Creating a NFS File Server

A PROJECT REPORT

Jessenth Ebenezer - 19BCE1462

Nathamayil.N - 19BCE1468

Rakul.S.K - 19BCE1613

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Science and Engineering

Vellore Institute of Technology

Vandalur - Kelambakkam Road, Chennai - 600 127

June 2021

Abstract

Unshared joy is an unlighted candle, says an old Spanish proverb. This applies even to the modern day world where one could find almost everything with the help of a search engine. But a few years ago, when this was still under construction and development, sharing wasn't an easy task. The Network File System(NFS) was used in those times as a means of sharing information between Linux/Unix Operating Systems. And we, as students who are still getting to know the basics of Operating Systems, we decided to create and explore how Network File Systems worked.

Introduction

NFS, or Network File System, was designed in 1984 by Sun Microsystems. This distributed file system protocol allows a user on a client computer to access files over a network in the same way they would access a local storage file. Because it is an open standard, anyone can implement the protocol.
NFS started in-system as an experiment but the second version was publicly released after the initial success.

To access data stored on another machine (i.e. a server) the server would implement NFS daemon processes to make data available to clients. The server administrator determines what to make available and ensures it can recognize validated clients.From the client's side, the machine requests access to exported data, typically by issuing a mount command. If successful, the client machine can then view and interact with the file systems within the decided parameters.

Project Description

The project involves creating a NFS Server and Client using CentOS and establishing a connection between them, and sharing the files in that system and verifying it.

Tools Used

2 CentOS machines used in Virtual Box

Sample Code

Setting up an internal network to mimic a DHCP connection

VBoxManage dhcpserver add --netname intnet --ip 10.0.1.1 --netmask 255.255.255.0 --lowerip 10.0.1.2 --upperip 10.0.1.200 --enable

On Command Prompt opened in Virtualbox Directory (./VBoxManage on Powershell)

On both machines, we set up the Internet Connection.

nmcli d
enps08
enps03

enps08 will be connected, but enps03 will be disconnected

vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

We open this file and change onboot to yes

systemctl restart network

Now we check ip using ip addr show

CentOS-1: 10.0.1.2
CentOS-2: 10.0.1.3

On CentOS-1: ping 10.0.1.3
On CentOS-2: ping 10.0.1.2

We ping each other to make sure everything works correctly before we work on the NFS Server and client on CentOS 1 and 2 respectively

Setting up NFS Server: Machine #1

On CentOS-1, we'll install the utilities and libraries for an NFS server and start it using rpcbind

```
yum install nfs-utils nfs-utils-lib -y
systemctl start rpcbind nfs-server
systemctl enable rpcbind nfs-server
```

Now, we create a new empty folder that will be our shared folder.

```
mkdir /nfs
vi /etc/exports
```

We write the name of the shared folder and the IP addresses that we want the shared folder to be shared inside this file.

Shell

```
/nfs 10.0.1.3(rw,sync,no_root_squash,no_subtree_check)
```

```
/nfs
10.0.1.3(rw,sync,no_root_squash,no_subtr
ee_check)
```

```
exportfs -a
```

We have to change the firewall to allow NFS and complimenting services.

```
firewall-cmd --permanent --zone=public --add-service=nfs
firewall-cmd --permanent --zone=public --add-service=mountd
firewall-cmd --permanent --zone=public --add-service=rpc-bind
firewall-cmd --reload
systemctl restart nfs
```

Setting up NFS Client: Machine #2

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

And we make the NFS folder where we mount the server

```
mkdir -p /nfs
```

Check if we can access the server properly

```
showmount -e 10.0.1.2
rpcinfo -p 10.0.1.2
mount 10.0.1.2:/nfs /nfs
```

With df -h, we see that 10.0.1.2:/nfs mount has been created at the bottom.

