

TP7 : NFS partage réseau sous Linux

NFS est l'abréviation de [Network File System](#), c'est-à-dire [système de fichiers réseau](#).

Ce système de fichiers en réseau permet de [partager des données](#) principalement entre systèmes de type UNIX mais des versions existent également pour Microsoft Windows™ et Mac. NFS est compatible avec IPv6 sur la plupart des systèmes.

L'intérêt de NFS est de permettre l'utilisation d'un répertoire ou de tout le système de fichiers d'un ordinateur de manière totalement transparente, comme s'il s'agissait d'un disque dur connecté directement à votre ordinateur.

Pré-requis

Deux ordinateurs au moins

- Un [serveur](#) sous GNU/Linux (ici Ubuntu) où se trouvent les fichiers à partager.
- Un [client](#) qui doit pouvoir accéder aux fichiers du Serveur.

Ne pas avoir peur de la ligne de commande

Avoir les droits d'administration sur les deux machines

- C'est-à-dire de pouvoir installer des paquets et utiliser la commande *sudo*

Etre sur un réseau de confiance

[NFS](#) surtout avant la version 4 (incompatible avec les précédentes) ne doit être déployé que sur des réseaux [sûrs](#). Quelle que soit la version, NFS est à déployer dans un [réseau local](#) et n'a pas de vocation à être ouvert sur [internet](#). On peut utiliser par exemple NFS sur son réseau local derrière une *Box* assurant le rôle de [routeur NAT](#).

Connaître les adresses IP locales de Serveur et de Client

Dans le [terminal](#) lancer « ifconfig », l'adresse est à la ligne « inet adr: », paragraphe « eth » si vous êtes connecté en ethernet, « wlan » pour une connexion Wi-Fi. Vous pouvez aussi connaître l'IP de vos PC via l'applet [network-manager](#).

Pour la suite de cet exemple, on considérera que Serveur et Client ont une adresse locale fixe :

- Serveur : **192.168.0.10**
- Client : **192.168.0.20**

Il peut être important de fixer les adresses IP de vos ordinateurs pour être sûr qu'elles ne changeront pas :

- Soit à partir de l'interface du routeur.
- Soit directement à partir d'Ubuntu via [Network-Manager](#) ou [par la configuration fixe](#)

Les machines	IP locales	Points de montage des fichiers
<i>Votre serveur</i>	<i>IP de votre serveur</i>	Exemple : <i>/home/Votre_Nom_d'Utilisateur_sur_Votre_Serveur</i>
Serveur	192.168.0.10	/Fichiers_à_partager_sur_Serveur
<i>Votre client</i>	<i>IP de votre client</i>	Exemple : <i>/home/Nom_d'Utilisateur_sur_le_Client/Partage</i>
Client	192.168.0.20	/Point_de_montage_sur_Client

Vous n'êtes pas obligé de connaître les IP de tous vos clients si vous en avez beaucoup ou si leurs IP sont amenées à changer. En effet avec cette méthode, les ordinateurs de tout votre réseau local pourront sans distinction utiliser NFS.

Création d'un Serveur NFS

Installation

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

Configuration

La configuration d'un 'export' NFS se fait en éditant le fichier **/etc/exports**

```
# Un exemple à adapter:  
/Dossier/à/partager/  
192.168.0.0/24(rw,all_squash,anonuid=1000,anongid=1000,sync,no_subtree_check)  
/Dossier/numero02/ master(rw) trusty(rw,no_root_squash)
```

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 défini 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 ou 192.168.0.*
- **<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 aux requêtes 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*).
 - **subtree_check** : Si un sous-répertoire dans un système de fichiers est partagé, mais que le système de fichiers ne l'est pas, alors chaque fois qu'une requête NFS arrive, le serveur doit non seulement vérifier que le fichier accédé est dans le système de fichiers approprié (ce qui est facile), mais aussi qu'il est dans l'arborescence partagée (ce qui est plus compliqué). Cette vérification s'appelle *subtree_check*.
 - **no_subtree_check** : Cette option neutralise la vérification de sous-répertoires, ce qui a des subtiles implications au niveau de la sécurité, mais peut améliorer la fiabilité dans certains cas.

Ce qui pourrait donner par exemple:

```
/media/NFS 192.168.0.0/24(rw,all_squash,anonuid=1000,anongid=1000,sync, no_subtree_check)
```

Si vous obtenez l'erreur suivante au montage : **mount.nfs4: access denied by server while mounting**, vérifiez les droits d'accès au dossier partagé (le dossier est peut être en mode interdit pour "autres" ce qui le rend impossible à lire pour le serveur NFS)

Relancer le service

Après avoir défini vos partages dans le fichier **/etc/exports** il suffit de relancer le service nfs:

```
sudo service nfs-kernel-server reload
```

Cette commande ne coupe pas les transferts en cours si la nouvelle configuration permet toujours leur accès au serveur. Vous pouvez donc la lancer plus ou moins à n'importe quel moment.

