

# GLUSTERFS

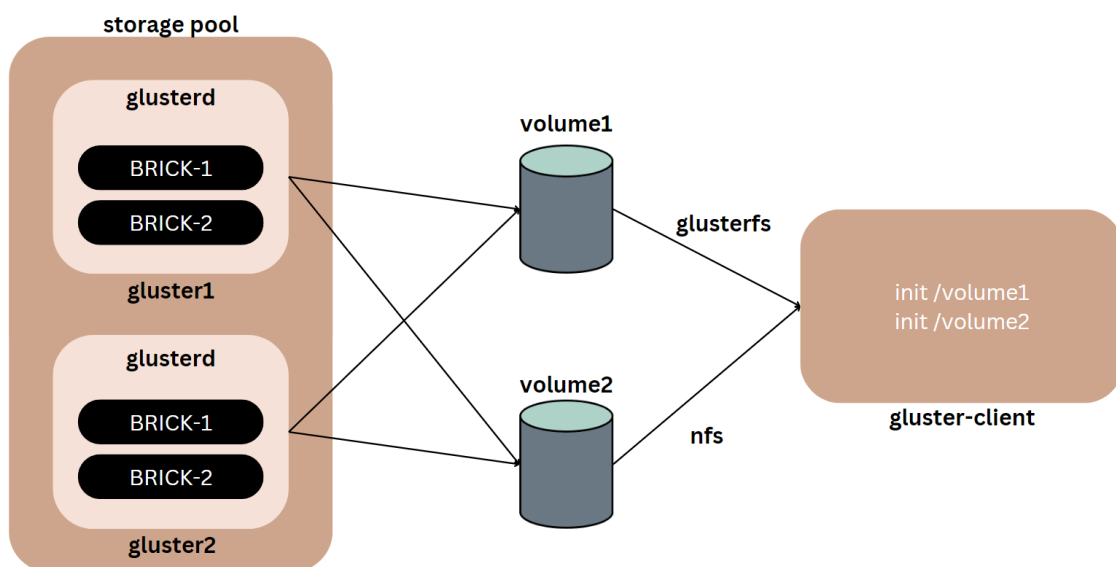
TEAM: 121AD0012, 121AD0014, 121AD0019, 121AD0020

## What is GlusterFS and why do we need it?

GlusterFS (GFS) is an open-source distributed file system designed to provide scalable and high-performance storage for various types of data. It is part of the Gluster storage software suite, initially developed by Gluster Inc. and later acquired by Red Hat. GlusterFS is built to run on commodity hardware, allowing users to create a distributed storage infrastructure that can easily scale as their storage needs grow. Main advantages to note are:

1. Distributed architecture
2. Flexibility
3. Scalability
4. High Availability
5. Elasticity

BASIC ARCHITECTURE USED:



This is to be implemented on the Ubuntu systems. We will have three machines, two of which will be servers and one client.

