[how to mount a directory from another server](https://serverfault.com/questions/410588/how-to-mount-a-directory-from-another-server)

|  |
| --- |
| On the 'server' add an entry into your [/etc/exports](http://linux.die.net/man/5/exports)to share the directory e.g.  /directory/with/data secondserver.tld(rw)  which allows secondserver read/write access to the directory/with/data  then use [exportfs](http://linux.die.net/man/8/exportfs) to share the directory  exportfs -r  and you can verify your export with  exportfs  /directory/with/data secondserver.tld  You can now [mount](http://linux.die.net/man/8/mount) your directory on secondserver  mount server:/directory/with/data /mnt  and you can verify the mount  mount -t nfs  server:/directory/with/data on /mnt type nfs (rw,addr=192.168.254.196)  You may want to add an entry to your [fstab](http://linux.die.net/man/5/fstab) to have it mount on boot too  server:/directory/with/data /mnt nfs rw 0 0 |

# How to Setup NFS (Network File System) on RHEL/CentOS/Fedora and Debian/Ubuntu

## Important commands for NFS

|  |
| --- |
| Some more important commands for **NFS**.   1. **showmount -e** : Shows the available **shares** on your local machine 2. **showmount -e** <**server-ip or hostname**>: Lists the available **shares** at the **remote** server 3. **showmount -d** : Lists all the **sub directories** 4. **exportfs -v** : Displays a list of shares **files** and **options** on a server 5. **exportfs -a** : Exports all shares listed in **/etc/exports**, or given name 6. **exportfs -u** : Unexports all shares listed in **/etc/exports**, or given name 7. **exportfs -r** : Refresh the server’s list after modifying **/etc/exports** |

## 1.Benefits of NFS

|  |
| --- |
| 1. **NFS** allows local access to remote files. 2. It uses standard **client**/**server** architecture for file sharing between all \***nix** based machines. 3. With **NFS** it is not necessary that both machines run on the same **OS**. 4. With the help of **NFS** we can configure **centralized storage** solutions. 5. Users get their **data** irrespective of physical location. 6. No manual **refresh** needed for new files. 7. Newer version of **NFS** also supports **acl**, **pseudo** root mounts. 8. Can be secured with **Firewalls** and **Kerberos**. |

## 2.NFS Services

|  |
| --- |
| Its a System V-launched service. The NFS server package includes three facilities, included in the portmap and nfs-utils packages.   1. portmap : It maps calls made from other machines to the correct RPC service (not required with NFSv4). 2. nfs: It translates remote file sharing requests into requests on the local file system. 3. rpc.mountd: This service is responsible for mounting and unmounting of file systems. |

## 3.Important Files for NFS Configuration

|  |
| --- |
| 1. /etc/exports : Its a main configuration file of NFS, all exported files and directories are defined in this file at the NFS Server end. 2. /etc/fstab : **To mount a NFS directory on your system across the reboots**, we need to make an entry in /etc/fstab. 3. /etc/sysconfig/nfs : Configuration file of NFS to control on which port rpc and other services are listening. |

## 4.Setup and Configure NFS Mounts on Linux Server

## 4A.Server Side

|  |
| --- |
| 1. **NFS Server**: nfsserver.example.com with IP-192.168.0.100 2. **NFS Client** : nfsclient.example.com with IP-192.168.0.101 |

### 4.1.Installing NFS Server and NFS Client

|  |
| --- |
| We need to install NFS packages **on all NFS Server and NFS Client machine**. We can install it via “yum” (Red Hat Linux) and “apt-get” (Debian and Ubuntu) package installers.  [root@nfsserver ~]# yum install nfs-utils nfs-utils-lib  [root@nfsserver ~]# yum install portmap (not required with NFSv4)  [root@nfsserver ~]# apt-get install nfs-utils nfs-utils-lib  [root@nfsserver ~]# rpm –qa | grep nfs => to check if packages was installed |

|  |
| --- |
| Now **start the services on both machines**.  [root@nfsserver ~]# /etc/init.d/portmap start  [root@nfsserver ~]# /etc/init.d/nfs start  [root@nfsserver ~]# chkconfig --level 35 portmap on  [root@nfsserver ~]# chkconfig --level 35 nfs on  After installing packages and starting services on both the machines, we need to configure both the machines for file sharing. |

