

NSD Project1 DAY03

- 1. [案例1：Keepalived高可用](#)
- 2. [案例2：部署Ceph分布式存储](#)

1 案例1：Keepalived高可用

1.1 问题

部署两台代理服务器，实现如下效果：

- 利用keepalived实现两台代理服务器的高可用
- 配置VIP为192.168.4.80
- 修改对应的域名解析记录

1.2 方案

实验拓扑如图-1所示，做具体实验前请先配置好环境。

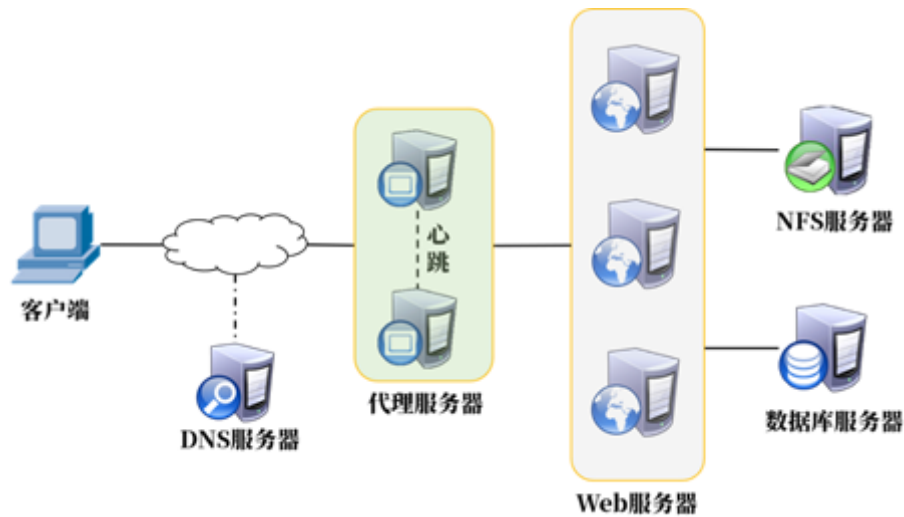


图-1

备注：实际操作中DNS服务代理服务器部署在同一台主机上（节约虚拟机资源）。
主机配置如表-1所示。

表-1

主机角色	主机名称	IP 地址
client	room9pc01	private2 (192.168.2.254/24)
代理服务器 DNS 服务器	proxy	eth0(192.168.4.5/24) eth1(192.168.2.5/24)
代理服务器 2	proxy2	eth0(192.168.4.6/24) eth1(192.168.2.6/24)
Web1 服务器	web1	eth1(192.168.2.11/24)
Web2 服务器	web2	eth1(192.168.2.12/24)
Web3 服务器	web3	eth1(192.168.2.13/24)
数据库服务器	database	eth1(192.168.2.21/24)
NFS 服务器	nfs	eth1(192.168.2.31/24)

[Top](#)

1.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置第二台代理服务器

1) 部署HAProxy

安装软件，手动修改配置文件，添加如下内容。

```
01. [root@proxy2 ~]# yum -y install haproxy
02. [root@proxy2 ~]# vim /etc/haproxy/haproxy.cfg
03. listen wordpress *:80
04.     balance roundrobin
05.     server web1 192.168.2.11:80 check inter 2000 rise 2 fall 3
06.     server web2 192.168.2.12:80 check inter 2000 rise 2 fall 3
07.     server web3 192.168.2.13:80 check inter 2000 rise 2 fall 3
08.
09. [root@proxy2 ~]# systemctl start haproxy
10. [root@proxy2 ~]# systemctl enable haproxy
```

步骤二：为两台代理服务器配置keepalived

1) 配置第一台代理服务器proxy (192.168.4.5)。

```
01. [root@proxy ~]# yum install -y keepalived
02. [root@proxy ~]# vim /etc/keepalived/keepalived.conf
03. global_defs {
04.     router_id proxy1                //设置路由ID号
05. }
06. vrrp_instance VI_1 {
07.     state MASTER                    //主服务器为MASTER (备服务器需要修改为BACKUP)
08.     interface eth0                 //定义网络接口
09.     virtual_router_id 51
10.     priority 100                   //服务器优先级,优先级高优先获取VIP (实验需要修改)
11.     advert_int 1
12.     authentication {
13.         auth_type pass
14.         auth_pass 1111              //主备服务器密码必须一致
15.     }
16.     virtual_ipaddress {             //谁是主服务器谁获得该VIP (实验需要修改)
17.         192.168.4.80
18.     }
```

[Top](#)

- ```

19. }
20. [root@proxy ~]# systemctl start keepalived
21. [root@proxy ~]# iptables -F #清空防火墙规则

