Niveau : Utilisateur	Numéro de la Leçon	Dernière Modification
1/4	<progrecss 1/5 style=inline />	2020/01/30 03:28

Système de Fichiers

Le système de fichiers de Linux est organisé autour d'une arborescence unique ayant un point de départ appelé la **racine**, représenté par le caractère **/**. En dessous de cette racine se trouvent des répertoires contenant fichiers et sous-répertoires. L'organisation des répertoires est conforme à un standard, appelé le **Linux File Hierarchy System**.

Linux File Hierarchy System

- **/bin** : est une abréviation de ***bin****ary* ou binaires. Il contient des programmes tels ls.
- **/boot** : contient les fichiers nécessaires au démarrage du système.
- **/dev** : contient les nœuds utilisés pour accéder à tout type de matériel tel /dev/fd0 pour le lecteur de disquette. C'est le binaire *udev* qui se charge de créer et supprimer d'une manière dynamique les nœuds.
- **/etc** : contient des fichiers de configuration tels passwd pour les mots de passe et fstab qui est la liste des systèmes de fichiers à monter lors du démarrage du système.
- **/home** : contient les répertoires de chaque utilisateur sauf l’utilisateur root.
- **/lib** : contient les bibliothèques communes utilisées par les programmes ainsi que les modules.
- **/lost+found** : contient des fragments de fichiers endommagés et retrouvés par la commande *fsck*.
- **/media** : contient des répertoires pour chaque système de fichiers monté (accessible au système linux) tels floppy, cdrom etc.
- **/mnt** : contient des répertoires pour chaque système de fichiers monté temporairement par root.
- **/opt** : contient des applications optionnelles.
- **/proc** : contient un système de fichiers virtuel qui extrait de la mémoire les informations en cours de traitement. Le contenu des fichiers est créé dynamiquement lors de la consultation. Seul root peut consulter la totalité des informations dans le répertoire /proc.
- **/root** : le home de root, l’administrateur système
- **/sbin** : contient des binaires, donc programmes, pour l’administration du système local.
- **/selinux** : contient des fichiers propres à l'implémentation de SELINUX.
- **/srv** : contient des données pour les **s**e**rv**ices hébergés par le système tels ftp, bases de données, web etc.
- **/sys** : contient un système de fichiers virtuel dont le rôle est de décrire le matériel pour udev.

- **/tmp** : stocke des fichiers temporaires créés par des programmes.
- **/usr** : contient des commandes des utilisateurs dans /usr/bin, les HOWTO dans /usr/share/doc, les manuels dans /usr/share/man ainsi que d'autres entrées majeures.
- **/var** : contient des fichiers de taille variable.

Il existe trois types majeurs de fichier sous le système Linux :

- les fichiers normaux (ordinary files)
- les répertoires (directories)
- les fichiers spéciaux (special files ou Devices)

Le fichiers normaux sont des fichiers textes, des tableaux ou des exécutables.

La longueur du nom de fichier est limité à 255 caractères.

Il y a une distinction entres les majuscules et le minuscules.

Si le nom d'un fichier commence par un **.**, le fichier devient caché.

La commande mount

Pour que Linux soit informé de la présence d’un système de fichiers, ce système doit être monté. Pour monter un système de fichiers, on utilise la commande **mount** :

```
# mount /dev/<fichier_spécial> /mnt/<répertoire_cible>
```

ou **/dev/<fichier_spécial>** est le périphérique à monter et **/mnt/<répertoire_cible>** est le répertoire qui servira comme «fenêtre» pour visionner le contenu du système de fichiers. Ce répertoire doit impérativement exister avant d’essayer de monter le système de fichiers.

<note> Connectez-vous à votre machine virtuelle openSUSE en tant que **trainee** avec le mot de passe **trainee**. Ouvrez un terminal via les menus **Ordinateur > Plus d'applications > Accessoires > Terminal GNOME**. Tapez la commande **su -** et appuyez sur la touche ↵ Entrée. Indiquez le mot de passe **fenestros**. Vous êtes maintenant connecté en tant que l'administrateur **root** et vous pouvez reproduire les exemples qui suivent.

