**分布式miner组网方案**

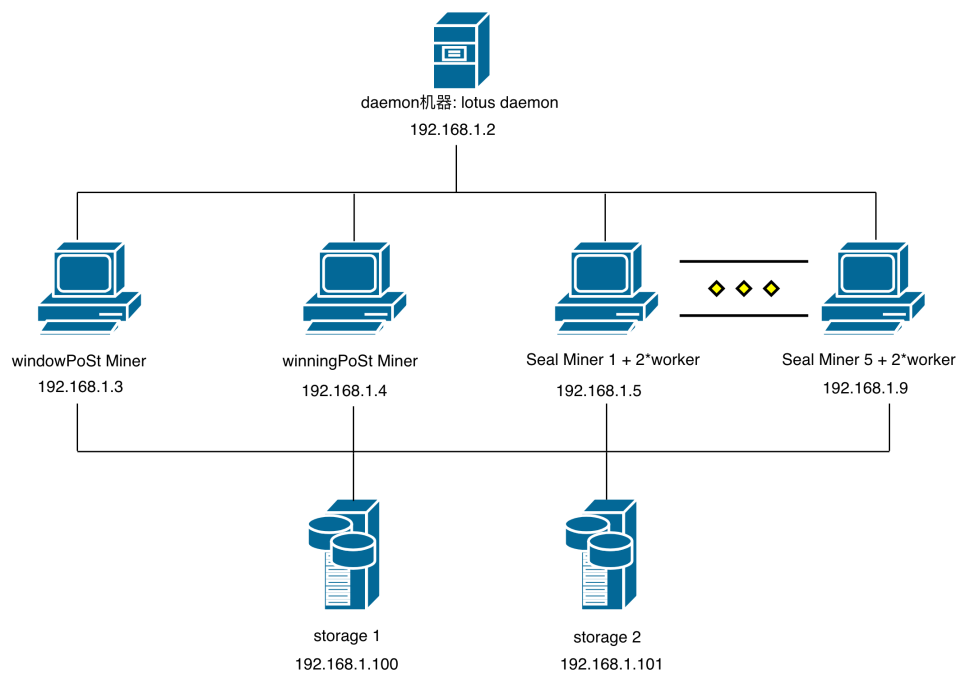
**By 默然**

## 组网方案

**1、机器配置**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能** | **数量** | **内网IP** | **配置信息** |
| daemon机器 | 1 | 192.168.1.2 | 7302 U + 512G内存 + 7T NVME |
| miner机器 | 2 | 192.168.1.3~4 | 7302 U + 512G内存 + 2080TI\*2 + 7T NVME |
| 算力机器 | 5 | 192.168.1.5~9 | 7302 U\*2 + 2T内存 + 2080TI\*2 + 28T NVME |
| 存储机器 | 2 | 192.168.1.100～101 | 10T \* 24盘 (接交换机40G口) |

**2、组网架构**



## 全部机器环境设置

1. **安装操作系统ubuntu18.04 server版本**

sudo useradd -d "/home/filecoin" -m -s "/bin/bash" filecoin # 建立filecoin用户

usermod -u 1010 filecoin # 确保各台机器的用户权限一致，nfs挂载读写所需(用root用户重新登陆执行)

groupmod -g 1010 filecoin # 确保各台机器的用户组权限一致，nfs挂载读写所需(用root用户重新登陆执行)

sudo chmod 744 /etc/sudoers && sudo sed -i '$a\\nfilecoin ALL=(ALL) ALL' /etc/sudoers && sudo chmod 440 /etc/sudoers # 管理权限

sudo chmod 744 /etc/sudoers && sudo sed -i '$a\\nfilecoin ALL=(ALL) NOPASSWD : ALL' /etc/sudoers && sudo chmod 440 /etc/sudoers # 免密设置

**2、安装依赖环境**

sudo apt update

sudo apt install make git gcc bzr jq pkg-config mesa-opencl-icd ocl-icd-opencl-dev llvm clang curl numactl

sudo apt install tmux libhwloc-dev hwloc nfs-kernel-server nfs-common

sudo ln -s /usr/lib/x86\_64-linux-gnu/libhwloc.so.5.7.6 /usr/lib/x86\_64-linux-gnu/libhwloc.so.15

sudo timedatectl set-timezone Asia/Shanghai # 设置中国时区

1. **设置环境变量**

在/etc/profile文件末加入下面内容：

ulimit -SHn 1048576 # 设置打开文件句柄数量

export NVME\_PATH=/nvme # 设置快速缓存盘所在的目录

export LOTUS\_PATH=$NVME\_PATH/.lotus # 设置daemon的配置目录

export LOTUS\_STORAGE\_PATH=$NVME\_PATH/.lotusstorage # 设置miner的配置目录

export WORKER\_PATH=$NVME\_PATH/.lotusworker # 设置worker的配置目录

export TMPDIR=$NVME\_PATH/tmp # 设置临时缓存文件目录

export FIL\_PROOFS\_PARENT\_CACHE=$NVME\_PATH/filecoin-proof-parameters/ # 设置sdr cache文件目录

