

Chapitre 6

Partage de dossiers et d'imprimantes

Le partage de dossiers et d'imprimantes permet de :

- réduire le coût d'investissement ;
- mutualiser les ressources.

Pour partager des dossiers entre des machines Unix/Linux on utilise le protocole NFS (Network File System).

Pour partager des dossiers et des imprimantes entre des machines Unix/Linux et des machines Windows on utilise le service samba.

Le protocole NFS

C'est un protocole qui fonctionne suivant le modèle client/serveur :

- un serveur met des dossiers à la disposition des machines sur le réseau suivant des droits d'accès.
- d'autres machines peuvent monter ces dossiers. Qui seront vus comme des dossiers locaux.

Il faut installer le serveur NFS, en tapant la commande :

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

Configuration

Le fichier de configuration est **/etc/exports**. On indique dans ce fichier la liste des répertoires à exporter (partagés) et les noms des machines autorisées à les utiliser.

Exemple :

```
/home machine1(rw,sync,no_subtree_check)
/home *(ro,sync,no_subtree_check)
/TP machine1(rw,sync,no_root_squash,no_subtree_check)
/test *(ro,sync,no_subtree_check)
```

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;
- toutes les autres machines du réseau peuvent monter **/home** en lecture seulement (**ro**) ;

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;
- toutes les autres machines du réseau peuvent monter **/home** en lecture seulement (**ro**) ;
- **machine1** peut monter **/TP** en lecture/écriture (**rw**) ;

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;
- toutes les autres machines du réseau peuvent monter **/home** en lecture seulement (**ro**) ;
- **machine1** peut monter **/TP** en lecture/écriture (**rw**) ;
- toutes les machines du réseau peuvent monter **/test** en lecture seule (**ro**).

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;
- toutes les autres machines du réseau peuvent monter **/home** en lecture seulement (**ro**) ;
- **machine1** peut monter **/TP** en lecture/écriture (**rw**) ;
- toutes les machines du réseau peuvent monter **/test** en lecture seule (**ro**).

On peut utiliser des noms ou adresses IP pour les machines.

Par exemple :

```
/home 192.10.1.1(rw, sync, no_subtree_check)
/TP   smi.ump.ma(ro, sync, no_subtree_check)
```

Une fois le fichier **/etc/exports** bien configurer il faut redémarrer (ou relancer) **nfs** :

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

ou bien

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

Affichage des répertoires partagés

La commande **exportfs** permet d'afficher les répertoires partagés.

Exemple :

```
# exportfs
```

affichera

```
/etc          192.168.56.1  
/var          192.168.56.1  
/test         <world>
```

Options

Principales options du fichier exports :

Option	Signification
ro	read-only (accès en lecture seule au répertoire exporté)

Principales options du fichier exports :

Option	Signification
ro	read-only (accès en lecture seule au répertoire exporté)
rw	read-write : le client accède au répertoire en lecture/écriture

Principales options du fichier exports :

Option	Signification
ro	read-only (accès en lecture seule au répertoire exporté)
rw	read-write : le client accède au répertoire en lecture/écriture
root_squash	convertit les UID/GID root, en utilisateur anonyme. L'administrateur de la machine cliente ne peut pas modifier le contenu des répertoires et des fichiers.

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.
all_squash	convertit tous les UID/GID en utilisateurs anonymes. Utile pour exporter avec NFS des répertoires publics.

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.
all_squash	convertit tous les UID/GID en utilisateurs anonymes. Utile pour exporter avec NFS des répertoires publics.
sync	ne répondre aux requêtes qu'après l'exécution de tous les changements sur le support réel.

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.
all_squash	convertit tous les UID/GID en utilisateurs anonymes. Utile pour exporter avec NFS des répertoires publics.
sync	ne répondre aux requêtes qu'après l'exécution de tous les changements sur le support réel.
no_subtree_check	annule la vérification des sous-répertoires

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.
all_squash	convertit tous les UID/GID en utilisateurs anonymes. Utile pour exporter avec NFS des répertoires publics.
sync	ne répondre aux requêtes qu'après l'exécution de tous les changements sur le support réel.
no_subtree_check	annule la vérification des sous-répertoires
fsid	définit le système de fichiers racine (fs : filesystem)

Les options doivent être séparées par des virgules, SANS ESPACE.
Pour plus d'options, veuillez consulter le manuel du fichier **exports**
(**man exports**).

