Introduction au service NFS

Le service NFS (Network File System),
permet le partage d'un système de fichiers sur un réseau Linux

Généralités

Les versions 1 et 2 sont non sécurisées, prévues pour fonctionner sur UDP.

La version 3 est étendue pour prendre en charge TCP.

À ce stade la gestion de la sécurité reste élémentaire et souffre d'importantes lacunes. Le système est sans état (*stateless*) et ne permet pas la reprise sur incidents.

- Il s'agit du protocole standard de partage réseau entre machines Unix, créé par SUN vers 1980. Il comprend l'ajout de fonctionnalités supplémentaires (dans la couche session au dessus de TCP/IP), les **RPC** =(*Remote Procedure Calls*)
- Donc une machine joue le rôle de serveur de fichiers. Elle est appelée **`serveur NFS`**, et
  - on dit qu'elle **exporte** tout (arborescence racine /) ou partie de son système de fichiers,
  - en le partageant par une liste de stations accessibles par réseau,
  - en installant toutefois des restrictions d'accès.
- Comme toute ressource extérieure doit être intégrée dans le système de fichiers Linux, cet accès ne pourra être permis qu'à l'aide d'un **processus de montage** : une partie de l'arborescence d'une machine Linux "serveur", est exportée ce qui lui permet d'être intégré dans le système de fichiers d'une machine Linux "cliente".
- L'utilisateur peut monter cette arborescence exportée par le serveur, sur un point de montage, de façon tout-à-fait semblable au montage des systèmes de fichiers des divers périphériques.
  Le montage peut s'effectuer en cours de session de travail par la commande interactive `mount`.
- Mais dans un cadre de travail stable, où le serveur est dédié, il est souhaitable de monter la ressources `NFS` au démarrage.
  Il suffit pour cela d'inclure la description du montage sur une ligne de **/etc/fstab**. On peut comparer le processus à la "connexion à un lecteur réseau" sur d'autres systèmes.
- Dès lors, pour l'utilisateur sur la machine cliente, la ressource est accessible comme si elle résidait sur un périphérique local.

Configuration de NFS

Mise en place d un serveur NFS

```
apt-get install nfs-kernel-server nfs-common portmap
```

Partage simple

Le partage “simple” tel qu’il est présenté ci-dessous est un partage utilisant l’interface graphique fournie avec Ubuntu.

Côté serveur

À partir de la version Hardy d'Ubuntu, un clic droit sur le dossier permet d'accéder à un menu **Options de partage**.

Dans ce menu il suffit de cocher **Partager ce dossier** pour activer le partage.

Ce système de partage se base sur les nom et mot de passe des utilisateurs locaux, donc

- le nom et mot de passe de l'utilisateur créateur du partage permettent un accès complet
- le nom des autres utilisateurs donne un accès restreint, en lecture seule ou avec l'écriture si on sélectionne la case correspondante
- pour les autres on choisit ou non d'activer `l'accès d'évaluation`.

Configuration manuelle

Nous allons définir les dossiers à partager. Toutes les informations concernant le partage de dossier pour NFS se trouvent dans le fichier `/etc/exports`. Pour l'éditer, utilisez votre éditeur de texte favori ou *vi* par le biais de la commande suivante :

```
$ sudo vi /etc/exports
```

Dans ce fichier, chaque ligne est définie comme ceci :

```
<dossier partagé> <hôte>(<options>) <hôte2>(<options>)...
```

Les informations se trouvant sur une ligne sont les suivantes :