export FIL\_PROOFS\_PARAMETER\_CACHE=$NVME\_PATH/filecoin-proof-parameters/ # 设置证明参数文件目录

export IPFS\_GATEWAY=https://proof-parameters.s3.cn-south-1.jdcloud-oss.com/ipfs/ # 设置证明参数文件下载网关

export RUST\_LOG=info # 设置打印消息级别为info

export FIL\_PROOFS\_USE\_MULTICORE\_SDR=1 # 设置p1是否启用多核心

export FIL\_PROOFS\_MULTICORE\_SDR\_PRODUCERS=1 # 设置p1启用多核心后使用多少个核心运行

export FIL\_PROOFS\_MAXIMIZE\_CACHING=1 # 新版本已无实际用途

export FIL\_PROOFS\_USE\_GPU\_COLUMN\_BUILDER=1 # 设置p2是否启用GPU计算列hash

export FIL\_PROOFS\_USE\_GPU\_TREE\_BUILDER=1 # 设置p2是否启用GPU计算hash树

export BELLMAN\_CUSTOM\_GPU="GeForce RTX 2080 Ti:4352" # 设置GPU类型

export TRUST\_PARAMS=1 # 设置启动miner或worker时跳过证明参数文件验证

## 三、存储机器(NFS服务设置)

sudo mkdir -p /mnt/disk0 # 建立挂载目录

sudo mkfs.xfs -f /dev/sda # 格式化sda盘

echo /dev/disk/by-uuid/$(lsblk -f | grep sda | awk -F ' ' '{print $3}') /mnt/disk0 xfs defaults 0 0 | sudo tee -a /etc/fstab # 设置开机启动

sudo mount -av # 挂载sda盘

echo "/mnt/disk0 \*(rw,no\_root\_squash,no\_subtree\_check,async)" | sudo tee -a /etc/exports # 配置nfs服务信息

sudo service nfs-kernel-server restart # 重启nfs服务

## 四、管理机器

**1、**设置管理机免钥登陆需管理的机器

ssh-keygen # 生成私钥（不存在的情况下需生成）

ssh-copy-id 192.168.1.\* # 推送私钥到需要管理的机器上(推送时需密码)

**2、安装编译环境**

**1) 安装设置golang1.15.6版本**

1.1）下载安装golang

wget https://dl.google.com/go/go1.15.6.linux-amd64.tar.gz

sudo tar -zxvf go1.15.6.linux-amd64.tar.gz -C /usr/local

1.2）设置golang环境（/etc/profile文件末添加）

export GOROOT=/usr/local/go

export PATH=$PATH:/usr/local/go/bin

export GOPATH=$HOME/go

export PATH=$PATH:$GOPATH/bin

export GO111MODULE=on

export GOPROXY=https://goproxy.cn

1.3）使golang环境变量生效

source /etc/profile

1. **安装nightly最新版本的rust**

2.1）下载安装rust

curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh # 按提示进行安装（可能需要梯子）

2.2）设置rust下载源，创建 $HOME/.cargo/config文件，写入下面内容：

[source.crates-io]

replace-with = 'sjtu'

[source.sjtu]

registry = "https://mirrors.sjtug.sjtu.edu.cn/git/crates.io-index"

2.3）使rust生效

source $HOME/.cargo/env

**3、编译lotus**

git clone <https://github.com/filecoin-project/lotus.git> && cd lotus/ # 此处为官方程序，需换成优化过的版本

FFI\_BUILD\_FROM\_SOURCE=1 RUSTFLAGS='-C target-cpu=native' make clean deps all # 生成lotus、lotus -miner、lotus-worker

**4、下发挖矿程序**

将上一步生成的lotus、lotus -miner、lotus-worker拷贝到每台机器的/usr/local/bin/目录下，并修改为filecoin用户的执行权限

## 五、daemon机器(部署节点)

1、挂载存储机器共享的NFS

sudo mkdir -p $NVME\_PATH/tmp # 建立临时文件缓存目录

sudo mkdir -p /mnt/192.168.1.100/disk0 # 建立存储机器1.100的nfs挂载目录

sudo mkdir -p /mnt/192.168.1.101/disk0 # 建立存储机器1.101的nfs挂载目录

sudo chown -R filecoin.filecoin/mnt/192.168.1.100/disk0 # 改变挂载目录的所有者为用户filecoin

sudo chown -R filecoin.filecoin/mnt/192.168.1.101/disk0 # 改变挂载目录的所有者为用户filecoin

sudo mount -t nfs -o noatime 192.168.1.100:/mnt/disk0 /mnt/192.168.1.100/disk0 # 挂载1.100的nfs共享存储

sudo mount -t nfs -o noatime 192.168.1.101:/mnt/disk0 /mnt/192.168.1.101/disk0 # 挂载1.101的nfs共享存储

