

Dernière mise-à-jour : 2020/08/07 16:37

LDF307 - Gestion du Serveur NFS

Contenu du Module

- **LDF307 - Gestion du Serveur NFS**
 - Présentation
 - Les Services et Processus du Serveur NFSv3
 - Les Services RPC
 - Options d'un Partage NFS
 - Commandes de Base
 - Installation
 - Mise en Place
 - Configuration du Serveur
 - Configuration du Client
 - Surveillance du Serveur
 - La Commande rpcinfo
 - La Commande nfsstat

Présentation

Quand on parle de NFS, on parle d'**exportation** d'un répertoire sur le serveur afin que celui-ci puisse être vu par des clients sur le réseau. Ces clients peuvent ensuite monter le répertoire et l'utiliser comme si celui-ci faisait partie de son propre filesystem.

Le Network File System (NFS) est le protocole de partage de fichiers historique sur des systèmes Unix. Lors de l'introduction de Samba, NFS a vu sa popularité diminuée, essentiellement parce que la connexion est non-sécurisée :

- le partage ainsi que ses caractéristiques sont configurés par rapport à l'adresse IP du client, or l'IP Spoofing est de plus en plus répandu,
 - aucun mot de passe n'est demandé lors de la connexion d'un utilisateur à une ressource car le serveur NFS présume que l'utilisateur *jean* distant
-

est le même utilisateur du compte *jean* sur le serveur NFS.

Cependant l'arrivée sur le marché de serveurs NAS domestiques ainsi que l'utilisation de la virtualisation dans le milieu professionnel fait que NFS connaît un regain d'intérêt en tant que stockage mutualisé raid, simple à mettre en œuvre.

Il existe actuellement 3 versions de NFS :

Version	Protocole Utilisé	Dépendance
NFSv2	TCP et UDP	portmap
NFSv3	TCP et UDP	portmap
NFSv4	TCP	Aucune - les fonctions de portmap sont incluses dans NFSv4

La version utilisée par défaut sous CentOS/Redhat est **NFSv3**.

Les Services et Processus du Serveur NFSv3

La version NFSv3 utilise les services suivants :

Services	Fonction
nfs	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

Les Services RPC

Les services RPC (*Remote Procedure Calls* ou appel de procédures distantes) ont été inventé par SUN Microsystem pour faciliter le travail des développeurs de pour des échanges entre machines distantes. Les RPC s'appuient sur des numéros de programmes.

Lorsque le client veut faire une requête à un service RPC, il contacte en premier lieu le service **rpcbind** qui assigne un numéro de port au delà du numéro **32768** à un numéro de programme RPC.

Options d'un Partage NFS

Certaines options, appliquées à un partage, modifient le comportement du serveur NFS 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

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

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

Installation

Installez le serveur et le client NFS en utilisant apt-get :

```
root@debian8:~# apt-get install nfs-kernel-server nfs-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
nfs-common is already the newest version.
The following NEW packages will be installed:
  nfs-kernel-server
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 115 kB of archives.
After this operation, 515 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ftp.fr.debian.org/debian/ jessie/main nfs-kernel-server amd64 1:1.2.8-9 [115 kB]
Fetched 115 kB in 0s (318 kB/s)
Selecting previously unselected package nfs-kernel-server.
(Reading database ... 82450 files and directories currently installed.)
Preparing to unpack .../nfs-kernel-server_1%3a1.2.8-9_amd64.deb ...
Unpacking nfs-kernel-server (1:1.2.8-9) ...
Processing triggers for man-db (2.7.0.2-5) ...
Processing triggers for systemd (215-17+deb8u4) ...
Setting up nfs-kernel-server (1:1.2.8-9) ...

Creating config file /etc/exports with new version

Creating config file /etc/default/nfs-kernel-server with new version
Processing triggers for systemd (215-17+deb8u4) ...
```

Mise en Place

Configuration du Serveur



Important : Arrêtez votre VM. Dans la fenêtre de Oracle VM VirtualBox, cliquez sur **Fichier > Paramètres > Réseau** et créez un réseau NAT appelé **NatNetwork**. Dans les paramètres de votre VM, cliquez sur **Réseau** et configurez la Carte 2 en Réseau NAT dans le réseau NatNetwork. Démarrez votre VM.

