Dernière mise-à-jour : 2024/11/21 15:28

RH13408 - Gestion du Stockage en Réseau

Contenu du Module

- **RH13408 - Gestion du Stockage en Réseau**
  - Contenu du Module
  - Présentation du Network File System (NFS)
    - NFSv3
      - Les Services et Processus Principaux du Serveur NFSv3
      - Options d'un Partage NFSv3
      - Commandes de Base de NFSv3
    - NFSv4
      - Différences entre NFSv4 et NFSv3
  - LAB #1 - Configuration du Serveur NFSv4
    - 1.1 - Désactivation de NFSv3
    - 1.2 - Définition d'un Partage
  - LAB #2 - Configuration du Client
    - 2.1 - Montage Éphémère
    - 2.2 - Montage Persistant
  - LAB #3 - Configuration de NFSv4 avec Automounter
    - Présentation
    - Le Service autofs
    - 3.1 - Création d'un Fichier de Mappage Indirect
      - Configuration du Serveur
      - Configuration du Client
    - 3.2 - Création d'un Fichier de Mappage Indirect
      - Configuration du Serveur
      - Configuration du Client

Présentation du Network File System (NFS)

Le Network File System (NFS) est un protocole Internet standard que Linux, UNIX et d'autres systèmes d'exploitation similaires utilisent comme système de fichiers réseau natif. NFS est une norme ouverte qui prend en charge les permissions Linux natives et les attributs du système de fichiers.

Par défaut, Red Hat 9 utilise la version 4.2 de NFS. RHEL prend entièrement en charge les protocoles NFSv3 et NFSv4. NFSv3 peut utiliser un protocole de transport TCP ou UDP, mais NFSv4 n'autorise que les connexions TCP.

Les serveurs NFS exportent des répertoires. Les clients NFS montent les répertoires exportés dans un répertoire local existant. Les clients NFS peuvent monter les répertoires exportés de plusieurs manières :

- Manuellement en utilisant la commande mount,
- De manière persistante au démarrage en configurant des entrées dans le fichier /etc/fstab,
- À la demande, en configurant une méthode de d'Automount.

Les méthodes d'Automount, comprennent :

- le service autofs,
- systemd.automount.

Il convient d'installer le paquet **nfs-utils** afin d'obtenir les outils clients pour le montage manuel, ou pour le montage automatique, des répertoires exportés.

Red Hat 9 prend également en charge le montage de répertoires partagés à partir de systèmes Microsoft Windows(R) en utilisant les mêmes méthodes que pour le protocole NFS, en utilisant les protocoles **Server Message Block** (SMB) ou **Common Internet File System** (CIFS). Les options de montage sont spécifiques au protocole et dépendent de la configuration du Windows ou du Samba.

NFSv3

Le protocole NFS a changé de manière significative entre NFSv3 et NFSv4. La méthode d'interrogation pour afficher les exportations disponibles est différente pour chaque version du protocole. NFSv3 utilise le protocole RPC, qui nécessite un serveur de fichiers prenant en charge les connexions NFSv3 pour exécuter le service rpcbind. Un client NFSv3 se connecte au service rpcbind sur le port 111 du serveur pour demander le service NFS. Le serveur répond en indiquant le port actuel du service NFS. La commande **showmount** est utilisée pour interroger les exportations disponibles sur un

serveur NFSv3 basé sur RPC.

Les Services et Processus Principaux du Serveur NFSv3

La version NFSv3 utilise les services suivants :

Services	Fonction
nfsd	Démarre le service NFS ainsi que les processus RPC pour recevoir et traiter les demandes des clients
nfslock	Démarre les processus RPC qui permettent aux clients de verrouiller les fichiers sur le serveur
portmap	Gestion des réservations des ports pour les services RPC locaux afin que les services RPC distants puissent se connecter

Options d'un Partage NFSv3

Certaines options, appliquées à un partage, modifient le comportement du serveur NFSv3 pour le partage concerné lors de son démarrage :

Option	Comportement
ro	Accès en lecture seule
rw	Accès en lecture / écriture
sync	Ecriture synchrone (écriture immédiate sur disque)
async	Ecriture asynchrone (écriture sur disque en utilisant une cache)
root_squash	Root perd ses prérogatives sur le partage concerné
no_root_squash	Root garde ses prérogatives sur le partage concerné
no_lock	Pas de verrous sur les fichiers accédés
all_squash	Force la mapping de tous les utilisateurs vers l'utilisateur **nobody**
anonuid	Fixe l'UID de l'utilisateur anonyme
anongid	Fixe le GID de l'utilisateur anonyme

Important : Si plusieurs options sont spécifiées, celles-ci doivent être séparées par des virgules.