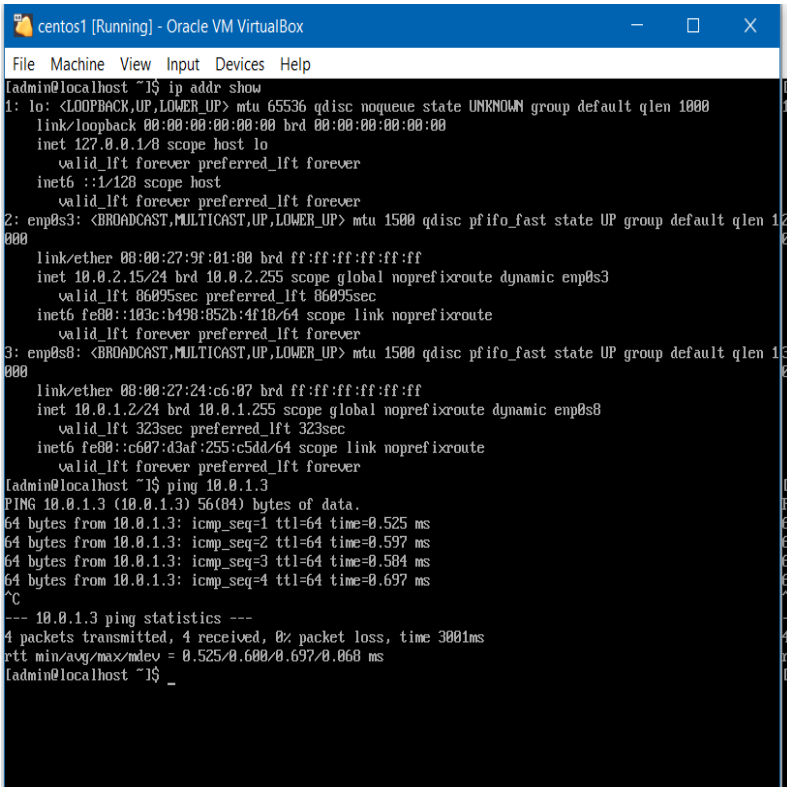
```
df -h
```

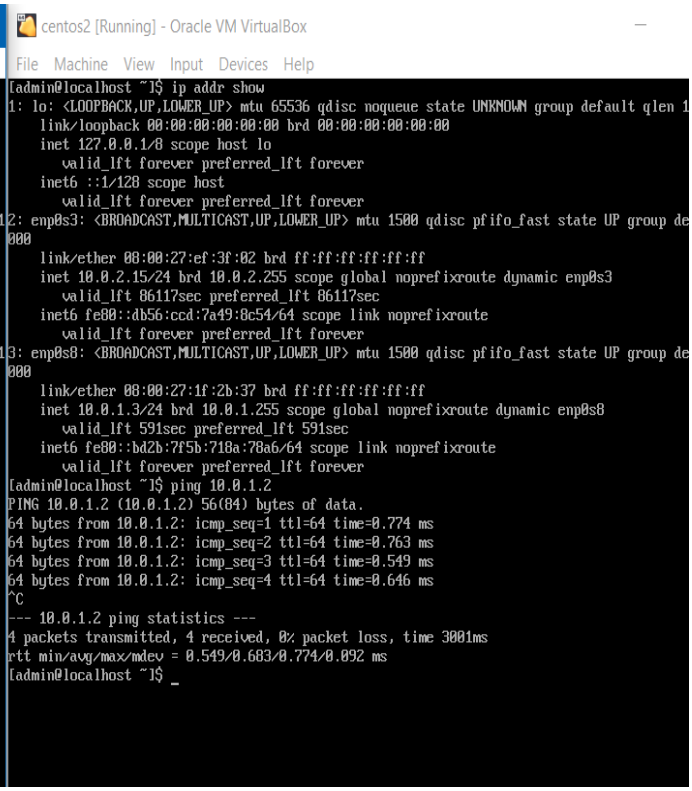
Now, we test that the shared folder actually works.

```
cd /nfs
```