Pour pouvoir monter des répertoires, il faut installer la package **nfs-common** :

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

En tant qu'administrateur du système, pour monter un répertoire distant, il faut utiliser la commande **mount** avec l'option **-t nfs**.

Exemple :

```
#mount -t nfs nom_machine:/home /test
```

ou

```
#mount -t nfs 192.168.56.2:/home /test
```

Remarque

L'option **-t nfs** n'est pas obligatoire :

```
#mount 192.168.56.2:/home /test
```


Remarque

L'option **-t nfs** n'est pas obligatoire :

```
#mount 192.168.56.2:/home /test
```

La commande **df -h** permet de d'afficher les répertoires montés (locaux et distants)

Création des répertoires :

- /partageNFS
- /partageNFS/TP
- /partageNFS/cours
- /partageNFS/documents

Ajouter dans le fichier **/etc/exports** les lignes suivantes :

```
/partageNFS *(ro,fsid=0,no_subtree_check)
/partageNFS/TP 192.168.56.0/24(rw,sync,no_subtree_check)
/partageNFS/cours 192.168.56.0/24(rw,sync,no_subtree_check)
/partageNFS/documents 192.168.56.0/24(rw,sync,no_subtree_check)
```

Pour activer les modifications, il suffit d'utiliser la commande :

sudo exportfs -a

Montage des répertoires

Au lieu d'utiliser la commande

```
#mount 192.168.56.2:/partageNFS/documents /documents
```

Il suffit de faire :

```
#mount 192.168.56.2:/documents /documents
```

puisque, le répertoire **/partageNFS** est défini comme racine.

Montage au démarrage

Pour monter un répertoire au démarrage du système, il suffit d'ajouter les renseignements nécessaire au fichier **/etc/fstab**

Par exemple :

```
nom_machine:/home /home nfs auto,rw,user 0 0
```

NFS n'est pas un protocole très sécurisé :

- l'authentification des clients repose uniquement sur le nom de domaine ou l'adresse IP ;

NFS n'est pas un protocole très sécurisé :

- l'authentification des clients repose uniquement sur le nom de domaine ou l'adresse IP ;
- l'identification des utilisateurs repose sur le « user id » sur le poste client => usurpation possible ;

NFS n'est pas un protocole très sécurisé :

- l'authentification des clients repose uniquement sur le nom de domaine ou l'adresse IP ;
- l'identification des utilisateurs repose sur le « user id » sur le poste client => usurpation possible ;
- le transfert des données est non crypté ;

NFS n'est pas un protocole très sécurisé :

- l'authentification des clients repose uniquement sur le nom de domaine ou l'adresse IP ;
- l'identification des utilisateurs repose sur le « user id » sur le poste client => usurpation possible ;
- le transfert des données est non crypté ;
- utilisation recommandée en intranet isolé, protégé de l'internet par un Firewall.

Le protocole SAMBA

Samba peut être utilisé pour partager des dossiers et des imprimantes entre des machines Unix/Linux et des machines Windows.

En ligne de commandes, il suffit de taper la commande :

```
#sudo apt-get install samba
```

Le fichier principal de configuration de samba est :
/etc/samba/smb.conf.

Avant de modifier le fichier de configuration, il faut le sauvegarder par prudence (`cp smb.conf smb.conf.old`).

Ce fichier est organisé en sections. L'administrateur **root** peut éditer, modifier et ajouter des sections, pour définir de nouvelles ressources à partager.

Une section commence par un mot entre crochets et se termine lorsqu'une autre section commence.

Exemple de sections :

[global]

#ensemble de directives

[homes]

#ensemble de directives

Remarque :

D'une façon générale, les permissions de partage définies dans les sections ne peuvent pas dépasser les permissions des fichiers du serveur hôte.

Pour plus d'informations concernant le fichier `smb.conf`, veuillez consulter le manuel en ligne :

```
man smb.conf
```

Vérification des changements

L'outil **testparm**, permet de tester la syntaxe du fichier de configuration et de détecter les erreurs. Il est recommandé de le lancer systématiquement lors de la modification de **smb.conf**.

