

"Mountages" en tout genre

Olivier Hoarau (olivier.hoarau@funix.org)

V2.1 du 4 septembre 2010

1 Historique du document.....	2
2 Préambule.....	2
3 Présentation.....	3
4 La commande mount.....	3
5 Le fichier /etc/fstab.....	4
6 Montage NFS.....	5
6.1 Présentation.....	5
6.2 Configurer la partie serveur NFS.....	5
6.2.1 Installation pour une Mandriva.....	5
6.2.2 Installation pour une (k)ubuntu.....	5
6.2.3 Configurer la partie serveur NFS.....	5
6.3 Configuration de la partie cliente NFS.....	6
7 Monter des partages Windows.....	6
8 Automontage.....	7
8.1 Présentation.....	7
8.2 Installation.....	8
8.3 Configuration de la machine serveur.....	8
8.4 Configuration de l'automontage sur la machine cliente.....	8
8.5 Utilisation.....	9

1 Historique du document

V2.1	4.09.10	modification suite passage nfs V4
V2.0	03.11.06	Installation automontage sous ubuntu
V1.9	26.07.06	Installation du serveur NFS sous ubuntu
V1.8	04.05.03	Légères modifications suite passage Mandrake 9.1
V1.7	24.12.02	Légères modifications suite passage Mandrake 9.0 (packages pour smbmount et supermount)
V1.6	09.06.02	Légères modifications suite passage Mandrake 8.2 (supermount, syntaxe de smbmount)
V1.5	22.02.02	Activer le supermount avec la Mandrake 8.1
V1.4	16.12.01	Légères modifications suite passage à Mandrake 8.1
V1.3	17.06.01	Légères modifications suite passage à Mandrake 8.0
V1.2	3.12.00	Modification de version des différents packages.

2 Préambule

Ce document présente les différents type de montage de filesystem qu'on peut trouver sous Linux.

La dernière version de ce document est téléchargeable à l'URL <http://www.funix.org>.

Ce document est sous licence Creative Commons Attribution-ShareAlike 3.0 Unported, le détail de la licence se trouve sur le site <http://creativecommons.org/licenses/by-sa/3.0/legalcode>. Pour résumer, vous êtes libres

- de reproduire, distribuer et communiquer cette création au public
- de modifier cette création

suivant les conditions suivantes:

- **Paternité** — Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'oeuvre).
- **Partage des Conditions Initiales à l'Identique** — Si vous transformez ou modifiez cette oeuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

Par ailleurs ce document ne peut pas être utilisé dans un but commercial sans le consentement de son auteur. Ce document vous est fourni "dans l'état" sans aucune garantie de toute sorte, l'auteur ne saurait être tenu responsable des quelconques misères qui pourraient vous arriver lors des manipulations décrites dans ce document.

3 Présentation

Le montage permet d'accéder à des systèmes de fichiers (filesystem), ceux-ci peuvent être des partitions locales, mais aussi distantes, des disquettes, des jaz, des cdrom et j'en passe. Dans cette page je présente le moyen de monter un filesystem local ou distant manuellement ou automatiquement (automount), que ce soit un filesystem linux ou windows.

4 La commande mount

Ce paragraphe n'a pas pour objet de présenter le principe du "montage", pour cela voir mon cours d'administration système HP-UX, mais plutôt de présenter les options les plus intéressantes. La syntaxe de la commande est la suivante:

mount -f type-de-filesystem -o options filesystem-à-monter répertoire-de-montage

Les types les plus courants sont:

- **ext2** (type par défaut), c'est le filesystem linux par défaut,
- **vfat**, FAT16 ou FAT32 de Win95 ou Win98,
- **nfs**, filesystem distant se trouvant sur une autre machine,

Sans la mention de **-o** options, on a les options par défaut, qui sont:

- **rw**, lecture et écriture
- **suid**, prise en compte les bits Set-UID ou Set-GID des fichiers se trouvant sur le système monté,
- **dev**, interpréter les fichiers spéciaux de périphériques présents sur le système de fichiers (voir le cours unix pour la signification de fichiers spéciaux),
- **exec**, permet l'exécution de fichiers binaires,
- **async** toutes les entrées/sorties sur le système de fichiers seront asynchrones (comprenez qui pourra ;-) ...)