Configurez votre interface réseau #2 (eth1) :

```
root@debian8:~# nmcli connection add con-name ip_fixe ifname eth1 type ethernet ip4 10.0.2.16/24 gw4 10.0.2.2
Connection 'ip_fixe' (d61c409c-d334-44db-a2fd-a7456562e63a) successfully added.
root@debian8:~# nmcli connection mod ip_fixe ipv4.dns 8.8.8.8
root@debian8:~# nmcli connection up ip_fixe
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/2)
root@debian8:~# nmcli c show
```

NAME	UUID	TYPE	DEVICE
eth1	e9fe77dc-2750-47ba-a3ef-50a8c5c8e22a	802-3-ethernet	--
eth0	02dd6b3d-097b-4243-8f86-eeb71ff27736	802-3-ethernet	eth0
Ifupdown (eth0)	681b428f-beaf-8932-dce4-687ed5bae28e	802-3-ethernet	--
ip_fixe	d61c409c-d334-44db-a2fd-a7456562e63a	802-3-ethernet	eth1

```
root@debian8:~# systemctl restart NetworkManager.service
root@debian8:~# nmcli c show
```

NAME	UUID	TYPE	DEVICE
eth0	e7181c1d-73c3-4c7e-89f2-7902c130ec9e	802-3-ethernet	eth0
ip_fixe	d61c409c-d334-44db-a2fd-a7456562e63a	802-3-ethernet	--
Ifupdown (eth0)	681b428f-beaf-8932-dce4-687ed5bae28e	802-3-ethernet	--
eth1	d33363de-a915-4030-8cea-0d64997ae669	802-3-ethernet	eth1

Ajoutez une autre adresse IP pour le NFS :

```
root@debian8:~# nmcli connection mod ip_fixe +ipv4.addresses 192.168.1.2/24
```

Continuez maintenant par la mise en place du service **nfs** :

```
root@debian8:~# systemctl enable nfs-kernel-server
Synchronizing state for nfs-kernel-server.service with SysVinit using update-rc.d...
Executing /usr/sbin/update-rc.d nfs-kernel-server defaults
Executing /usr/sbin/update-rc.d nfs-kernel-server enable
```

La mise en place d'un partage ponctuel se fait en utilisant la commande **exportfs** en indiquant en argument le répertoire sous la forme de *adresse_ip_du_serveur:chemin_du_partage* :

```
root@debian8:~# exportfs
root@debian8:~# exportfs 192.168.1.1:/home/trainee
root@debian8:~# exportfs
/home/trainee 192.168.1.1
```

Afin de mettre en place un ou des partages **permanents**, il est nécessaire d'éditer le fichier **/etc/exports** :

```
root@debian8:~# vi /etc/exports
root@debian8:~# cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#      to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/home/trainee 192.168.1.1
/tmp          *(fsid=0)
```



Important : Dans ce cas, nous avons partagé le répertoire **/home/trainee** pour la seule adresse IP 192.168.1.1.

Démarrez maintenant le service nfs afin que le fichier **/etc/exports** soit re-lu :

```
root@debian8:~# systemctl reload nfs-kernel-server
root@debian8:~# systemctl start nfs-kernel-server
root@debian8:~# systemctl status nfs-kernel-server
● nfs-kernel-server.service - LSB: Kernel NFS server support
   Loaded: loaded (/etc/init.d/nfs-kernel-server)
   Active: active (exited) since Thu 2018-03-22 10:04:01 GMT; 22min ago
   Process: 10475 ExecReload=/etc/init.d/nfs-kernel-server reload (code=exited, status=0/SUCCESS)

Mar 22 10:04:01 debian8 nfs-kernel-server[506]: Not starting NFS kernel daemon: no exports. ... (warning).
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: Re-exporting directories for NFS kernel daemon...exportfs: No
options for /home/t...warning
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: exportfs: /etc/exports [1]: Neither 'subtree_check' or
'no_subtree_check' specifi...ainee".
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: Assuming default behaviour ('no_subtree_check').
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: NOTE: this default has changed since nfs-utils version 1.0.x
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: exportfs: /etc/exports [2]: Neither 'subtree_check' or
'no_subtree_check' specifi...:/tmp".
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: Assuming default behaviour ('no_subtree_check').
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: NOTE: this default has changed since nfs-utils version 1.0.x
Mar 22 10:25:59 debian8 nfs-kernel-server[10475]: .
Hint: Some lines were ellipsized, use -l to show in full.
```