### 4.2. Setting Up the NFS Server

|  |  |
| --- | --- |
| First we will be configuring the **NFS** server. **Configure Export directory**  |  | | --- | | => creating a new directory named “**nfsshare**” in “**/**” partition to share with **client server**, or can share an already existing directory with NFS.  [root@nfsserver ~]# mkdir /Server/shared/mount/point  => make an entry in “**/etc/exports**” and **use command exportfs –ra to take the change affect**. If still not ok restart service nfs  [root@nfsserver ~]# vi /etc/exports  /Server/shared/mount/point 192.168.0.101(rw,sync,no\_root\_squash)  #192.168.0.101: Client IP  [root@nfsserver ~]# **exportfs –ra => r: refresh server list; a: export all shared lists** | |

### NFS Options

|  |
| --- |
| 1. ro: With the help of this option we can provide read only access to the shared files i.e client will only be able to read. 2. rw: This option allows the client server to both read and write access within the shared directory. 3. sync: Sync confirms requests to the shared directory only once the changes have been committed. 4. no\_subtree\_check: This option prevents the subtree checking. When a shared directory is the subdirectory of a larger file system, nfs performs scans of every directory above it, in order to verify its permissions and details. Disabling the subtree check may increase the reliability of NFS, but reduce security. 5. no\_root\_squash: This phrase allows root to connect to the designated directory. |

## 4B.Client Side

|  |
| --- |
| After configuring the **NFS** server, we need to **mount** that shared directory or partition in the **client** server. |

### 4.1. Show Mount Shared Directories on NFS Client

|  |
| --- |
| Now at the NFS client end, we need to mount that directory in our server to access it locally. To do so, first we need to find out that shares available on the remote server or NFS Server.  [root@nfsclient ~]# showmount -e 192.168.0.100 => Server IP: 192.168.0.100  Export list for 192.168.0.100:  /Server/shared/mount/point 192.168.0.101  Above command shows that a directory named “nfsshare” is available at “192.168.0.100” to share with your server. |

### 4.2. Mount Shared NFS Directory

|  |
| --- |
| To mount that shared NFS directory we can use following mount command.  [root@nfsclient ~]# mount -t nfs 192.168.0.100:/Server/shared/mount/point /Client/mount/point  The above command will mount that shared directory in “/mnt/nfsshare” on the client server. You can verify it following command.  [root@nfsclient ~]# mount | grep nfs  sunrpc on /var/lib/nfs/rpc\_pipefs type rpc\_pipefs (rw)  nfsd on /proc/fs/nfsd type nfsd (rw)  192.168.0.100:/Server/shared/mount/point on /Client/mount/point type nfs (rw,addr=192.168.0.100)  The above mount command mounted the nfs shared directory on to nfs client temporarily, to mount an NFS directory permanently on your system across the reboots, we need to make an entry in “/etc/fstab“.  [root@nfsclient ~]# vi /etc/fstab  Add the following new line as shown below.  192.168.0.100:/Server/shared/mount/point /Client/mount/point nfs defaults 0 0 |

### 5.Result in client side:

|  |
| --- |
| Filesystem Size Used Avail Use% Mounted on  172.23.32.10:/hnas1-data01  200T 180T 21T 90% /NAS\_INGEST01  172.23.32.11:/hnas1-data02  200T 70T 131T 35% /NAS\_INGEST02  172.23.32.12:/hnas2-data01  200T 102T 99T 51% /NAS\_DIST01  172.23.32.13:/hnas2-data02  200T 107T 94T 54% /NAS\_DIST02 |