Activation des changements

A chaque changement effectué dans **smb.conf**, il faut relancer les démons **smbd** et **nmdbd** :

```
#service smbd restart
```

suivie de :

```
#service nmdbd restart
```

Activation des changements

A chaque changement effectué dans **smb.conf**, il faut relancer les démons **smbd** et **nmdbd** :

```
#service smbd restart
```

suivie de :

```
#service nmdbd restart
```

Pour les nouveaux systèmes :

```
systemctl restart smbd
```

suivie de

```
systemctl restart nmdbd
```


Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client
guest ok = yes no	no	permettre l'accès sans authentification

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client
guest ok = yes no	no	permettre l'accès sans authentification
valid users =	tous	liste des utilisateurs autorisés à se connecter à la ressource

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client
guest ok = yes no	no	permettre l'accès sans authentification
valid users =	tous	liste des utilisateurs autorisés à se connecter à la ressource
printable = true false	false	partage d'un service d'impression et non d'un répertoire.

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client
guest ok = yes no	no	permettre l'accès sans authentification
valid users =	tous	liste des utilisateurs autorisés à se connecter à la ressource
printable = true false	false	partage d'un service d'impression et non d'un répertoire.
writable = yes no	no	permet ou non l'écriture sur le répertoire, contraire de read only

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
browseable =	yes	visibilité du partage par tous, même les utilisateurs non autorisés

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
browseable =	yes	visibilité du partage par tous, même les utilisateurs non autorisés
create mask =	0744	droits maxi accordés à un fichier créé dans la ressource ces droits seront en intersection (and) avec les droits Linux (umask)

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
browseable =	yes	visibilité du partage par tous, même les utilisateurs non autorisés
create mask =	0744	droits maxi accordés à un fichier créé dans la ressource ces droits seront en intersection (and) avec les droits Linux (umask)
directory mask =	0755	droits maxi accordés à un répertoire créé dans la ressource ces droits seront en intersection (and) avec les droits Linux (umask)

Commande umask

L'umask permet d'attribuer des permissions aux fichiers et répertoires créés par l'utilisateur. Il se présente sous la forme de 4 chiffres. La valeur par défaut de l'umask est 0022. Pour obtenir les permissions qui seront utilisées, il faut appliquer la règle suivante :

- pour les fichiers, il faut soustraire le umask de 666.
Par exemple $666 - 0022 = 644$ ce qui donne les droits
rw-r--r--

Commande umask

L'umask permet d'attribuer des permissions aux fichiers et répertoires créés par l'utilisateur. Il se présente sous la forme de 4 chiffres. La valeur par défaut de l'umask est 0022. Pour obtenir les permissions qui seront utilisées, il faut appliquer la règle suivante :

- pour les fichiers, il faut soustraire le umask de 666.
Par exemple $666 - 0022 = 644$ ce qui donne les droits
rw-r--r--
- pour les répertoires, il faut soustraire le umask de 777.
Par exemple $777 - 0022 = 755$ ce qui donne les droits
rwxr-xr-x

Commande umask (suite)

Si l'utilisateur veut que les nouveaux fichiers soient créés avec les droits **rw-----** et que les nouveaux répertoires soient créés avec les droits **rw-x-----**, il doit utiliser le masque **0077**. Pour cela, il doit taper la commande :

```
umask 0077
```

ou tout simplement :

```
umask 77
```

Commande umask (suite)

Si l'utilisateur veut que les nouveaux fichiers soient créés avec les droits **rw-----** et que les nouveaux répertoires soient créés avec les droits **rw-x-----**, il doit utiliser le masque **0077**. Pour cela, il doit taper la commande :

```
umask 0077
```

ou tout simplement :

```
umask 77
```

Remarque

umask accepte les symboles (r, w et x) comme **chmod**

umask 77 peut être utilisée comme suit :

```
umask u=rwx,g=,o=
```

La section globale

```
[global]
```

```
# donner le meme nom de groupe de travail que celui  
des stations Windows (Voisinage reseau/  
identification)
```

```
workgroup = SMI
```

```
# restreindre par sécurité les sous-réseaux autorisés à se connecter  
au serveur
```

```
# ici on se limite aux adresses réseau privé 192.168.1.0 et à  
l'interface "loopback"
```

```
hosts allow = 192.168.1. 127.
```

```
# on peut exclure des machines de l'accès au réseau
```

```
hosts allow = 192.168.1. EXCEPT 192.168.1.125
```

```
# d'autres possibilités existent : voir le manuel man smb
```

Le répertoire personnel

[homes]

#accès au répertoire personnel de chaque utilisateur.