```

## 2) 配置第二台代理服务器proxy (192.168.4.6) 。

- ```

01.  [root@proxy2 ~]# yum install -y keepalived
02.  [root@proxy2 ~]# vim /etc/keepalived/keepalived.conf
03.  global_defs {
04.      router_id proxy2          //设置路由ID号
05.  }
06.  vrrp_instance VI_1 {
07.      state BACKUP              //主服务器为MASTER (备服务器需要修改为BACKUP)
08.      interface eth0           //定义网络接口
09.      virtual_router_id 51
10.      priority 50              //服务器优先级,优先级高优先获取VIP
11.      advert_int 1
12.      authentication {
13.          auth_type pass
14.          auth_pass 1111       //主备服务器密码必须一致
15.      }
16.      virtual_ipaddress {      //谁是主服务器谁获得该VIP
17.          192.168.4.80
18.      }
19.  }
20.  [root@proxy2 ~]# systemctl start keepalived
21.  [root@proxy2 ~]# iptables -F          #清空防火墙规则

```

步骤三：修改DNS服务器

- 1) 修改网站域名对应的解析记录，解析到新的VIP地址。
192.168.4.5为DNS服务器。

- ```

01. [root@proxy ~]# vim /var/named/lab.com.zone
02. $TTL 1D
03. @ IN SOA @ name.invalid. (
04. 0 ; serial
05. 1D ; refresh

```

[Top](#)

- 06.1H; retry
- 07.1W; expire
- 08.3H); minimum
09. @ NS dns.lab.com.
10. dns A 192.168.4.5
11. www A 192.168.4.80

2) 重启DNS服务

```
01. [root@proxy ~]# systemctl restart named
```

2 案例2：部署Ceph分布式存储

2.1 问题

部署Ceph分布式存储，实现如下效果：

- 使用三台服务器部署Ceph分布式存储
- 实现Ceph文件系统共享
- 将网站数据从NFS迁移到Ceph存储

2.2 方案

实验拓扑如图-2所示，做具体实验前请先配置好环境。

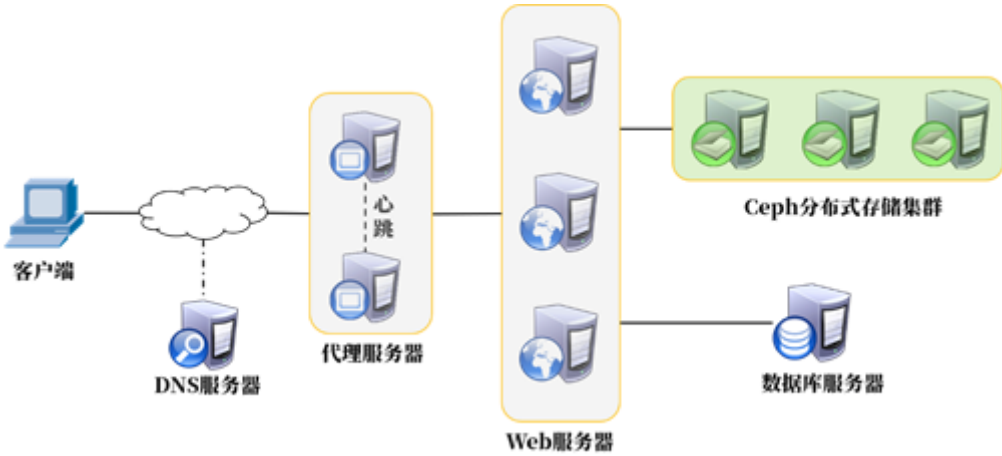


图-2

备注：实际操作中DNS服务代理服务器部署在同一台主机上（节约虚拟机资源）。  
主机配置如表-2所示。

表-2

| 主机角色      | 主机名称  | IP 地址                 |
|-----------|-------|-----------------------|
| Ceph 节点 1 | node1 | eth1(192.168.2.41/24) |
| Ceph 节点 2 | node2 | eth1(192.168.2.42/24) |
| Ceph 节点 3 | node3 | eth1(192.168.2.43/24) |

[Top](#)

## 2.3 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：准备实验环境

1) 物理机为所有节点配置yum源服务器。

提示：ceph10.iso在/linux-soft/02目录。

```
01. [root@room9pc01 ~]# mkdir /var/ftp/ceph
02. [root@room9pc01 ~]# mount ceph10.iso /var/ftp/ceph/
```

2) 在node1配置SSH密钥，让node1可用无密码连接node1,node2,node3

```
01. [root@node1 ~]# ssh-keygen -f /root/.ssh/id_rsa -N ''
02. [root@node1 ~]# for i in 41 42 43
03. do
04. ssh-copy-id 192.168.2.$i
05. done
```

