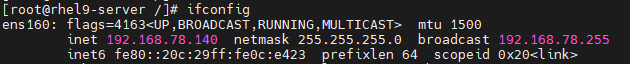
**AutoFS NFS**

For autofs lab setup, I am taking one server & one client as shown-



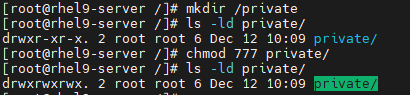


**At Server Side:**

1. Next, we will install nfs packages in server using below command-

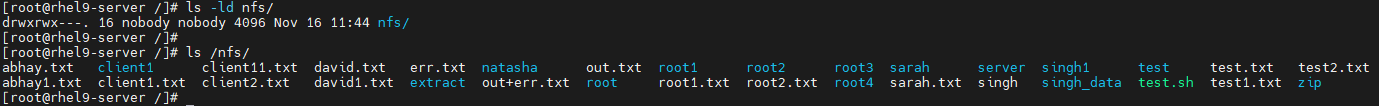
yum install nfs\* -y

2. We will create first directory “private” with 777 permission & add some content in it to share with client-

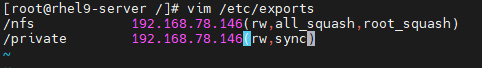




3. Second directory “nfs” with 770 permission-



4. Now we will add entries for these two directories in /etc/exports-



5. Then we will exports these two share-

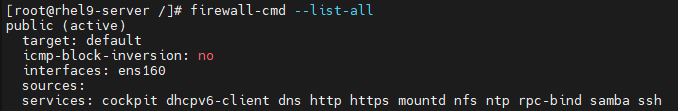


6. Next, we will add nfs, rpc-bind & mountd service in firewall as –

firewall-cmd --permanent --add-service={nfs,rpc-bind,mountd}

firewall-cmd --reload

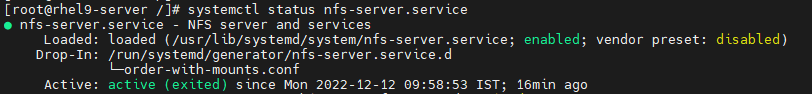
7. We can verify these as-



8. Start & enable nfs-server service-

systemctl enable nfs --now

9. Check nfs-server service is up & running-

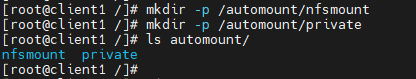


**At Client Side:**

1. First, we will install few packages at client-

yum install nfs-utils nfs4-acl-tools autofs -y

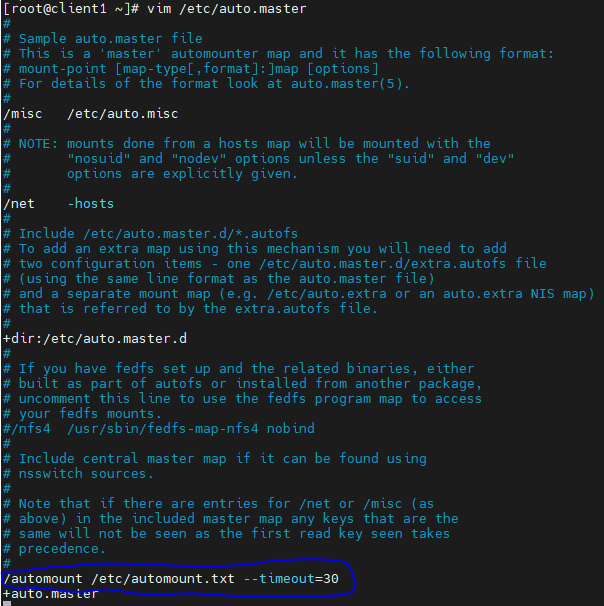
2. Next, we will create two directories inside parent directory as shown-



Here automount directory (parent) will be used for autofs. Shares will be mounted inside this directory.

nfsmount will be used for nfs share & private will be used for private share.

3. To create autofs, we will edit auto.master file inside /etc as highlighed-



Here, we will mention that parent directory used for autofs at client side. It will be used to mount shared directories.

Timeout is 30 seconds. After this time duration, shared directory will be unmount automatically if not in use (If we are outside this created child directory at client i.e nfsmount & private).

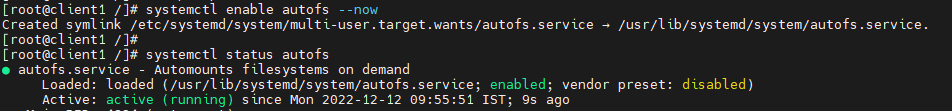
4. Next, we will create automount.txt in /etc. It will contain detail of exported file system-



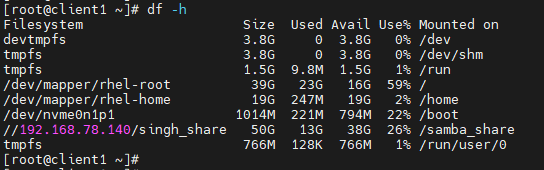
Here, private & nfsmount are client’s child directories inside parent directory automount on which corresponding server’s share will be mounted.