## Installing glusterfs on server and client nodes

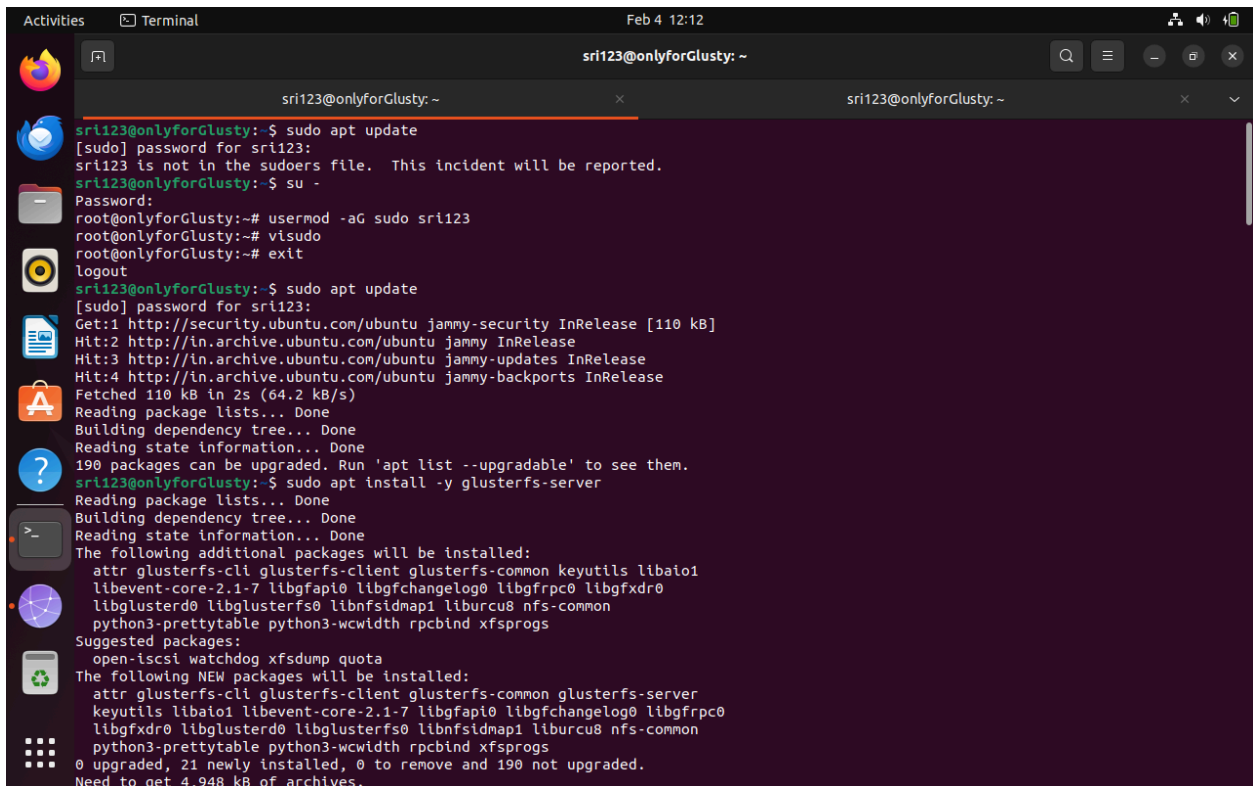
Command: `lsblk`, gives the disks and partitions on the machine.

On both server nodes, first we execute the following commands:

```
sudo apt update
sudo apt install -y glusterfs-server
sudo systemctl start glusterd
sudo systemctl enable glusterd
sudo systemctl status glusterd
```