Commandes de Base de NFSv3

Plusieurs commandes permettent de gérer et de s'informer sur l'activité du serveur NFSv3 :

Commande	Comportement
exportfs	Affiche les partages actifs sur le serveur courant
nfsstat	Affiche les statistiques de l'activité NFS
rpcinfo	Affiche les démons gérés en effectuant une requête RPC sur le serveur courant
showmount	Affiche les partages actifs sur un serveur distant
mount	Permet de monter un partage distant sur un répertoire local

NFSv4

Différences entre NFSv4 et NFSv3

Le protocole NFSv4 a éliminé l'utilisation de l'ancien protocole RPC pour les transactions NFS. L'utilisation de la commande showmount sur un serveur qui ne prend en charge que NFSv4 s'interrompt sans recevoir de réponse car le service rpcbind n'est pas exécuté sur le serveur. Cependant, l'interrogation d'un serveur NFSv4 est plus simple que d'interroger un serveur NFSv3.

NFSv4 a introduit une arborescence d'exportation qui contient tous les chemins d'accès aux répertoires exportés par le serveur. Pour afficher tous les répertoires exportés, il convient de monter la racine (/) de l'arborescence d'exportation du serveur. Le montage de la racine de l'arborescence d'exportation permet de parcourir les chemins d'accès de tous les répertoires exportés, en tant qu'enfants du répertoire racine de l'arborescence mais ne monte aucun des répertoires exportés.

Le format de cette commande est le suivant :

```
# mkdir /mountpoint

# mount -t nfs -o rw,sync server:/export /mountpoint
```

Important : L'option **-o sync** spécifie que toutes les transactions vers le système de fichiers exporté sont effectuées de manière synchrone, ce qui est fortement recommandée pour tous les montages réseau de production où les transactions doivent réussir sous peine d'être renvoyées en cas d'échec. Notez que seul, root, peut procéder aux montages.

Pour monter une exportation NFSv4 tout en parcourant l'arborescence de l'exportation montée, il convient de changer de répertoire pour prendre un chemin d'accès au répertoire exporté. Il est aussi possible utiliser la commande mount avec le nom complet du chemin d'accès d'un répertoire exporté pour monter un seul répertoire exporté.

A noter que, les répertoires exportés qui utilisent la sécurité Kerberos n'autorisent pas le montage ou l'accès à un répertoire lorsque vous parcourez une arborescence d'exportation, même si vous pouvez voir le chemin d'accès à l'exportation. Le montage de partages protégés par Kerberos nécessite une configuration supplémentaire du serveur et l'utilisation des informations d'identification de l'utilisateur Kerberos. Ce cas, ne fait pas partie de la certification RH134 mais est couvert dans la formation **RH362 - Red Hat Security: Identity Management and Active Directory Integration**.

LAB #1 - Configuration du Serveur NFSv4

Vérifiez l'état des machines virtuelles :

```
[root@redhat9 ~]# virsh list --all
 Id   Name      State
-------------------------
 5    testvm1   running
 8    testvm2   running
```

Connectez-vous à machine virtuelle **testvm2** et activez et démarrez le service **nfs-server** :

```
[root@redhat9 ~]# virsh console testvm2
Connected to domain 'testvm2'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm2 ~]# systemctl enable --now nfs-server
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/nfs-server.service → /usr/lib/systemd/system/nfs-
server.service.
[root@testvm2 ~]# systemctl status nfs-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; prese>
     Active: active (exited) since Mon 2024-10-28 12:08:12 CET; 3s ago
       Docs: man:rpc.nfsd(8)
             man:exportfs(8)
    Process: 5309 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUC>
    Process: 5310 ExecStart=/usr/sbin/rpc.nfsd (code=exited, status=0/SUCCESS)
    Process: 5329 ExecStart=/bin/sh -c if systemctl -q is-active gssproxy; then>
   Main PID: 5329 (code=exited, status=0/SUCCESS)
        CPU: 31ms

Oct 28 12:08:12 testvm2.ittraining.loc systemd[1]: Starting NFS server and serv>
Oct 28 12:08:12 testvm2.ittraining.loc systemd[1]: Finished NFS server and serv>
```

1.1 - Désactivation de NFSv3

Le serveur NFS fonctionne en mode mixte NFSv3/NFSv4. Ceci est visible grâce à la sortie de la commande **nfsstat** :

```
[root@testvm2 ~]# nfsstat
Server rpc stats:
calls      badcalls   badfmt     badauth    badclnt
0          0          0          0          0
```