Puisque aucune option ne soit spécifiée pour les montages, ceux-ci ont été exportés avec des option par défaut. En utilisant l'option **-v** de la commande **exportfs**, il est possible de consulter ces options :

```
root@debian8:~# exportfs -v
/home/trainee 192.168.1.1(ro,wdelay,root_squash,no_subtree_check,sec=sys,ro,root_squash,no_all_squash)
```

```
/tmp <world>(ro,wdelay,root_squash,no_subtree_check,fsid=0,sec=sys,ro,root_squash,no_all_squash)
```

Configuration du Client



Important : Arrêtez votre VM. Créez une clône de votre VM. Démarrez la VM clonée.

Re-configuez ensuite l'interface réseau de votre VM Client :

```
root@debian8:~# nmcli connection del ip_fixe
root@debian8:~# nmcli connection show ip_fixe
Error: ip_fixe - no such connection profile.
root@debian8:~# nmcli connection add con-name ip_fixe ifname eth1 type ethernet ip4 10.0.2.17/24 gw4 10.0.2.2
Connection 'ip_fixe' (5f81cfd6-9ccb-4828-a9ff-aa610719de31) successfully added.
root@debian8:~# nmcli connection mod ip_fixe ipv4.dns 8.8.8.8
root@debian8:~# nmcli connection mod ip_fixe +ipv4.addresses 192.168.1.1/24
root@debian8:~# nmcli connection up ip_fixe
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/3)
root@debian8:~# systemctl restart NetworkManager.service
```



Important : Démarrez la VM d'origine.

A partir de votre client, consultez les répertoire exportés du serveur :

```
root@debian8:~# showmount --exports 192.168.1.2
Export list for 192.168.1.2:
/tmp *
/home/trainee 192.168.1.1
```

Créez maintenant le répertoire **/nfs** dans le client et montez le partage **192.168.1.2:/home/trainee** :

```
root@debian8:~# mkdir /nfs
root@debian8:~# mount -t nfs 192.168.1.2:/home/trainee /nfs
```

Surveillance du Serveur

La Commande rpcinfo

La commande **rpcinfo** permet de faire une requête RPC sur le serveur et de voir les démons gérés :

```
root@debian8:/# rpcinfo
  program version netid    address                service    owner
  100000    4      tcp6    :::0.111               portmapper superuser
  100000    3      tcp6    :::0.111               portmapper superuser
  100000    4      udp6    :::0.111               portmapper superuser
  100000    3      udp6    :::0.111               portmapper superuser
  100000    4      tcp     0.0.0.0.0.111         portmapper superuser
  100000    3      tcp     0.0.0.0.0.111         portmapper superuser
  100000    2      tcp     0.0.0.0.0.111         portmapper superuser
  100000    4      udp     0.0.0.0.0.111         portmapper superuser
  100000    3      udp     0.0.0.0.0.111         portmapper superuser
  100000    2      udp     0.0.0.0.0.111         portmapper superuser
  100000    4      local   /run/rpcbind.sock     portmapper superuser
  100000    3      local   /run/rpcbind.sock     portmapper superuser
  100024    1      udp     0.0.0.0.203.188       status     107
  100024    1      tcp     0.0.0.0.148.150       status     107
  100024    1      udp6    :::169.248            status     107
  100024    1      tcp6    :::161.44             status     107
  100003    2      tcp     0.0.0.0.8.1           nfs        superuser
  100003    3      tcp     0.0.0.0.8.1           nfs        superuser
```