On client node:

```
sudo apt update
sudo apt install -y glusterfs-client
```



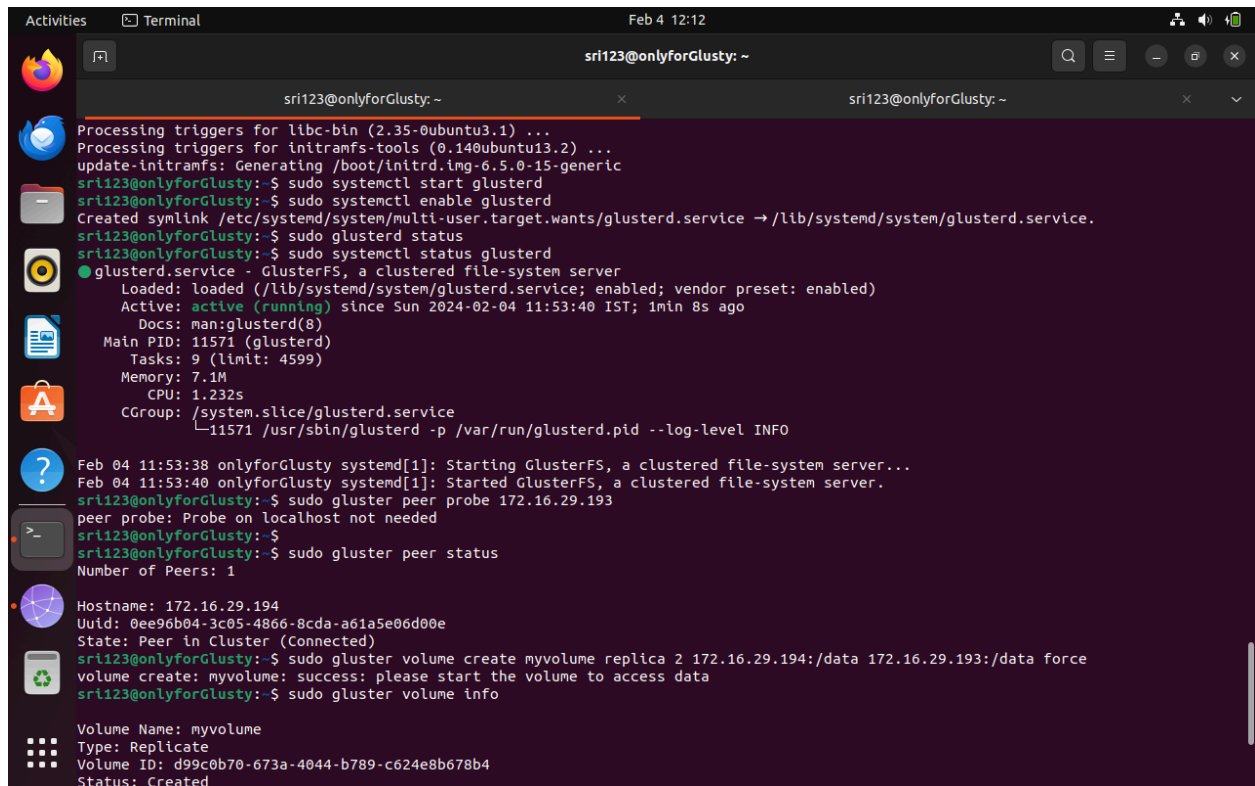
```
Activities Terminal Feb 4 12:12
sri123@onlyforGlusty: ~
sri123@onlyforGlusty: ~
sri123@onlyforGlusty:~$ sudo apt update
[sudo] password for sri123:
sri123 is not in the sudoers file. This incident will be reported.
sri123@onlyforGlusty:~$ su -
Password:
root@onlyforGlusty:~# usermod -aG sudo sri123
root@onlyforGlusty:~# visudo
root@onlyforGlusty:~# exit
logout
sri123@onlyforGlusty:~$ sudo apt update
[sudo] password for sri123:
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 110 kB in 2s (64.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
190 packages can be upgraded. Run 'apt list --upgradable' to see them.
sri123@onlyforGlusty:~$ sudo apt install -y glusterfs-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  attr glusterfs-cli glusterfs-client glusterfs-common keyutils libaio1
  libevent-core-2.1-7 libgfapi0 libgfchangelog0 libgfrpc0 libgfxdr0
  libglusterd0 libglusterfs0 libnfsidmap1 liburcu8 nfs-common
  python3-pytable python3-wcwidth rpcbind xfsprogs
Suggested packages:
  open-iscsi watchdog xfsdump quota
The following NEW packages will be installed:
  attr glusterfs-cli glusterfs-client glusterfs-common glusterfs-server
  keyutils libaio1 libevent-core-2.1-7 libgfapi0 libgfchangelog0 libgfrpc0
  libgfxdr0 libglusterd0 libglusterfs0 libnfsidmap1 liburcu8 nfs-common
  python3-pytable python3-wcwidth rpcbind xfsprogs
0 upgraded, 21 newly installed, 0 to remove and 190 not upgraded.
Need to get 4,948 kB of archives.
```

## Configuring the servers as a single storage pool

Now on one of the servers, we need to configure them to create a trusted storage pool. In order to do so: `sudo gluster peer probe <server1>` is executed, where the server1 is another machine server along with the one on which this command is being run.

Command to check status:

`sudo gluster peer status`

A terminal window titled 'sri123@onlyforGlusty: ~' showing the configuration of GlusterFS. The user runs several commands: 'sudo systemctl start glusterd', 'sudo systemctl enable glusterd', 'sudo glusterd status', and 'sudo gluster peer status'. The output shows that the glusterd service is active and running. The peer status command shows one peer (localhost) is not needed. The user then runs 'sudo gluster volume create myvolume replica 2 172.16.29.194:/data 172.16.29.193:/data force', which successfully creates a volume named 'myvolume' with two replicas on the specified paths. Finally, the user runs 'sudo gluster volume info', which displays details about the volume: Name: myvolume, Type: Replicate, Volume ID: d99c0b70-673a-4044-b789-c624e8b678b4, Status: Created.

```
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for initramfs-tools (0.140ubuntu13.2) ...
update-initramfs: Generating /boot/initrd.img-6.5.0-15-generic
sri123@onlyforGlusty:~$ sudo systemctl start glusterd
sri123@onlyforGlusty:~$ sudo systemctl enable glusterd
Created symlink /etc/systemd/system/multi-user.target.wants/glusterd.service → /lib/systemd/system/glusterd.service.
sri123@onlyforGlusty:~$ sudo glusterd status
glusterd.service - GlusterFS, a clustered file-system server
Loaded: loaded (/lib/systemd/system/glusterd.service; enabled; vendor preset: enabled)
Active: active (running) since Sun 2024-02-04 11:53:40 IST; 1min 8s ago
Docs: man:glusterd(8)
Main PID: 11571 (glusterd)
Tasks: 9 (limit: 4599)
Memory: 7.1M
CPU: 1.232s
CGroup: /system.slice/glusterd.service
└─11571 /usr/sbin/glusterd -p /var/run/glusterd.pid --log-level INFO

Feb 04 11:53:38 onlyforGlusty systemd[1]: Starting GlusterFS, a clustered file-system server...
Feb 04 11:53:40 onlyforGlusty systemd[1]: Started GlusterFS, a clustered file-system server.
sri123@onlyforGlusty:~$ sudo gluster peer probe 172.16.29.193
peer probe: Probe on localhost not needed
sri123@onlyforGlusty:~$ sudo gluster peer status
Number of Peers: 1

Hostname: 172.16.29.194
Uuid: 0ee96b04-3c05-4866-8cda-a61a5e06d00e
State: Peer in Cluster (Connected)
sri123@onlyforGlusty:~$ sudo gluster volume create myvolume replica 2 172.16.29.194:/data 172.16.29.193:/data force
volume create: myvolume: success: please start the volume to access data
sri123@onlyforGlusty:~$ sudo gluster volume info

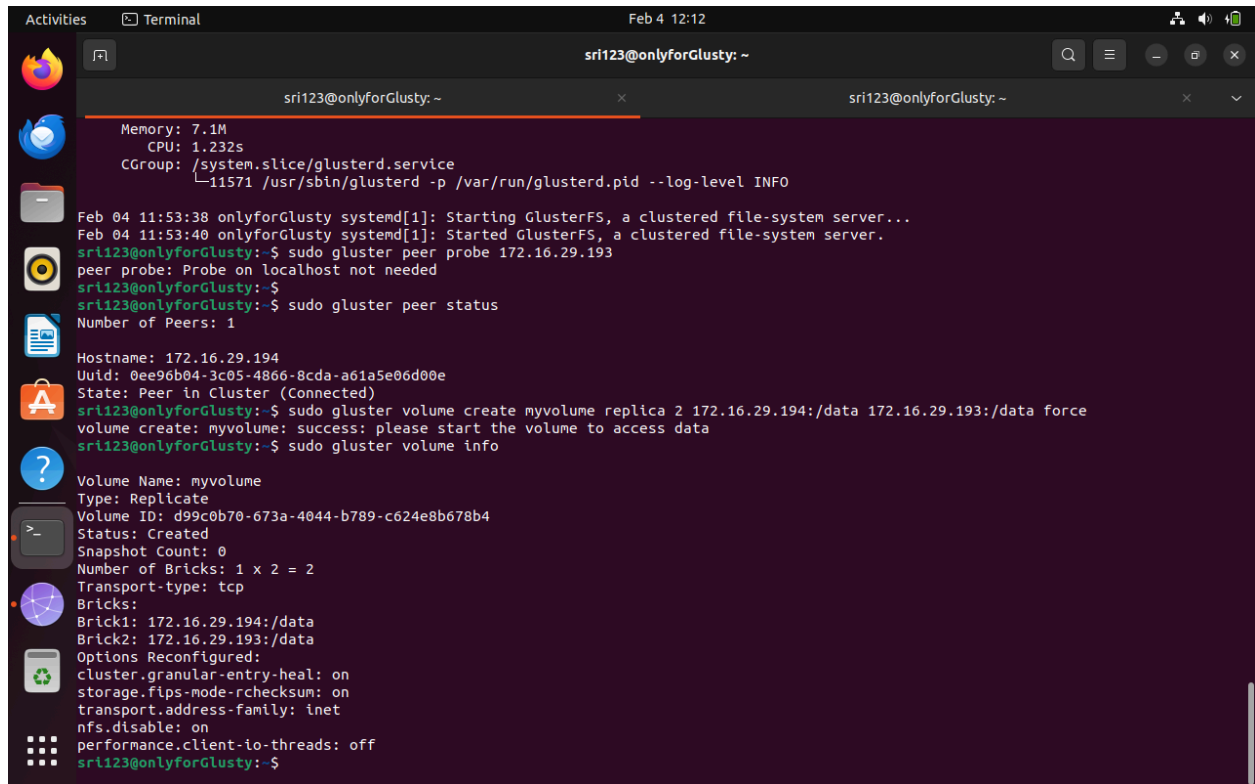
Volume Name: myvolume
Type: Replicate
Volume ID: d99c0b70-673a-4044-b789-c624e8b678b4
Status: Created
```

