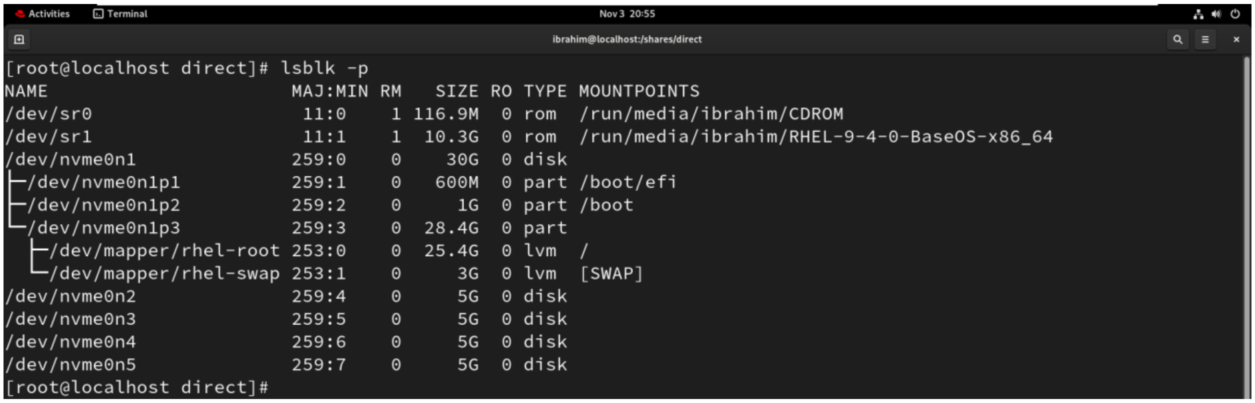
1- Logical Volume Management

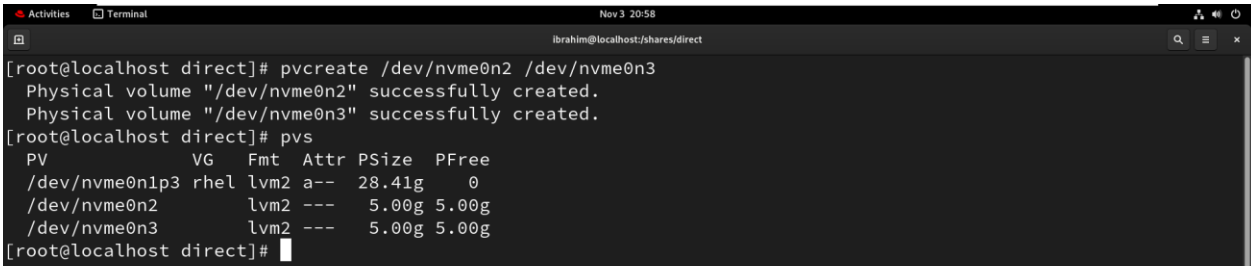
a- Create a volume group and logical volume. Format it, mount it, and extend it by 1GB.

>> At first you must add a volumes from the vmware



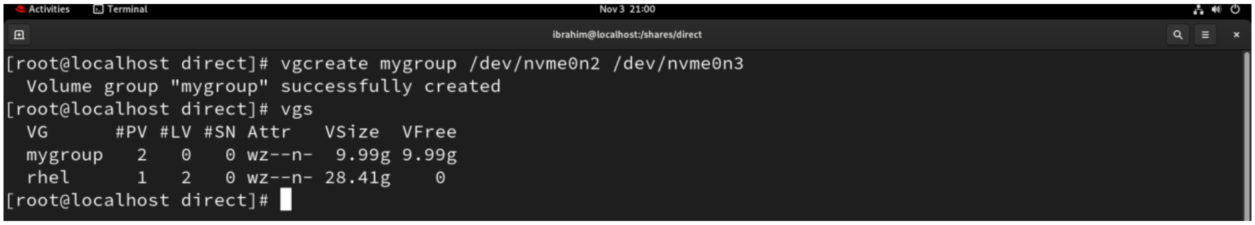
```
[root@localhost direct]# lsblk -p
NAME                         MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
/dev/sr0                      11:0    1 116.9M  0 rom  /run/media/ibrahim/CDROM
/dev/sr1                      11:1    1  10.3G  0 rom  /run/media/ibrahim/RHEL-9-4-0-BaseOS-x86_64
/dev/nvme0n1                 259:0    0    30G  0 disk
├─/dev/nvme0n1p1             259:1    0   600M  0 part /boot/efi
├─/dev/nvme0n1p2             259:2    0     1G  0 part /boot
└─/dev/nvme0n1p3             259:3    0  28.4G  0 part
  ├─/dev/mapper/rhel-root 253:0    0  25.4G  0 lvm  /
  └─/dev/mapper/rhel-swap 253:1    0     3G  0 lvm  [SWAP]
/dev/nvme0n2                 259:4    0     5G  0 disk
/dev/nvme0n3                 259:5    0     5G  0 disk
/dev/nvme0n4                 259:6    0     5G  0 disk
/dev/nvme0n5                 259:7    0     5G  0 disk
[root@localhost direct]#
```

>>Then you need to label these devices as a pv



```
[root@localhost direct]# pvcreate /dev/nvme0n2 /dev/nvme0n3
  Physical volume "/dev/nvme0n2" successfully created.
  Physical volume "/dev/nvme0n3" successfully created.
[root@localhost direct]# pvs
  PV             VG   Fmt  Attr PSize  PFree
  /dev/nvme0n1p3 rhel lvm2 a--  28.41g    0
  /dev/nvme0n2        lvm2 ---   5.00g 5.00g
  /dev/nvme0n3        lvm2 ---   5.00g 5.00g
[root@localhost direct]#
```