100003	4	tcp	0.0.0.0.8.1	nfs	superuser
100227	2	tcp	0.0.0.0.8.1	-	superuser
100227	3	tcp	0.0.0.0.8.1	-	superuser
100003	2	udp	0.0.0.0.8.1	nfs	superuser
100003	3	udp	0.0.0.0.8.1	nfs	superuser
100003	4	udp	0.0.0.0.8.1	nfs	superuser
100227	2	udp	0.0.0.0.8.1	-	superuser
100227	3	udp	0.0.0.0.8.1	-	superuser
100003	2	tcp6	:::8.1	nfs	superuser
100003	3	tcp6	:::8.1	nfs	superuser
100003	4	tcp6	:::8.1	nfs	superuser
100227	2	tcp6	:::8.1	-	superuser
100227	3	tcp6	:::8.1	-	superuser
100003	2	udp6	:::8.1	nfs	superuser
100003	3	udp6	:::8.1	nfs	superuser
100003	4	udp6	:::8.1	nfs	superuser
100227	2	udp6	:::8.1	-	superuser
100227	3	udp6	:::8.1	-	superuser
100021	1	udp	0.0.0.0.209.119	nlockmgr	superuser
100021	3	udp	0.0.0.0.209.119	nlockmgr	superuser
100021	4	udp	0.0.0.0.209.119	nlockmgr	superuser
100021	1	tcp	0.0.0.0.197.239	nlockmgr	superuser
100021	3	tcp	0.0.0.0.197.239	nlockmgr	superuser
100021	4	tcp	0.0.0.0.197.239	nlockmgr	superuser
100021	1	udp6	:::207.89	nlockmgr	superuser
100021	3	udp6	:::207.89	nlockmgr	superuser
100021	4	udp6	:::207.89	nlockmgr	superuser
100021	1	tcp6	:::237.162	nlockmgr	superuser
100021	3	tcp6	:::237.162	nlockmgr	superuser
100021	4	tcp6	:::237.162	nlockmgr	superuser
100005	1	udp	0.0.0.0.227.242	mountd	superuser
100005	1	tcp	0.0.0.0.225.144	mountd	superuser
100005	1	udp6	:::154.158	mountd	superuser
100005	1	tcp6	:::215.191	mountd	superuser

100005	2	udp	0.0.0.0.185.106	mountd	superuser
100005	2	tcp	0.0.0.0.212.122	mountd	superuser
100005	2	udp6	::.136.22	mountd	superuser
100005	2	tcp6	::.193.198	mountd	superuser
100005	3	udp	0.0.0.0.133.232	mountd	superuser
100005	3	tcp	0.0.0.0.232.4	mountd	superuser
100005	3	udp6	::.193.195	mountd	superuser
100005	3	tcp6	::.198.65	mountd	superuser

La Commande nfsstat

La Commande **nfsstat** permet de vérifier l'activité sur le serveur NFS :

```
root@debian8:/# nfsstat
Server rpc stats:
calls      badcalls   badclnt    badauth    xdrcll
0          0          0          0          0

Client rpc stats:
calls      retrans    authrefrsh
77         0          77

Client nfs v3:
null       getattr    setattr    lookup     access     readlink
0          0% 10      27% 0        0% 2        5% 8        22% 0        0%
read       write      create     mkdir      symlink    mknod
0          0% 0        0% 1        2% 0        0% 0        0% 0        0%
remove     rmdir     rename     link       readdir    readdirplus
0          0% 0        0% 0        0% 0        0% 0        6        16%
fsstat     fsinfo    pathconf   commit
0          0% 6        16% 3        8% 0        0%
```

Client nfs v4:

null	read	write	commit	open	open_conf
0	0% 0	0% 0	0% 0	0% 0	0% 0
open_noat	open_dgrd	close	setattr	fsinfo	renew
0	0% 0	0% 0	0% 0	0% 6	16% 0
setclntid	confirm	lock	lockt	locku	access
3	8% 3	8% 0	0% 0	0% 0	0% 3
getattr	lookup	lookup_root	remove	rename	link
3	8% 3	8% 3	8% 0	0% 0	0% 0
symlink	create	pathconf	statfs	readlink	readdir
0	0% 0	0% 3	8% 0	0% 0	0% 0
server_caps	delegreturn	getacl	setacl	fs_locations	rel_lkowner
9	25% 0	0% 0	0% 0	0% 0	0% 0
secinfo	exchange_id	create_ses	destroy_ses	sequence	get_lease_t
0	0% 0	0% 0	0% 0	0% 0	0% 0
reclaim_comp	layoutget	getdevinfo	layoutcommit	layoutreturn	getdevlist
0	0% 0	0% 0	0% 0	0% 0	0% 0
(null)					
0	0%				

<html>

Copyright © 2020 Hugh Norris

</html>