Pour désactiver le serveur NFSv3, éditez la section **[nfsd]** du fichier **/etc/nfs.conf** :

```
[root@testvm2 ~]# cat /etc/nfs.conf
...
[nfsd]
# debug=0
# threads=8
```

```
# host=
# port=0
# grace-time=90
# lease-time=90
# udp=n
# tcp=y
# vers3=y
# vers4=y
# vers4.0=y
# vers4.1=y
# vers4.2=y
rdma=y
rdma-port=20049
...
```

```
[root@testvm2 ~]# vi /etc/nfs.conf
[root@testvm2 ~]# cat /etc/nfs.conf
...
[nfsd]
# debug=0
# threads=8
# host=
# port=0
# grace-time=90
# lease-time=90
# udp=n
# tcp=y
vers3=n
# vers4=y
# vers4.0=y
# vers4.1=y
vers4.2=y
rdma=y
rdma-port=20049
```

```
...
```

Désactivez tous les services NFSv3 :

```
[root@testvm2 ~]# systemctl mask --now rpc-statd.service rpcbind.service rpcbind.socket
Created symlink /etc/systemd/system/rpc-statd.service → /dev/null.ervice rpcbind.socket
Created symlink /etc/systemd/system/rpcbind.service → /dev/null.
Created symlink /etc/systemd/system/rpcbind.socket → /dev/null.
```

Redémarrez le service **nfs-server** :

```
[root@testvm2 ~]# systemctl restart nfs-server
```

Vérifiez la désactivation de NFSv3 en consultant le fichier **/proc/fs/nfsd/versions** :

```
[root@testvm2 ~]# cat /proc/fs/nfsd/versions
-3 +4 +4.1 +4.2
```

Créez ensuite le fichier **/etc/systemd/system/nfs-mountd.service.d/v4only.conf** :

```
[root@testvm2 ~]# mkdir /etc/systemd/system/nfs-mountd.service.d

[root@testvm2 ~]# vi /etc/systemd/system/nfs-mountd.service.d/v4only.conf

[root@testvm2 ~]# cat /etc/systemd/system/nfs-mountd.service.d/v4only.conf
[Service]
ExecStart=
ExecStart=/usr/sbin/rpc.mountd --no-tcp --no-udp
```

Rechargez le service **nfs-mountd** :

```
[root@testvm2 ~]# systemctl daemon-reload
```

```
[root@testvm2 ~]# systemctl restart nfs-mountd

[root@testvm2 ~]# systemctl status nfs-mountd
● nfs-mountd.service - NFS Mount Daemon
     Loaded: loaded (/usr/lib/systemd/system/nfs-mountd.service; static)
    Drop-In: /etc/systemd/system/nfs-mountd.service.d
             └─v4only.conf
     Active: active (running) since Mon 2024-10-28 12:44:22 CET; 10s ago
       Docs: man:rpc.mountd(8)
    Process: 1410 ExecStart=/usr/sbin/rpc.mountd --no-tcp --no-udp (code=exited>
   Main PID: 1412 (rpc.mountd)
      Tasks: 1 (limit: 11096)
     Memory: 940.0K
        CPU: 6ms
     CGroup: /system.slice/nfs-mountd.service
             └─1412 /usr/sbin/rpc.mountd --no-tcp --no-udp

Oct 28 12:44:22 testvm2.ittraining.loc systemd[1]: Starting NFS Mount Daemon...
Oct 28 12:44:22 testvm2.ittraining.loc rpc.mountd[1410]: mountd: No V2 or V3 li>
Oct 28 12:44:22 testvm2.ittraining.loc rpc.mountd[1412]: Version 2.5.4 starting
Oct 28 12:44:22 testvm2.ittraining.loc systemd[1]: Started NFS Mount Daemon.
```

1.2 - Définition d'un Partage

Créez le répertoire **/mountpoint** :

```
[root@testvm2 ~]# mkdir /mountpoint
```

Modifez les permissions et le groupe du répertoire :

```
[root@testvm2 ~]# chmod 2770 /mountpoint

[root@testvm2 ~]# chgrp users /mountpoint
```

```
[root@testvm2 ~]# ls -ld /mountpoint/
drwxrws---. 2 root users 6 Oct 28 13:05 /mountpoint/
```

Créez ensuite les fichiers **test1.txt** et **test2.txt** dans le répertoire **/mountpoint** :

```
[root@testvm2 mountpoint1]# cd /mountpoint

[root@testvm2 mountpoint]# touch test1.txt test2.txt

[root@testvm2 mountpoint1]# cd ~
```

Editez ensuite le fichier **/etc/exports** afin d'exporter le répertoire **/mountpoint** :

```
[root@testvm2 ~]# vi /etc/exports

[root@testvm2 ~]# cat /etc/exports
/mountpoint     192.168.56.0/24(rw)     2001:db8::/32(rw)
```

