变，交易处于pending状态，要使交易被处理则要重新启动挖矿



完成交易操作，再次检查账户余额

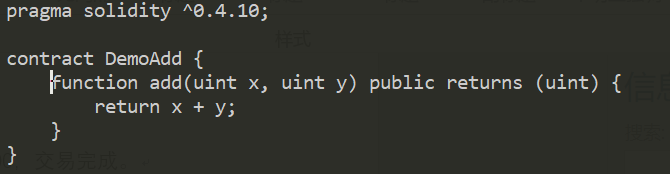


此时coinbase账户余额被扣除100，另一个账户余额增加100，交易完成。

1. 编写智能合约

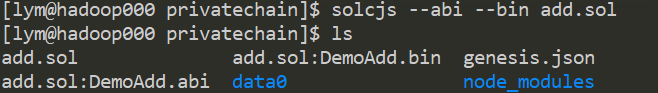
编写一个求加法计算的智能合约

1. 在线编辑器remix中编写智能合约代码，文件名为add.sol



1. 编译add.sol文件

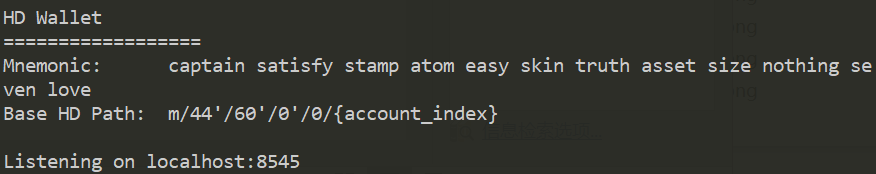
执行 solcjs --abi --bin add.sol



执行完成后目录下多了bin和abi文件

1. 准备部署环境，安装ethereumjs-testrpc

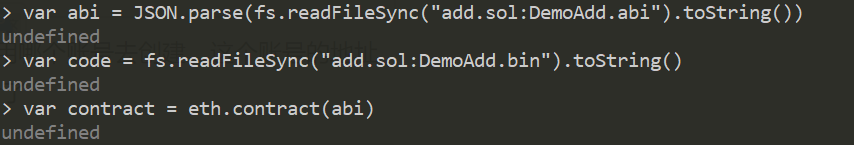
安装完成后输入命令testrpc



1. 在node环境中链接testrpc



1. 开始部署，读取abi文件和bin文件

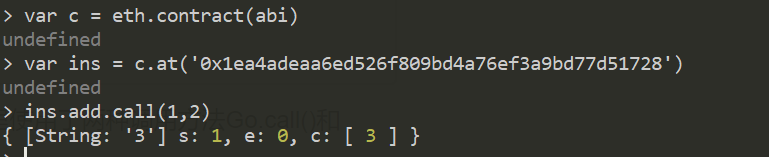




1. 获取部署实例地址



1. 使用智能合约



得到正确结果。