

TP NFS

1 PREREQUIS

1.1 Configuration des IP

À cause du pare-feu de l'Efrei, nous ne pouvions pas ping nos machine entre nous.

Nous utilisons donc notre machine (avec WSL) et une machine virtuelle.

Sur WSL (Ubuntu) :

```
# TP Admin linux
192.168.46.2    tp
```

```
thibaut@DESKTOP-U29NSUE:~$ ping tp
PING tp (192.168.46.2) 56(84) bytes of data.
64 bytes from tp (192.168.46.2): icmp_seq=1 ttl=63 time=1.44 ms
64 bytes from tp (192.168.46.2): icmp_seq=2 ttl=63 time=1.37 ms
```

Sur la machine virtuelle VirtualBox (Ubuntu) :

```
# TP Admin linux
192.168.46.1    tp
```

```
thibaut@Ubuntu:~$ ping tp
PING tp (192.168.46.1) 56(84) bytes of data.
64 bytes from tp (192.168.46.1): icmp_seq=1 ttl=128 time=0.533 ms
64 bytes from tp (192.168.46.1): icmp_seq=2 ttl=128 time=1.08 ms
```

1.2 Ajout des utilisateurs

Sur les deux machines, nous ajoutons les utilisateurs :

```
root@Ubuntu:~# useradd tann -u 1100
root@Ubuntu:~# useradd colnot -u 1101
```

On définit les mots de passes avec passwd

```
# passwd tann
# passwd colnot
```

Sur la machine virtuelle :

```
root@Ubuntu:~# getent passwd | grep tann
tann:x:1100:1100::/home/tann:/bin/sh
root@Ubuntu:~# getent passwd | grep colnot
colnot:x:1101:1101::/home/colnot:/bin/sh
```

Sur WSL :

```
thibaut@DESKTOP-U29NSUE:~$ getent passwd | grep tann
tann:x:1100:1100::/home/tann:/bin/sh
thibaut@DESKTOP-U29NSUE:~$ getent passwd | grep colnot
colnot:x:1101:1101::/home/colnot:/bin/sh
```

1.3 Configuration connexion ssh

Installation d'un serveur ssh : sur chaque machine

```
sudo apt install openssh-server # installe le serveur
sudo ufw allow ssh # autorise le pare-feu
```

2 CONFIGURATION SERVEUR

2.1 Installation de NFS

Installons d'abord NFS sur les deux machines. Nous suivons pour ceci la documentation ubuntu.

```
sudo apt install nfs-common
sudo apt install nfs-kernel-server
```

Normalement, `rpcbind` est déjà installé sur les deux.

2.2 Vérification de l'installation des paquets et de l'état des services

Vérifions-le :

```
thibaut@ubuntu:~$ dpkg --get-selections | grep rpcbind
ii rpcbind 1.2.6-2build1 amd64 converts RPC program numbers into universal addresses
thibaut@ubuntu:~$ dpkg --get-selections | grep nfs-common
ii nfs-common 1:2.0-1ubuntu1.2 amd64 NFS support files common to client and server
thibaut@ubuntu:~$ dpkg --get-selections | grep nfs-kernel-server
ii nfs-kernel-server 1:2.0-1ubuntu1.2 amd64 support for NFS kernel server
```