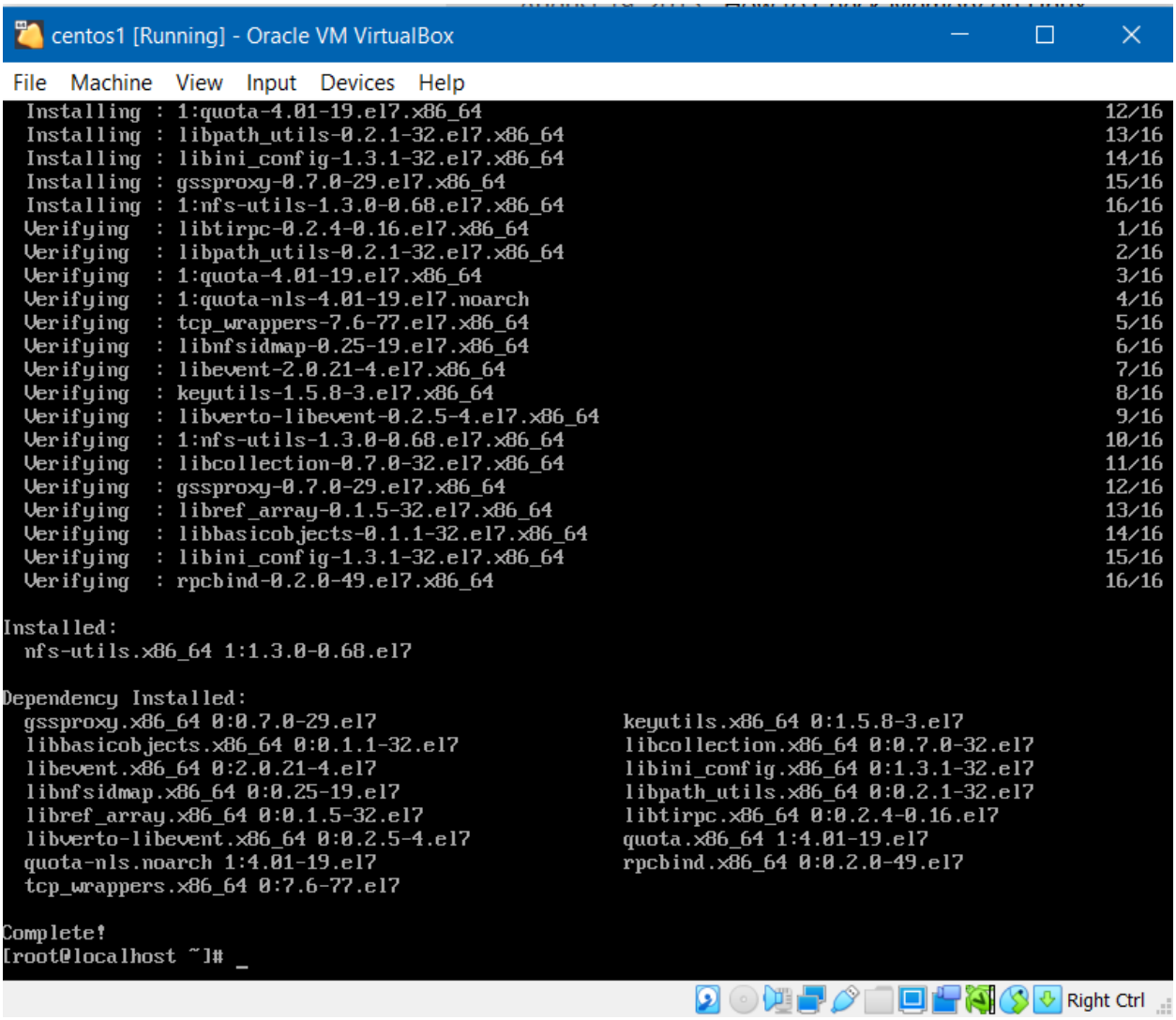
Screenshots have the rest of the information.

SCREEN SHOTS:



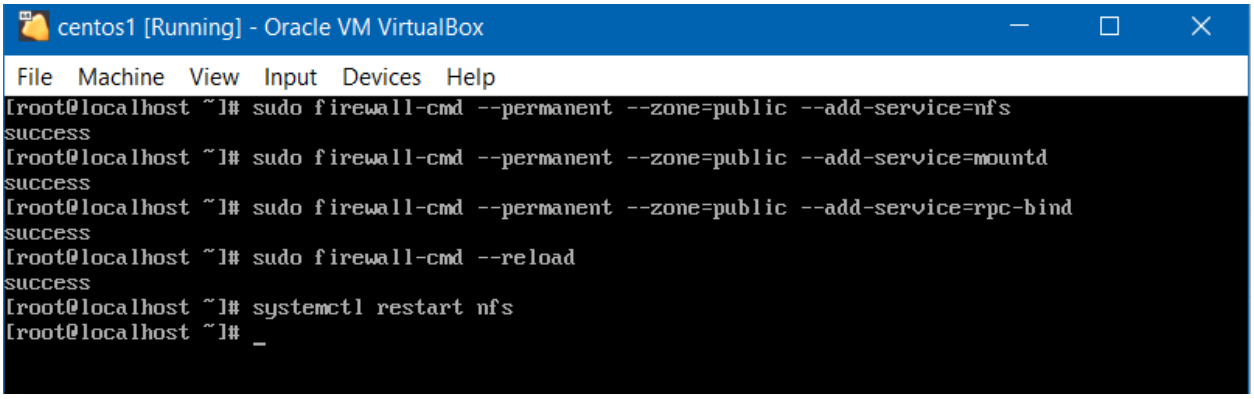


Internet connection within the internal network works perfectly, allowing us to ping both servers

