GETTING STARTED

WITH

NETWORK FILE SYSTEM

➢ NETWORK FILE SYSTEM:

- There are many different ways you can run share files with users on your network.
- NFS provides a way for other systems on the network to store files in a centralized place.
- The NFS protocol works great when it comes to Linux systems because it allows for client flexibility, centralized management of files, and some other great features.
- By using NFS protocol, remote users can mount shared directories over a network and use them as they were mounted locally. This enables you to consolidate resources onto centralized servers on the network.

NFSV4:

- There are four different versions of NFS; Version4 is the most current. Although you can disable what versions the server listens for, the client actually determines which version it will use when connecting to the server.
- NFSv4 version brings enhancements aimed at improving performance and security. Use these improvements to utilize the full potential of NFSv4, ensuring efficient and reliable file sharing across networks.
- It works through firewalls and on the Internet, no longer requires a **rpcbind** service, supports **Access Control Lists (ACLs)**, and utilizes stateful operations.

NFS VERSION 3 (NFSV3):

- Supports safe **Asynchronous** writes and is more robust at error handling.
- Supports **64-bit file sizes** and offsets, allowing clients to access **more than 2 GB** of file data.

NFS FEATURES:

- Everyone can access same data.
- Reduces storage cost.
- Can be acted as centralized storage solutions.
- Provides data consistency.
- Reduces the system administrator overhead.

❖ NFS SERVER CONFIGURATION:

PRE-REQUISITES:

▪ **Package name**	: nfs-utils
▪ **Shared File Location**	: /etc/exports
▪ **Ports Control config file**	: /etc/sysconfig/nfs
▪ **Main Config file**	: /etc/nfs.conf
▪ **Log File**	: /var/log/messages
▪ **Currently exported resources**	: /var/lib/nfs/etab
▪ **Remotely mounted resources**	: /var/lib/nfs/rmtab
▪ **Ports**	: NFS – 2049, PORTMAP - 111
▪ **Services**	: rpcbind, nfs-server

- **Daemons:**

  NFSD : Starts the NFS server

  RPCBIND : Forward incoming requests to the appropriate subservice

  RPC.MOUNTD : nfs server to process mount request from clients.

  RPC.NFSD : Responds to clients request for file access.

  LOCKD : Manages file locks and release in case of client disconnect.

  RPC.STATD : Works with rpc.lockd to provide recovery service

  RPC.RQUOTAD : Provides statistics on disk quotas to clients.

  RPC.IDMAPD : Provides NFSV4 client and server upcalls, which map b/w on the NFSv4 names & local UID's and GID's.

THE /ETC/EXPORTS CONFIGURATION FILE:

- The /etc/exports file controls which directories the server exports. Each line contains an export point, a whitespace-separated list of clients that are allowed to mount the directory, and options for each of the clients:

  SYNTAX: **<directory>   <host_or_network_1>(<options_1>)…**

  ro : Exports the directory in read-only mode.

  sync : The NFS server does not reply to requests before changes made by previous requests are written to disk.

  root_squash : Prevents that the root user on clients has root permissions on an exported directory. With root_squash enabled, the NFS server maps access from root to the user nobody.

→ **Install the nfs-utils package:**

```
#dnf install nfs-utils
```

→ **Edit the /etc/nfs.conf file, and make the following changes:**

Disable the vers3 parameter in the [nfsd] section to disable NFSv3:

```
        vers3=n
```

→ **Reload the systemd configuration and restart the nfs service:**

```
# systemctl daemon-reload
# systemctl restart nfs-server
```

→ **Enable and verifying nfs server status:**

```
#systemctl enable nfs-server
#systemctl status nfs-server
```

→ **To verify the port numbers:**

```
#netstat -pantl
```

→ **To verify nfs version status:**

```
# rpcinfo -p
```

→ **Create a directory that you want to share, for example:**

```
#mkdir /cloud-data
#cd /cloud-data
#touch aws azure gcp
```

→ **Add an export point for each directory that you want to share:**

```
#vim /etx/exports
  /cloud-data     192.168.10.0/24(ro,sync)
```

→ **Run the command to update /etc/exports file:**

```
#exportfs -avf
```

→ **verify the NFS shared resources:**

```
#showmount -e 0
```

mountstats **:** Shows information about mounted NFS shares.
nfsstat **:** Shows statistics of exported resources.
nfsiostat **:** Shows statistics of NFS mounted shares.

❖ NFS CLIENT CONFIGURATION:

→ **Install the nfs-utils package:**

```
#dnf install nfs-utils
```

→ **Verify nfs server shared resources:**

#showmount -e <server-ip/hostname>

```
#showmount -e 192.168.10.254
```

→ **create a directory for nfs mount:**

```
#mkdir /nfs-share
```

→ **Mount nfs share permanently:**

```
#vim /etc/fstab
  192.168.10.254:/cloud-data   /nfs-share   nfs   defaults   0 0
```

→ **Reload the daemon of system:**

```
#systemctl daemon-reload
#mount /nfs-share
#df -h
```

→ **List shared mount resources:**

```
#ls /nfs-share
```