01-24-2021
https://youtu.be/t8z-na-ZFbI

NFS – NETWORK FILE SYSTEM

Network File System – it is a client/server application that lest a computer user view and store and update files on remote system as though they were on the user's own computer. The NFS protocol is one of the several distributed files system standards for NAS – Network Attached Storage

Clone a host and name it NFS - Server

Hostname

nfs01.zmpt.com

Install packages

```
[root@nfs01 ~]# yum install nfs-utils –y
[root@nfs01 ~]# systemctl start nfs-server
[root@nfs01 ~]# systemctl enable nfs-server
Created symlink from /etc/systemd/system/multi-user.target.wants/nfs-server.service to
/usr/lib/systemd/system/nfs-server.service.
```

Create share

```
[root@nfs01 ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda               8:0    0   16G  0 disk
├─sda1            8:1    0    1G  0 part /boot
└─sda2            8:2    0   15G  0 part
  ├─centos-root   253:0    0 13.4G  0 lvm  /
  └─centos-swap   253:1    0  1.6G  0 lvm  [SWAP]
sdb               8:16   0   16G  0 disk
sdc               8:32   0    8G  0 disk
├─zmpt1-Accounting 253:2    0    4G  0 lvm  /accounting
└─zmpt1-HR        253:4    0    2G  0 lvm  /hr
sdd               8:48   0    8G  0 disk
└─zmpt1-Finance   253:3    0    6G  0 lvm  /finance
sde               8:64   0    8G  0 disk
└─zmpt1-Recruiting 253:5    0    4G  0 lvm
sr0              11:0    1 1024M  0 rom
```

```
[root@nfs01 ~]# pvcreate /dev/sdb
  Physical volume "/dev/sdb" successfully created.

[root@nfs01 ~]# vgcreate SHARE-01 /dev/sdb
  Volume group "SHARE-01" successfully created

[root@nfs01 ~]# lvcreate -n NFS_SHARE -L 15G SHARE-01
  Logical volume "NFS_SHARE" created.

[root@nfs01 ~]# mkfs.xfs /dev/mapper/SHARE--01-NFS_SHARE

[root@nfs01 ~]# lsblk

sdb                 8:16   0   16G  0 disk
└─SHARE--01-NFS_SHARE 253:6    0   15G  0 lvm
```

Mount to the directory

```
[root@nfs01 ~]# mount /dev/mapper/SHARE--01-NFS_SHARE /SHARED/
```

Make /etc/fstab entries

```
/dev/mapper/SHARE--01-NFS_SHARE /SHARED xfs     defaults        0 0
```

Vi /etc/default/nfs-share – create file if not present

Make entry as shown

```
NEED_IDMAPD=YES
```

vi /etc/default/idmapd.conf

```
#type exactly as shown

nfs01.zmpt.com
```

Change the permissions for the shared folder

```
[root@nfs01 ~]# chmod 777 /SHARED/
[root@nfs01 ~]# ls -ld /SHARED/
drwxrwxrwx. 2 root root 6 Jan 24 14:26 /SHARED/
```

Enter vi /etc/exports

```
#enter the list of servers or client to grant access
/SHARED 192.168.56.117(rw,async)
/SHARED 192.168.56.120(rw,async)
```

Test

```
[root@nfs01 ~]# showmount -e
Export list for nfs01.zmpt.com:
[root@nfs01 ~]#
[root@nfs01 ~]# exportfs -a
[root@nfs01 ~]# exportfs –r

[root@nfs01 ~]# showmount -e
Export list for nfs01.zmpt.com:
/SHARED 192.168.56.120,192.168.56.117
```

Open NFS port in firewall

```
[root@nfs01 ~]# firewall-cmd --permanent --add-port=2049/tcp
success
[root@nfs01 ~]# firewall-cmd --list-ports

[root@nfs01 ~]# firewall-cmd --reload
success
[root@nfs01 ~]# firewall-cmd --list-ports
2049/tcp

[root@nfs01 ~]# rpcinfo -p | grep nfs
   100003    3   tcp   2049  nfs
   100003    4   tcp   2049  nfs
   100227    3   tcp   2049  nfs_acl
   100003    3   udp   2049  nfs
   100003    4   udp   2049  nfs
   100227    3   udp   2049  nfs_acl
```

NFS Server is done configuration

Enter these configuration on any other server

You can use anisble to make entires

Install the needed package

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "yum install nfs-utils -y"

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "systemctl start nfs-server"

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "systemctl enable nfs-server"

Make the directory

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "mkdir /NETWORK_FOLDER"

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "chmod -R 777 /NETWORK_FOLDER

Mount NFS

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "mount -t nfs 192.168.56.126:/SHARED /NETWORK_FOLDER"

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "df -h"

Make entry for /etc/fstab

[root@ansiblemaster ~]# ansible all -i nfs-clients -m shell -a "echo '192.168.56.126:/SHARED /NETWORK_FOLDER nfs defaults 0 0' >> /etc/fstab"

What is the difference between hard mount and soft mount?

The NFS mount can be a “soft mount” or a “hard mount” – mount option define the way how NFS client should handle NFS crash/failure

Soft mount: suppose you have mounted the NFS by using ‘soft mount’ when a application request a file from NFS server, NFS server Deamon will try to retrieve the data from the NFS server. If it doesn’t get any response from NFS server due to failure or crash on the NFS server. Then NFS client report an error to the process on the client machine requesting the file access.

- Advantage: fast response, it doesn’t wait for the NFS server to respond.
- Disadvantage is this method is data corruption or data loss – so this option is not Recommended

Hard Mount: if you have mounted the NFS by ‘hard mount’, during crash it will repeatedly try to

connect to the NFS server. Once the server is back online the application will continue to execute undisturbed where it was during the crash. You can add mount option ‘intr’ which allows NFS request to interrupt if the server goes down or cannot be accessible.