- `<dossier partagé>` : chemin menant au dossier partagé.
- `<hôte>` : indique quel est l'hôte qui peut accéder à ce partage, l'hôte peut être définit de plusieurs manières :
    - une IP : on indique simplement l'adresse IP de la machine pouvant accéder à ce partage.
- un nom d'hôte : on indique le nom complet de l'hôte (pour peu qu'il soit connu du système au travers d'un DNS ou du fichier `hosts`).
- un nom de groupe réseau NIS (NIS netgroup) qui s'indique sous la forme `@<netgroup>`.
- un domaine avec un joker qui indique les machines d'un domaine ou sous-domaine; par exemple : `*.ubuntu-fr.org`.
- un intervalle d'IP avec le masque de sous-réseau; par exemple : `192.168.0.0/24`.
- `<options>` : indique les options de partage; nous n'allons pas parcourir toutes les options ensemble mais uniquement les plus importantes.
    - `rw` : permet la lecture et l'écriture sur un partage pour l'hôte défini (par défaut, les partages sont en mode `ro`; c'est-à-dire en lecture seule).
- `async` : permet au serveur NFS de violer le protocole NFS et de répondre au requête avant que les changements effectués par la requête aient été appliqués sur l'unité de stockage. Cette option améliore les performances mais a un coût au niveau de l'intégrité des données (données corrompues ou perdues) en cas de redémarrage non-propre (par exemple en cas de crash système).
- `sync` : est le contraire de `async`. Le serveur NFS respecte le protocole NFS.
- `root_squash` : force le *mapping* de l'utilisateur `root` vers l'utilisateur anonyme (option par défaut).
- `no_root_squash` : n'effectue pas de *mapping* pour l'utilisateur `root`.

- `all_squash` : force le *mapping* de tous les utilisateurs vers l'utilisateur anonyme.
- `anonuid` : indique au serveur NFS l'UID de l'utilisateur anonyme (considéré comme tel dans les précédentes options de *mapping*).
- `anongid` : indique au serveur NFS le GID de l'utilisateur anonyme (considéré comme tel dans les précédentes options de *mapping*).

http://myweb.worldnet.net/~lmsoft/linux/indexnet.php3?page=nis#ss6
et http://smhteam.info/wiki/index.linux.php5?wiki=NFS

Exemple d'exportation déclarées dans le fichier **`/etc/exports`**`sur le serveur p00`

```
# repertoire   liste-machines(liste-options)
/home/jean  pc2(ro)  pc3(rw)
/usr/bin  pc2(ro) pc3(ro)
/var/www/html  *.fctice.ac-creteil.fr (ro) pc3 (rw)
/usr/share/doc (ro)
```

Pour valider un changement opéré dans ce fichier de configuration, faire appel à la commande :

```
exportfs -a # sous root
```

autres exemples:

- Le fichier /etc/exports de la machine serveur NFS (et NIS) qui a pour nom saturne, contiendra les noms des répertoires et disques à exporter et leur destination (vers quelles machines), par exemple :
  Vous voulez(saturne) partager, le répertoire /home en lecture/écriture avec orion mais en lecture seule avec neptune, le répertoire ftp en lecture pour tous et le cdrom en lecture avec toutes les machines du réseau clearsoft.fr .

```
# /etc/exports du serveur saturne.clearsoft.fr
#
/home   orion(rw) neptune(ro)
#/home  (ro,insecure,root_squash)
/home/ftp       (ro)
/mnt/cdrom *.clearsoft.fr(ro)
```

- Partage de home en lecture seule pour la machine 192.168.0.2 et 192.168.0.3.

```
/home/franck 192.168.0.2(ro,sync) 192.168.0.3(ro,sync)
```

- Partage de home en lecture seule pour le reseau 192.168.0.0/255.255.255.0

```
/home/franck 192.168.0.0/255.255.255.0(ro,sync)
```

- Répertoire en lecture-écriture pour la machine 192.168.0.2. Toutes les requêtes qui seront envoyées au serveur seront associées a l'utilisateur et au groupe anonyme grâce à la directive all_squash.

```
/nfs/export 192.168.0.2(rw,sync,all_squash)
```

- Répertoire en lecture-écriture pour tout le réseau local définit par 192.168.0.0/255.255.255.0. Toutes les requêtes qui seront envoyées au serveur seront associées a l'utilisateur et au groupe anonyme, puis redirigées vers l'utilisateur dont l'uid est 1001 et le groupe dont le gid est 1002.

```
/nfs/export 192.168.0.0/255.255.255.0(rw,sync,all_squash,anonuid=1001,anongid=1002)
```

L'identification des ordinateurs pour obtenir un accès aux ressources est très limité. Le serveur fait confiance au client NFS pour authentifier les utilisateurs. Comme le mettent en evidence les exemples ci-dessus, les seules restrictions qui peuvent etre imposées par le serveur sont les adresses IP des machines se connectant, et les droits associés aux requêtes, peu importe qui se connecte.