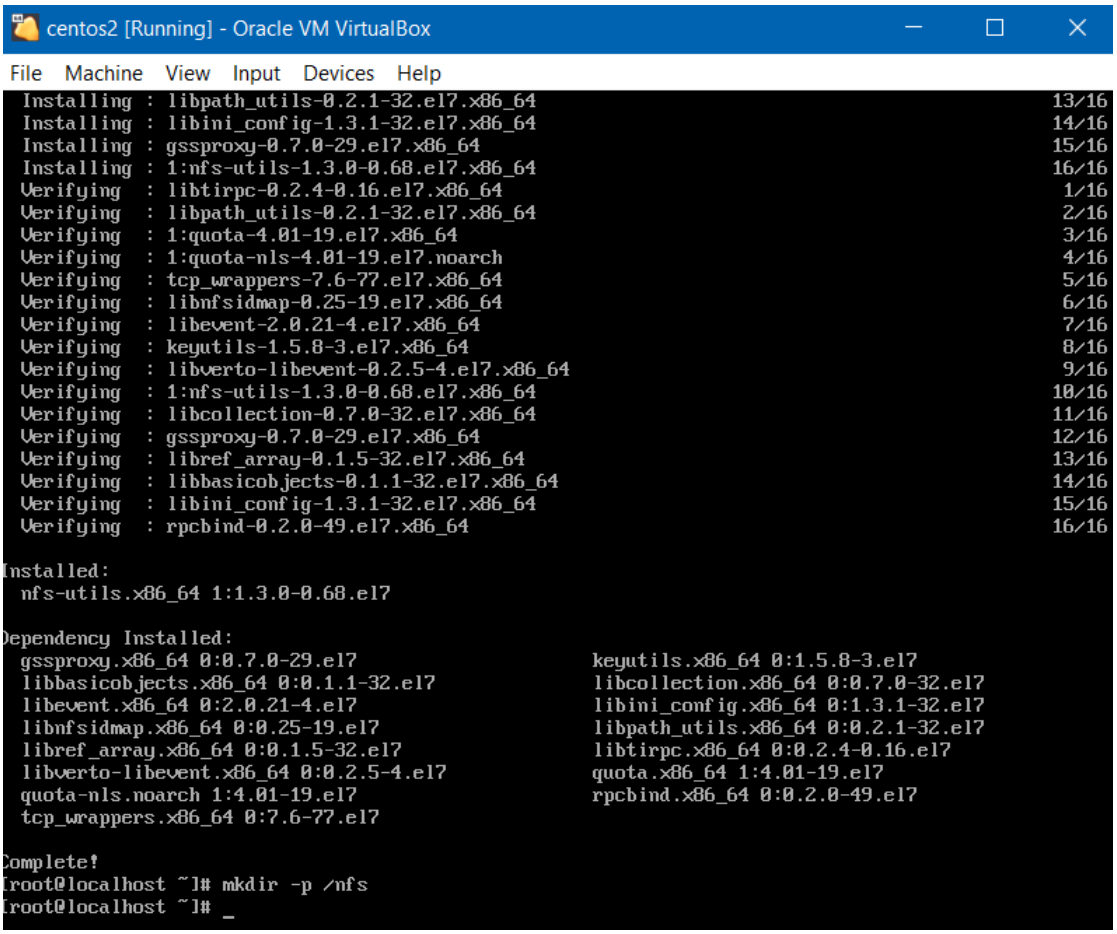


Installing the NFS utilities on CentOS 1

Enabling the server using a remote procedure call



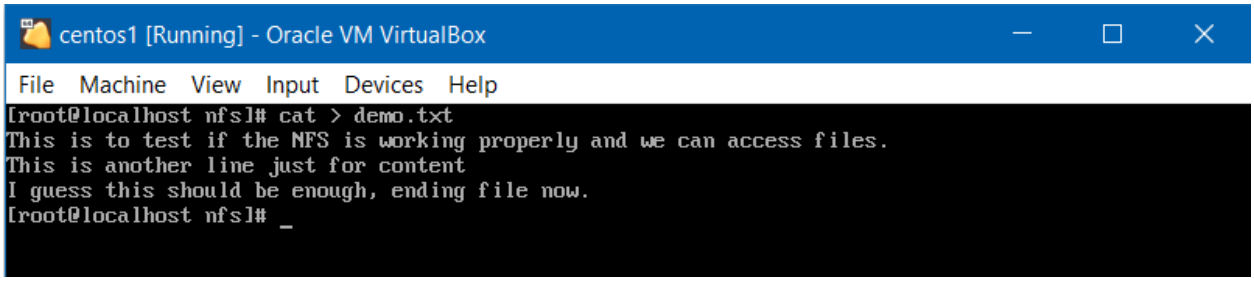
Providing access to the server through the System Firewall



