## How To Use Autofs To Mount NFS Shares In Linux – (autofs In Linux)

NFS shares can also be made persistent or mounted on the client by using automount (autofs services) which offers a better performance compared to using the fstab file.

Before we see the step by step guide of how to do this, let’s understand how automount works with NFS in Linux.

## How automount/autofs Works In Linux

NFS shares can automatically be mounted on the client whenever it is required (on demand) and be unmounted whenever it is inactive or no longer required with the use of the autofs service, hence offering better performance;

one of which is reducing the booting time of the system because the system won’t find the NFS entries in the fstab file while booting. More so, if the NFS server is down, it won’t affect the system from booting/entering a maintenance mode.

The autofs service needs to be installed, and then configured on the system by using a **master map file**, and in turn, the master map file will reference a **map file**.

The master map file can be defined in one of the autofs configuration files (/etc/auto.master).

However, it is recommended most times that the master map file is defined as a drop-in file in (/etc/auto.master.d) because the files in the main configuration file (/etc/auto.master) sometimes may be overwritten if an update is done on the system.

The drop-in file must have an extension of (.autofs) and the same format/syntax that is going to be used in the configuration file of (/etc/auto.master) will also be used in the drop-in file entry.

Let’ see a format of a master map file entry in (/etc/auto.master) configuration file.

[root@DRDEV1 ~]# cat /etc/auto.master

A screenshot of a computer program

Description automatically generated

Automount maps can be direct or indirect. Hence, In a master map file entry, you can configure a direct or an indirect map mount type.

For a direct map, the client path’s mount point specified in the **map file** will be absolute.

For an indirect map, the client path’s mount point specified in the map file will not be absolute. A parent directory will be specified in the **master map file** while a sub-directory will be specified in the **map file** the master map file is referencing.

We will see how to specify this syntax in the step by step guide.

## Step By Step Guide On How To Automount NFS Share In Linux – RHEL/CentOS 7&8

configure automount on NFS client (HQDEV1) using an indirect map and “/software/site1” as the mount point for the client. The NFS share is /softwaredownload

Similarly, use a direct map and “/direct/software” as the mount point for the client. The NFS share is /softwaredownload.

**1. Install autofs**

[root@HQDEV1 ~]# yum install autofs

Updating Subscription Management repositories.

Last metadata expiration check: 0:01:55 ago on Fri 18 Dec 2020 11:18:01 AM GMT.

Dependencies resolved.

Installing:

........

**2. verify if you can connect to the NFS server and see the export lists from the client**

[root@HQDEV1 ~]# showmount -e 192.168.170.172

Export list for 192.168.170.172:

/softwaredownload 192.168.170.171

/sofwaredownload 192.168.170.178,192.168.170.177,192.168.170.176,192.168.170.175,192.168.170.174

where 192.168.170.172 is the NFS server’s IP. The host name can also be used provided DNS resolution is working.

If you are having difficulties listing the exports list, you can test by mounting the shared directory on the client.

[root@HQDEV1 ~]# mount -t nfs 192.168.170.172:/ /mnt

NB: only the shared directories will be mounted and not the entire root filesystem.

[root@HQDEV1 ~]# cd /mnt/

[root@HQDEV1 mnt]# ls

softwaredownload

[root@HQDEV1 ~]# df -h |grep mnt

192.168.170.172:/softwaredownload 13G 5.9G 6.7G 48% /mnt

[root@HQDEV1 ~]# umount /mnt

**3. create a master map file**

**Note: (This step is for an Indirect mapping)**

create a drop in file in (/etc/auto.master.d)

NB: the file must have “.autofs” extension

[root@HQDEV1 ~]# vim /etc/auto.master.d/tekneed.autofs

/software /etc/auto.tekneed

NB: This entry in master map file **(/etc/auto.master.d)** tells autofs to look in the map file entry (**/etc/auto.tekneed**) and create mount-points in the **/**software directory if there is an activity in the /software directory.

**4. put in the entry for “/etc/auto.tekneed”**

NB2: tekneed.autofs must correspond with auto.tekneed. Whatever name you used as the master map file/drop-in file must correspond with the name that will be used in the drop-in file entry.

The entries here will reference step 3

[root@HQDEV1 ~]# vim /etc/auto.tekneed

site1 -rw,sync 192.168.170.172:/softwaredownload

The format for the entry of this file is:

<mount point> <-mount options> <server-location>

which is similar to the fstab file format. This creates a new mount point at “/software/site1” and mounts the NFS share directory (/softwaredownload) exported by the server (192.168.170.172) with read and write permission & with immediate synchronization.

For an indirect mapping, a parent directory will be specified in the master map file while a sub-directory will be specified in the map file. autofs mounts the shares from the parent directory to the sub-directory.

Also note that the permission on the NFS share supersedes/takes precedence over the one specified with the mount option. Other mount options just as it can be used in the fstab can also be used here.

**5. start and enable autofs**

[root@HQDEV1 ~]# systemctl enable --now autofs

**6. Test your configuration**

[root@HQDEV1 ~]# cd /software/site1

[root@HQDEV1 site1]# ls

image\_download iso\_download

### **(For a direct mapping)**

From step 3, the master map file will be created as

[root@HQDEV1 ~]# vim /etc/auto.master.d/tekneed.autofs

/- /etc/auto.tekneed

This entry in master map file (**/etc/auto.master.d)** tells autofs to look in the map file entry (**/etc/auto.tekneed)** and create mount-points in the absolute path specified in the entry for **/etc/auto.tekneed** file. That’s what direct map do.

4. **put in the entry for “/etc/auto.tekneed”**

[root@HQDEV1 ~]# vim /etc/auto.tekneed

/direct/software -rw,sync,fstype=nfs4 192.168.170.172:/softwaredownload

This creates a new mount point at /direct/software and mounts the NFS share directory (/softwaredownload) exported by the server (192.168.170.172) with read and write permission with immediate synchronization and with nfs version 4 filesystem type.

Other filesystem type can be specified as (-fstype=cifs, -fstype=xfs), etc.

**5. start and enable autofs**

[root@HQDEV1 ~]# systemctl enable --now autofs

NB: You may restart autofs if it is already started

**6. Test your configuration**.

[root@HQDEV1 ~]# cd /direct/software/

[root@HQDEV1 software]# ls

image\_download iso\_download