Pour vérifier que l'export a bien eu lieu, taper sur le serveur NFS la commande :

```
showmount -e
```

Si l'export n'est pas effectif , il faut faire restart du service NFS mais attention cela peut interrompre les transferts en cours :

```
sudo service nfs-kernel-server restart
```

Un peu de sécurité avec TCPwrapper

Pas indispensable si vous avez un bon firewall entre internet et votre réseau local, NFS peut être protégé par le serveur le cas échéant. (testez tout d'abord votre configuration sans protection pour voir si tout fonctionne si vous comptez utiliser ce genre de protection dans le futur)

Si vous utilisez [denyhosts](#) sur le serveur, il vous faudra ajouter les lignes suivantes dans le fichier `/etc/hosts.allow` :

/etc/hosts.allow

```
portmap: <ip de la machine cliente>
lockd: <ip de la machine cliente>
nfsd: <ip de la machine cliente>
mountd: <ip de la machine cliente>
rquotad: <ip de la machine cliente>
statd: <ip de la machine cliente>
```

et également afin de sécuriser un peu plus votre installation, ajouter dans le fichier `/etc/hosts.deny` :

/etc/hosts.deny

```
# Tout interdire sur portmap, nfsd et mountd
portmap:ALL
nfsd:ALL
mountd:ALL
```

Client NFS

Pour accéder à un NFS, il faut le "monter". En gros il s'agit de dire au service qui s'occupe des disques durs que ce serveur NFS est comme un disque dur et qu'il serait bête de ne pas s'en servir !

Il n'existe pas d'autoconfiguration pour NFS, il faudra donc le faire manuellement.

Installation

Le paquet nécessaire pour accéder à un NFS est [nfs-common](#)

Après avoir installé ce paquet, il existe deux solutions pour monter son NFS sur un ordinateur. La première 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) saisissez dans un [terminal](#) la commande suivante:

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

Modifier /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
```

Vous pouvez bien entendu créer un répertoire du nom de votre choix; dans ce cas adaptez les instructions suivantes au besoin.

Pour ce faire, 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,noatime,intr 0 0
```

- **192.168.0.10** étant l'IP de votre serveur
- **<Dossier_à_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,noatime,intr** sont des options de montage, vous pouvez ajouter **,ro** à la fin pour ne pas autoriser l'écriture.
- **0 0** dit à Linux de ne pas vérifier si le disque contient des erreurs (le serveur s'en occupera)

l'option **intr** est dépréciée depuis la version de kernel 2.6.25 donc [Lucid 10.04 LTS](#) comme le stipule la manpage [NFS](#)

Optimisation possible:

- Si vous utilisez un réseau Gigabit ethernet (les dernier switches, routeurs, box), vous pouvez ajouter en option de montage: **,rsize=16384,wsiz=16384**

Ces options de montage permettront de booster légèrement à énormément (suivant la carte réseau) la vitesse de transfert sur les gros fichiers.

- Si votre serveur est très peu puissant (un Pentium 2 ou 3 par exemple plus de 12 ans d'âge, bon millésime) vous pouvez augmenter les performances en ajoutant en "options de montage" (voir plus haut) **,vers=3,udp** ceci aura pour effet d'utiliser la version 3 de NFS ainsi que le protocole UDP.

Vous devriez maintenant utiliser moins de processeur mais ceci peut ne pas marcher très bien si votre liaison réseau est de mauvaise qualité (**ne surtout pas utiliser ces options sur les clients en Wifi !**)

Utiliser Autofs

- [Autofs](#) : Logiciel pour monter facilement votre partage NFS.

Résoudre les problèmes

Rétrograder en NFSv3

Si vous obtenez un message comme: "mount.nfs: access denied by server while mounting 'ipserveur'", c'est certainement que vous avez entré les bonnes adresses ip autorisés dans export, cependant, NFSv4 n'utilise pas le système d'authentification par ip, il est donc utile de rétrograder en version 3 pour que le serveur se base sur l'ip

Côté serveur

Modifier le fichier **/etc/default/nfs-kernel-server** et éditer la ligne

```
RPCNFSDCOUNT='8'
```

par

```
RPCNFSDCOUNT='8 --no-nfs-version 4'
```

Côté client

Ajoutez **nfsvers=3** au fichier **/etc/fstab**

Utilisez **nfsstat** pour vérifier que la modification à été prise en compte.

Résoudre le problème de verrouillage de fichiers avec LibreOffice

Commentez les lignes suivantes dans les fichiers **/usr/bin/libreoffice** et **/usr/bin/soffice** (ajouter un # devant les deux lignes)

```
SAL_ENABLE_FILE_LOCKING=1
export SAL_ENABLE_FILE_LOCKING
```

Résoudre le problème de temps de montage sur les clients

Les clients Ubuntu 13&+ présentent un bug connu temps de latence très long pour le montage

Pour le résoudre, sur le client Modifier le fichier **/etc/default/nfs-common** et éditer la ligne

```
# Do you want to start the gssd daemon? It is required for Kerberos mounts.
NEED_GSSD=
```

par

```
# Do you want to start the gssd daemon? It is required for Kerberos mounts.
NEED_GSSD=yes
```

puis blacklister le module 'rpcsec_gss_krb5' en éditant le fichier **/etc/modprobe.d/blacklist.conf** en y ajoutant à la fin la ligne :

```
#To improve NFS mounting time
blacklist 'rpcsec_gss_krb5'
```