After that, we provide server’s shared directories location.

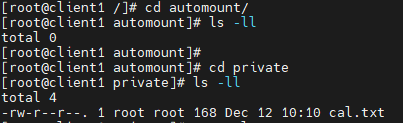
5. Start & enable autofs service & verify it-



6. Now, we will verify for current mounted directory on this client before mounting these new shares-

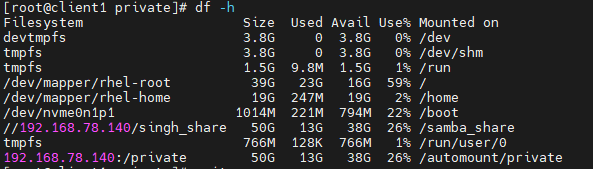


7. Next, go to parent directory automount & go inside child directories to see whether we are able to see shared files/directories by server-



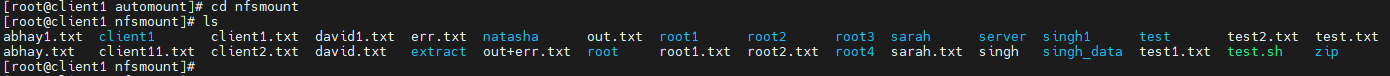
Here, cal.txt is visible which was created at server side inside their private directory.

8. Now, we will verify this mount point at client-

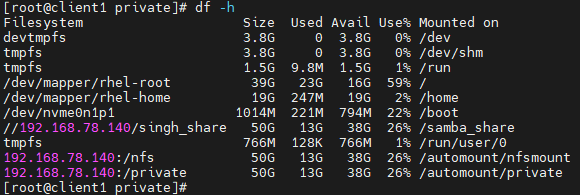


Here it is showing as mounted in last line.

9. In the same way, we will check for another share-

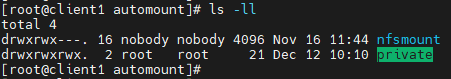


10. Now verify all new mount points-

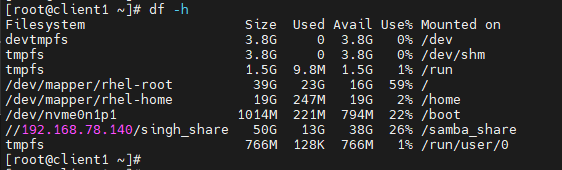


Last two lines shows our required mount point.

11. If we long list parent automount directory, we will see both child directories which with server’s shared directory permission-



12. After 30 seconds of timeout duration, these mounted directories will be unmounted if not in use-

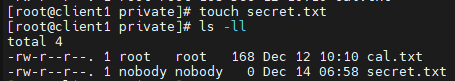


13. We can verify the same using long listing parent automount directory-

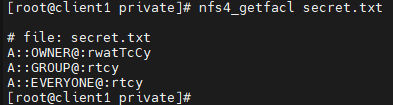


**ACL on NFS-Share:**

1. First create any file/dir on which we will play ACL-



2. Check current ACL applied on this secret.txt file-



3. We want to give read, write & execute access to abhay user at this file. So first check its user id-

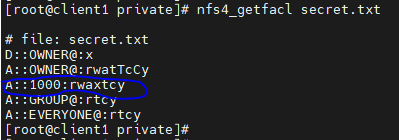


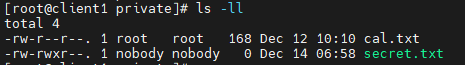
Initially, we will not be able to edit this file by abhay user as it has read-only access.

4. Next, set ACL permission for abhay user-



5. Verify this ACL-





Now Abhay will be able to do anything with this file.

6. To remove ACL, we will use either of two ways-

(i)

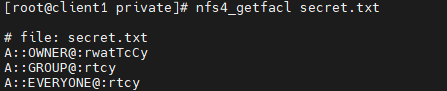


It will get open in vim editor. We will simply delete that line.

(ii)



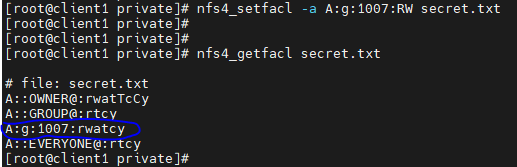
7. Verify the same-



8. If we want to give read, write access to cricbuzz group, then first check group id-



9. Next, add ACL for this group & verify-



10. Now remove ACL permission from group in any of the two ways-