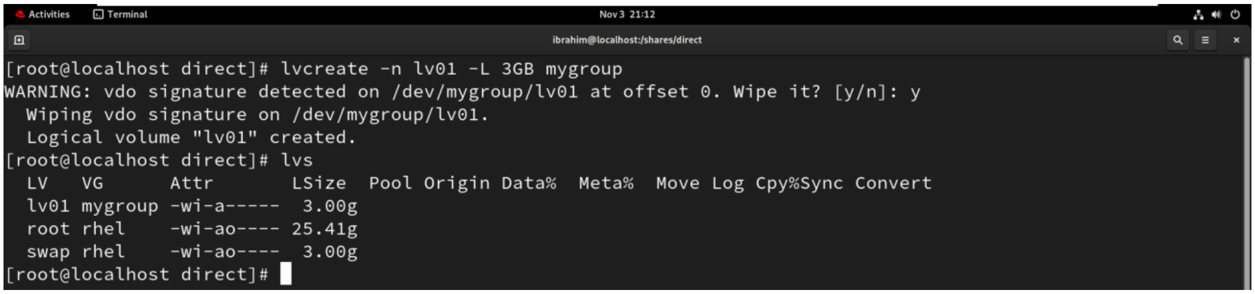
>> Add these devices to a volume group



```
[root@localhost direct]# vgcreate mygroup /dev/nvme0n2 /dev/nvme0n3
  Volume group "mygroup" successfully created
[root@localhost direct]# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  mygroup   2   0   0 wz--n-  9.99g 9.99g
  rhel      1   2   0 wz--n- 28.41g    0
[root@localhost direct]#
```

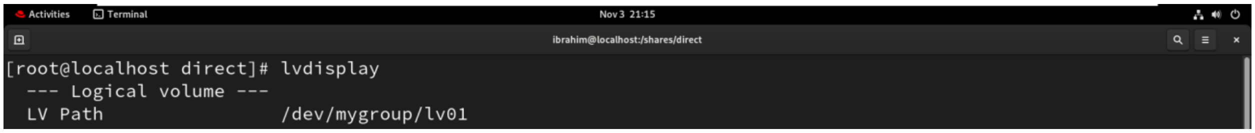
>> Here each device is a 5GB so the group of these devices is about 10GB

>>Create a logical volume from the available free space on the volume group

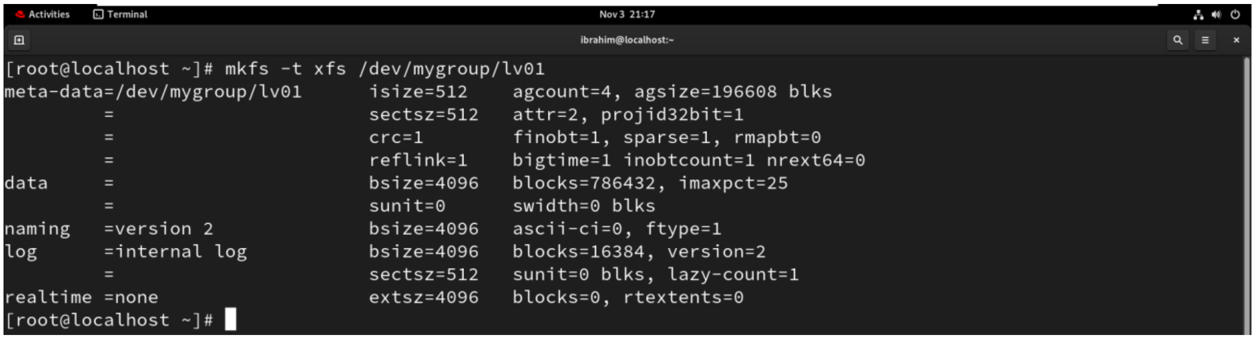


```
[root@localhost direct]# lvcreate -n lv01 -L 3GB mygroup
WARNING: vdo signature detected on /dev/mygroup/lv01 at offset 0. Wipe it? [y/n]: y
  Wiping vdo signature on /dev/mygroup/lv01.
  Logical volume "lv01" created.
[root@localhost direct]# lvs
  LV   VG      Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  lv01 mygroup -wi-a-----  3.00g
  root rhel    -wi-ao---- 25.41g
  swap rhel    -wi-ao----  3.00g
[root@localhost direct]#
```

>> First get the path of the logical volume and Choose a file system (xfs) to be assigned to this logical volume's path

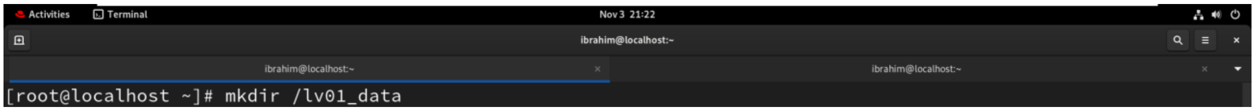


```
[root@localhost direct]# lvdisplay
  --- Logical volume ---
  LV Path                /dev/mygroup/lv01
```

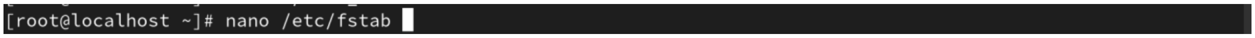


```
[root@localhost ~]# mkfs -t xfs /dev/mygroup/lv01
meta-data=/dev/mygroup/lv01      isize=512    agcount=4, agsize=196608 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
data     =                       bsize=4096   blocks=786432, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@localhost ~]# 
```

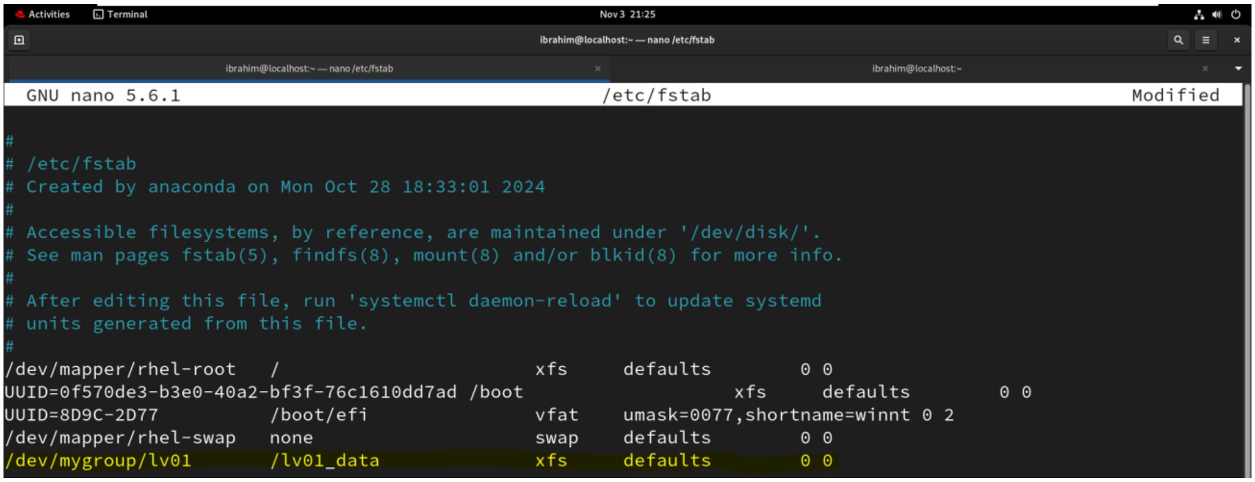
>> Now you can put this file system and its logical volume in the fstab file to mount this logical volume and its file system on a mount point (lv01_data)and make it permanent across system reboots



```
[root@localhost ~]# mkdir /lv01_data
```

