

Linux搭建nfs共享服务

原创 心雨先生 2019-04-15 10:59:27 6215 ☆ 收藏 5

展开

NFS(network file system)网络文件系统，类似Windows中的文件夹共享，如下有三台机器A, B, C，它们需要访问同一个目录，目录中都是图片，传统的做法是把这些图片分别放到A, B, C。但是使用NFS只需要放到A上，然后A共享给B和C即可。访问的时候，B和C是

通过网络的方式去访问A上的那个目录。

server机 (A) : 192.168.227.3

Client机 (B, C) : 192.168.227.4 192.168.227.5

一、在A机上安装 NFS 服务器所需的软件包：

```
yum -y install nfs-utils
```

(实际上需要安装两个包nfs-utils和rpcbind, 不过当使用yum安装nfs-utils时会把rpcbind一起安装上)

二、编辑exports文件，添加从机

```
vim /etc/exports
```

```
/opt/nfs 192.168.227.4(rw,sync,fsid=0) 192.168.227.5(rw,sync,fsid=0)
```

配置说明：

这一行分为三个部分：

第一部分：/opt/nfs，这个是本地要共享出去的目录。

第二部分：192.168.227.0/24，允许访问的主机，可以是一个IP：192.168.227.4，也可以是一个IP段：192.168.227.0/24

第三部分：括号中部分。

- rw表示可读写，ro只读；
- sync：同步模式，内存中数据时时写入磁盘；async：不同步，把内存中数据定期写入磁盘中；
- no_root_squash：加上这个选项后，root用户就会对共享的目录拥有至高的权限控制，就像是对本机的目录操作一样。不安全，不建议使用；root_squash：和上面的选项对应，root用户对共享目录的权限不高，只有普通用户的权限，即限制了root；all_squash：不管使用NFS的用户是谁，他的身份都会被限定成为一个指定的普通用户身份；
- anonuid/anongid：要和root_squash 以及all_squash一同使用，用于指定使用NFS的用户限定后的uid和gid，前提是本机的/etc/passwd中存在这个uid和gid。
- fsid=0表示将/opt/nfs整个目录包装成根目录

这个配置文件也可以这样写：

```
/opt/test/ 192.168.227.0/24(rw,no_root_squash,no_all_squash,sync,anonuid=501,anongid=501)
```

三、启动A机上nfs服务

先为rpcbind和nfs做开机启动：

Centos 6X：

```
chkconfig --list rpcbind
```

```
chkconfig --list nfs
```

chkconfig nfs on

Centos 7X:

systemctl enable rpcbind.service

systemctl enable nfs-server.service

然后分别启动rpcbind和nfs服务:

Centos 7X:

systemctl start rpcbind.service

systemctl start nfs.service

Centos 6X:

service rpcbind start

service nfs start

启动之前记得创建你得共享文件

确认NFS服务器启动成功:

rpcinfo -p

通过查看service列中是否有nfs服务来确认NFS是否启动。

showmount -e 192.168.227.3

查看可挂载目录及可连接的IP

```
[root@node1 opt]# showmount -e 192.168.227.3
Export list for 192.168.227.3:
/opt/nfs 192.168.227.5,192.168.227.4
```

四、关闭A机上的防火墙或者给防火墙配置nfs的通过规则

还是关闭简单啊!

systemctl stop firewalld.service

五、在B, C机上配置clinet端

1、安装nfs, 并启动服务。

yum install -y nfs-utils

systemctl enable rpcbind.service

systemctl start rpcbind.service

客户端不需要启动nfs服务, 只需要启动rpcbind服务。

2、检查 NFS 服务器端是否有目录共享

showmount -e 192.168.227.3

3、使用 mount 挂载A服务器端的目录/opt/nfs到客户端B的目录/opt/nfs下

```
[root@localhost ~]# mkdir /opt/nfs
```

```
[root@localhost ~]# mount -t nfs 192.168.227.3:/opt/nfs/ /opt/nfs/
```

```
[root@localhost ~]# df -h
```

文件系统	容量	已用	可用	已用%	挂载点
/dev/mapper/centos-root	11G	1.3G	9.1G	13%	/
devtmpfs	911M	0	911M	0%	/dev
tmpfs	921M	0	921M	0%	/dev/shm
tmpfs	921M	8.5M	912M	1%	/run
tmpfs	921M	0	921M	0%	/sys/fs/cgroup
/dev/sda1	497M	170M	328M	35%	/boot
tmpfs	185M	0	185M	0%	/run/user/0
192.168.227.3:/opt/nfs	11G	1.3G	9.1G	13%	/opt/nfs

```
[root@node2 ~]# mount -t nfs 192.168.227.3:/opt/nfs/ /opt/nfs/
[root@node2 ~]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/mapper/vg_node1-lv_root
                           28G       5.2G   21G   21% /
tmpfs                      491M        0  491M    0% /dev/shm
/dev/sda1                  477M       28M  425M    7% /boot
192.168.227.3:/opt/nfs/
                           28G       7.5G   19G   29% /opt/nfs
[root@node2 ~]#
```

4、挂载完成，可以正常访问本机下的/opt/nfs，如果在服务端A在共享目录/opt/nfs中写入文件，B、C机上可以看到，但是不能在这个目录中写入文件。

六、在服务器端A再建立一个共享目录，并且可以允许A、B、C写入共享目录

1、在B、C机上取得root用户ID号

```
[root@localhost home]# id root
```

uid=0(root) gid=0(root) 组=0(root)

2、在A服务器上再建立一个共享目录

```
mkdir /opt/nfs1
```

```
# vim /etc/exports
```

```
/opt/nfs 192.168.227.4(rw,sync,fsid=0) 192.168.227.5(rw,sync,fsid=0)
```

```
/opt/nfs1 192.168.227.0/24(rw,sync,all_squash,anonuid=0,anongid=0)
```

加入第二行，anonuid=0,anongid=0即为root用户id。

3、让修改过的配置文件生效

```
exportfs -arv
```

使用exportfs命令，当改变/etc/exports配置文件后，不用重启nfs服务直接用这个exportfs即可，它的常用选项为[-aruv]。

-a : 全部挂载或者卸载;

-r : 重新挂载;

-u : 卸载某一个目录;

-v : 显示共享的目录;

4、查看新的可挂载目录及可连接的IP

```
showmount -e 192.168.227.3
```

5、在B、C client端新挂载一个目录

```
showmount -e 192.168.227.3 #查看新的挂载共享目录是否有了。
```

```
mkdir nfs1
```

```
mount -t nfs 192.168.227.3:/opt/nfs1/ /opt/nfs1/
```

```
ll / >/opt/nfs1/ll.txt #测试向新的共享目录中可以写入文件了。
```

(卸载挂载: umount /home/nfs1/)

6、想在客户机B、C上实现开机挂载,则需要编辑/etc/fstab:

```
vim /etc/fstab
```

加入以下内容:

```
192.168.227.3:/opt/nfs      /opt/nfs  nfs  nolock  0 0
```

```
192.168.227.3:/opt/nfs1    /opt/nfs1 nfs  nolock  0 0
```

保存后, 重新挂载

```
mount -a
```