#la valeur du champ "comment" apparaîtra dans le voisinage réseau

#inutile pour cette section de préciser le path, c'est celui de
l'utilisateur, en fait /home/%u

browsable = no

writable = yes

create mode = 0700

Rendre un répertoire public en lecture seule

Pour rendre un répertoire accessible par tous le monde, il faut tout d'abord le créer ou vérifier qu'il existe.

```
# mkdir /home/partage
```

```
ls -ld /home/partage
```

doit renvoyer les droits par défaut `drwxr-xr-x`, sinon il faut les changer en tapant la commande :

```
chmod 755 /home/partage
```

ou son équivalent

```
chmod u=rwx,go=rx /home/partage
```

pour y ajouter les permissions d'accès et de lecture pour tous.

Rendre un répertoire public en lecture seule

Ensuite, il faut ajouter une nouvelle section « **[partage]** » comme suit :

```
[partage]  
path = /home/partage  
browsable = yes  
writable = no  
guest ok = yes
```

Rendre un répertoire public en lecture et écriture

Si on veut rendre ce répertoire partagé en écriture aussi, il faut modifier les droits d'accès du répertoire et modifier la section [partage] :

```
chmod 777 /home/partage
```

```
[partage]  
path = /home/partage  
browsable = yes  
writable = yes  
guest ok = yes  
create mode = 0755
```

`testparm` : permet la validation du fichier de configuration de Samba.

- `testparm` : permet la validation du fichier de configuration de Samba.
- `smbclient` : client Linux/Unix similaire à FTP permettant de se connecter à des partages Samba.

- `testparm` : permet la validation du fichier de configuration de Samba.
- `smbclient` : client Linux/Unix similaire à FTP permettant de se connecter à des partages Samba.
- `smbpasswd` : permet à un administrateur de modifier les mots de passe chiffrés utilisés par Samba.

- testparm** : permet la validation du fichier de configuration de Samba.
- smbclient** : client Linux/Unix similaire à FTP permettant de se connecter à des partages Samba.
- smbpasswd** : permet à un administrateur de modifier les mots de passe chiffrés utilisés par Samba.
- smbstatus** : dresse l'état des connexions aux partages d'un serveur Samba.

Ajout d'un utilisateur samba

Pour permettre à un utilisateur de se connecter à son répertoire personnel à partir d'autres machines, il faut l'ajouter en tant qu'utilisateur samba en tapant la commande suivante :

smbpasswd -a utilisateur

Par exemple :

smbpasswd -a smi

Problème de connexion avec Windwos

Sous Windows, si on se connecte avec un utilisateur 1 (par exemple **smi**) et on veut se connecter avec un autre utilisateur (par exemple **sma**), la connexion ne réussisse pas. Pour cela, il faut supprimer la connexion à l'utilisateur 1, en tapant la commande (sous un invite de commande) :

```
net use \\nom-partage\utilisateur /delete
```

Par exemple :

```
net use \\Ubuntu\smi /delete
```

Ou **Ubuntu** est le nom de partage et **smi** est le nom de l'utilisateur déjà connecté.

Connexion à partir d'un client Linux

Pour se connecter à partir d'un client Linux en utilisant la commande **smbclient**, il faut taper la commande

```
smbclient //nom-machine/repertoire
```

Par exemple :

```
smbclient //192.168.56.2/partage
```

Connexion à partir d'un client Linux

Pour se connecter à partir d'un client Linux en utilisant la commande **smbclient**, il faut taper la commande

```
smbclient //nom-machine/repertoire
```

Par exemple :

```
smbclient //192.168.56.2/partage
```

Pour se connecter en utilisant le compte d'un utilisateur qui s'appelle **smi**, il faut taper la commande :

```
smbclient //192.168.56.2/smi -U smi
```

ensuite, on saisit le mot de passe.

Connexion à partir d'un client Linux

On peut utiliser la commande **mount** pour monter un répertoire partagé (fonctionne sous root) :

```
mount -o username=smi //192.168.56.2/smi Rep
```

Ou **Rep** est le répertoire de montage.