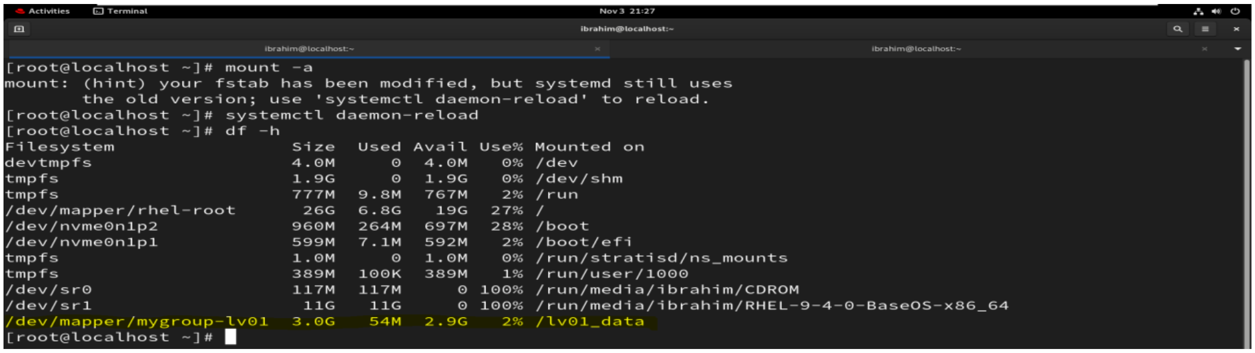


```
[root@localhost ~]# nano /etc/fstab 
```


Activities
Terminal
Nov 3 21:25
ibrahim@localhost:~ — nano /etc/fstab
ibrahim@localhost:~ — nano /etc/fstab
ibrahim@localhost:~

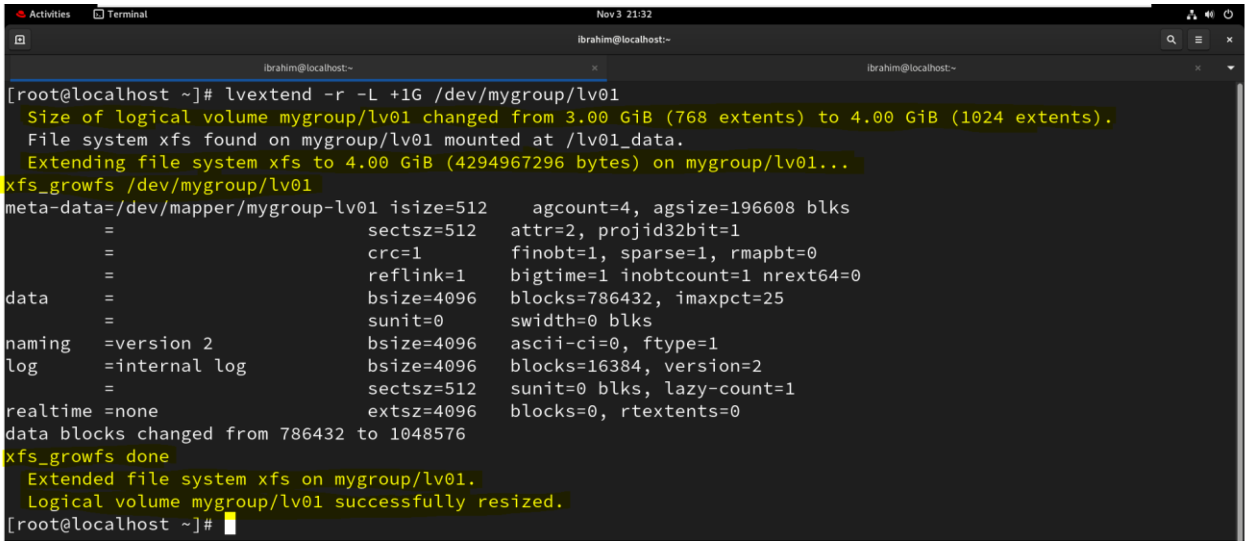
```
  GNU nano 5.6.1                    /etc/fstab                              Modified


#
# /etc/fstab
# Created by anaconda on Mon Oct 28 18:33:01 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rhel-root   /                       xfs     defaults        0 0
UUID=0f570de3-b3e0-40a2-bf3f-76c1610dd7ad /boot                   xfs     defaults        0 0
UUID=8D9C-2D77          /boot/efi               vfat    umask=0077,shortname=winnt 0 2
/dev/mapper/rhel-swap   none                    swap    defaults        0 0
/dev/mygroup/lv01       /lv01_data              xfs     defaults        0 0
```

