

File Server (NFS) on Oracle Linux

- Node2 change mac add to separate the VMs = **NFS server**
- Node1 = **NFS client**
- `server` → NFS **host** (exports the shared folder)
- `client` → NFS **client** (mounts and accesses that folder)

Why NFS

- Shared storage between multiple VMs (centralized data).
- Easy file transfer or config sharing.
- Simpler backup management (only on server).

Step 0 — Make sure both can reach each other

On **client**:

```
ping -c3 <server ip>
```

On **server**:

```
ping -c3 <client ip >
```

configure the NFS export on Server and mount it on Node1.

Step 1 — Install NFS packages

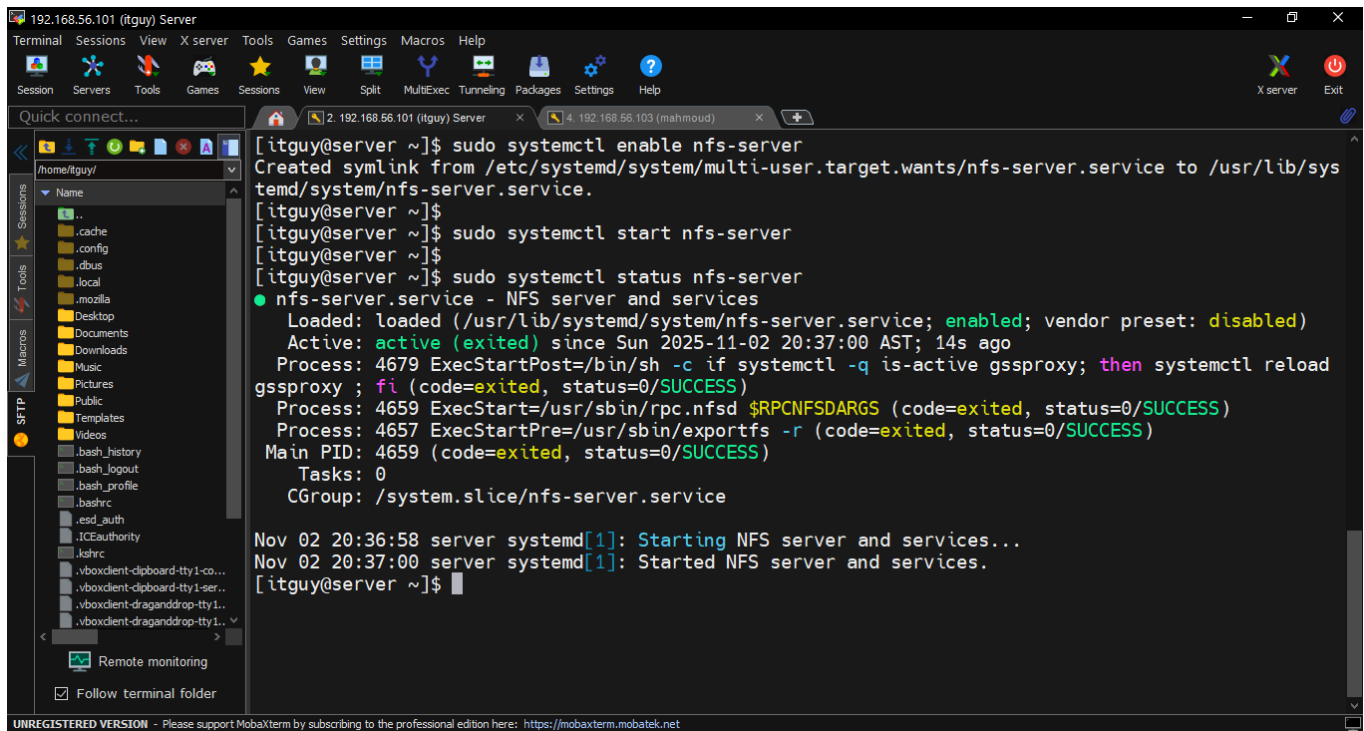
On the **Server VM**

```
sudo dnf install nfs-utils -y
```

On the **Client VM**

```
sudo dnf install nfs-utils -y
```

Step 2 — Enable and start NFS services (on the server) :



