**Creating a Private Ethereum Chain**

In this post, let us see how to set up your own private Etherum blockchain. Since the chain is private, one can easily develop/test with this network as the chain syncs up fast.

**Versions used**

Ubuntu – 14.04.1 and geth  1.6.1-unstable version is used in this tutorial.

If you have not installed geth, use the below link and download the latest geth version

<https://github.com/ethereum/go-ethereum/releases>

**Installation instructions**

<https://github.com/ethereum/go-ethereum/wiki/Installation-Instructions-for-Ubuntu>

Execute geth command and make sure it is working fine.

**Step 1 :**

Let us create appropriate directories and accounts to set up this private chain.

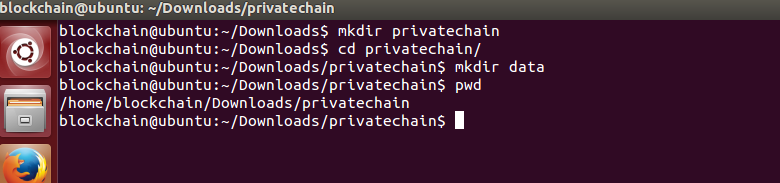
**Directory creation**

Execute the following commands to setup the directory. You can create choice of your own

mkdir privatechain

cd privatechain

mkdir data



**Account creation**

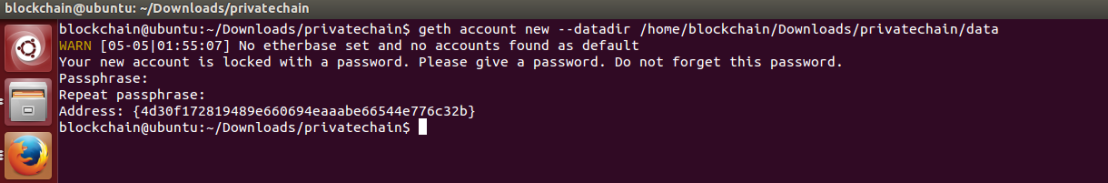
Now let us create accounts using the following command

 geth account new –datadir /home/blockchain/Downloads/privatechain/data

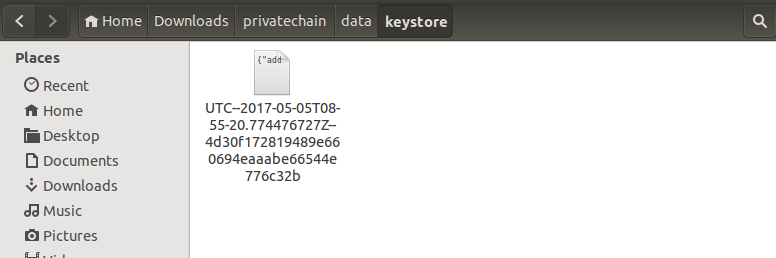
Enter the password and now we are all set with the account. Here the account address is

4d30f172819489e660694eaaabe66544e776c32b

Note down your account address



After account creation, you could notice keystore directory inside the data folder



**Step 2 :**

In order to set up a private chain, one has to configure a genesis file which will be used by the private network.

**WARN**: Configuration file used in the previous version differ from the current one.

This is our configuration file and let us name it as custom\_genesis.json and save it under

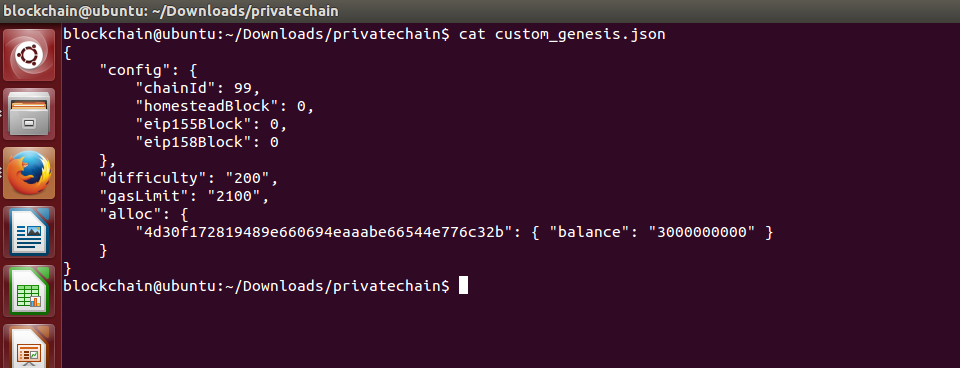
/home/blockchain/Downloads/privatechain

{  
“config”: {  
“chainId”: 99,  
“homesteadBlock”: 0,  
“eip155Block”: 0,  
“eip158Block”: 0  
},  
“difficulty”: “200”,  
“gasLimit”: “2100”,  
“alloc”: {  
“4d30f172819489e660694eaaabe66544e776c32b”: { “balance”: “3000000000” }  
}  
}

**Note**:

alloc section is to allocate few ethers on load. This can be omitted.