>> Save and exit

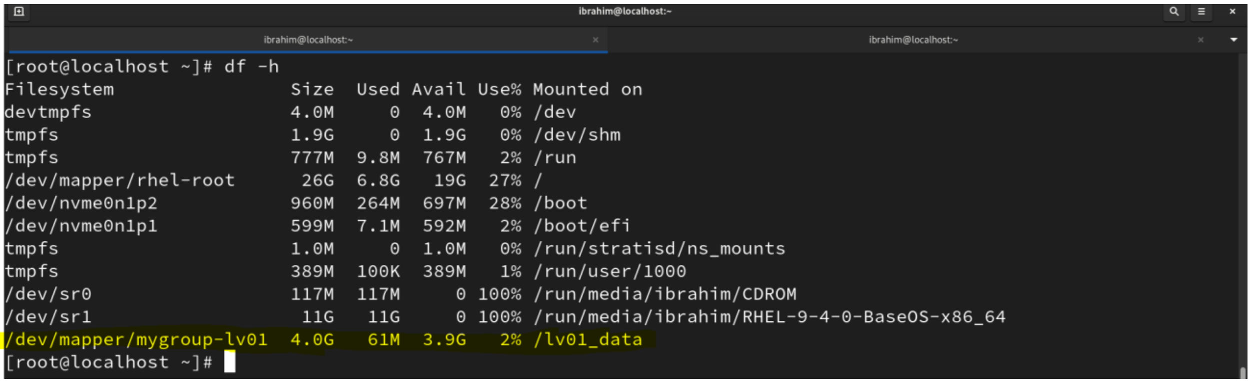


```
[root@localhost ~]# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
[root@localhost ~]# systemctl daemon-reload
[root@localhost ~]# df -h
Filesystem                 Size  Used Avail Use% Mounted on
devtmpfs                   4.0M     0  4.0M   0% /dev
tmpfs                      1.9G     0  1.9G   0% /dev/shm
tmpfs                      777M  9.8M  767M   2% /run
/dev/mapper/rhel-root       26G  6.8G   19G  27% /
/dev/nvme0n1p2             960M  264M  697M  28% /boot
/dev/nvme0n1p1             599M  7.1M  592M   2% /boot/efi
tmpfs                      1.0M     0  1.0M   0% /run/stratisd/ns_mounts
tmpfs                      389M  100K  389M   1% /run/user/1000
/dev/sr0                   117M  117M     0 100% /run/media/ibrahim/CDROM
/dev/sr1                    11G   11G     0 100% /run/media/ibrahim/RHEL-9-4-0-BaseOS-x86_64
/dev/mapper/mygroup-lv01   3.0G   54M  2.9G   2% /lv01_data
[root@localhost ~]# 
```

>>To extend this logical volume by 1GB

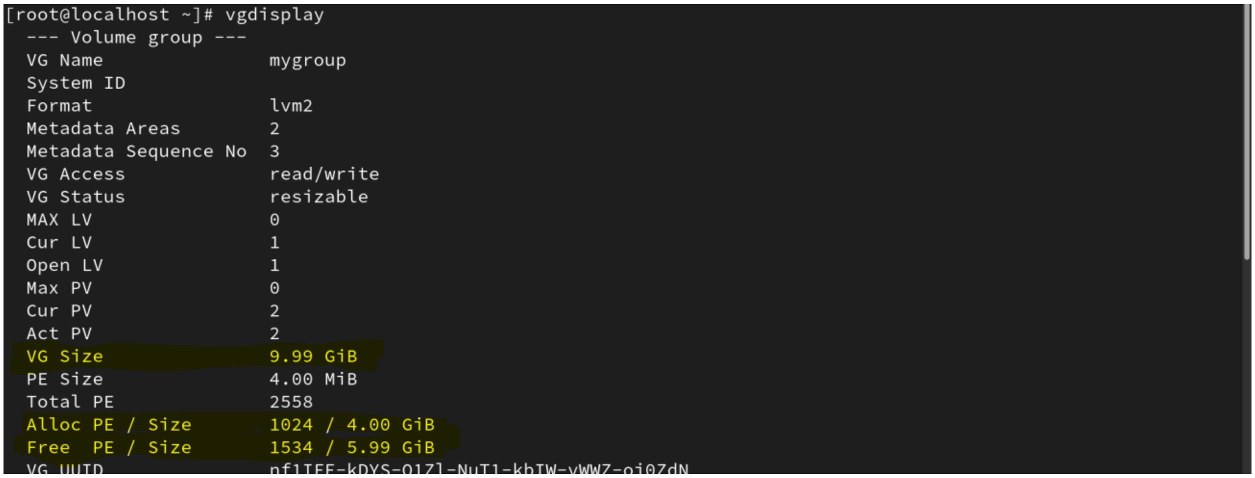


```
[root@localhost ~]# lvextend -r -L +1G /dev/mygroup/lv01
  Size of logical volume mygroup/lv01 changed from 3.00 GiB (768 extents) to 4.00 GiB (1024 extents).
  File system xfs found on mygroup/lv01 mounted at /lv01_data.
  Extending file system xfs to 4.00 GiB (4294967296 bytes) on mygroup/lv01...
xfs_growfs /dev/mygroup/lv01
meta-data=/dev/mapper/mygroup-lv01 isize=512    agcount=4, agsize=196608 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
data     =                       bsize=4096   blocks=786432, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 786432 to 1048576
xfs_growfs done
  Extended file system xfs on mygroup/lv01.
  Logical volume mygroup/lv01 successfully resized.
[root@localhost ~]# 
```



```
[root@localhost ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  4.0M     0  4.0M   0% /dev
tmpfs                     1.9G     0  1.9G   0% /dev/shm
tmpfs                     777M  9.8M  767M   2% /run
/dev/mapper/rhel-root      26G  6.8G   19G  27% /
/dev/nvme0n1p2            960M  264M  697M  28% /boot
/dev/nvme0n1p1            599M  7.1M  592M   2% /boot/efi
tmpfs                     1.0M     0  1.0M   0% /run/stratisd/ns_mounts
tmpfs                     389M  100K  389M   1% /run/user/1000
/dev/sr0                  117M  117M     0 100% /run/media/ibrahim/CDROM
/dev/sr1                   11G   11G     0 100% /run/media/ibrahim/RHEL-9-4-0-BaseOS-x86_64
/dev/mapper/mygroup-lv01  4.0G   61M  3.9G   2% /lv01_data
[root@localhost ~]# 
```

>> Here is the verification about the storage used and the free storage in the vg



```
[root@localhost ~]# vgdisplay
  --- Volume group ---
  VG Name               mygroup
  System ID
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               9.99 GiB
  PE Size               4.00 MiB
  Total PE              2558
  Alloc PE / Size       1024 / 4.00 GiB
  Free  PE / Size       1534 / 5.99 GiB
  VG UUID               pf1IEE-kDYS-Q1Zl-NuT1-kbIW-vWWZ-oj0ZdN
```

2- NFS Setup

a- Install and configure an NFS server and client. Share a directory and verify accessibility from the client machine.