Les services `rpcbind` ainsi que `nfs-server` sont bien actifs :

```
thibaut@DESKTOP-U29NSUE:~$ systemctl status rpcbind
● rpcbind.service - RPC bind portmap service
   Loaded: loaded (/lib/systemd/system/rpcbind.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-05-16 10:18:08 CEST; 32min ago
   TriggeredBy: ● rpcbind.socket
     Docs: man:rpcbind(8)
    Main PID: 37028 (rpcbind)
       Tasks: 1 (limit: 9062)
      Memory: 612.0K
    CGroup: /system.slice/rpcbind.service
            └─37028 /sbin/rpcbind -f -w

May 16 10:18:08 DESKTOP-U29NSUE systemd[1]: Starting RPC bind portmap service...
May 16 10:18:08 DESKTOP-U29NSUE systemd[1]: Started RPC bind portmap service.
thibaut@DESKTOP-U29NSUE:~$ systemctl status nfs-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled; vendor preset: enabled)
   Active: active (exited) since Thu 2024-05-16 10:47:21 CEST; 3min 16s ago
   Main PID: 49725 (code=exited, status=0/SUCCESS)

May 16 10:47:21 DESKTOP-U29NSUE systemd[1]: Starting NFS server and services...
May 16 10:47:21 DESKTOP-U29NSUE exportfs[49724]: exportfs: can't open /etc/exports for reading
May 16 10:47:21 DESKTOP-U29NSUE systemd[1]: Finished NFS server and services.
```

2.3 Export des répertoires home

À partir de ce point, nous considérons WSL comme le client et la machine VirtualBox comme le serveur.

On édite d'abord `/etc/exports` afin d'exporter les répertoires home à tous les IP.

```
GNU nano 6.2 /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/krb5l(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5l(rw,sync,no_subtree_check)
#
/home/tann *(rw)
/home/colnot *(rw)
```

Ensuite, on exporte et on redémarre les services :

```
root@Ubuntu:~# exportfs -a
exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check' specified for export "*/home/tann".
    Assuming default behaviour ('no_subtree_check').
    NOTE: this default has changed since nfs-utils version 1.0.x

exportfs: /etc/exports [3]: Neither 'subtree_check' or 'no_subtree_check' specified for export "*/home/colnot".
    Assuming default behaviour ('no_subtree_check').
    NOTE: this default has changed since nfs-utils version 1.0.x

root@Ubuntu:~# systemctl restart rpcbind
root@Ubuntu:~# systemctl restart nfs-server
```

Vérifions enfin les exports avec `showmount` :

```
root@Ubuntu:~# showmount --export
Export list for Ubuntu:
/home/colnot *
/home/tann *
```

Note : nous aurions également pu utiliser `exportfs` sans arguments.

3 CONFIGURATION CLIENT

Le paquet `rpcbind` est déjà installé et le service est actif :

```
root@DESKTOP-U29NSUE:~# dpkg -l | grep rpcbind
ii rpcbind 1.2.6-2build1 amd64 converts RPC program to root@DESKTOP-U29NSUE:~#
root@DESKTOP-U29NSUE:~# systemctl status rpcbind
● rpcbind.service - RPC bind portmap service
   Loaded: loaded (/lib/systemd/system/rpcbind.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-05-16 11:12:18 CEST; 12min ago
     TriggeredBy: ● rpcbind.socket
    Docs: man:rpcbind(8)
   Main PID: 60180 (rpcbind)
      Tasks: 1 (limit: 9062)
     Memory: 560.0K
    CGroup: /system.slice/rpcbind.service
            └─60180 /sbin/rpcbind -f -w

May 16 11:12:18 DESKTOP-U29NSUE systemd[1]: Starting RPC bind portmap service...
May 16 11:12:18 DESKTOP-U29NSUE systemd[1]: Started RPC bind portmap service.
```

Créons le répertoire de destination `/mnt/FromNFS`.

Enfin, nous réalisons une opération de montage par NFS :

```
thibaut@Ubuntu:~$ sudo mount -t nfs tp:/home/tann /mnt/FromNFS
```

```
thibaut@Ubuntu:~$ sudo -u tann ls -l /mnt/FromNFS
total 0
-rw-r--r-- 1 tann tann 0 mai 16 11:05 test-nfs
thibaut@Ubuntu:~$
```

→ Le fichier « test-nfs » est bien présent.