Configurez SELinux en mode **permissive** et arrêtez le service **firewalld** :

```
[root@testvm2 ~]# setenforce permissive

[root@testvm2 ~]# systemctl stop firewalld
```

Redémarrez le service **nfs-server** :

```
[root@testvm2 ~]# systemctl restart nfs-server

[root@testvm2 ~]# systemctl status nfs-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; prese>
     Active: active (exited) since Mon 2024-10-28 13:10:35 CET; 10s ago
       Docs: man:rpc.nfsd(8)
             man:exportfs(8)
```

```
    Process: 1475 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUC>
    Process: 1476 ExecStart=/usr/sbin/rpc.nfsd (code=exited, status=0/SUCCESS)
    Process: 1486 ExecStart=/bin/sh -c if systemctl -q is-active gssproxy; then>
   Main PID: 1486 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Oct 28 13:10:34 testvm2.ittraining.loc systemd[1]: Starting NFS server and serv>
Oct 28 13:10:35 testvm2.ittraining.loc systemd[1]: Finished NFS server and serv>
```

Ajoutez le groupe et l'utilisateur **trainee** :

```
[root@testvm2 ~]# groupadd trainee && useradd trainee -c Trainee -d /home/trainee -g trainee -G users -s
/bin/bash
```

Définissez le mot de passe **trainee** pour l'utilisateur **trainee** :

```
[root@testvm2 ~]# passwd trainee
Changement de mot de passe pour l'utilisateur trainee.
Nouveau mot de passe : trainee
MOT DE PASSE INCORRECT : Le mot de passe comporte moins de 8 caractères
Retapez le nouveau mot de passe : trainee
passwd : mise à jour réussie de tous les jetons d'authentification.
```

LAB #2 - Configuration du Client NFSv4

Déconnectez-vous de la machine virtuelle **testvm2** :

```
[root@testvm2 ~]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm1** :

```
[root@redhat9 ~]# virsh console testvm1
Connected to domain 'testvm1'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm1 /]#
```

Installez le paquet **nfs-utils** :

```
[root@testvm1 /]# dnf install -y nfs-utils
```

Ajoutez le groupe et l'utilisateur **trainee** :

```
[root@testvm1 /]# groupadd trainee && useradd trainee -c Trainee -d /home/trainee -g trainee -G users -s
/bin/bash
```

Définissez le mot de passe **trainee** pour l'utilisateur **trainee** :

```
[root@testvm1 /]# passwd trainee
Changement de mot de passe pour l'utilisateur trainee.
Nouveau mot de passe : trainee
MOT DE PASSE INCORRECT : Le mot de passe comporte moins de 8 caractères
Retapez le nouveau mot de passe : trainee
passwd : mise à jour réussie de tous les jetons d'authentification.
```

2.1 - Montage Ephémère

Montez le partage **192.168.56.100:/mountpoint** sur le répertoire **/mnt** :

```
[root@testvm1 /]# mount -t nfs -o rw,sync 192.168.56.100:/mountpoint /mnt
[172524.919003] FS-Cache: Loaded
[172525.206140] Key type dns_resolver registered
[172525.606078] NFS: Registering the id_resolver key type
```

```
[172525.606089] Key type id_resolver registered
[172525.606090] Key type id_legacy registered
```

Vérifiez que le montage a été effectué :

```
[root@testvm1 /]# mount | grep mountpoint
192.168.56.100:/mountpoint on /mnt type nfs4
(rw,relatime,sync,vers=4.2,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,client
addr=192.168.56.50,local_lock=none,addr=192.168.56.100)
```

Essayez de vous placer dans le répertoire **/mnt** :

```
[root@testvm1 /]# ls -l /mnt
ls: impossible d'ouvrir le répertoire '/mnt': Permission non accordée
```

L'utilisateur **root** ne peut pas se placer dans /mnt parce que root ne fait pas partie du groupe **users**. Devenez donc l'utilisateur **trainee** et lister le contenu du répertorie **/mnt** :

```
[root@testvm1 /]# su - trainee

[trainee@testvm1 ~]$ ls -l /mnt
total 0
-rw-r--r--. 1 root users 0 28 oct.  13:59 test1.txt
-rw-r--r--. 1 root users 0 28 oct.  13:59 test2.txt
```

Créez maintenant le fichier **/mnt/test3.txt** :

```
[trainee@testvm1 ~]$ touch /mnt/test3.txt

[trainee@testvm1 ~]$ ls -l /mnt
total 0
-rw-r--r--. 1 root    users 0 28 oct.  13:59 test1.txt
-rw-r--r--. 1 root    users 0 28 oct.  13:59 test2.txt
```