>> In order to achieve this requirement I have a two vms up and running

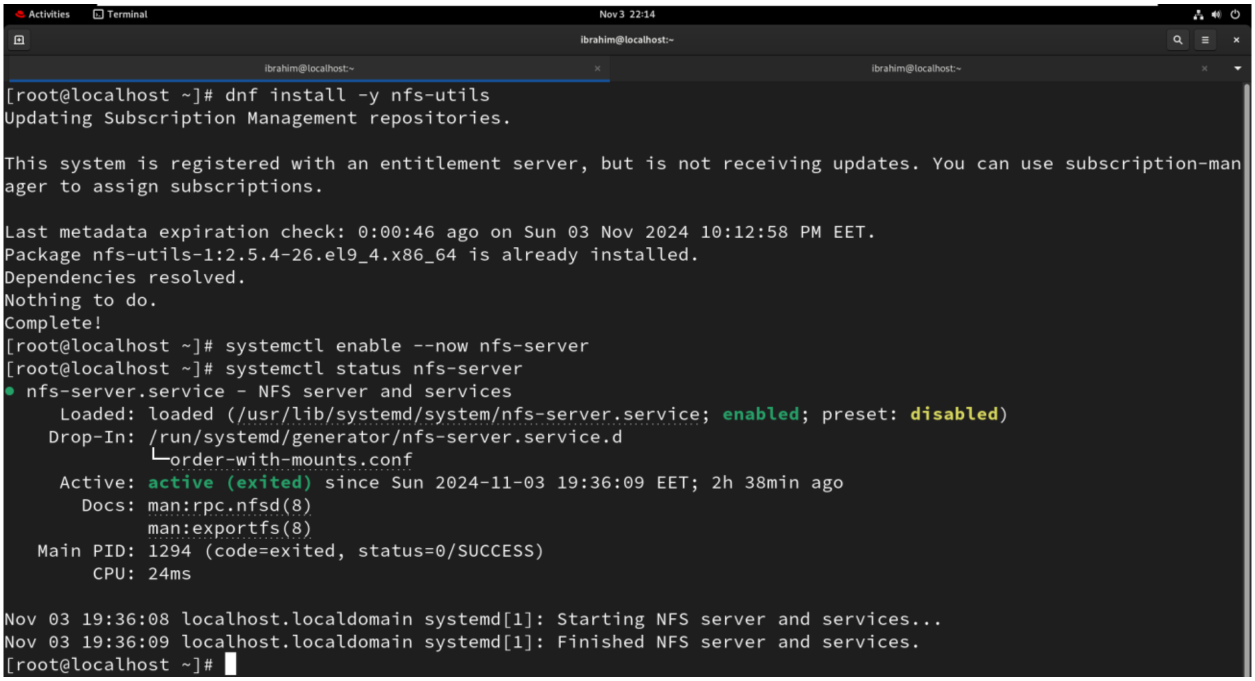
Ubuntu as a NFS client

RHEL 9.4 as NFS server

>> The common step must be executed on the two vm is installing the NFS package.