</note>

Dans le cas où la commande **mount** est utilisée sans options, le système retourne une liste de tous les systèmes de fichiers actuellement montés :

```
opensuse:/ # mount
devtmpfs on /dev type devtmpfs (rw,relatime,size=500816k,nr_inodes=125204,mode=755)
tmpfs on /dev/shm type tmpfs (rw,relatime)
devpts on /dev/pts type devpts (rw,relatime,gid=5,mode=620,ptmxmode=000)
/dev/sda2 on / type ext3 (rw,relatime,errors=continue,user_xattr,acl,commit=15,barrier=1,data=ordered)
proc on /proc type proc (rw,relatime)
sysfs on /sys type sysfs (rw,relatime)
debugfs on /sys/kernel/debug type debugfs (rw,relatime)
/dev/sda1 on /boot type ext3 (rw,relatime,errors=continue,user_xattr,acl,commit=15,barrier=1,data=ordered)
securityfs on /sys/kernel/security type securityfs (rw,relatime)
gvfs-fuse-daemon on /home/trainee/.gvfs type fuse.gvfs-fuse-daemon
(rw,nosuid,nodev,relatime,user_id=1000,group_id=100)
```

Dans le cas où la commande **mount** est utilisée avec l’option **–a**, tous les systèmes de fichiers mentionnés dans un fichier spécial dénommé **/etc/fstab** seront montés en même temps :

```
opensuse:/ # cat /etc/fstab
/dev/disk/by-id/ata-VBOX_HARDDISK_VBb97f4bdd-e14419fa-part3 swap                swap       defaults
0 0
/dev/disk/by-id/ata-VBOX_HARDDISK_VBb97f4bdd-e14419fa-part2 /                   ext3       acl,user_xattr
1 1
/dev/disk/by-id/ata-VBOX_HARDDISK_VBb97f4bdd-e14419fa-part1 /boot               ext3       acl,user_xattr
1 2
proc                 /proc                proc       defaults              0 0
sysfs                /sys                 sysfs      noauto                0 0
debugfs              /sys/kernel/debug    debugfs    noauto                0 0
usbfs                /proc/bus/usb        usbfs      noauto                0 0
devpts               /dev/pts             devpts     mode=0620,gid=5       0 0
```

Options de la commande

Les options de la commande **mount** sont :

```
trainee@opensuse:~> mount --help
Utilisation : mount -V                 : afficher la version
              mount -h                 : afficher cette aide
              mount                    : lister les systèmes de fichiers montés
              mount -l                 : idem, incluant les étiquettes de volumes
Cela pour la partie informative. Suit ce qui porte sur le montage.
La commande est « mount [-t type-sys-fichier]  quoi  où ».
Les détails se trouvant dans /etc/fstab peuvent être omis.
              mount -a [-t|-O] ...     : monter tout ce qui est listé dans /etc/fstab
              mount périphérique       : monter le périphérique à l'endroit connu
              mount répertoire         : monter le périphérique connu ici
              mount -t type périph rép : commande de montage ordinaire
Noter qu'on ne monte pas réellement un périphérique, on monte
un système de fichiers (d'un type donné) trouvé sur le périphérique.
On peut aussi monter une arborescence de rép. déjà visible ailleurs :
              mount --bind ancien-rép nouveau-rép
ou déplacer une sous-arborescence:
              mount --move ancien-rép nouveau-rép
On peut changer le type de montage d'un rép. :
              mount --make-shared rép.
              mount --make-slave rép.
              mount --make-private rép.
              mount --make-unbindable rép.
On peut changer le type de tous les points de montage d'une
sous-arborescence contenue dans le rép.:
              mount --make-rshared rép.
              mount --make-rslave rép.
              mount --make-rprivate rép.
              mount --make-runbindable rép.
```

```
Un périphérique peut être nommé, comme /dev/hda1 ou /dev/cdrom, ou
repéré par l'étiquette, avec -L étiqu. ou par UUID, avec -U uuid .
Autres options: [-nfFrsvw] [-o options] [-p descr_fic_mots_passe].
Pour plus de précisions, consultez « man 8 mount ».
```

La commande umount

Pour démonter un système de fichiers, on utilise la commande umount :

```
# umount /mnt/<répertoire_cible>
```

