## 以太坊私有链部署

1. 操作系统准备
2. golang安装
3. 下载以太坊
4. 安装以太坊
5. 创世区块文件的准备
6. 创世区块初始化
7. 以太坊启动

### **1. 操作系统准备**

我们使用的操作系统是CentOS7，安装之后需要系统能够上网，因为后面的步骤需要联网安装以太坊所需的环境。系统安装好之后，执行 cat /etc/redhat-release确认系统版本，以下是我们执行的结果： su root 进入root权限

[a1@localhost ~]$ cat /etc/redhat-releaseCentOS Linux release 7.0.1406 (Core)

### **2. golang安装（我在这里停留了很久）**

1)首先执行安装yum源。

rpm -ivh http://ftp.riken.jp/Linux/fedora/epel/6/i386/epel-release-6-8.noarch.rpm

yum源安装成功后就可以使用下面的命令安装golang了

yum install golang

2)下载golang，然后解压，复制到指定路径

cp -r go... /usr/local

进去该文件增加路径配置

vi /etc/profile在文件最后面加

export GOPATH=/usr/local/pathgo

export GOROOT=/usr/local/go

export PATH=$GOROOT/bin:$PATH

source一下

source /etc/profile

之后就可以查看go是否安装成功，以及查看go版本 go version

-bash-4.2# go\> Go is a tool for managing Go source code.

Usage:

go command [arguments]

### **3. 下载以太坊源代码**

从https://github.com/ethereum/go-ethereum/releases/tag/v1.8.3 这个地址下载的zip压缩格式的源代码，为防止将来该地址不再能用，我们将zip压缩格式的源代码放到本站，地址是：<https://github.com/ethereum/go-ethereum/archive/v1.8.3.zip>

### **4. 安装以太坊**

将下载的zip格式的源码包解压缩并进入解压缩后的目录。

unzip go-ethereum.zip(看自己的文件名解压)

对源码编译一下

make

### **5. 创世区块文件的准备**

在go-ethereum-1.8.3/build/bin目录下创建vi init.json的文本文件，内容如下，这是新版本的创始块，参数有所改变：

vi init.json

{

"nonce": "0x0000000000000042",

"difficulty": "0x400",

"mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",

"coinbase": "0x0000000000000000000000000000000000000000",

"timestamp": "0x00",

"parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",

"extraData": "0x11bbe8db4e347b4e8c937c1c8370e4b5ed33adb3db69cbdb7a38e1e50b1b82fa",

"gasLimit": "0x1388",

"alloc": {

"3282791d6fd713f1e94f4bfd565eaa78b3a0599d": {

"balance": "1337000000000000000000"

},

"17961d633bcf20a7b029a7d94b7df4da2ec5427f": {

"balance": "229427000000000000000"

}

},

"config": {

"chainId": 15,

"homesteadBlock": 0,

"eip155Block": 0,

"eip158Block": 0

}}

### **6. 创世区块初始化**

在go-ethereum-1.8.3/build/bin目录下执行以下命令以完成创世区块的创建：

./geth --datadir "/app/chain" init init.json

注意：上面命令中–datadir后面的 /app/chain可以任意指定，无需提前创建，但是一定要保证有足够的磁盘空间。init.json是我们在上一步创建的初始块文件，注意文件名要一致。

执行完成后，将会出现类似以下信息：

I0927 14:31:07.374864 ethdb/database.go:82] Alloted 16MB cache and 16 file handles to /app/chain/chaindata

I0927 14:31:07.402544 cmd/geth/main.go:299] successfully wrote genesis block and/or chain rule set: 244469eaad4c9c7a245fa4962a6ed7346e9925317f222b580e7f3113f9722657

### **7. 以太坊启动**

仍然在go-ethereum-1.8.3/build/bin目录下执行，执行的命令是：

./geth --rpc --rpccorsdomain "\*" --datadir "/app/chain" --port "30303" --rpcapi "db,eth,net,web3" --networkid 100000 console

注意：上面命令中–datadir 后的"/app/chain"要跟我们上一步的–datadir 参数一致。解释一下区块里面的参数https://bitshuo.com/topic/58ce7b2d0a3de8932e6f75ba

执行完此命令后将会出现类似以下信息

Welcome to the Geth JavaScript console!

自动进入geth的命令行则说明以太坊私有链安装成功了。