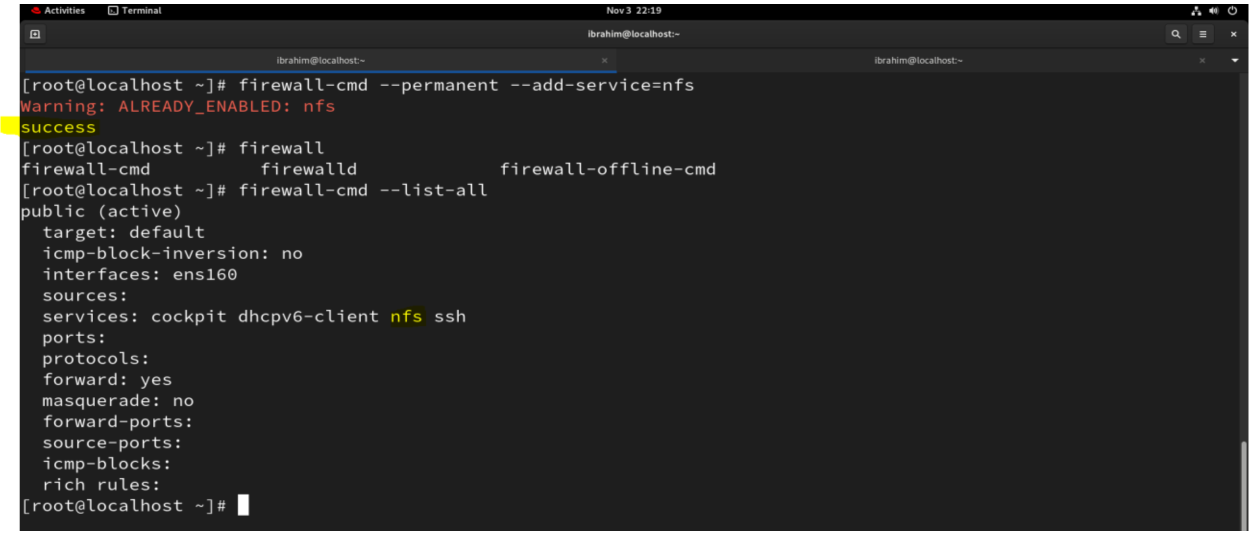
>> First let's setup the server side

>>install and enable the nfs-server



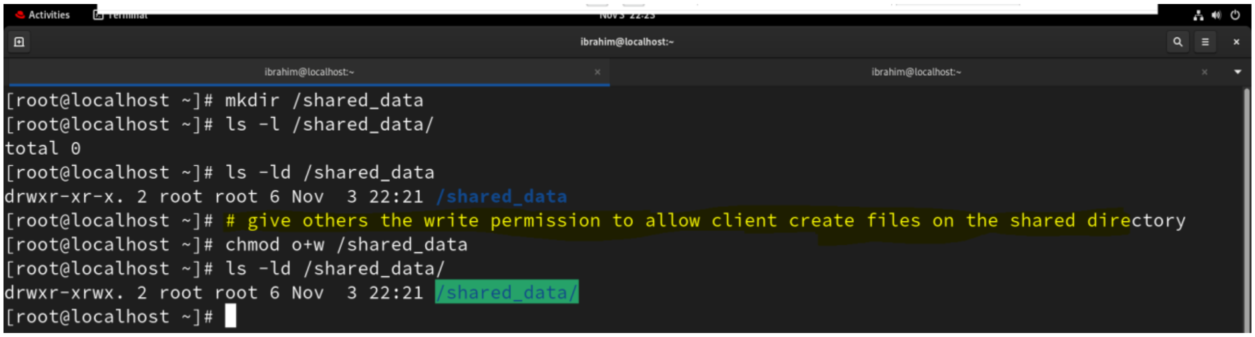
```
[root@localhost ~]# dnf install -y nfs-utils
Updating Subscription Management repositories.

This system is registered with an entitlement server, but is not receiving updates. You can use subscription-man
ager to assign subscriptions.

Last metadata expiration check: 0:00:46 ago on Sun 03 Nov 2024 10:12:58 PM EET.
Package nfs-utils-1:2.5.4-26.el9_4.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@localhost ~]# systemctl enable --now nfs-server
[root@localhost ~]# systemctl status nfs-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; preset: disabled)
    Drop-In: /run/systemd/generator/nfs-server.service.d
             └─order-with-mounts.conf
     Active: active (exited) since Sun 2024-11-03 19:36:09 EET; 2h 38min ago
       Docs: man:rpc.nfsd(8)
             man:exportfs(8)
   Main PID: 1294 (code=exited, status=0/SUCCESS)
        CPU: 24ms

Nov 03 19:36:08 localhost.localdomain systemd[1]: Starting NFS server and services...
Nov 03 19:36:09 localhost.localdomain systemd[1]: Finished NFS server and services.
[root@localhost ~]# 
```

>> Enable the nfs-servie on the firewall



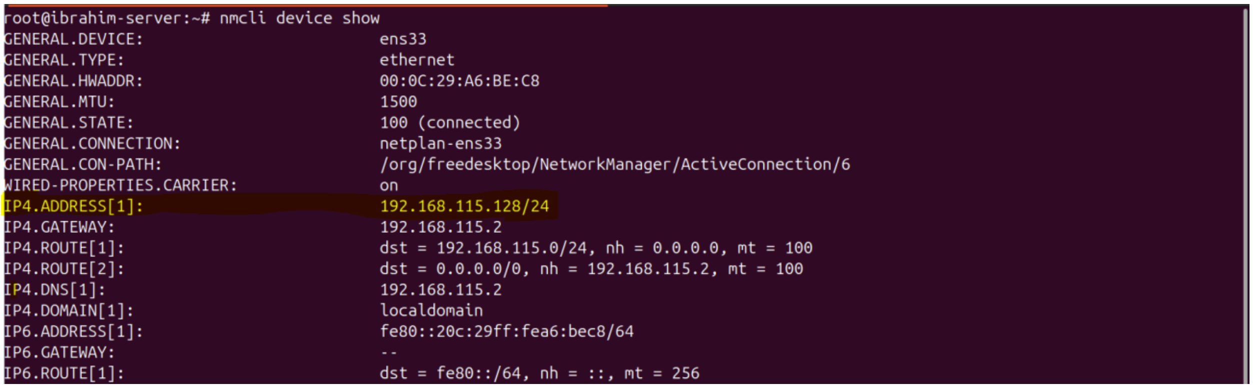
```
[root@localhost ~]# firewall-cmd --permanent --add-service=nfs
Warning: ALREADY_ENABLED: nfs
success
[root@localhost ~]# firewall
firewall-cmd          firewalld             firewall-offline-cmd
[root@localhost ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens160
  sources:
  services: cockpit dhcpv6-client nfs ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[root@localhost ~]# 
```

>>Create the shared directory

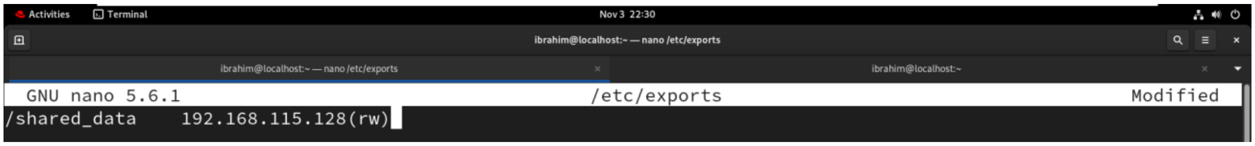


```
[root@localhost ~]# mkdir /shared_data
[root@localhost ~]# ls -l /shared_data/
total 0
[root@localhost ~]# ls -ld /shared_data
drwxr-xr-x. 2 root root 6 Nov  3 22:21 /shared_data
[root@localhost ~]# # give others the write permission to allow client create files on the shared directory
[root@localhost ~]# chmod o+w /shared_data
[root@localhost ~]# ls -ld /shared_data/
drwxr-xrwx. 2 root root 6 Nov  3 22:21 /shared_data/
[root@localhost ~]# 
```