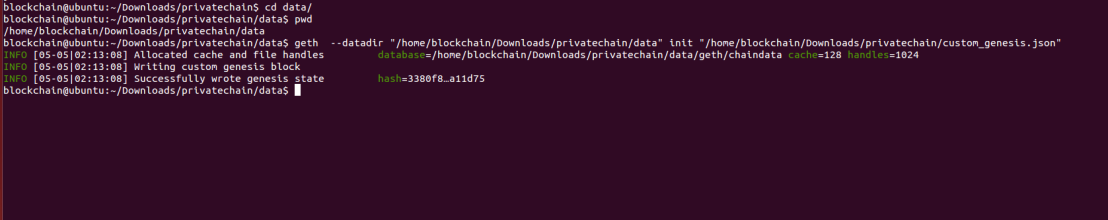
Also, don’t forget to mention your account address



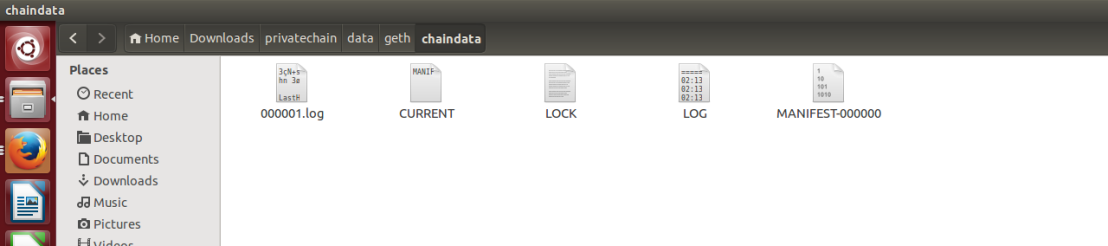
**Step 3:**

Execute the following command to initialize the chain with our custom genesis file

geth  –datadir “/home/blockchain/Downloads/privatechain/data” init “/home/blockchain/Downloads/privatechain/custom\_genesis.json”



Once done, chain related files gets created

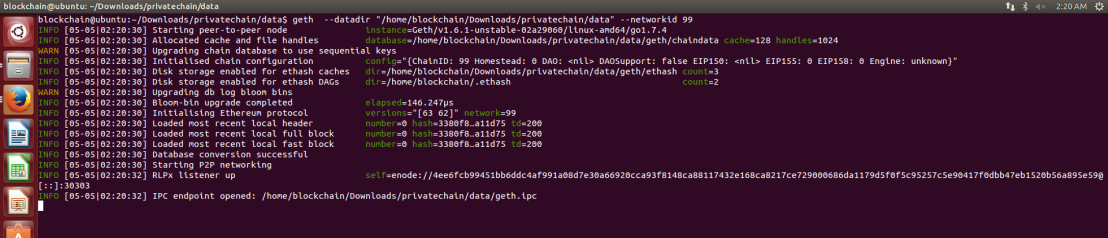


**Step 4:**

Let us start the chain now. For the case of simplicity, we are starting our private chain with simple options.

geth  –datadir “/home/blockchain/Downloads/privatechain/data” –networkid 99

You should be able to see something like this. Private chain has started in network 99

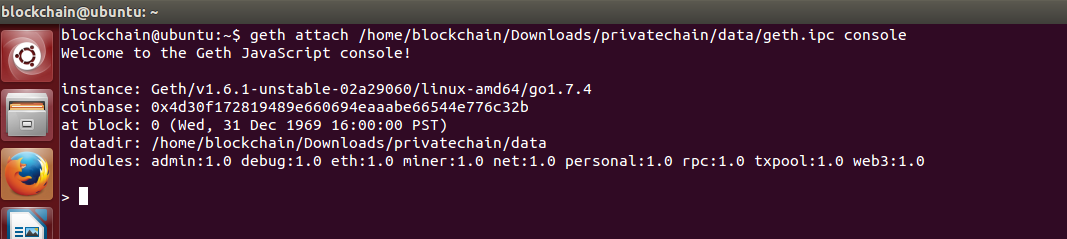


**Step 5:**

Let us ensure mining process works fine. This is required because in Ethereum blockchain world all the transactions has to be mined

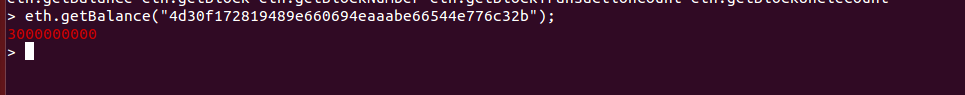
a) Open another Terminal window and execute the following command. This opens a simple Javascript session. We can use this session to interact with our private chain

geth attach /home/blockchain/Downloads/privatechain/data/geth.ipc console



b) Let us check the balance of our account

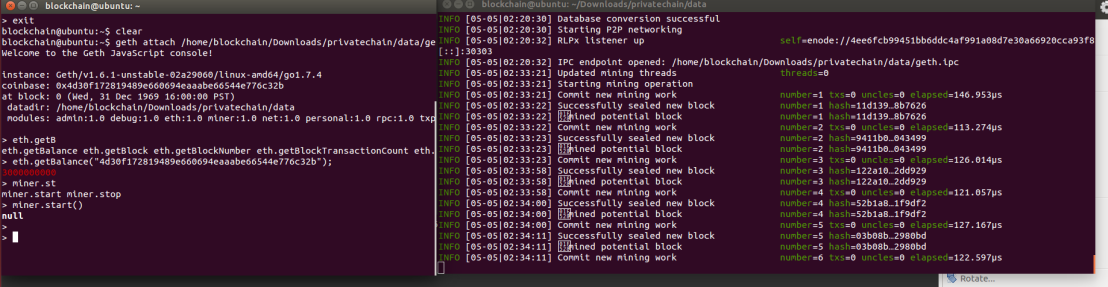
 eth.getBalance(“4d30f172819489e660694eaaabe66544e776c32b”);



c) Now let us start mining. Execute the following command in Javascript session to start mining

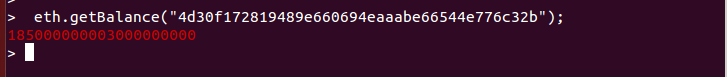
miner.start()

In another window, you can see mining process has started.



d) Let us check the balance once again

 eth.getBalance(“4d30f172819489e660694eaaabe66544e776c32b”);



You can notice the balance amount has increased due to mining.

In order to stop the mining process, execute the following command

miner.stop()