## Creating and mounting Volume

`sudo gluster volume create ourvolume replica 2 <server1>:/data <server2>:/data force`

This command creates a volume or a storage space and its replica on both the servers. This command when run on one of the servers makes changes to both of the server nodes. To check this fact, we can execute:

`sudo gluster volume info`

A terminal window titled 'Terminal' with a dark background. The prompt is 'sri123@onlyforGlusty: ~'. The output shows system resources (Memory: 7.1M, CPU: 1.232s, CGroup: /system.slice/glusterd.service), log messages for starting GlusterFS, and the execution of 'sudo gluster peer probe 172.16.29.193'. It then shows 'sudo gluster peer status' output (Number of Peers: 1) and 'sudo gluster volume create myvolume replica 2 172.16.29.194:/data 172.16.29.193:/data force' output (volume create: myvolume: success: please start the volume to access data). Finally, 'sudo gluster volume info' is run, showing details for 'myvolume' (Type: Replicate, Volume ID: d99c0b70-673a-4044-b789-c624e8b678b4, Status: Created, Number of Bricks: 1 x 2 = 2, Transport-type: tcp, Bricks: 172.16.29.194:/data, 172.16.29.193:/data, Options Reconfigured: cluster.granular-entry-heal: on, storage.fips-mode-rchecksum: on, transport.address-family: inet, nfs.disable: on, performance.client-io-threads: off).

```
Memory: 7.1M
CPU: 1.232s
CGroup: /system.slice/glusterd.service
        └─11571 /usr/sbin/glusterd -p /var/run/glusterd.pid --log-level INFO

Feb 04 11:53:38 onlyforGlusty systemd[1]: Starting GlusterFS, a clustered file-system server...
Feb 04 11:53:40 onlyforGlusty systemd[1]: Started GlusterFS, a clustered file-system server.
sri123@onlyforGlusty:~$ sudo gluster peer probe 172.16.29.193
peer probe: Probe on localhost not needed
sri123@onlyforGlusty:~$ sudo gluster peer status
Number of Peers: 1

Hostname: 172.16.29.194
Uuid: 0ee96b04-3c05-4866-8cda-a61a5e06d00e
State: Peer in Cluster (Connected)
sri123@onlyforGlusty:~$ sudo gluster volume create myvolume replica 2 172.16.29.194:/data 172.16.29.193:/data force
volume create: myvolume: success: please start the volume to access data
sri123@onlyforGlusty:~$ sudo gluster volume info

Volume Name: myvolume
Type: Replicate
Volume ID: d99c0b70-673a-4044-b789-c624e8b678b4
Status: Created
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 172.16.29.194:/data
Brick2: 172.16.29.193:/data
Options Reconfigured:
cluster.granular-entry-heal: on
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
sri123@onlyforGlusty:~$
```

## Testing the setup

With the above commands we have finished the setting of two server nodes and one client node. Now we need to check whether the server has the client files or not. For this we need to first make the volume active or running:

```
sudo gluster volume start ourvolume
```

After executing this command on one of the servers, we may check the volume status on both the servers to be active or running:

```
sudo gluster volume status
```

In this the volume should be 'running' in order for the mounting data files by the client.