D'autres options existent qu'on peut illustrer par cet exemple:

mount -t vfat -o uid=5001,gid=5000,umask=022 /dev/hda1 /windoze

Cette commande permet de monter la partition Windows identifiée par **/dev/hda1** dans le répertoire **/windoze**, l'utilisateur avec l'uid 5001, et appartenant au groupe de gid 5000, sera propriétaire de tous les fichiers. Quand ce dernier créera un fichier, celui-ci aura pour droit 755 (rwxr-xr-x) qui correspond à un umask de 022 en notation octale, pour simplifier 0 correspond à 7 (rwx), 2 à 5(r-x), 1 à 6(rw-), et 3 à 4(r--).

Pour monter un file system **nfs** (répertoire **/alphonse** de la station distante **obelix**) sur le répertoire d'accrochage **/alphonse**, on tapera:

mount -t nfs obelix:/alphonse /alphonse

Pour monter une disquette DOS, on tapera:

mount -t vfat /dev/fd0 /mnt/floppy

NOTE: Le répertoire d'accrochage doit exister, vous devez préalablement le créer avant montage.

La commande umount sert à démonter le filesystem, elle a deux syntaxes possibles:

umount filesystem

ou

umount répertoire-d'accrochage

Généralement on appellera **mount** dans une ligne de commande que dans des cas bien précis, on peut tout mettre dans un fichier appelé au boot ou à l'action de la commande **mount -a**.

5 Le fichier /etc/fstab

Le fichier possède autant de lignes qu'il y a de filesystem à monter. La syntaxe pour une ligne est la suivante:

filesystem **répertoire-d'accrochage** **type de filesystem** **options de montage**
paramètre pour dump (commande de sauvegarde) **paramètre pour fsck**

Pour le paramètre de **dump**, si on a 0 cela signifie que le filesystem en question ne sera pas sauvegardé à l'appel de dump. Il le sera dans le cas contraire.

Le dernier paramètre correspond à l'ordre dans lequel **fsck** devra vérifier les filesystems, le premier (celui avec numéro 1), doit être nécessairement la partition /, les autres peuvent être à 2, pour ne demander de vérification, il suffit de mettre 0.

Quelques exemples maintenant:

Montage d'une partition windows dans le répertoire **/windoze**, l'utilisateur ayant l'uid 5001 et appartenant au groupe d'uid 5000 en sera le propriétaire, les fichiers créés dans cette partition auront pour droit 755 (umask=022 voir plus haut). Les options par défaut sont: **rw,suid, dev, exec, auto, nouser**, et **async**.

```
/dev/hda1            /windoze            vfat    uid=5001,gid=5000,umask=022,defaults    0 0
```

Montage d'une disquette dans le répertoire **/mnt/floppy**:

```
/dev/fd0            /mnt/floppy            auto    sync,user,noauto,nosuid,nodev,unhide    0 0
```

Le type de filesystem **auto** permet en fait de tester les types suivants : **minix, ext, ext2, xia, iso9660**, il n'y a pas **vfat** dans la liste, c'est pourquoi je vous conseille de changer **auto** en **vfat**, si vous n'avez l'habitude de monter que des disquettes DOS.

L'option **user** permet qu'un utilisateur ordinaire non root, puisse monter la disquette, pour cela il lui suffira de taper:

```
mount /dev/fd0 ou /mount /mnt/floppy
```

L'option **noauto** permet que la disquette ne soit pas montée au boot, il faut explicitement taper une des deux commandes citées précédemment. On rajoute les options **nodev** (pas d'interprétation des fichiers spéciaux) et **nosuid** (pas de prise en compte des bit suid) pour des raisons de sécurité. L'option **unhide** permet de visualiser aussi les fichiers cachés.