```
-rw-r--r--. 1 trainee users 0 28 oct.  14:06 test3.txt
```

2.2 - Montage Permanent

Démontez le point de montage **/mnt** :

```
[root@testvm1 /]# umount /mnt

[root@testvm1 /]# mount | grep mountpoint

[root@testvm1 /]# ls /mnt
```

Editez ensuite le fichier **/etc/fstab** en y ajoutant la ligne **192.168.56.100:/mountpoint /mnt nfs rw,sync 0 0** :

```
[root@testvm1 /]# vi /etc/fstab

[root@testvm1 /]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Fri Oct 25 15:41:24 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=e778bb41-0528-4e4b-9b6b-b80070117978 /                       xfs     defaults        0 0
UUID=6d55a589-bef2-4a9b-941f-d2f288057104 /boot                   xfs     defaults        0 0
UUID=11306088-b22e-4906-9c31-b2f1a8b416fd none                    swap    defaults        0 0
192.168.56.100:/mountpoint      /mnt    nfs     rw,sync 0 0
```

Exécutez la commande **systemctl daemon-reload** pour une prise en compte des modifications :

```
[root@testvm1 /]# systemctl daemon-reload
[175309.751966] systemd-rc-local-generator[4833]: /etc/rc.d/rc.local is not marked executable, skipping.
```

Appliquez maintenant le fichier **/etc/fstab** :

```
[root@testvm1 /]# mount -a

[root@testvm1 /]# mount | grep mountpoint
192.168.56.100:/mountpoint on /mnt type nfs4
(rw,relatime,sync,vers=4.2,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,client
addr=192.168.56.50,local_lock=none,addr=192.168.56.100)
```

Devenez l'utilisateur **trainee** et vérifiez la présence des trois fichiers :

```
[root@testvm1 /]# su - trainee
Dernière connexion : lundi 28 octobre 2024 à 14:04:58 CET sur ttyS0

[trainee@testvm1 ~]$ ls -l /mnt
total 0
-rw-r--r--. 1 root    users 0 28 oct.  13:59 test1.txt
-rw-r--r--. 1 root    users 0 28 oct.  13:59 test2.txt
-rw-r--r--. 1 trainee users 0 28 oct.  14:06 test3.txt

[trainee@testvm1 ~]$ exit
déconnexion
```

Déconnectez-vous de la machine virtuelle **testvm1** :

```
[root@testvm1 /]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm2** :

```
[root@redhat9 ~]# virsh console testvm2
Connected to domain 'testvm1'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm2 ~]#
```

Vérifiez la présence des trois fichiers dans le répertoire **/mountpoint** :

```
[root@testvm2 ~]# ls -l /mountpoint
total 0
-rw-r--r--. 1 root    users 0 Oct 28 13:59 test1.txt
-rw-r--r--. 1 root    users 0 Oct 28 13:59 test2.txt
-rw-r--r--. 1 trainee users 0 Oct 28 14:06 test3.txt
```

LAB #3 - Configuration de NFSv4 avec Automounter

Présentation

L'**Automounter** est un service,appelé **autofs**, qui monte automatiquement les systèmes de fichiers et les exportations NFS à la demande et démonte automatiquement les systèmes de fichiers et les exportations NFS lorsque les ressources montées ne sont plus utilisées.

Automounter a été créée pour résoudre le problème des utilisateurs non privilégiés qui ne disposent pas des autorisations suffisantes pour utiliser la commande **mount**. Sans l'utilisation de la commande mount,les utilisateurs normaux ne peuvent pas accéder aux supports amovibles tels que les CD, les DVD et les lecteurs de disques amovibles.

En outre, si un système de fichiers local ou distant n'est pas monté au démarrage à l'aide de la configuration **/etc/fstab**, un utilisateur normal ne peut pas monter et accéder à ces systèmes de fichiers non montés.

Les fichiers de configuration d'Automounter contiennent des informations sur le montage du système de fichiers, de la même manière que les entrées du fichier /etc/fstab.

Bien que les systèmes de fichiers spécifiés dans /etc/fstab se montent au démarrage du système restent montés jusqu'à l'arrêt du système ou toute

autre intervention, les systèmes de fichiers contrôlés par l'Automounter se montent à la demande, lorsqu'un utilisateur ou une application tente d'entrer dans le point de montage du système de fichiers pour accéder aux fichiers.