D'autres options sont disponibles. Pour de plus amples informations, vous pouvez consulter la man page de exportfs.

portmap(S11) doit être lancé avant nfs(S60) .
Les commandes pour lancer les démons sont :

```
/etc/rc.d/init.d/portmap start
/etc/rc.d/init.d/nfs start
```

remarque:
* Si le serveur répond : **permission denied** , il est possible qu'il faille editer /**etc/exports** sur le pc serveur,

```
/home/ols/Shared_Linux  192.168.0.100(rw)
```

en :

```
/home/ols/Shared_Linux  192.168.0.100/255.255.255.0(rw)
```

et ensuite redémarrer le service NFS :

Mise en route

Il ne vous reste plus qu'à démarrer votre nouveau serveur de fichier avec la commande :

```
sudo /etc/init.d/nfs-kernel-server start
```

Lorsque vous modifiez le fichier de configuration, n'oubliez pas d'entrer la commande suivante pour le recharger :

```
sudo /etc/init.d/nfs-kernel-server reload
```

Pour redémarrer le serveur NFS complètement, vous devez utiliser cette commande:

```
sudo /etc/init.d/nfs-kernel-server restart
```

Pour voir les différents démons RPC utilisés par vôtre système, tapez : ls /usr/sbin/rpc.*
chez moi, cela a donné:

```
/usr/sbin/rpc.mountd
/usr/sbin/rpc.nfsd
/usr/sbin/rpc.rstatd
/usr/sbin/rpc.yppasswdd
/usr/sbin/rpc.ypxfrd
```

Les fichiers /etc/hosts.allow et /etc/hosts.deny

Le système NFS est basé sur l'utilisation de cinq daemons. Ceux-ci gèrent entre autre le verrouillage des fichiers, les quotas ou encore le montage des dossiers exportés. Tout cela pour en venir au fait que securiser votre serveur NFS, vous impose aussi de limiter les accès exterieurs a ces daemons. Pour ce faire, il faut utiliser les fichiers /etc/hosts.allow et /etc/hosts.deny qui specifient quels ordinateurs sont autorisés à utiliser tels ou tels services sur votre machine.

Le principe est le suivant:

1. Dans un premier temps, on verifie si l'IP du client est située dans le fichier /etc/hosts.allow. Si oui, le client est autorisé à utiliser le ou les services associés.
2. Dans le cas contraire, on verifie le fichier /etc/hosts.deny. Si le client est mentionné pour un ou plusieurs services, celui-ci se verra refuser l'accès a ce ou ces derniers.
3. Le point important est que si l'IP du client n'est presente dans aucun de ces deux fichiers, alors le serveur considere que le client est autorise a utiliser les services.

Par consequent, il est préferable de renseigner ces deux fichiers pour éviter toute intrusion non souhaitée.

Le fichier /etc/hosts.allow devrait se presenter de la manière suivante :

```
portmap: IP_client
lockd: IP_client
mountd: IP_client
rquotad:IP_client
statd: IP_client
```

et en ce qui concerne le fichier /etc/hosts.deny :

```
portmap: ALL
lockd: ALL
mountd: ALL
rquotad:ALL
statd: ALL
```

Configuration des clients NFS :

Les packages

```
apt-get install nfs-common portmap
```

Sur la station cliente

- On crée un rép. de montage, situé pour des tests, dans **`/mnt`**, par exemple sur la machine `pc3` :
  [root@pc3 /]# **`mkdir /mnt/nfs`** (vide avant montage !!!)
- Puis on effectue le montage, sur le point de montage précédent, de la ressource `/var/www/html` (qui a été exportée par **`p00`** [root@pc3 /]
  # **`mount -t nfs p00:/home/httpd/html /mnt/nfs`**
  L'utilisateur sur pc3 pourra alors mettre à jour le site WEB distant sur `p00`
- Syntaxe générale
  `mount -t nfs nom-serveur(ip):arborescence point-montage`