Options de la commande

Les options de la commande **umount** sont :

```
trainee@opensuse:~> umount --help
Utilisation : umount -h | -V
       umount -a [-d] [-f] [-r] [-n] [-v] [-t typevfs] [-O opts]
       umount [-d] [-f] [-r] [-n] [-v] spécial | noeud...
```

Systèmes de fichiers Unix

Chaque partition sous un système Unix peut héberger une des structures suivantes :

- superbloc
- inode
- bloc de données
- blocs d'indirections

Superbloc

Le superbloc contient :

- la taille des blocs
- la taille du système de fichiers
- le nombre de montages effectués pour ce système de fichiers
- un pointeur vers la racine du système de fichiers
- les pointeurs vers la liste des inodes libres
- les pointeurs vers la liste des blocs de données libres

Le Superbloc est dupliqué tous les 8 ou 16Mo sur le système de fichiers.

Pour visualiser l'emplacement du Superbloc primaire et ses sauvegardes, utilisez la commande suivante :

```
opensuse:~ # dumpe2fs /dev/sda1 | grep -i superbloc
dumpe2fs 1.41.14 (22-Dec-2010)
  Primary superblock at 1, Group descriptors at 2-2
  Backup superblock at 8193, Group descriptors at 8194-8194
  Backup superblock at 24577, Group descriptors at 24578-24578
  Backup superblock at 40961, Group descriptors at 40962-40962
  Backup superblock at 57345, Group descriptors at 57346-57346
  Backup superblock at 73729, Group descriptors at 73730-73730
```

Pour réparer un système de fichiers en restaurant un Superbloc, utilisez la commande suivante :

```
# e2fsck -f -b 8193 /dev/sda1 [Enter]
```

Inodes

Chaque fichier est représenté par un **inode**. L'inode contient :

- le type de fichier, soit **-**, **d**, **l**, **b**, **c**, **p**, **s**
- les droits d'accès, par exemple **rwx rw- r–**
- le nombre de liens physiques soit le nombre de noms
- l'UID du créateur ou l'UID affecté par la commande **chown** s'il y a eu une modification
- le GID du processus créateur ou le GID affecté par la commande **chgrp**
- la taille du fichier en octets
- la date de dernière modification de l'inode, soit le **ctime**
- la date de dernière modification du fichier, soit le **mtime**
- la date du dernier accès, soit le **atime**
- les adresses qui pointent vers les blocs de données du fichier

Graphiquement, on peut schématiser cette organisation de la façon suivante :

Pour mieux comprendre, tapez la commande suivante :

```
# ls -ld /dev/console /dev/initctl /dev/loop0 /etc /etc/passwd [Entrée]
```

Vous obtiendrez un résultat similaire à la suivante :

```
opensuse:/ # ls -ld /dev/console /dev/initctl /dev/loop0 /etc /etc/passwd
crw-------   1 root root  5, 1 Oct 20 15:01 /dev/console
prw-------   1 root root     0 Oct 20 15:00 /dev/initctl
brw-rw----   1 root disk  7, 0 Oct 20 15:00 /dev/loop0
drwxr-xr-x 115 root root 12288 Oct 20 15:01 /etc
-rw-r--r--   1 root root  1372 Jul 28 11:04 /etc/passwd
```

Le premier caractère de chaque ligne peut être un des suivants :

- **-** - un fichier
- **d** - un répertoire
- **l** - un lien symbolique
- **b** - un périphérique du type bloc

- **c** - un périphérique du type caractère
- **p** - un tube nommé pour la communication entre processus
- **s** - un socket dans un contexte réseau

Pour visualiser le numéro d'inode, utilisez l'option **-i** :

```
opensuse:/ # ls -ldi /dev/console /dev/initctl /dev/loop0 /etc /etc/passwd
  869 crw-------   1 root root  5, 1 Oct 20 15:01 /dev/console
 2317 prw-------   1 root root     0 Oct 20 15:00 /dev/initctl
 2334 brw-rw----   1 root disk  7, 0 Oct 20 15:00 /dev/loop0
24001 drwxr-xr-x 115 root root 12288 Oct 20 15:01 /etc
28932 -rw-r--r--   1 root root  1372 Jul 28 11:04 /etc/passwd
```

Blocs de données

Les données sont stockées dans des blocs de données. Dans le cas d'un répertoire, le bloc de données contient une table qui référence les inodes et les noms des fichiers dans le répertoire. Cette table s'appelle une **table catalogue**.