L'utilisation des ressources pour les systèmes de fichiers Automounter est équivalente à celle des systèmes de fichiers montés au démarrage car un système de fichiers n'utilise des ressources que lorsqu'un programme lit et écrit des fichiers ouverts. Les systèmes de fichiers montés mais inactifs et les systèmes de fichiers non montés utilisent la même quantité de ressources : presque aucune.

L'avantage de l'Automounter est qu'en démontant le système de fichiers chaque fois qu'il n'est plus utilisé, le système de fichiers est protégé contre toute corruption inattendue,

Lorsque le système de fichiers est à nouveau monté, le service autofs utilise la configuration de montage la plus récente, contrairement à un montage dans le fichier /etc/fstab, qui peut encore utiliser une configuration montée il y a plusieurs mois lors du dernier démarrage du système. En outre, si la configuration de serveur NFS comprend des serveurs et des chemins d'accès redondants, l'Automounter peut utiliser la configuration de montage la plus récente.

Le Service autofs

Le service autofs prend en charge les mêmes systèmes de fichiers locaux et distants que le fichier /etc/fstab, y compris les protocoles de partage de fichiers NFS et SMB, et prend en charge les mêmes options de montage spécifiques au protocole, y compris les paramètres de sécurité. Les systèmes de fichiers montés par l'intermédiaire de l'Automounter sont accessibles par défaut à tous les utilisateurs, mais peuvent être restreints par des options d'autorisation d'accès.

Comme l'Automounter est une configuration côté client qui utilise les commandes standard **mount** et **umount** pour gérer les systèmes de fichiers, les systèmes de fichiers montés automatiquement présentent un comportement identique aux systèmes de fichiers montés à l'aide du fichier **/etc/fstab**.

La différence est qu'un système de fichiers Automounter reste démonté jusqu'à ce que l'on accède au point de montage, ce qui entraîne le montage immédiat du système de fichiers, qui reste monté tant que le système de fichiers est en cours d'utilisation. Lorsque tous les fichiers du système de sont fermés et que tous les utilisateurs et processus quittent le répertoire du point de montage, Automounter démonte le système de fichiers après un délai minimal.

3.1 - Création d'un Fichier de Mappage Indirect

On parle de montage indirect lorsque l'emplacement du point de montage n'est pas connu tant que la demande de montage n'a pas eu lieu. Un exemple de montage indirect est la configuration des répertoires personnels montés à distance, où le répertoire personnel d'un utilisateur inclut son nom d'utilisateur dans le chemin d'accès au répertoire. Le système de fichiers distant de l'utilisateur est monté sur son répertoire personnel, uniquement après qu'Automounter a appris quel utilisateur a spécifié de monter son répertoire personnel, et qu'il a déterminé que le système de fichiers distant est monté sur son répertoire personnel. Bien que des points de montage indirects semblent exister, le service autofs les crée lorsque la demande de montage se produit et les supprime à nouveau lorsque la demande est terminée et que le système de fichiers est démonté.

Configuration du Serveur

Editez le fichier **/etc/exports** :

```
[root@testvm2 ~]# vi /etc/exports

[root@testvm2 ~]# cat /etc/exports
/mountpoint     192.168.56.0/24(rw)     2001:db8::/32(rw)
/home           192.168.56.0/24(rw)     2001:db8::/32(rw)
```

Redémarrez le service **nfs-server** :

```
[root@testvm2 ~]# systemctl restart nfs-server
```

Devenez l'utilisateur **trainee** et créez le fichier **test4.txt** :

```
[root@testvm2 ~]# su - trainee
Last login: Tue Oct 29 13:56:07 CET 2024 on ttyS0

[trainee@testvm2 ~]$ touch test4.txt
```

Configuration du Client

Déconnectez-vous de la machine virtuelle **testvm2** :

```
[root@testvm2 ~]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm1** :

```
[root@redhat9 ~]# virsh console testvm1
Connected to domain 'testvm1'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm1 /]#
```

Configurez SELinux en mode **permissive**

```
[root@testvm1 /]# setenforce permissive
```

Arrêtez le service **firewalld** :

```
[root@testvm1 /]# systemctl stop firewalld
```

Installez le paquet **autofs** :

```
[root@testvm1 /]# dnf install autofs -y
```

Créez ensuite le fichier de mappage indirect maître **/etc/auto.master.d/home.autofs** :

```
[root@testvm1 /]# vi /etc/auto.master.d/home.autofs

[root@testvm1 /]# cat /etc/auto.master.d/home.autofs
/home   /etc/auto.home
```

Créez le fichier de mappage **/etc/auto.home** :

