NFS

https://www.howtoforge.com/nfs-server-and-client-on-centos-7

1. Preliminary setup

    a. Virtual machines - you will need at least 2 virtual machines, one to act as the NFS server and one or more to act as the NFS clients.

    b. Network - Ensure that the network for Host-Only networking has been configured and that all of the VMs are connected to the Host-Only network. You will need to know the IP address of VM that will be the NFS server.

2. Setup the NFS Server

    a. Install the NFS package (and dependencies)
    ```
    yum install nfs-utils
    ```

    b. Create the directory that will be shared on the network
    ```
    mkdir /var/nfsshare
    ```

    c. Change the permissions and ownership of this folder
    ```
    chmod -R 755 /var/nfsshare
    chown nfsnobody:nfsnobody /var/nfsshare
    ```

    d. Start and enable the nfs services
    ```
    systemctl start rpcbind
    systemctl start nfs-server
    systemctl start nfs-lock
    systemctl start nfs-idmap
    systemctl enable rpcbind
    systemctl enable nfs-server
    systemctl enable nfs-lock
    systemctl enable nfs-idmap
    ```

    e. Edit the NFS configuration file /etc/exports to instruct NFS to export, or share, the directory you created above. You can use the editor of your choice. Add a line with the following:

    ```
    directoryToExport clientIPAddress(rw,sync,no_root_squash,no_all_squash)
    ```

The directoryToExport is the full path to the directory you created in step 2a, and the serverIPAddress is the IP address of the NFS client. For example, if the directory you are exporting is /var/nfsshare and the IP address of the client is 192.168.149.20 this line must be:

```
/var/nfsshare    192.168.149.20(rw,sync,no_root_squash,no_all_squash)
```

You can use wildcards in the NFS client IP addresses. For example, to allow any client in the 192.168.149 network use 192.168.149.*

```
/var/nfsshare    192.168.149.*(rw,sync,no_root_squash,no_all_squash)
```

f. Restart the NFS service so that it will read the /etc/exports file
systemctl restart nfs-server

systemctl restart nfs-server

g. Edit the firewall rules to allow the NFS traffic

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

h. Reload the firewall so that it reads the new rules

firewall-cmd --reload

3. Setup the NFS Client
   a. Install the NFS package (and dependencies). Yes, this is the same package that we installed for the server.
   `yum install nfs-utils`

   b. Create a directory/folder that will be used as the mount point for the network file.

   mkdir -p /mnt/nfsshare

   With Windows we would assign a drive letter like E: to the network drive. However since UNIX/Linux uses one directory tree and does not use driver letters, we have to pick a directory to use as the mount point or attachment point for the network drive. While the network drive can be mounted anywhere in the file system the mount points are typically created under the /mnt directory. Note that the folder name does not need to be exactly the same as the name of the folder you exported on the server. However if you're like me and easily confused

then using the same name should help minimize confusion, but it's not a requirement. The only problem with this scheme comes if you're mounting multiple network drives, only one can be called nfsshare, so you'll have to come up with some other naming convention and way to track which network file system you're mounting.

c. Manually mount the network drive

   mount -t nfs ***serverIP***:***exportedDir  mountPoint***

   Where *serverIP* is the IP address of the NFS server
   *exportedDir* is the directory exported by the NFS server
   *mountPoint* is the mount point on this client

   For example, if the server IP is 192.168.149.60, the directory exported by the server was /var/nfsshare, and the mount point is /mnt/nfsshare then the command would be:

   mount -t nfs  192.168.149.60:/var/nfsshare  /mnt/nfsshare

d. Test so see if the network directory was successfully mounted by running the df command. The output should show something similar to:

e. Edit the file /etc/fstab so that the network directory is mounted everytime the system boots. Add a line to the bottom of the file like the following:

   ***serverIP***:***exportedDir    mountPoint*** nfs defaults 0 0

   As before *serverIP* is the IP address of the NFS server
   *exportedDir* is the directory exported by the NFS server
   *mountPoint* is the mount point on this client

   For example, if the server IP is 192.168.149.60, the directory exported by the server was /var/nfsshare, and the mount point is /mnt/nfsshare then the line would read:

   192.168.149.60:/var/nfsshare    /mnt/nfsshare  nfs defaults 0 0

4. Acceptance Test
   a. Check that the client is able to mount the network share
      i. Reboot the client
      ii. Run the df command

b. Check that the client is able to write to the network share
    i. Login to the NFS client
    ii. Change to the folder that is being used as the mount point. For example:

        cd /mnt/nfsshare

    iii. Create a new file in this folder. Add some text to the file saying “This was created on the NFS client”

c. Check that the server is able to read the new file created on the client
    i. Login to the NFS server
    ii. Change to the folder that has been exported. For example:
    iii. cd /var/nfsshare
    iv. Use the ls command to list the files in the directory. Ensure that the file you created in the previous step exists.
    v. Display the contents of the file using and ensure that it contains the text you added from the client

d. Check that the NFS server is able to create a file that can be read by the NFS client by reversing the process in the two previous steps.

e. Test the NFS file locking
    i. Ensure that you are logged in to the NFS client
    ii. Open one of the files in the shared network folder in vi (or the editor of your choice)
    iii. Log in to the NFS server
    iv. Start the editor and try to open the same file that you have open in the NFS client.