

部署Minio高可用群集

环境准备

虚拟机及操作系统基础环境

服务器信息

主机名	IP地址	操作系统	挂载磁盘
minio01	192.168.100.181	RedHat8.6	/dev/sdb1
minio02	192.168.100.182	RedHat8.6	/dev/sdb1
minio03	192.168.100.183	RedHat8.6	/dev/sdb1
minio04	192.168.100.184	RedHat8.6	/dev/sdb1

修改计算机名称

```
# 分别在每个节点上运行
hostnamectl set-hostname minio01
hostnamectl set-hostname minio02
hostnamectl set-hostname minio03
hostnamectl set-hostname minio04
# 运行完成以上命令后重启节点生效
```

修改hosts文件，实现内网互通

```
cat >> /etc/hosts << EOF
192.168.100.181 minio01
192.168.100.182 minio02
192.168.100.183 minio03
192.168.100.184 minio04
EOF
```

初始化并挂载磁盘

```
# 查看磁盘情况
fdisk -l
Disk /dev/sda: 127 GiB, 136365211648 bytes, 266338304 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: dos
```

```
Disk identifier: 0x5997ed9a
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	2099199	2097152	1G	83	Linux
/dev/sda2		2099200	266338303	264239104	126G	8e	Linux LVM

```
Disk /dev/sdb: 200 GiB, 214748364800 bytes, 419430400 sectors
```

```
Units: sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 4096 bytes
```

```
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

由于MBR分区格式有一定限制我们这里使用GPT分区格式

```
# 下载工具gdisk
yum install -y gdisk
# 运行gdisk进行分区
gdisk /dev/sdb
```

输入? 获取命令

```
[root@minio02 ~]# gdisk /dev/sdb
GPT fdisk (gdisk) version 1.0.3

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.

Command (? for help): ?
b      back up GPT data to a file
c      change a partition's name
d      delete a partition
i      show detailed information on a partition
l      list known partition types
n      add a new partition
o      create a new empty GUID partition table (GPT)
p      print the partition table
q      quit without saving changes
r      recovery and transformation options (experts only)
s      sort partitions
t      change a partition's type code
v      verify disk
w      write table to disk and exit
x      extra functionality (experts only)
?      print this menu

Command (? for help):
```

输入n 创建分区 (一直保持默认即可)

```
Command (? for help): n
Partition number (1-128, default 1): 1
First sector (34-419430366, default = 2048) or {+}-size{KMGT}:
Last sector (2048-419430366, default = 419430366) or {+}-size{KMGT}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'
```

输入w 保存分区

```
Command (? for help): w
```

```
Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING  
PARTITIONS!!
```

最后使用命令lsblk查看块设备信息，此时已经可以看到刚新建的分区了

```
[root@minio02 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0  127G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0  126G  0 part
│   ├─rhel-root 253:0    0    70G  0 lvm /
│   ├─rhel-swap 253:1    0     4G  0 lvm [SWAP]
│   └─rhel-home 253:2    0  52.1G  0 lvm /home
sdb          8:16    0  200G  0 disk
└─sdb1       8:17    0  200G  0 part
sr0         11:0    1 1024M  0 rom
```

格式化分区为xfs

```
mkfs -t xfs /dev/sdb1
```

创建minio数据目录，然后挂载上面的磁盘

```
# 创建minio目录
mkdir -p /data/minio-data
# 把上面操作的磁盘挂载到此目录
mount /dev/sdb1 /data/minio-data
# 编辑/etc/fstab文件，永久挂载此目录
cat >> /etc/fstab << EOF
/dev/sdb1 /data/minio-data xfs defaults 0 0
EOF
```

系统配置

关闭防火墙

```
systemctl stop firewalld
systemctl disable firewalld
systemctl status firewalld
```

禁用 selinux

```
Edit the /etc/selinux/config file and change SELINUX=enforcing to SELINUX=disabled
.
Reboot the system. $ sudo reboot.
```

修改系统最大文件数

```
# 查看最大连接数
ulimit -n
ulimit -a
echo "* soft nofile 65535" >> /etc/security/limits.conf
echo "* hard nofile 65535" >> /etc/security/limits.conf
sysctl -p
reboot
```

创建minio启动脚本和配置文件目录

```
mkdir -p /data/minio/run && mkdir -p /etc/minio
```

下载minio到/data/minio/run目录下

```
cd /data/minio/run && wget https://dl.min.io/server/minio/release/linux-amd64/minio
```

部署minio

编写集群启动脚本

所有节点配置文件相同

启动脚本/data/minio/run/run.sh

```
#!/bin/bash

#其中, "MINIO_ACCESS_KEY"为用户名, "MINIO_SECRET_KEY"为密码, 密码不能设置过于简单, 不然minio会启动失败
#export MINIO_ACCESS_KEY=minio
#export MINIO_SECRET_KEY=miniostorage

export MINIO_ROOT_USER=minio
export MINIO_ROOT_PASSWORD=miniostorage

# 4 个节点, 每个节点各有两个目录
/data/minio/run/minio server --config-dir /etc/minio --address ":9000" --console-address ":9001" \
http://192.168.100.181/data/minio-data/data1 http://192.168.100.181/data/minio-data/data2 \
http://192.168.100.182/data/minio_data/data1 http://192.168.100.182/data/minio-data/data2 \
```

```
http://192.168.100.183/data/minio-data/data1 http://192.168.100.183/data/minio-  
data/data2 \  
http://192.168.100.184/data/minio-data/data1 http://192.168.100.184/data/minio-  
data/data2
```

创建systemd配置文件minio.service

```
cat > /usr/lib/systemd/system/minio.service <<EOF  
[Unit]  
Description=Minio service  
Documentation=https://docs.minio.io/  
  
[Service]  
WorkingDirectory=/data/minio/run/  
ExecStart=/data/minio/run/run.sh  
  
Restart=on-failure  
RestartSec=5  
  
[Install]  
WantedBy=multi-user.target  
EOF
```

启动

修改权限

```
chmod +x /usr/lib/systemd/system/minio.service && chmod +x /data/minio/run/minio  
&& chmod +x /data/minio/run/run.sh
```

依次启动每个服务器的minio

```
systemctl daemon-reload  
systemctl enable minio && systemctl start minio  
systemctl status minio
```

使用 journalctl -xe 查看日志