autre exemple:
La commande pour monter, par exemple le répertoire /home de la machine saturne sur le répertoire /users de la machine orion, aura cette forme, depuis la machine orion :

mount -t nfs saturne:/home /users

Vous pouvez éditer et ajouter des lignes dans le fichier /etc/fstab, en plus du montage des partitions ext2, vfat ...déjà existantes :

```
# /etc/fstab sur orion.clearsoft.fr
#
# Montage partitions NFS de saturne.clearsoft.fr
saturne:/home/ftp          /users/ftp      nfs     noauto,soft,intr,ro  0 0
saturne:/home          /users/home     nfs     noauto,hard,intr,rw  0 0
saturne:/mnt/cdrom     /users/cdrom    nfs     noauto,soft,intr,ro  0 0
```

Pour prendre en compte le fichier fstab après modification (si le montage ne comporte pas l'option noauto), tapez :

mount -a

Pour prendre connaissance des différentes options utilisables dans /etc/fstab, voir nfs_options.txt .
et utilisez les pages de manuel pour plus d'info : man nfs
ATTENTION! Les répertoires sous /users/ doivent être vide avant montage !!!

Complements de dernière minute

Mise en place de partages NFS v4 :

NFS v4 est la nouvelle version du protocole de partages de fichiers historique pour *NIX Cette version apporte de nombreuses améliorations telles que :

- sécurité via l'utilisation de kerberos pour l'authentification
- la fiabilité avec l'utilisation de TCP par défaut
- le passage à travers un firewall est beaucoup plus simple en utilisant par défaut le port 2049
- le support de l'ipv6
- réplication, failover et récupération des sessions en cas de panne du serveur

Cette version du protocole est incompatible avec les anciennes versions mais, cette incompatibilité est largement compensée par les améliorations apportées et la migration de l'une à l'autre est relativement simple à effectuer.

Avant de commencer la configuration il faut s'assurer que les deux lignes suivantes soient présentes dans la sortie de mount :

```
nfsd on /proc/fs/nfsd type nfsd (rw)
rpc_pipefs on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
```

Sans ces deux lignes, NFS v4 ne pourra pas être mis en place. nfsd est monté par le script d'init du paquet nfs-kernel-server et rpc_pipefs est quant à lui monté par le script d'init de nfs-common.

NFS 4 permet maintenant de monter des partages en fonction d'une racine virtuelle. Ce partage racine est remarquable dans le fichier /etc/exports car il contient fsid=0. Par exemple pour définir la racine des répertoires NFS partagés sous /exports il faut ajouter dans le fichier /etc/exports :

```
/export *(rw,fsid=0,insecure,no_subtree_check)
```

La grande différence entre NFS v3 et NFS v4 est là, pour monter sur un client la racine des partage en NFS v3 il fallait faire :

```
mount -t nfs server:/export /mnt
```

alors qu'avec NFS v4 la commande devient :

```
mount -t nfs4 server:/ /mnt
```

Chaque sous-répertoire du répertoire /export sera partagé en fonction de la racine virtuelle. Donc pour partager un répertoire à l'extérieur de cette racine virtuelle, vous pouvez utiliser l'option --bind de mount(1). Par exemple pour partager les répertoires utilisateurs, utilisez la commande suivante pour ajouter le répertoire à la racine virtuelle :

```
mount --bind /home /export/home
```

un répertoire /export/home est alors présent dans /export. Pour exporter ce répertoire, vous pouvez utiliser la syntaxe habituelle des exports NFS.

Configuration du client :

Le montage d'un export NFS v4 sur un client requiert un util-linux récent (ici 2.12r-11) et nfs-common mais il reste une anomalie (#389557) dans le paquet Debian d'util-linux et génère cet avertissement :

```
Warning: rpc.idmapd appears not to be running.
All uids will be mapped to the nobody uid.
```

Pour monter un partage NFS, en supposant que les partages du serveur se situent dans /export, il faut utiliser la commande :

```
mount -t nfs4 serveur:/ /mnt
```

alors qu'avec NFS v3 et inférieure la commande aurait été :

```
mount -t nfs server:/export /mnt
```

Dans le fichier /etc/fstab le point de montage sera de cette forme :

```
serveur:/ /mnt nfs4 wsize=32768,rsize=32768 0 0
```

Services obsolètes :

L'arrivée de NFS v4 rend obsolète les services suivants :

- portmapper
- rpc.lockd
- rpc.quotad

Trucs

Ajouter

```
NEED_IDMAPD=yes
```

dans

```
/etc/default/nfs-common
```

et lancer un

```
/etc/init.d/nfs-common restart
```

Vous pourrez tranquillement profiter par la suite de votre beau NFS v4 !