>>Get the client IP and Export the shared directory



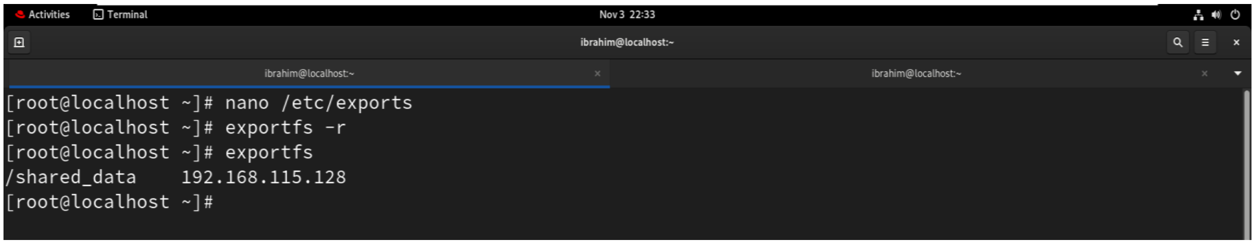
```
root@ibrahim-server:~# nmcli device show
GENERAL.DEVICE:                         ens33
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         00:0C:29:A6:BE:C8
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     netplan-ens33
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/6
WIRED-PROPERTIES.CARRIER:               on
IP4.ADDRESS[1]:                         192.168.115.128/24
IP4.GATEWAY:                            192.168.115.2
IP4.ROUTE[1]:                           dst = 192.168.115.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh = 192.168.115.2, mt = 100
IP4.DNS[1]:                             192.168.115.2
IP4.DOMAIN[1]:                          localdomain
IP6.ADDRESS[1]:                         fe80::20c:29ff:fea6:bec8/64
IP6.GATEWAY:                            --
IP6.ROUTE[1]:                           dst = fe80::/64, nh = ::, mt = 256
```


Activities
Terminal
Nov 3 22:30
ibrahim@localhost:~ — nano /etc/exports
ibrahim@localhost:~ — nano /etc/exports
ibrahim@localhost:~

```
  GNU nano 5.6.1                    /etc/exports                    Modified
/shared_data     192.168.115.128(rw)
```

>>save and exit

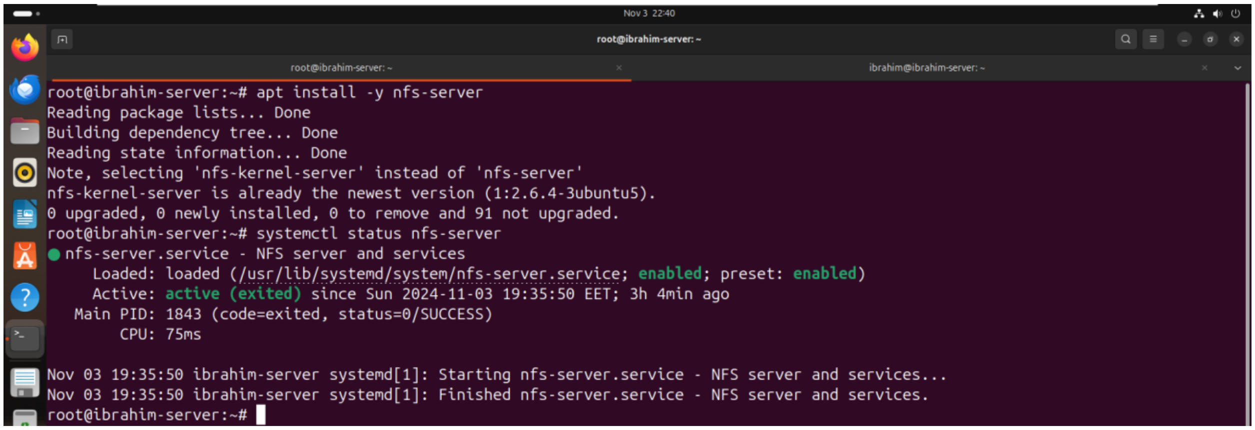
>> Now use the exportfs -r to make the server re read the configuration file without terminating any connections



```
[root@localhost ~]# nano /etc/exports
[root@localhost ~]# exportfs -r
[root@localhost ~]# exportfs
/shared_data     192.168.115.128
[root@localhost ~]#
```

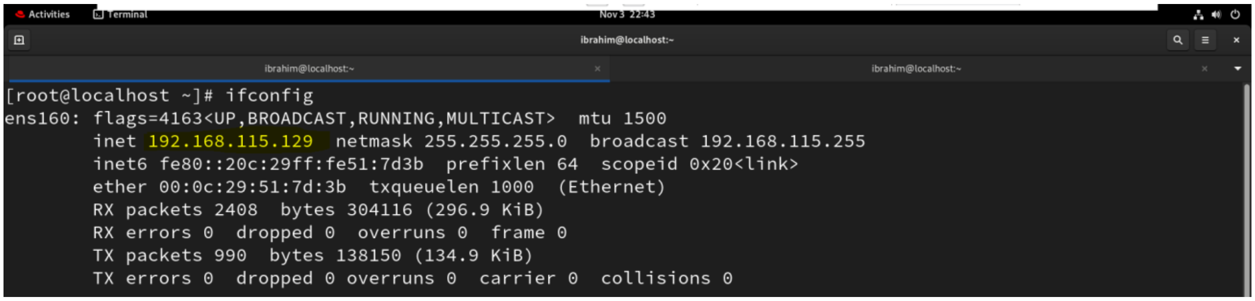
>> Now let’s configure the client side

>> Install the NFS package

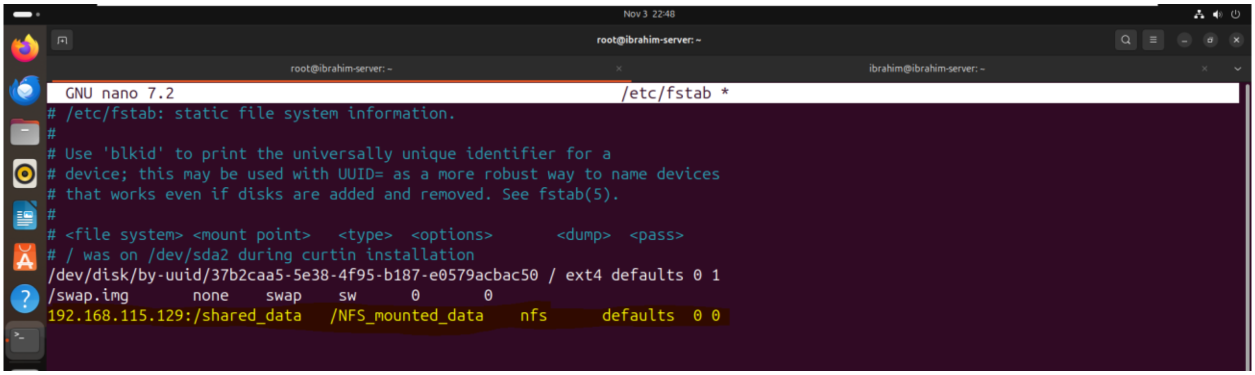


```
root@ibrahim-server:~# apt install -y nfs-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'nfs-kernel-server' instead of 'nfs-server'
nfs-kernel-server is already the newest version (1:2.6.4-3ubuntu5).
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ibrahim-server:~# systemctl status nfs-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; preset: enabled)
     Active: active (exited) since Sun 2024-11-03 19:35:50 EET; 3h 4min ago
   Main PID: 1843 (code=exited, status=0/SUCCESS)
        CPU: 75ms

Nov 03 19:35:50 ibrahim-server systemd[1]: Starting nfs-server.service - NFS server and services...
Nov 03 19:35:50 ibrahim-server systemd[1]: Finished nfs-server.service - NFS server and services.
root@ibrahim-server:~# 
```