```
[root@testvm1 /]# vi /etc/auto.home
```

```
[root@testvm1 /]# cat /etc/auto.home
*       -fstype=nfs4,rw,sync    192.168.56.100:/home/&
```

Important : Notez que le format du nom du fichier de mappage maître est **nom**.autofs et que le format du nom du fichier de mappage est auto.**nom**. Les deux valeurs de **nom** doivent être identique.

Afin d'éviter des erreurs liées à la non-implémentation de **sss**, supprimer **sss** de la ligne **automount** du fichier **/etc/nsswitch.conf** :

```
[root@testvm1 /]# vi /etc/nsswitch.conf

[root@testvm1 /]# cat /etc/nsswitch.conf
...
# In order of likelihood of use to accelerate lookup.
passwd:     sss files systemd
shadow:     files
group:      sss files systemd
hosts:      files dns myhostname
services:   files sss
netgroup:   sss
automount:  files
...
```

Activez et démarrez le service **autofs** :

```
[root@testvm1 /]# systemctl enable --now autofs

[root@testvm1 /]# systemctl status autofs
● autofs.service - Automounts filesystems on demand
     Loaded: loaded (/usr/lib/systemd/system/autofs.service; enabled; preset: d>
     Active: active (running) since Tue 2024-10-29 13:55:10 CET; 19min ago
```

```
   Main PID: 3425 (automount)
      Tasks: 7 (limit: 23172)
     Memory: 6.1M
        CPU: 68ms
     CGroup: /system.slice/autofs.service
             └─3425 /usr/sbin/automount --systemd-service --dont-check-daemon

oct. 29 13:55:10 testvm1.ittraining.loc systemd[1]: Starting Automounts filesys>
oct. 29 13:55:10 testvm1.ittraining.loc systemd[1]: Started Automounts filesyst>
```

Vérifiez la prise en compte de la configuration :

```
[root@testvm1 /]# mount | tail
none on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs
(ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/vda1 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
none on /run/credentials/systemd-tmpfiles-setup.service type ramfs
(ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
192.168.56.100:/mountpoint on /mnt type nfs4
(rw,relatime,sync,vers=4.2,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,client
addr=192.168.56.50,local_lock=none,addr=192.168.56.100)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=164980k,nr_inodes=41245,mode=700,inode64)
/etc/auto.misc on /misc type autofs
(rw,relatime,fd=6,pgrp=3425,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=35290)
-hosts on /net type autofs (rw,relatime,fd=9,pgrp=3425,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=33681)
/etc/auto.home on /home type autofs
(rw,relatime,fd=12,pgrp=3425,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=35294)
```

Devenez l'utilisateur **trainee** :

```
[root@testvm1 /]# su - trainee
```

```
Dernière connexion : mardi 29 octobre 2024 à 13:55:17 CET sur ttyS0
```

Exécutez la commande **ls**. Vous devez voir le fichier **test4.txt** qui a été créé dans la machine virtuelle **testvm2** :

```
[trainee@testvm1 ~]$ ls
test4.txt

[trainee@testvm1 ~]$ exit
```

Déconnectez-vous de la machine virtuelle **testvm1** :

```
[root@testvm1 /]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm2** :

```
[root@redhat9 ~]# virsh console testvm2
Connected to domain 'testvm2'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm2 ~]#
```

Devenez l'utilisateur **trainee** et créez le fichier **test5.txt** :

```
[root@testvm2 ~]# su - trainee
Last login: Tue Oct 29 13:56:07 CET 2024 on ttyS0

[trainee@testvm2 ~]$ touch test5.txt

[trainee@testvm2 ~]$ ls
test4.txt  test5.txt
```

Déconnectez-vous de la machine virtuelle **testvm2** :

```
[root@testvm2 ~]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm1** :

```
[root@redhat9 ~]# virsh console testvm1
Connected to domain 'testvm1'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm1 /]#
```

Devenez l'utilisateur **trainee** et exécutez la commande **ls**. Vous devez voir le fichier **test5.txt** qui a été créé dans la machine virtuelle **testvm2** :

```
[trainee@testvm1 ~]$ ls
test4.txt  test5.txt

[trainee@testvm1 ~]$ exit
```

3.2 - Création d'un Fichier de Mappage Direct

On parle de montage direct lorsqu'un système de fichiers est monté sur un point de montage immuable et connu. Un point de montage direct existe en tant que répertoire permanent, comme les autres répertoires normaux.

Déconnectez-vous de la machine virtuelle **testvm1** :

```
[root@testvm1 /]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm2** :

```
[root@redhat9 ~]# virsh console testvm2
Connected to domain 'testvm2'
```

```
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm2 ~]#
```

