

Les Services Réseaux

Université Assane Seck
UFR Sciences et technologie
Département Informatique

1 Les services de partage de fichiers

1 Les services de partage de fichiers

Introduction

- Le partage de fichiers est une pratique très répandue dans les réseaux
- Il permet à des machines de partager des informations à travers des infrastructures réseaux
- Dans Linux on peut distinguer:
 - NFS (Network File System), c'est-à-dire système de fichiers réseau entre machine Linux
 - Samba système de fichier réseau entre machine Linux et Windows

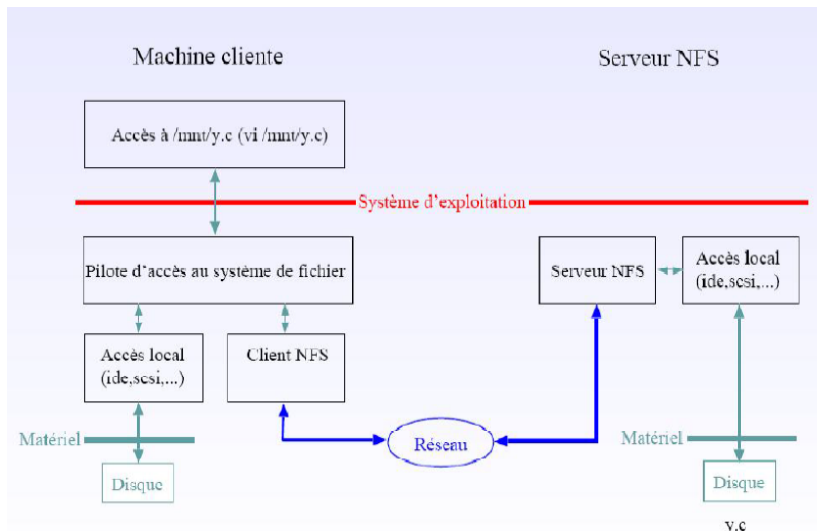
Le système NFS

- **NFS (Network File System)** est un protocole qui permet d'accéder à un système de fichiers à distance par le réseau
- NFS est pris en charge par tous les systèmes Unix
- L'utilisation du protocole NFS doit se limiter à un réseau local de confiance car les données qui circulent sur le réseau ne sont pas chiffrées
- Les droits d'accès sont accordés en fonction de l'adresse IP du client (qui peut être usurpée)

Le système NFS

- Objectifs de NFS
 - Avoir un système de fichiers distants
 - Accès transparent à ses données pour l'utilisateur
 - Tout se passe comme en local (ls, mv, etc.)!
- Pourquoi faire ?
 - Pouvoir accéder à ses données de plusieurs endroits
 - Administration
 - Plusieurs machines, un seul compte
- Protocole peu performant
 - Bon pour des réseaux locaux,
 - Très inconfortable pour les liaisons modem

Fonctionnement de NFS



NFS et les RPC

NFS repose sur les **RPC (Remote Procedure Calls)**

- Utilisation du **portmapper (programme portmap de Linux)**
- Portmapper = conversion des n° de prog RPC en n° de ports

Déroulement d'une RPC:

- Serveur RPC
 - Indique à portmap le port qu'il utilise et les n° de prog RPC qu'il gère
- Envoi d'une requête RPC par un client :
 - Il contacte portmap du serveur pour connaître le numéro de port du programme souhaité
 - Il envoie les données au port correspondant

NFS en pratique

NFS est composé de 4 protocoles utilisant les RPC

- nfs = programme **nfsd**
 - Authentification + Création, recherche, lecture et écriture de fichiers
- **mountd**
 - Montage des systèmes exportés (mount et umount)
- **nsm (Network Status Monitor)** = programme statd
 - Surveillance des noeuds du réseau (redémarrages...)
- **nlm (Network Lock Manager)** = programme lockd
 - Section critique (lock les fichiers utilisés)

Installation NFS

Installation des paquets

- **nfs-kernel-server**
- **nfs-common**

Lancement des démons

- Vérifier portmap : **rpcinfo -p**
- Lancer mountd et nfs s'ils ne le sont pas

```
# rpcinfo -p
program no_version protocole  no_port
100000      2             tcp      111  portmapper
100000      2             udp      111  portmapper
100005      2             udp      745  mountd
100005      1             tcp      747  mountd
100003      2             udp      2049  nfs
100003      1             tcp      2049  nfs
```

Configuration du serveur

Dans le fichier `/etc/exports`

- Chaque système de fichiers exporté doit avoir sa propre ligne
- Toutes les listes d'hôtes autorisés placées après un système de fichiers exporté doivent être séparées par des espaces
- Les options pour chacun des hôtes doivent être placées entre parenthèses directement après l'identificateur d'hôte
- La ligne pour un système de fichiers exporté a la structure suivante

```
<export> <host1>(<options>) <hostN>(<options>)...
```