3)修改/etc/hosts域名解析记录（不要删除原有的数据），同步给所有ceph节点。

```
01. [root@node1 ~]# vim /etc/hosts
02. 192.168.2.41 node1
03. 192.168.2.42 node2
04. 192.168.2.43 node3
05.
06. [root@node1 ~]# for i in 41 42 43
07. do
08. scp /etc/hosts 192.168.2.$i:/etc
09. done
```

4) 为所有ceph节点配置yum源，并将配置同步给所有节点

```
01. [root@node1 ~]# cat /etc/yum.repos.d/ceph.repo
02. [mon]
03. name=mon
04. baseurl=ftp://192.168.2.254/ceph/MON
05. gpgcheck=0
```

[Top](#)

```

06. [osd]
07. name=osd
08. baseurl=ftp://192.168.2.254/ceph/OSD
09. gpgcheck=0
10. [tools]
11. name=tools
12. baseurl=ftp://192.168.2.254/ceph/Tools
13. gpgcheck=0
14. [root@node1 ~]# yum repolist #验证YUM源软件数量
15. 源标识 源名称 状态
16. Dvd redhat 9,911
17. Mon mon 41
18. Osd osd 28
19. Tools tools 33
20. repolist: 10,013
21.
22. [root@node1 ~]# for i in 41 42 43
23. do
24. scp /etc/yum.repos.d/ceph.repo 192.168.2.$i:/etc/yum.repos.d/
25. done

```

5) 所有节点主机与真实主机的NTP服务器同步时间。

提示：默认真实物理机已经配置为NTP服务器。

```

01. [root@node1 ~]# vim /etc/chrony.conf
02.
03. server 192.168.2.254 iburst
04.
05.
06. [root@node1 ~]# for i in 41 42 43
07. do
08. scp /etc/chrony.conf 192.168.2.$i:/etc/
09. ssh 192.168.2.$i "systemctl restart chronyd"
10. done

```

6) 使用virt-manager为三台ceph虚拟机添加磁盘。

每台虚拟机添加3块20G的磁盘。

[Top](#)

## 步骤二：部署ceph集群

1) 给node1主机安装ceph-deploy，创建工作目录，初始化配置文件。

01. [root@node1 ~]# yum -y install ceph-deploy
02. [root@node1 ~]# mkdir ceph-cluster
03. [root@node1 ~]# cd ceph-cluster
04. [root@node1 ceph-cluster]# ceph-deploy new node1 node2 node3

## 2) 给所有ceph节点安装ceph相关软件包

01. [root@node1 ceph-cluster]# for i in node1 node2 node3
02. do
03. ssh \$i "yum -y install ceph-mon ceph-osd ceph-mds"
04. done
- 05.
06. [root@node1 ceph-cluster]# ceph -s #查看结果
07. cluster 9f3e04b8-7dbb-43da-abe6-b9e3f5e46d2e
08. health HEALTH\_ERR
09. monmap e2: 3 mons at
10. {node1=192.168.2.41:6789/0,node2=192.168.2.42:6789/0,node3=192.168.2.43
- 11.

## 3) 准备磁盘分区，创建journal盘，并永久修改设备权限。

01. [root@node1 ceph-cluster]# for i in node1 node2 node3
02. do
03. ssh \$i "parted /dev/vdb mklabel gpt"
04. ssh \$i "parted /dev/vdb mkpart primary 1 50%"
05. ssh \$i "parted /dev/vdb mkpart primary 50% 100%"
06. done

提示：下面的步骤在所有主机都需要操作（node1，node2，node3）

01. #临时修改权限：
02. [root@node1 ceph-cluster]# chown ceph.ceph /dev/vdb1
03. [root@node1 ceph-cluster]# chown ceph.ceph /dev/vdb2
04. #永久修改权限：
05. [root@node1 ceph-cluster]# vim /etc/udev/rules.d/70-vdb.rules
06. ENV{DEVNAME}==" /dev/vdb1",OWNER="ceph",GROUP="ceph"

[Top](#)

07. ENV{DEVNAME}="/dev/vdb2",OWNER="ceph",GROUP="ceph"

4) 使用ceph-deploy工具初始化数据磁盘（仅node1操作）。

```
01. [root@node1 ceph-cluster]# ceph-deploy disk zap node1:vdc node1:vdd
02. [root@node1 ceph-cluster]# ceph-deploy disk zap node2:vdc node2:vdd
03. [root@node1 ceph-cluster]# ceph-deploy disk zap node3:vdc node3:vdd
```