Configuration du Serveur

Éditez le fichier **/etc/exports** en y ajoutant la ligne **/mountpoint1 192.168.56.0/24(rw) 2001:db8::/32(rw)** :

```
[root@testvm2 ~]# vi /etc/exports

[root@testvm2 ~]# cat /etc/exports
/mountpoint     192.168.56.0/24(rw)     2001:db8::/32(rw)
/home           192.168.56.0/24(rw)     2001:db8::/32(rw)
/mountpoint1    192.168.56.0/24(rw)     2001:db8::/32(rw)
```

Redémarrez le service **nfs-server** :

```
[root@testvm2 ~]# systemctl restart nfs-server
```

Créez le répertoire **/mountpoint1** et modifiez les permissions :

```
[root@testvm2 ~]# mkdir /mountpoint1

[root@testvm2 ~]# chmod 777 /mountpoint1
```

Configuration du Client

Déconnectez-vous de la machine virtuelle **testvm2** :

```
[root@testvm2 ~]# [CTRL]+[VERR MAJ]+[5]
```

```
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm1** :

```
[root@redhat9 ~]# virsh console testvm1
Connected to domain 'testvm1'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm1 /]#
```

Créez le fichier de mappage direct maître **/etc/auto.master.d/mountpoint1.autofs** :

```
[root@testvm1 /]# vi /etc/auto.master.d/mountpoint1.autofs

[root@testvm1 /]# cat /etc/auto.master.d/mountpoint1.autofs
/-      /etc/auto.mountpoint1
```

Créez le fichier de mappage **/etc/auto.mountpoint1** :

```
[root@testvm1 /]# vi /etc/auto.mountpoint1

[root@testvm1 /]# cat /etc/auto.mountpoint1
/mountpoint1    -fstype=nfs4,rw,sync    192.168.56.100:/mountpoint1
```

Redémarrez le service **autofs** :

```
[root@testvm1 /]# systemctl restart autofs

[root@testvm1 /]# systemctl status autofs
● autofs.service - Automounts filesystems on demand
     Loaded: loaded (/usr/lib/systemd/system/autofs.service; enabled; preset: d>
     Active: active (running) since Tue 2024-10-29 14:40:16 CET; 8s ago
   Main PID: 3612 (automount)
      Tasks: 8 (limit: 23172)
```

```
     Memory: 5.9M
        CPU: 29ms
     CGroup: /system.slice/autofs.service
             └─3612 /usr/sbin/automount --systemd-service --dont-check-daemon

oct. 29 14:40:16 testvm1.ittraining.loc systemd[1]: Starting Automounts filesys>
oct. 29 14:40:16 testvm1.ittraining.loc systemd[1]: Started Automounts filesyst>
```

Vérifiez la pris en compte de la configuration :

```
[root@testvm1 /]# mount | tail
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/vda1 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
none on /run/credentials/systemd-tmpfiles-setup.service type ramfs
(ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
192.168.56.100:/mountpoint on /mnt type nfs4
(rw,relatime,sync,vers=4.2,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,client
addr=192.168.56.50,local_lock=none,addr=192.168.56.100)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=164980k,nr_inodes=41245,mode=700,inode64)
/etc/auto.misc on /misc type autofs
(rw,relatime,fd=6,pgrp=3612,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=34473)
-hosts on /net type autofs (rw,relatime,fd=9,pgrp=3612,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=34474)
/etc/auto.home on /home type autofs
(rw,relatime,fd=12,pgrp=3612,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=34475)
/etc/auto.mountpoint1 on /mountpoint1 type autofs
(rw,relatime,fd=15,pgrp=3612,timeout=300,minproto=5,maxproto=5,direct,pipe_ino=35368)
```

Créez le fichier **/mountpoint1/test6.txt** :

```
[root@testvm1 /]# touch /mountpoint1/test6.txt

[root@testvm1 /]# ls /mountpoint1/
```

```
test6.txt
```

Déconnectez-vous de la machine virtuelle **testvm1** :

```
[root@testvm1 /]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Connectez-vous à la machine virtuelle **testvm2** :

```
[root@redhat9 ~]# virsh console testvm2
Connected to domain 'testvm2'
Escape character is ^] (Ctrl + ])
[Enter]
[root@testvm2 ~]#
```

Consultez ls contenu du répertoire **/mountpoint1** vous devez voir le fichier **test6.txt** :

```
[root@testvm2 ~]# ls -l /mountpoint1
total 0
-rw-r--r--. 1 nobody nobody 0 Oct 29 14:43 test6.txt
```

Déconnectez-vous de la machine virtuelle **testvm2** :

```
[root@testvm2 ~]# [CTRL]+[VERR MAJ]+[5]
[root@redhat9 ~]#
```

Copyright © 2024 Hugh Norris