Setting up the client machine and making the connected directory

```
[root@localhost ~]# mount 10.0.1.2:/nfs /nfs
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
devtmpfs                 232M     0  232M   0% /dev
tmpfs                    244M     0  244M   0% /dev/shm
tmpfs                    244M  4.6M  239M   2% /run
tmpfs                    244M     0  244M   0% /sys/fs/cgroup
/dev/mapper/centos-root  6.2G  1.4G  4.9G  22% /
/dev/sda1               1014M  137M  877M  14% /boot
tmpfs                     49M     0   49M   0% /run/user/0
10.0.1.2:/nfs            6.2G  1.4G  4.9G  22% /nfs
[root@localhost ~]# cd /nfs
[root@localhost nfs]#
```

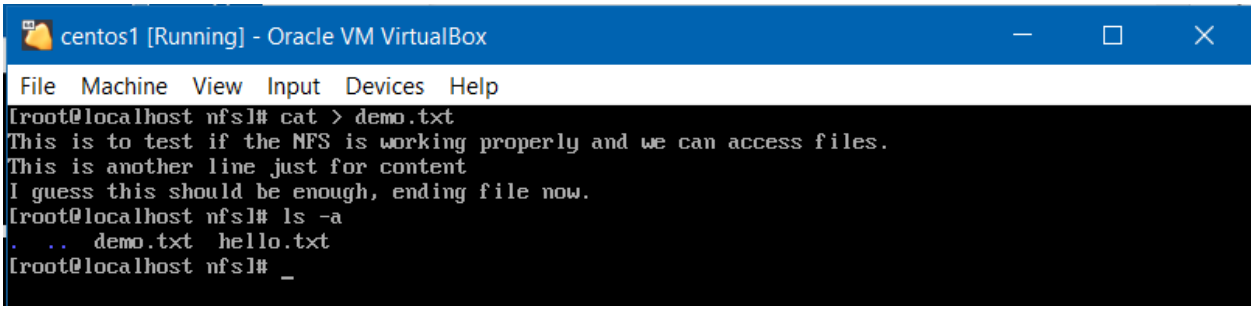
Verifying that the connection is stable



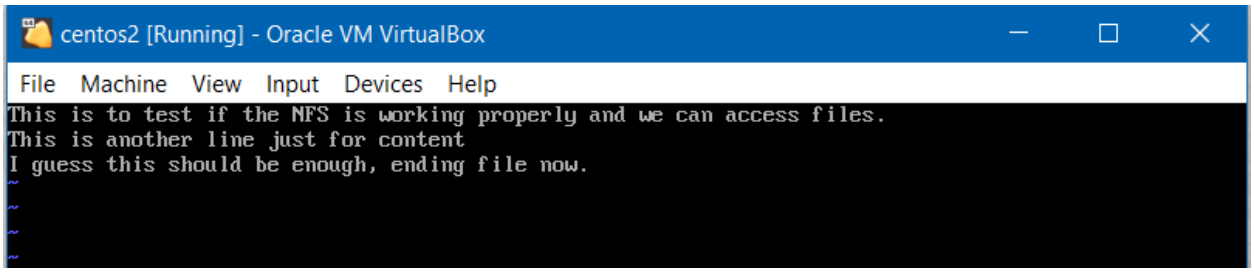
Writing into a demo.txt file using vi in server machine

```
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
devtmpfs                 232M     0  232M   0% /dev
tmpfs                    244M     0  244M   0% /dev/shm
tmpfs                    244M  4.6M  239M   2% /run
tmpfs                    244M     0  244M   0% /sys/fs/cgroup
/dev/mapper/centos-root  6.2G  1.4G  4.9G  22% /
/dev/sda1               1014M  137M  877M  14% /boot
tmpfs                     49M     0   49M   0% /run/user/0
10.0.1.2:/nfs            6.2G  1.4G  4.9G  22% /nfs
[root@localhost ~]# cd /nfs
[root@localhost nfs]# touch demo.txt
[root@localhost nfs]# touch hello.txt
[root@localhost nfs]# _
```

Creating another file on the client machine to verify bidirectional file transmission on the NFS



Verified that the file exists in server machine



Verifying that the edited file exists in client machine

CONCLUSION

The NFS was created successfully and the files could be shared in it between the servers and clients as well.

REFERENCES

[1]https://www.cs.cornell.edu/courses/cs6411/2018sp/papers/nfs.pdf

[2]https://dl.acm.org/doi/abs/10.1145/502034.502052