Autre remarque trouvée sur le net

apparemment kerberos n'aime pas les machines avec une majuscule...
J'ai simplement changé mon Linux1.example.com en linux1.example.com et tout fonctionne. pourtant, c'était bel et bien, à la casse près, le bon nom dans le krb5.keytab

NFSv4Howto

https://help.ubuntu.com/community/NFSv4Howto

Installation

- NFSv4 client

  ```
  # apt-get install nfs-common
  ```

- NFSv4 server

  ```
  # apt-get install nfs-kernel-server
  ```

Après installation, on peut avoir un échec au démarrage de nfs-kernel-server du à l'abscence d'entrées dans le fichiers in /etc/exports.

NFSv4 simple

NFSv4 Server

NFSv4 exports existe dans un pseudo système de fichiers lié aux vrais répertoires montés avec l'option `--bind`.

- `Supposons qu'on exporte notre répertoire /home dans /home/users`. Premièrement, un crée un système de fichiers à exporter:

  ```
  # mkdir /export
  # mkdir /export/users
  ```

  et on mount le vrai users directory avec:

  ```
  # mount --bind /home/users /export/users
  ```

  On peut éviter de retaper cela à chaque démarrage par

  une ligne dans `/etc/fstab`

  ```
  /home/users    /export/users   none    bind  0  0
  ```

- Dans `/etc/default/nfs-kernel-server` on place:

  ```
  NEED_SVCGSSD=no
  ```

  car on ne va pas activer le module de sécurité de NFSv4 pour le moment.
- `Dans/etc/default/nfs-common` on ajoute:

  ```
  NEED_IDMAPD=yes
  NEED_GSSD=no
  ```

- Pour exporter nos répertoires dans un réseau local 192.198.1.0/24

  `nous ajoutons ces deux lignes dans/etc/exports`

  ```
  /export       192.168.1.0/24(rw,fsid=0,insecure,no_subtree_check,async)
  /export/users 192.168.1.0/24(rw,nohide,insecure,no_subtree_check,async)
  ```

- On redémarre le service

```
# /etc/init.d/nfs-kernel-server restart
```

NFSv4 Client

- Sur le client on peut monter le système de fiché complet par une commande:

```
# mount -t nfs4 -o proto=tcp,port=2049 nfs-server:/ /mnt
```

- On peut également monter un sous répertoire par :

```
# mount -t nfs4 -o proto=tcp,port=2049 nfs-server:/users /home/users
```

- On peut rendre la modification permanente par une ligne dans `/etc/fstab`:

```
nfs-server:/   /mnt   nfs4    _netdev,auto  0  0
```

  - où `l'option auto` option mount au démarrage et le paramètre `_netdev` attend jusqu'au démarrage du réseau.

- Si votre réseau est lent ou le démarrage difficile, changez par:

```
nfs-server:/    /mnt   nfs4    noauto  0  0
```

et axécutez le montage apres un certain temps.Ajoutez par exemple dans `/etc/rc.local`

- ```
  # sleep 5
  # mount /mnt
  ```

- Si vous avez un message d'erreur tel que

```
Warning: rpc.idmapd appears not to be running.
         All uids will be mapped to the nobody uid.
mount: unknown filesystem type 'nfs4'
```

  `Ajoutez la ligne suivante dans /etc/default/nfs-common`:

```
NEED_IDMAPD=yes
```

  et redémarrez nfs-common

```
# /etc/init.d/nfs-common restart
```

http://wiki.epfl.ch/nfsv4-clients-config/ubuntu-test
```