Now on client node:

```
sudo mkdir /mnt/glusterfs
```

```
sudo mount -t glusterfs <server_ip>:/myvolume /mnt/glusterfs
```

These commands will create a local directory that will be shared with both the servers. We can verify the mount using command: *df -h*

```
hemanth@hemanth:/data$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            726M  2.3M  724M   1% /run
/dev/nvme0n1p6  96G   18G   74G  19% /
tmpfs            3.6G   0  3.6G   0% /dev/shm
tmpfs            5.0M   20K   5.0M   1% /run/lock
/dev/nvme0n1p1  256M   90M  167M  35% /boot/efi
tmpfs            726M  104K  726M   1% /run/user/1000
```

```
chandu@chandu:/data$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            768M  2.3M  766M   1% /run
/dev/sda5       143G   37G   99G  27% /
tmpfs            3.8G   0  3.8G   0% /dev/shm
tmpfs            5.0M   8.0K   5.0M   1% /run/lock
/dev/sda4        1.1G   34M   1.1G   4% /boot/efi
tmpfs            768M  112K  768M   1% /run/user/1000
/dev/sdb1        15G   5.1G   9.4G  36% /media/chandu/SONY
```

```
chandu@chandu:~$ sudo gluster volume status
[sudo] password for chandu:
Status of volume: ourvolume
Gluster process                                TCP Port  RDMA Port  Online  Pid
-----
Brick 172.16.30.153:/data                      55810     0           Y      18642
Brick 172.16.28.242:/data                      60320     0           Y      14193
Self-heal Daemon on localhost                  N/A       N/A         Y      14210
Self-heal Daemon on 172.16.30.153              N/A       N/A         Y      18659

Task Status of Volume ourvolume
-----
There are no active volume tasks

chandu@chandu:~$
```

```
hemanth@hemanth:~$ sudo gluster volume status
[sudo] password for hemanth:
Status of volume: ourvolume
Gluster process                                TCP Port  RDMA Port  Online  Pid
-----
Brick 172.16.30.153:/data                      55810     0           Y      18642
Brick 172.16.28.242:/data                      60320     0           Y      14193
Self-heal Daemon on localhost                  N/A       N/A         Y      18659
Self-heal Daemon on 172.16.28.242              N/A       N/A         Y      14210

Task Status of Volume ourvolume
-----
There are no active volume tasks
```

```
chandu@chandu:~$ sudo gluster volume info
Volume Name: ourvolume
Type: Replicate
Volume ID: 640bfd62-1d57-448b-904f-0df66861c763
Status: Started
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 172.16.30.153:/data
Brick2: 172.16.28.242:/data
Options Reconfigured:
cluster.granular-entry-heal: on
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
chandu@chandu:~$
```

```
hemanth@hemanth:~$ sudo gluster volume info
Volume Name: ourvolume
Type: Replicate
Volume ID: 640bfd62-1d57-448b-904f-0df66861c763
Status: Created
Snapshot Count: 0
Number of Bricks: 1 x 2 = 2
Transport-type: tcp
Bricks:
Brick1: 172.16.30.153:/data
Brick2: 172.16.28.242:/data
Options Reconfigured:
cluster.granular-entry-heal: on
storage.fips-mode-rchecksum: on
transport.address-family: inet
nfs.disable: on
performance.client-io-threads: off
```

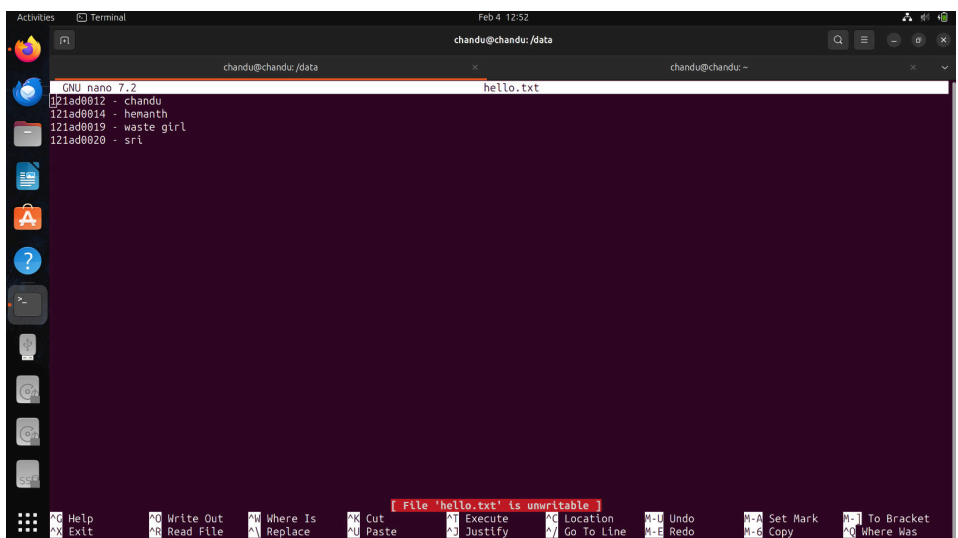
Now we create a file in client and save it to see whether the file is present in servers:

```
echo "Hello, chandu, hemanth!" | sudo tee /mnt/glusterfs/testfile.txt
```