5) 初始化OSD集群。

```
01. [root@node1 ceph-cluster]# ceph-deploy osd create \
02. node1:vdc:/dev/vdb1 node1:vdd:/dev/vdb2
03. //创建osd存储设备，vdc为集群提供存储空间，vdb1提供JOURNAL缓存，
04. //一个存储设备对应一个缓存设备，缓存需要SSD，不需要很大
05. [root@node1 ceph-cluster]# ceph-deploy osd create \
06. node2:vdc:/dev/vdb1 node2:vdd:/dev/vdb2
07. [root@node1 ceph-cluster]# ceph-deploy osd create \
08. node3:vdc:/dev/vdb1 node3:vdd:/dev/vdb2
09.
10. [root@node1 ceph-cluster]# ceph -s #查看集群状态
```

### 步骤三：部署ceph文件系统

1) 启动mds服务

```
01. [root@node1 ceph-cluster]# ceph-deploy mds create node3
```

2) 创建存储池（文件系统由inode和block组成）

```
01. [root@node1 ceph-cluster]# ceph osd pool create cephfs_data 128
02. [root@node1 ceph-cluster]# ceph osd pool create cephfs_metadata 128
03. [root@node1 ceph-cluster]# ceph osd lspools
04. 0 rbd,1 cephfs_data,2 cephfs_metadata
```

3) 创建文件系统

[Top](#)



01. [root@node1 ceph-cluster]# ceph fs new myfs1 cephfs\_metadata cephfs\_data
02. [root@node1 ceph-cluster]# ceph fs ls
03. name: myfs1, metadata pool: cephfs\_metadata, data pools: [cephfs\_data ]

#### 步骤四：迁移网站数据到ceph集群

1) 卸载web1，web2，web3的NFS共享。

暂停服务防止有人实时读写文件。

01. [root@web1 ~]# /usr/local/nginx/sbin/nginx -s stop
02. [root@web2 ~]# /usr/local/nginx/sbin/nginx -s stop
03. [root@web3 ~]# /usr/local/nginx/sbin/nginx -s stop
04. [root@web1 ~]# umount /usr/local/nginx/html
05. [root@web2 ~]# umount /usr/local/nginx/html
06. [root@web3 ~]# umount /usr/local/nginx/html
07. [root@web1 ~]# vim /etc/fstab
08. #192.168.2.31:/web\_share/html /usr/local/nginx/html/ nfs defaults 0 0
09. [root@web2 ~]# vim /etc/fstab
10. #192.168.2.31:/web\_share/html /usr/local/nginx/html/ nfs defaults 0 0
11. [root@web3 ~]# vim /etc/fstab
12. #192.168.2.31:/web\_share/html /usr/local/nginx/html/ nfs defaults 0 0

2) web服务器永久挂载Ceph文件系统（web1、web2、web3都需要操作）。

在任意ceph节点，如node1查看ceph账户与密码。

01. [root@node1 ~]# cat /etc/ceph/ceph.client.admin.keyring
02. [client.admin]
03. key = AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==

/etc/rc.local是开机启动脚本，任何命令放在该文件中都是开机自启。

01. [root@web1 ~]# mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
02. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==
03. [root@web1 ~]# echo 'mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
04. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==' >>> /etc/rc.local
05. [root@web1 ~]# chmod +x /etc/rc.local
- 06.

```

07.
08. [root@web2 ~]# mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
09. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==
10. [root@web2 ~]# echo 'mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
11. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==' >> /etc/rc.
12. [root@web2 ~]# chmod +x /etc/rc.local
13.
14. [root@web3 ~]# mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
15. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==
16. [root@web3 ~]# echo 'mount -t ceph 192.168.2.41:6789:/ /usr/local/nginx/html/ \
17. -o name=admin,secret=AQAOKtlcRGz5JxAA/KOAD/uNuLI1RqPsNGC7zg==' >> /etc/rc.
18. [root@web3 ~]# chmod +x /etc/rc.local

```

### 3) 迁移NFS服务器中的数据到Ceph存储

登陆NFS服务器备份数据，将备份数据拷贝给web1或web2或web3，tar备份数据时注意使用-f选项保留文件权限。

```

01. [root@nfs ~]# cd /web_share/html/
02. [root@nfs html]# tar -czpf /root/html.tar.gz ./*
03. [root@nfs html]# scp /root/html.tar.gz 192.168.2.11:/usr/local/nginx/html/

```

### 登陆web1将数据恢复到Ceph共享目录

```

01. [root@web1 html]# tar -xf html.tar.gz
02. [root@web1 html]# rm -rf html.tar.gz

```

### 3) 恢复web服务

```

01. [root@web1 ~]# /usr/local/nginx/sbin/nginx
02. [root@web2 ~]# /usr/local/nginx/sbin/nginx
03. [root@web3 ~]# /usr/local/nginx/sbin/nginx

```

[Top](#)