2、daemon使用

**1）启动节点守护程序**

rm -rf ~/.lotus # 如果之前已经运行过lotus并想删除所有以前的数据执行此命令进行操作

tmux new -s lotus -d -n daemon # 开启后台运行程序

tmux send-keys -t lotus:daemon "lotus daemon" C-m # 将守护进程放到后台运行, 执行后会生成~/.lotus

**2）设置节点监听地址并重启daemon**

tmux send-keys -t lotus:daemon "C-c" C-m # 发送ctrol + c 到后台daemon结束掉daemon

pkill -9 lotus # 再次杀死daemon

tmux send-keys -t lotus:daemon "lotus daemon" C-m # 重新启动daemon

**3）检查连接网络节点数量及检查区块同步状态**

lotus net peers | wc -l # 运行此命令前，需要daemon正常运行

lotus sync status # 查询区块同步情况(有网络连接节点才会同步)

**4）创建一个钱包地址(建立矿工需要)**

lotus wallet new bls

**5）发送币到刚建立的钱包地址(建立矿工需要)**

lotus send <walletaddress> <amount>

**6）检查钱包余额**

lotus wallet balance <f3...>

3、lotus-miner使用

**1）初始化矿工**

lotus-miner init --owner=<bls-address> --sector-size=64GiB # 保证区块同步、钱包有币,LOTUS\_STORAGE\_PATH目录会生成miner信息

**2）运行miner**

tmux new-window -t lotus -n miner # 开启miner后台运行程序

tmux send-keys -t lotus:miner "lotus-miner run" C-m # 后台运行miner

**3）设置miner的sealing目录属性**

sed -i 's/\"CanStore\": true/\"CanStore\": false/' $LOTUS\_STORAGE\_PATH/sectorstore.json # 设置miner的sealing目录属性

lotus-miner storage attach --init --store /mnt/192.168.1.100/disk0/storage # 设置miner的存储目录1

lotus-miner storage attach --init --store /mnt/192.168.1.101/disk0/storage # 设置miner的存储目录2

4、lotus-worker使用

**1）运行worker**

tmux new-window -t lotus -n worker # 开启miner后台运行程序

tmux send-keys -t lotus:worker "lotus-worker run" C-m # 后台运行worker，WORKER\_PATH目录会生成worker的配置信息

**2）设置worker的sealing目录属性**

cp $LOTUS\_STORAGE\_PATH/storage.json $WORKER\_PATH/ # 设置worker的sealing目录指向miner的sealing目录

5、停用daemon机器的miner与worker

tmux send-keys -t lotus:miner "C-c" C-m # 发送ctrol + c 到后台miner结束掉lotus-miner

tmux send-keys -t lotus:worker "C-c" C-m # 发送ctrol + c 到后台worker结束掉lotus-worker

pkill -9 lotus-miner # 再次杀死lotus-miner

pkill -9 lotus-worker # 再次杀死lotus-worker

tmux send-keys -t lotus:miner "exit" C-m # 退出后台miner

tmux send-keys -t lotus:worker "exit" C-m # 再次后台worker

## 六、miner机器与算力机器

1、挂载存储机器共享的NFS

sudo mkdir -p $NVME\_PATH/tmp # 建立临时文件缓存目录

sudo mkdir -p /mnt/192.168.1.100/disk0 # 建立存储机器1.100的nfs挂载目录

sudo mkdir -p /mnt/192.168.1.101/disk0 # 建立存储机器1.101的nfs挂载目录

sudo chown -R filecoin.filecoin/mnt/192.168.1.100/disk0 # 改变挂载目录的所有者为用户filecoin

sudo chown -R filecoin.filecoin/mnt/192.168.1.101/disk0 # 改变挂载目录的所有者为用户filecoin

sudo mount -t nfs -o noatime 192.168.1.100:/mnt/disk0 /mnt/192.168.1.100/disk0 # 挂载1.100的nfs共享存储

sudo mount -t nfs -o noatime 192.168.1.101:/mnt/disk0 /mnt/192.168.1.101/disk0 # 挂载1.101的nfs共享存储