Configuration du serveur

Dans cette structure,

- Remplacez <export> par le répertoire devant être exporté
- Remplacez <host1> par l'hôte ou le réseau vers lequel l'export est partagé
 - hôte simple — Où un hôte particulier est spécifié avec un FQDN, un nom d'hôte ou une adresse IP
 - réseaux IP — Autorisent la mise en correspondance d'hôtes en fonction de leur adresse IP dans un réseau
- Remplacez <options> par les options pour cet hôte ou ce réseau
 - ro — Les montages du système de fichiers exporté sont en lecture-seule
 - rw – lecture-écriture
 - root_squash — Retire au super-utilisateur en connexion distante tous les privilèges de son statut en lui assignant l'ID d'utilisateur nfsnobody (personne)

Quelques exemples de configuration

Exemple 1

```
/exported/directory bob.example.com
```

bob.example.com peut monter /exported/directory/

Exemple 2

```
/another/exported/directory 192.168.0.3(rw, sync)
```

192.168.0.3 peut monter /another/exported/directory/ en lecture/écriture et tous les transferts vers le disque sont validés avant que la requête d'écriture par le client ne soit achevée

Quelques exemples de configuration

Exemple 3

```
# fichier /etc/exports d'exemple
/          maître(rw) confiance(rw,no_root_squash)
/projects  proj*.local.domain(rw)
/usr       *.local.domain(ro) @trusted(rw)
/home/joe  pc001(rw,all_squash,anonuid=150,anongid=100)
/pub       (ro,insecure,all_squash)
```

Commentaire :

La première ligne exporte l'ensemble du système de fichiers vers les machines maître et confiance. En plus des droits d'écriture, toute conversion d'UID est abandonnée pour l'hôte confiance.

La deuxième et la troisième ligne montrent des exemples de noms d'hôtes génériques, et de sous-réseaux ('@trusted').

La quatrième ligne montre une entrée pour le client PC/NFS présenté plus haut.

La dernière ligne exporte un répertoire public de FTP, à tous les hôtes dans le monde, en effectuant les requêtes sous le compte anonyme. L'option *insecure* permet l'accès aux clients dont l'implémentation NFS n'utilise pas un port réservé.

Configuration coté client

Il existe plusieurs solutions pour monter son NFS sur un ordinateur

- Avec la commande mount pour un montage temporaire
- La première via /etc/fstab est plus adaptée à un Serveur allumé en permanence (ou presque)
- La seconde via autoFS est peut être plus tolérante à l'arrêt inopiné du serveur
- Pour parcourir les partages NFS d'un serveur (ici : serveur.local, que vous pouvez remplacer par l'IP)
- Un exemple

```
showmount -e serveur.local  
Export list for serveur.local:  
/mnt/nas/media *.local,192.168.0.0/255.255.0.0
```

Configuration coté client avec la commande mount

Pour monter un SF NFS

- `mount nom_server:/nom_rep /nom_point_de_montage`

Configuration coté client avec /etc/fstab

- Il faut tout d'abord créer un dossier dans lequel le NFS viendra se loger
- Ouvrez donc un terminal sur vos clients puis tapez
 - `sudo mkdir /media/NFS` (Ceci est un exemple)
- Il suffit de modifier le fichier `/etc/fstab` pour y ajouter la ligne

```
192.168.0.10:/<Dossier_à_partager>/ /media/NFS nfs defaults,user,auto,_netdev,bg 0 0
```

Configuration coté client avec /etc/fstab

- 192.168.0.10 étant l'IP de votre serveur
- *< Dossier__partager >* étant le dossier sur votre serveur que vous voulez recevoir (doit correspondre à celui que le serveur veut envoyer)
- /media/NFS On précise le point de montage (dossier que vous avez créé juste avant dans lequel vous souhaitez monter le partage NFS).
- nfs On précise le protocole utilisé.
- defaults,user,auto,_netdev sont décrite dans fstab, vous pouvez ajouter ro à la fin pour ne pas autoriser l'écriture.
- atime/noatime, diratime/nodiratime, relatime/norelatime, strictatime/nostrectatime mount n'as pas d'effet pour un montage nfs
- bg permet de ne pas bloquer le démarrage du client si le serveur n'est pas opérationnel.
- 0 0 dit à Linux de ne pas vérifier si le disque contient des erreurs (le serveur s'en occupera)

Configuration coté client avec

- Autofs : Logiciel pour monter facilement votre partage NFS
- Pour chaque type de montage via AutoFS, éditez le fichier (avec les droits d'administration) `/etc/auto.master`
- Ajouter une ligne concernant le type de système de fichier

Exemple complet

- nfs-serv veut partager le répertoire /nfs-save avec host01 pour un espace de sauvegarde
- Sur nfs-serv
 - Ajouter la ligne suivante à /etc/exports
 - /nfs-save host01 (rw)
 - Relancer nfs
 - /etc/init.d/nfsd restart ou exportfs
- Sur host01
 - Monter le répertoire /nfs-save dans /mnt/nfs_save
 - mkdir /mnt/nfs-save
 - mount nfs-serv:/nfs-save /mnt/nfs-save