Pour monter un filesystem **nfs**, on tapera la ligne:

```
obelix:/alphonse /alphonse    nfs    defaults 0 0
```

Une fois que **/etc/fstab** a été modifiée, il faut soit rebooter pour prendre en compte les modifications, ou tout simplement taper:

mount -a

Tous les filesystems indiqués dans le fichier seront montés sauf ceux qui possèdent l'option **noauto**, qui demande la saisie explicite de la commande **mount** (exemple plus haut avec la disquette).

6 Montage NFS

6.1 Présentation

NFS ou Network File System (initialement un produit SUN) permet de partager des disques entre stations UNIX, voire entre serveur NFS UNIX et client Windows PC-NFS.

ATTENTION: Vous devez faire en sorte que les uid et gid de vos utilisateurs soient strictement identiques entre votre serveur NFS et vos clients NFS, le mieux est de configurer un domaine NIS.

6.2 Configurer la partie serveur NFS

6.2.1 Installation pour une Mandriva

Pour cela on doit installer les packages **nfs-utils-clients** et **nfs-utils** par ailleurs il vous faudra installer préalablement **portmap**.

Pour lancer les services NFS il suffit maintenant de taper

```
/etc/rc.d/init.d/nfs-common start
```

```
/etc/rc.d/init.d/rpcbind start
```

```
/etc/rc.d/init.d/nfs-server start
```

6.2.2 Installation pour une (k)ubuntu

Pour une ubuntu vous devez installer les packages **nfs-kernel-server** et **nfs-common** avec **apt-get install** qui va lancer le daemon dans la foulée. Vous devez également manuellement créer le répertoire **/var/lib/nfs/v4recovery**.

6.2.3 Configurer la partie serveur NFS

Pour cela on doit configurer la partie serveur, le poste linux qui partagera ses répertoires.

Maintenant nous allons configurer les répertoires à exporter et vers quelles machines. Tout cela se passe dans le fichier **/etc/exports** de la machine serveur NFS qui a pour nom **obelix**, admettons que vouliez exporter "partager" votre répertoire **/alphonse** à la station **zoulou** en lecture écriture (rw) et en lecture seule (ro) à **shuttle**, le fichier contiendra donc:

```
/alphonse zoulou(rw) shuttle(ro)
```

Pour que le fichier soit pris en compte, en tant que root vous devez taper:

```
exportfs -a
```

NOTE: A chaque modification de ce fichier, vous devrez taper cette commande pour que la modification soit prise en compte. Pour avoir une idée des répertoires exportés, il suffit de taper

```
exportfs
```

6.3 Configuration de la partie cliente NFS

Sur une machine cliente NFS pour monter des filesystems exportés par un serveur NFS, vous devez modifier le fichier `/etc/fstab`. En reprenant notre exemple, sur la machine cliente zoulou, cela donnera la ligne supplémentaire (en plus des montages de partitions habituelles):

```
obelix:/alphonse /alphonse nfs defaults 0 0
```

`/alphonse` (deuxième champ de la ligne) étant le point de montage sur **zoulou**, c'est un répertoire à créer.

ATTENTION: Ce répertoire doit être vide avant tout montage.

Pour prendre en compte le montage vous devez taper:

```
mount -a
```

ATTENTION dans le cas de NFSv4 (mandriva 2010 spring par exemple) il faudra sur le client éditer le fichier `/etc/sysconfig/nfs-common` et modifiez la ligne suivante comme ceci

```
NEED_IDMAPD=yes
```

7 Monter des partages Windows

On a la possibilité de monter sous linux des partages déclarés sous Windows 95/98, en d'autres termes monter des partitions windows d'un poste windows distant. Tout d'abord déclarer le partage sous Windows, nom du poste **windoze**, aller dans le l'explorateur, cliquez sur la partition à partager (**d:** par exemple), puis sur le bouton droit de la souris, puis **partager**, donner comme nom **marcel**. Donner éventuellement un mot de passe pour l'accès complet (lecture et écriture). Si la commande **partager** n'existe pas, vous devez d'une part au niveau des propriétés réseau charger le service "**Partage des fichiers et imprimantes pour les réseaux Microsoft**", vous devez ensuite choisir une ouverture de session principale de type "**Ouverture de session windows**" et activer le partage de fichiers et imprimantes.