This creates a text file in the /mnt/glusterfs directory, and on going to /data it shows the file in servers as well.

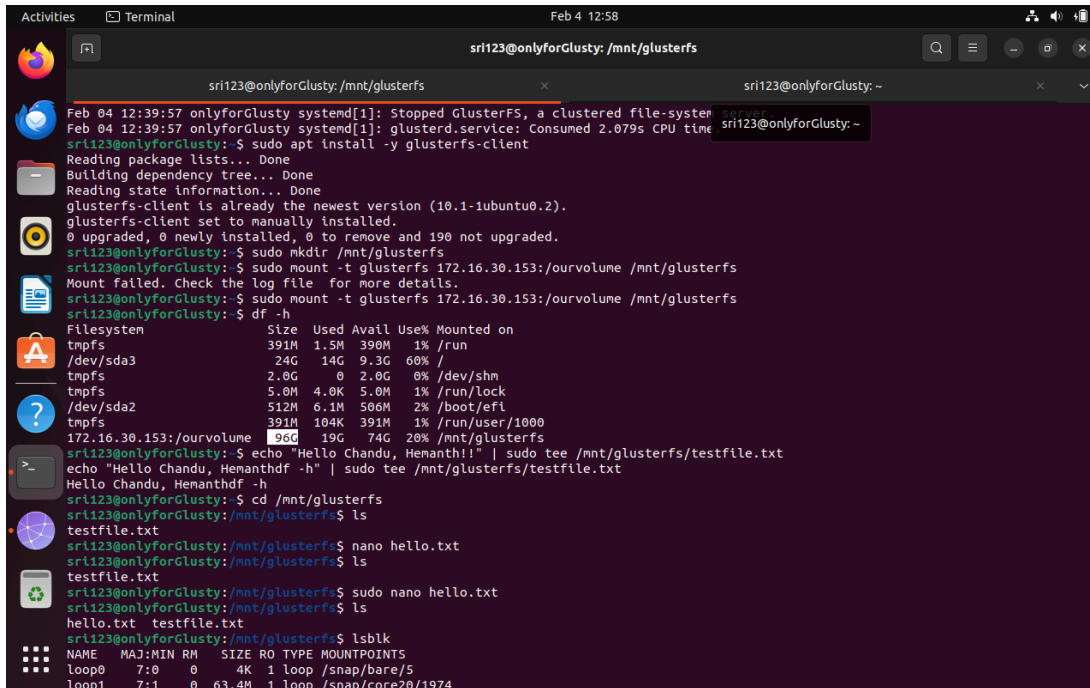
```
hemanth@hemanth:~$ ls /data
testfile.txt
hemanth@hemanth:~$
```

We notice that on accessing the file, we are only able to see and not to edit any of its contents on the servers.



Similarly on writing another file in the same directory, it is also showing in both the servers.

Client:

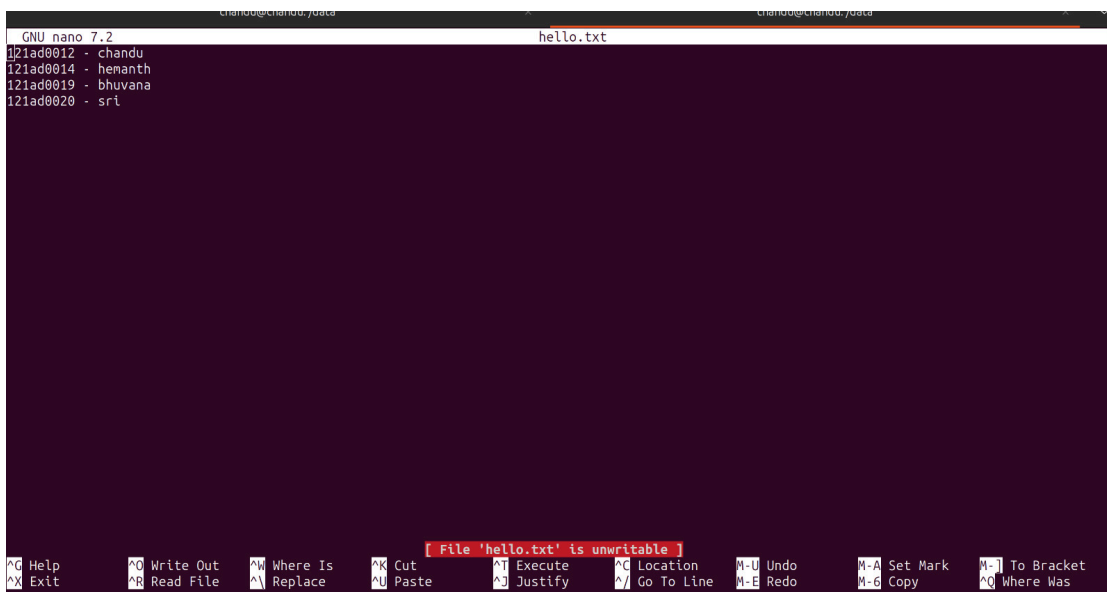


```
Feb 04 12:39:57 onlyforGlusty systemd[1]: Stopped GlusterFS, a clustered file-system
Feb 04 12:39:57 onlyforGlusty systemd[1]: glusterd.service: Consumed 2.079s CPU time
sri123@onlyforGlusty:~$ sudo apt install -y glusterfs-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
glusterfs-client is already the newest version (10.1-1ubuntu0.2).
glusterfs-client set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 190 not upgraded.
sri123@onlyforGlusty:~$ sudo mkdir /mnt/glusterfs
sri123@onlyforGlusty:~$ sudo mount -t glusterfs 172.16.30.153:/ourvolume /mnt/glusterfs
Mount failed. Check the log file for more details.
sri123@onlyforGlusty:~$ sudo mount -t glusterfs 172.16.30.153:/ourvolume /mnt/glusterfs
sri123@onlyforGlusty:~$ df -h
Filesystem                Size      Used Avail Use% Mounted on
tmpfs                      391M    1.5M  390M   1% /run
/dev/sda3                  24G     14G   9.3G  60% /
tmpfs                      2.0G     0   2.0G   0% /dev/shm
tmpfs                      5.0M   4.0K  5.0M   1% /run/lock
/dev/sda2                  512M   6.1M  506M   2% /boot/efi
tmpfs                      391M   104K  391M   1% /run/user/1000
172.16.30.153:/ourvolume  96G    19G   74G  20% /mnt/glusterfs
sri123@onlyforGlusty:~$ echo "Hello Chandu, Hemanth!!" | sudo tee /mnt/glusterfs/testfile.txt
Hello Chandu, Hemanth!!
sri123@onlyforGlusty:~$ cd /mnt/glusterfs
sri123@onlyforGlusty:/mnt/glusterfs$ ls
testfile.txt
sri123@onlyforGlusty:/mnt/glusterfs$ nano hello.txt
sri123@onlyforGlusty:/mnt/glusterfs$ ls
testfile.txt
sri123@onlyforGlusty:/mnt/glusterfs$ sudo nano hello.txt
sri123@onlyforGlusty:/mnt/glusterfs$ ls
hello.txt  testfile.txt
sri123@onlyforGlusty:/mnt/glusterfs$ lsblk
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0  7:0    0    4K  1 loop /snap/bare/5
loop1  7:1    0 63.4M  1 loop /snap/core20/1974
```

Servers:



```
chandu@chandu:~$ cd /data
chandu@chandu:/data$ ls
hello.txt  testfile.txt
chandu@chandu:/data$
```



```
GNU nano 7.2 hello.txt
121ad0012 - chandu
121ad0014 - hemanth
121ad0019 - bhuvana
121ad0020 - sri
[ File 'hello.txt' is unwritable ]
Help Write Out Where Is Cut Execute Location M-U Undo M-A Set Mark M-~ To Bracket
Exit Read File Replace Paste Justify Go To Line M-E Redo M-C Copy M-! Where Was
```