Remarque : tann est bien le propriétaire du répertoire FromNFS et test-nfs.

```
thibaut@Ubuntu:~$ sudo -u tann ls -lR /mnt
/mnt:
total 4
drwxr-x--- 4 tann tann 4096 mai 16 22:46 FromNFS

/mnt/FromNFS:
total 0
-rw-r--r-- 1 tann tann 0 mai 16 11:05 test-nfs
```

4 ADMINISTRATION ET SECURITE

Nous choisissons de mettre une sécurité sur notre serveur par le biais de `/etc/exports`. Maintenant, nous n'autorisons que la machine de notre binôme (nommée « tp ») à monter l'arborescence des répertoires `/home/tann` et `/home/colnot` via nfs :

```
thibaut@Ubuntu:/$ 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/tann        tp(rw)
/home/colnot      tp(rw)
thibaut@Ubuntu:/$ sudo systemctl restart nfs-kernel-server
thibaut@Ubuntu:/$ sudo exportfs
/home/tann        tp
/home/colnot      tp
thibaut@Ubuntu:/$
```

Nous observons ensuite sur le client que « root » ne peut pas accéder au système de fichier monté :

```
thibaut@Ubuntu:~$ sudo ls /mnt/FromNFS/
ls: impossible d'ouvrir le répertoire '/mnt/FromNFS/': Permission non accordée
thibaut@Ubuntu:~$ sudo ls -l /mnt | grep FromNFS
drwxr-x--- 4 tann tann 4096 mai 16 22:46 FromNFS
```


Nous rajoutons maintenant le paramètre `no_root_squash` sur le serveur pour indiquer que l'utilisateur root du client possèdera tous les droits sur le système de fichier monté à distance :

```
thibaut@Ubuntu:/$ 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/tann        tp(rw,no_root_squash)
/home/colnot      tp(rw,no_root_squash)
thibaut@Ubuntu:/$ sudo systemctl restart nfs-kernel-server
thibaut@Ubuntu:/$
```

Sur le client, vérifions que c'est le cas :

```
thibaut@Ubuntu:~$ sudo ls /mnt/FromNFS/
test-nfs
thibaut@Ubuntu:~$ sudo ls -lR /mnt
/mnt:
total 4
drwxr-x--- 4 tann tann 4096 mai 16 22:46 FromNFS

/mnt/FromNFS:
total 0
-rw-r--r-- 1 tann tann 0 mai 16 11:05 test-nfs
thibaut@Ubuntu:~$
```

Puis tentons de modifier le fichier en tant que « tann », cela est possible :

```
thibaut@Ubuntu:/$ su tann
Mot de passe :
$ cat /mnt/FromNFS/test-nfs
$ nano /mnt/FromNFS/test-nfs
$ cat /mnt/FromNFS/test-nfs
Cette ligne a ete rajoutee par tann
```

Enfin, nous allons faire le nécessaire sur le client pour que le montage se fasse automatiquement au démarrage du système.

Pour cela, nous modifions le fichier `/etc/fstab` qui liste les partitions qui seront montées automatiquement au démarrage ou à la connexion du périphérique.

tp:/home/tann	/mnt/FromNFS	nfs	defaults	0	2
---------------	--------------	-----	----------	---	---

- Colonne 1 : nous /home/tann depuis l'hôte tp
- Colonne 2 : Le montage se fait dans /mnt/FromNFS
- Colonne 3 : Le montage est de type NFS

- Colonne 4 : Nous laissons les options par défaut (rw,suid,dev,exec,auto,nouser,async)
- Colonne 5 : = 0 : Nous ne faisons pas de sauvegardes via l'utilitaire dump
- Colonne 6 : = 2 : Il s'agit d'une partition Linux « externe »

Après un redémarrage de la machine, le montage s'est bien fait :

```
thibaut@Ubuntu:~$ sudo ls -lR /mnt
[sudo] Mot de passe de thibaut :
/mnt:
total 4
drwxr-x--- 4 tann tann 4096 mai 16 23:42 FromNFS

/mnt/FromNFS:
total 4
-rw-r--r-- 1 tann tann 36 mai 16 23:42 test-nfs
```