Vous avez besoin du package **samba-client** pour bénéficier de la commande **smbmount**. Vous devez installer préalablement dans l'ordre **libattr1**, **libacl1** et **samba-common**.

Pour monter le partage sous `/mnt/marcel`, tapez maintenant en tant que root la commande:

Partage déclaré en **Accès complet** sans mot de passe

```
smbmount //windoze/marcel /mnt/marcel/
```

Un mot de passe vous sera demandé c'est celui de root, pour éviter d'avoir à rentrer un mot de passe il suffit de taper :

```
smbmount //windoze/marcel /mnt/marcel/ -o guest
```

Un rapide coup d'œil sous `/mnt/marcel` et on voit que les fichiers et répertoires sont à 755 et le propriétaire est root (de même que le groupe), en d'autres termes seul lui pourra écrire dans la partition.

Pour spécifier un utilisateur précis, par exemple celui qui a l'UID 500 et qui fait parti du groupe ayant le GID 500, on tapera :

```
smbmount //windoze/marcel /mnt/marcel/ -o uid=501,gid=500,guest
```

Si vous voulez que tous les utilisateurs du groupe 500, puissent écrire dans le partage monté, il faut changer les droits des fichiers (**fmask**) et ceux des répertoires (**dmask**), en les mettant à 775 par exemple.

```
smbmount //windoze/marcel /mnt/marcel/ -o uid=501,gid=500,fmask=775,dmask=775,guest
```

Un autre utilisateur pourra donc écrire dans le partage mais le fichier appartiendra toujours à l'utilisateur d'UID 500.

Partage déclaré en **Accès selon mot de passe** avec mot de passe

En récapitulant ce qu'on vient de dire précédemment, cela nous donne la commande :

```
smbmount //windoze/marcel /mnt/marcel/ -o password=mot-de-passe,uid=501,gid=500,fmask=775,dmask=775,guest
```

Pour démonter il suffira de taper :

```
umount /mnt/marcel
```

Il sera peut être nécessaire de spécifier le compte windows de connexion, comme ceci

```
smbmount //windoze/marcel /mnt/marcel/ -o username=compte-windows password=mot-de-passe,uid=501,gid=500,fmask=775,dmask=775,guest
```

NOTE Sur des anciennes versions de **smbmount** (2.0.3) qu'on retrouvait sur la mandrake 6.0 ou la Red Hat 6.0, la syntaxe est la suivante :

```
smbmount //windoze/marcel -U toto%mot-de-passe -c "mount /marcel -g 500 -u 501"
```

On va donc monter le partage dans le répertoire **/marcel**, l'utilisateur **toto** (uid 501, du groupe ayant le gid 500) sera propriétaire des fichiers. Avec la RH6.1 et la Mdk6.1, **smbmount** est passé à la version 2.0.5a, l'option **-c** a disparu, du coup pas moyen de lui spécifier un nom d'utilisateur propriétaire des fichiers.

Pour info **smbmount** se trouve dans le package **samba-client**.