2、配置miner、worker运行环境

**1）拷贝daemon的配置目录（目录内只需api、token文件）**

daemon机器：$LOTUS\_PATH ---copy to--> miner、算力机器：$LOTUS\_PATH

**2）拷贝miner的配置目录**

daemon机器：$LOTUS\_STORAGE\_PATH --copy to--> miner、算力机器：$LOTUS\_STORAGE\_PATH

**3）拷贝worker的配置目录**

daemon机器：$WORKER\_PATH --copy to--> miner、算力机器：$WORKER\_PATH

3、miner机器运行PoSt-Miner

**1）miner机器1（运行winning-PoSt-Miner与扇区分配服务）---192.168.1.3**

sed -i 's/# AllowAddPiece = true/AllowAddPiece = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的ap功能

sed -i 's/# AllowPreCommit1 = true/AllowPreCommit1 = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的p1功能

sed -i 's/# AllowPreCommit2 = true/AllowPreCommit2 = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的p2功能

sed -i 's/# AllowCommit = true/AllowCommit = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的c2功能

tmux new -s lotus -d -n miner

tmux send-keys -t lotus:miner "lotus-miner run --wdpost=false --wnpost=true --p2p=false --sctype=alloce --sclisten=192.168.1.3:1357" C-m

**2）miner机器2（运行window-PoSt-Miner）---192.168.1.4**

sed -i 's/# AllowAddPiece = true/AllowAddPiece = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的ap功能

sed -i 's/# AllowPreCommit1 = true/AllowPreCommit1 = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的p1功能

sed -i 's/# AllowPreCommit2 = true/AllowPreCommit2 = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的p2功能

sed -i 's/# AllowCommit = true/AllowCommit = false/' $LOTUS\_STORAGE\_PATH/config.toml # 禁用miner的c2功能

tmux new -s lotus -d -n miner

tmux send-keys -t lotus:miner "lotus-miner run --wdpost=true --wnpost=false --p2p=false" C-m

4、算力机器运行Seal-Miner及worker**---192.168.1.5 ~9**

**1）运行miner（仅做本机的调度及P1任务）---并行13个p1**

tmux new -s lotus -d -n miner

tmux send-keys -t lotus:miner "mkdir -p $TMPDIR/tmp0" C-m

tmux send-keys -t lotus:miner "mkdir -p $TMPDIR/tmp1" C-m

tmux send-keys -t lotus:miner "export FIL\_PROOFS\_MULTICORE\_SDR\_PRODUCERS=1 " C-m

tmux send-keys -t lotus:miner "export SDR\_CPU\_INFO='0 8 1 9 2 10 3 11 4 12 5 13 6 14 7 15' " C-m

tmux send-keys -t lotus:miner "lotus-miner run --wdpost=false --wnpost=false --sctype=get --sclisten=192.168.1.3:1357 --workername=p1 --ability=AP:0,PC1:13,PC2:0,C1:0,C2:0,FIN:5,GET:5,UNS:1,RD:1 " C-m

**2）运行worker0（仅做AP/P2/C1任务）---绑定1张显卡运行**

tmux new-window -t lotus -n worker0

tmux send-keys -t lotus:worker0 "export WORKER\_PATH=$NVME\_PATH/.worker0" C-m

tmux send-keys -t lotus:worker0 "export CUDA\_VISIBLE\_DEVICES=0" C-m

tmux send-keys -t lotus:worker0 "export TMPDIR=$TMPDIR/tmp0" C-m

tmux send-keys -t lotus:worker0 "taskset -c 36-63 lotus-worker run --listen=127.0.0.1:3450 --workername=0 --ability=AP:1,PC1:0,PC2:2,C1:1,C2:0 " C-m

**3）运行worker1（仅做AP/C2任务）---绑定另1张显卡运行**

tmux new-window -t lotus -n worker1

tmux send-keys -t lotus:worker1 "export WORKER\_PATH=$NVME\_PATH/.worker1" C-m

tmux send-keys -t lotus:worker1 "export CUDA\_VISIBLE\_DEVICES=1" C-m

tmux send-keys -t lotus:worker1 "export TMPDIR=$TMPDIR/tmp1" C-m

tmux send-keys -t lotus:worker1 "taskset -c 36-63 lotus-worker run --listen=127.0.0.1:3451 --workername=1 --ability=AP:1,PC1:0,PC2:0,C1:0,C2:1 " C-m

**4）运行AP任务命令**

lotus-miner sectors pledge # 此命令会产生需封装的原始垃圾数据，需设置脚本周期运行产生算力

## 七、其他常用命令

lotus-miner info # 查看矿工信息

lotus-miner sectors list # 查看扇区列表

lotus-miner sectors status --log <sectorid> # 查看某个扇区的封装过程信息

lotus-miner sealing jobs # 查看矿机运行的任务

lotus-miner sealing workers # 查看接入的矿机状态

lotus-miner proving deadlines # 查看proving状态的扇区证明窗口

lotus-miner sectors pledge # 封装垃圾数据