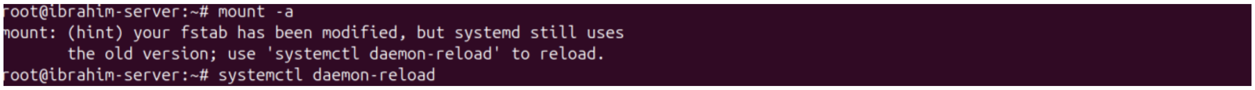
>> Now get the IP address of the NFS-server and create a mount point “/NFS_mounted_data” on the client then go to the fstab file and add the mount record to make it exist across system reboots



```
[root@localhost ~]# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.115.129  netmask 255.255.255.0  broadcast 192.168.115.255
        inet6 fe80::20c:29ff:fe51:7d3b  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:51:7d:3b  txqueuelen 1000  (Ethernet)
        RX packets 2408  bytes 304116 (296.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 990  bytes 138150 (134.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

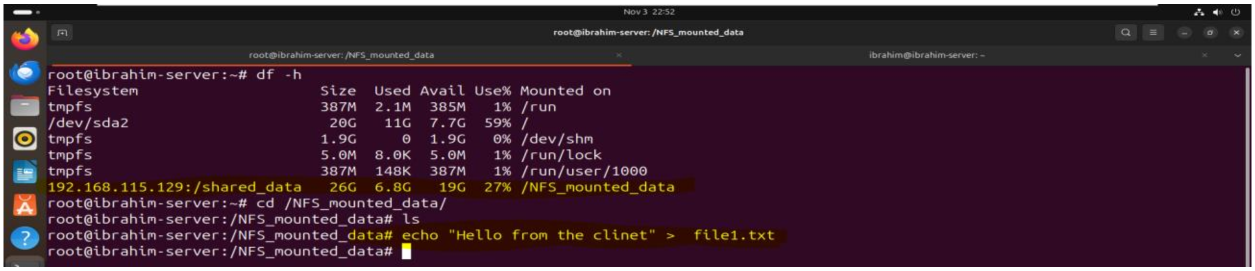


```
  GNU nano 7.2                                               /etc/fstab *
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/37b2caa5-5e38-4f95-b187-e0579acbac50 / ext4 defaults 0 1
/swap.img       none    swap    sw      0       0
192.168.115.129:/shared_data    /NFS_mounted_data     nfs      defaults  0 0
```



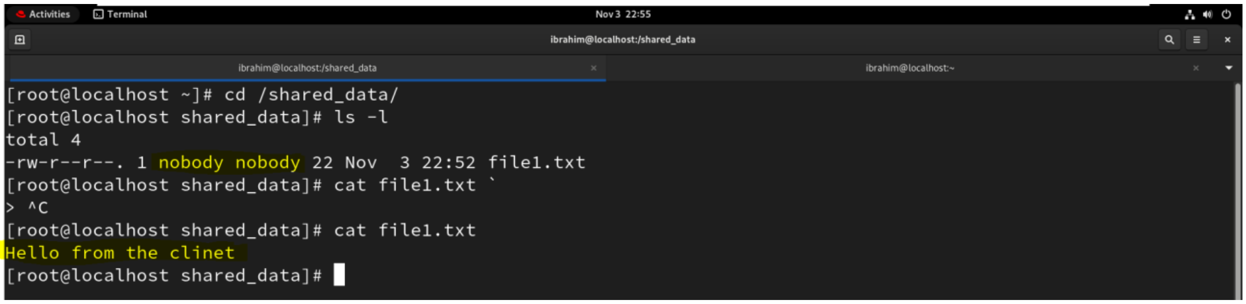
```
root@ibrahim-server:~# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
root@ibrahim-server:~# systemctl daemon-reload
```

>>Verfy that your client can reach the server’s shared file



```
root@ibrahim-server:~# df -h
Filesystem                     Size  Used Avail Use% Mounted on
tmpfs                          387M  2.1M  385M   1% /run
/dev/sda2                       20G   11G  7.7G  59% /
tmpfs                          1.9G     0  1.9G   0% /dev/shm
tmpfs                          5.0M  8.0K  5.0M   1% /run/lock
tmpfs                          387M  148K  387M   1% /run/user/1000
192.168.115.129:/shared_data    26G  6.8G   19G  27% /NFS_mounted_data
root@ibrahim-server:~# cd /NFS_mounted_data/
root@ibrahim-server:/NFS_mounted_data# ls
root@ibrahim-server:/NFS_mounted_data# echo "Hello from the clinet" >  file1.txt
root@ibrahim-server:/NFS_mounted_data# 
```

>> Go to the server and check if the file that created on the client device is correctly created or not


Activities
Terminal
Nov 3 22:55
ibrahim@localhost:/shared_data
ibrahim@localhost:/shared_data
ibrahim@localhost:~

```
[root@localhost ~]# cd /shared_data/
[root@localhost shared_data]# ls -l
total 4
-rw-r--r--. 1 nobody nobody 22 Nov  3 22:52 file1.txt
[root@localhost shared_data]# cat file1.txt `
> ^C
[root@localhost shared_data]# cat file1.txt
Hello from the clinet
[root@localhost shared_data]# 
```

>> The “nobody” user appears on the server is: Because of the client’s root user has created this file plus I have choosen the root squash option on the NFS-server for security reasons the client’s root user is assigned to “nobody” user which is a regular user on my NFS-server.