The screenshot shows a MobaXterm terminal window with the following commands and output:

```
[itguy@server ~]$ sudo 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.
[itguy@server ~]$ sudo systemctl start nfs-server
[itguy@server ~]$ sudo systemctl status nfs-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; vendor preset: disabled)
   Active: active (exited) since Sun 2025-11-02 20:37:00 AST; 14s ago
     Process: 4679 ExecStartPost=/bin/sh -c if systemctl -q is-active gssproxy; then systemctl reload gssproxy ; fi (code=exited, status=0/SUCCESS)
     Process: 4659 ExecStart=/usr/sbin/rpc.nfsd $RPCNFSDARGS (code=exited, status=0/SUCCESS)
     Process: 4657 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUCCESS)
    Main PID: 4659 (code=exited, status=0/SUCCESS)
       Tasks: 0
      CGroup: /system.slice/nfs-server.service

Nov 02 20:36:58 server systemd[1]: Starting NFS server and services...
Nov 02 20:37:00 server systemd[1]: Started NFS server and services.
[itguy@server ~]$
```

Step 3 — Enable and start NFS services (on the server)

```
sudo systemctl enable nfs-server
sudo systemctl start nfs-server
sudo systemctl status nfs-server
```

 It should show `active (running)`.

Step 4 — Create a shared directory

```
sudo mkdir -p /srv/nfs/shared
sudo chown -R nfsnobody:nfsnobody /srv/nfs/shared
sudo chmod 777 /srv/nfs/shared
ls -ld /srv/nfs/share
```

```
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Desktop
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Documents
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Downloads
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Music
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Pictures
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Public
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Templates
drwxr-xr-x. 2 itguy itguy 6 Oct 23 16:03 Videos
[itguy@server ~]$ sudo mkdir -p /srv/nfs/share
[itguy@server ~]$ sudo chown -R nfsnobody:nfsnobody /srv/nfs/share/
[itguy@server ~]$ sudo chmod 777 /srv/nfs/share/
[itguy@server ~]$ cd /srv/nfs
```

You can also put a test file there:

```
echo "Hello from NFS server" | sudo tee /srv/nfs/share/test 101.txt
```

```
[itguy@server nfs]$ ll
total 0
drwxrwxrwx. 2 nfsnobody nfsnobody 6 Nov  2 20:47 share
[itguy@server nfs]$ cd share
[itguy@server share]$ cd
[itguy@server ~]$ echo "Hello from the NFS Server" |sudo tee /srv/nfs/share/test101.txt
[sudo] password for itguy:
Hello from the NFS Server
[itguy@server ~]$
```

Step 5 — Configure NFS exports

Edit `/etc/exports`:

```
sudo nano /etc/exports
```

Add a line like this (replace IP if needed):

```
/srv/nfs/shared *(rw,sync,no_root_squash) #"*" means all clients ip

/srv/nfs_share 192.168.56.0/24(rw,sync,no_root_squash) #Limits access to
machines in that subnet (for example, your host-only network).
```

```
[itguy@server ~]$ cat /etc/exports
/srv/nfs/shared *(rw,sync,no_root_squash)
```

Then apply:

```
sudo exportfs -r
```

```
sudo exportfs -v
```

```
[itguy@server ~]$ sudo nano /etc/exports
[itguy@server ~]$ sudo exportfs -r
[itguy@server ~]$ sudo exportfs -v
/srv/nfs/share <world>(sync,wdelay,hide,no_subtree_check,sec=sys,rw,secure,no_root_squash,no_all_squash)
```

```
sudo showmount -e
```

You should see:

```
[itguy@server ~]$ sudo showmount -e
Export list for server:
/srv/nfs/share *
```

```
Export list for server:
/srv/nfs/share *
```

Step 6 — On the client side

1 Create a mount point:

```
sudo mkdir -p /mnt/nfs_share
```

2 Mount the NFS share:

```
sudo mount -t nfs <server ip> :/srv/nfs/share /mnt/nfs_share
```

3 Verify:

```
df -h | grep nfs
```

You should see the mount.

4 Check files:

```
ls /mnt/nfs_share
```

```
cat /mnt/nfs/test101.txt
```

Step 7 — Make it permanent (optional)

Add this line to `/etc/fstab` on the client:

```
server:/srv/nfs/shared /mnt/nfs nfs defaults 0 0
```

Then test:

```
sudo mount -a
```

Summary

Role	Command Highlights
Server	<code>yum install nfs-utils</code> , <code>mkdir /srv/nfs/shared</code> , <code>configure /etc/exports</code>
Client	<code>yum install nfs-utils</code> , <code>mount server:/srv/nfs/shared /mnt/nfs</code>
Verify	<code>showmount -e</code> , <code>df -h</code> , <code>cat /mnt/nfs/test.txt</code>

<https://github.com/Mahmoudsherif18>