Le nom d'un fichier n'est pas stocké dans l'inode mais dans une **table catalogue**. Cette particularité nous permet de donner deux noms différents au même fichier. Pour ajouter un nouveau nom à un fichier, il convient de créer un **lien physique**.

Liens Physiques

Un lien physique se crée en utilisant la commande suivante :

- ln nom_du_fichier nom_supplémentaire

Pour illustrer ce point, tapez la ligne de commande suivante :

```
# cd /tmp; mkdir inode; cd inode; touch fichier1; ls -ali [Entrée]
```

Vous obtiendrez un résultat similaire à celui-ci :

```
opensuse:/ # cd /tmp; mkdir inode; cd inode; touch fichier1; ls -ali
total 8
105204 drwxr-xr-x  2 root root 4096 Oct 20 15:47 .
104001 drwxrwxrwt 37 root root 4096 Oct 20 15:47 ..
106849 -rw-r--r--  1 root root    0 Oct 20 15:47 fichier1
```

Notez bien le numéro de l'inode du fichier **fichier1**. Notez aussi que le numéro dans le troisième champs de la ligne de fichier1 a la valeur **1** :

106849 -rw-r–r– 1 root root 0 Oct 20 15:47 fichier1

Créez maintenant un lien physique :

```
# ln fichier1 fichier2 [Entrée]
```

Visualisez le résultat :

```
opensuse:/tmp/inode # ln fichier1 fichier2
opensuse:/tmp/inode # ls -ali
total 8
105204 drwxr-xr-x  2 root root 4096 Oct 20 15:47 .
104001 drwxrwxrwt 37 root root 4096 Oct 20 15:47 ..
106849 -rw-r--r--  2 root root    0 Oct 20 15:47 fichier1
106849 -rw-r--r--  2 root root    0 Oct 20 15:47 fichier2
```

Notez les deux lignes suivantes :

106849 -rw-r–r– 2 root root 0 Oct 20 15:47 fichier1
106849 -rw-r–r– 2 root root 0 Oct 20 15:47 fichier2

Les deux fichiers, fichier1 et fichier2, sont référencés par le même inode. Le nombre de liens est donc augmenté de 1 (le numéro dans le troisième champs).

<note important> Un lien physique ne peut être crée que dans le cas où les deux fichiers se trouvent dans le même filesystem et que le fichier source existe. </note>

Liens Symboliques

Un lien symbolique est un **raccourci** vers un autre fichier ou répertoire. Un lien symbolique se crée en utilisant la commande suivante :

- ln -s nom_du_fichier nom_raccourci

Pour illustrer ce point, tapez la ligne de commande suivante :

```
# ln -s fichier1 fichier3 [Entrée]
```

Vous obtiendrez un résultat similaire à celui-ci :

```
opensuse:/tmp/inode # ln -s fichier1 fichier3
opensuse:/tmp/inode # ls -ali
total 8
105204 drwxr-xr-x  2 root root 4096 Oct 20 15:48 .
104001 drwxrwxrwt 37 root root 4096 Oct 20 15:47 ..
106849 -rw-r--r--  2 root root    0 Oct 20 15:47 fichier1
106849 -rw-r--r--  2 root root    0 Oct 20 15:47 fichier2
106847 lrwxrwxrwx  1 root root    8 Oct 20 15:48 fichier3 -> fichier1
```

Notez que le lien symbolique est référencé par un autre inode. Le lien symbolique pointe vers le fichier1.

<note important> Un lien symbolique peut être crée même dans le cas où les deux fichiers se trouvent dans deux filesystems différents et même dans le cas où le fichier source n'existe pas. </note>

~~DISCUSSION:off~~

Donner votre Avis

{(rater>id=opensuse_11_l101|name=cette page|type=rate|trace=user|tracedetails=1)}