8 Automontage

8.1 Présentation

L'automonteur est un process qui monte et démonte les file systems automatiquement sans avoir à taper la commande mount, les filesystems concernés peuvent être locaux (cdrom, disquette, zip, jaz) ou distants (partage NFS, c'est à dire filesystem d'un poste linux distant). Si le filesystem en question est démonté et qu'un utilisateur essaie d'y accéder, celui-ci sera automatiquement monté. D'un autre côté si le filesystem n'a pas été accédé depuis un certain temps, le daemon automonter le démontrera automatiquement. L'automontage est intéressant quand les filesystems concernés ne sont pas forcément disponibles (zip non présent, serveur NFS distant éteint, ...), avec un montage NFS classique vous auriez une erreur, voire un blocage complet de la machine en recherche du filesystem à monter, l'automontage permet d'éviter ces blocages, car les filesystems ne sont montés que quand on les utilise.

L'automonteur doit être installé sur la machine cliente, celle où on souhaite monter et démonter des filesystems (locaux ou distants). Sur la machine serveur, qui partage ses filesystems, on se contentera de configurer correctement **/etc/exports**.

8.2 Installation

Pour installer l'automonteur, vous devez installer le package **autofs** pour une Mandriva sur la machine cliente qui requière préalablement l'installation du package **nfs-utils-client**. Ce dernier en plus d'installer l'exécutable **/usr/sbin/automount**, installe le fichier de lancement **autofs** sous **/etc/rc.d/init.d**.

Pour ubuntu il faudra installer les packages **autofs**, **nfs-common** et **portmap**.

8.3 Configuration de la machine serveur

Vous devez configurer la machine qui souhaite partager ses filesystems, la configuration est en tout point identique à celle d'un serveur NFS. Vous devez installer le package NFS server (**nfs-utils** et **nfs-utils-client**) et configurer **/etc/exports**. Si l'on veut partager par exemple le répertoire **/alphonse** et **/roger** aux machines **zoulou** en lecture, écriture et le répertoire **/roger** en lecture seule à **shuttle**, et le **/mnt/cdrom** (le répertoire de montage du CD-ROM) à tout le monde, voici le contenu du dit-fichier.

```
/alphonse zoulou(rw)  
/roger shuttle(ro)  
/mnt/cdrom
```

N'oubliez pas de taper en tant que root **exportfs -a** pour que le fichier soit pris en compte.

NOTE: C'est particulièrement intéressant de pouvoir exporter des répertoires comme **/mnt/zip**, **/mnt/jaz** si vous voulez partager votre lecteur zip ou jaz.

8.4 Configuration de l'automontage sur la machine cliente

Admettons que le serveur NFS se nomme **obelix**, sur la machine cliente, on voudra monter les filesystems nfs **/alphonse**, **/roger** et **/mnt/cdrom** dans **/obelix**. On souhaite donc avoir l'arborescence suivante sur la machine cliente après automontage:

```
      roger  
/obelix  alphonse  
          mnt      cdrom
```

Pour cela on va créer un fichier **/etc/auto.master** contenant:

```
/obelix /etc/auto.obelix --timeout=60
```

/obelix est donc le point de montage des différents filesystems exportés par **obelix**, **timeout** (en secondes) est le temps au bout duquel les filesystems seront automatiquement démontés, s'ils sont non utilisés (non accédés).

On crée ensuite le fichier **/etc/auto.obelix** contenant:

```
roger      -fstype=nfs      obelix:/roger  
alphonse  -fstype=nfs      obelix:/alphonse  
cdrom    -fstype=nfs      obelix:/mnt/cdrom
```

Ca y est, on peut lancer l'automonter sur la machine cliente en lançant:

`/etc/rc.d/init.d/autofs start`

8.5 Utilisation

A l'utilisation l'automontage est particulièrement déconcertant, allez dans le répertoire **/obelix**, et tapez **ls**, vous verrez qu'il est vide! Les répertoires **roger**, **alphonse** et **mmt/cdrom** n'ont pas été montés. Et pour cause vous n'y avez pas encore accédé, vous devez taper explicitement **cd /obelix/roger** pour que le répertoire **/roger** se monte. Au bout de 60 secondes, d'inactivité, le répertoire sera automatiquement démonté et **obelix** paraîtra à nouveau vide.

Ceci impose de connaître au préalable les noms des filesystems exportés.