**453---SYSTEME D'EXPLOITATION - e-learning - Séquence 22**

**Gestion des disques**

**Légende :**

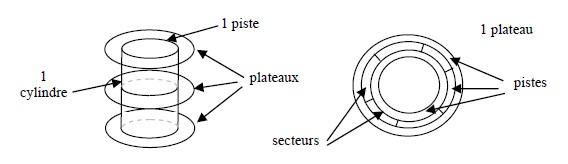
Explications sur fond blanc

Exercice à accomplir sur fond rouge clair

Une ou plusieurs solutions possibles pour l’exécution de commande sur fond vert clair

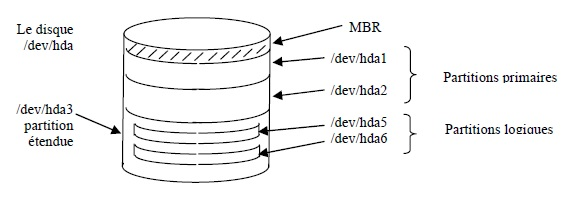
Lignes de configuration sur fond jaune clair

Un disque dur est constitué physiquement de n plateaux circulaires tournant autour du même axe de rotation. Les informations sont enregistrées sur les deux faces de chacun de ces n plateaux, le long de « pistes » concentriques (de quelques microns de largeur). Il y a donc 2n têtes de lecture/écriture. Sur un même disque, la capacité de chaque piste est constante.



Chaque piste est divisée en « secteurs ». Du point de vue physique, dans un disque dur, on peut distinguer comme unités d’allocation : le « Cylindre » (2n pistes superposées), la « Piste » (ensemble des secteurs sur une face à une distance radiale déterminée), le « Bloc » ou « Cluster » (un ou plusieurs secteurs consécutifs sur une piste) et le « Secteur » (nombre d’octets fixé au formatage).

Du point de vue logique, un disque contient des partitions. Jusqu’à 4 partitions « primaires » ou « principales ». Mais, afin de pouvoir outrepasser la limite des quatre partitions principales, l’une d’elles peut être érigée en partition « étendue », pouvant contenir plusieurs partitions « logiques ». En outre, un disque contient le secteur de démarrage (cylindre 0, tête 0 et secteur 1), appelé Master Boot Record ou MBR (contenant le chargeur qui permet de booter un système), ainsi que la table des partitions, et des informations d’exploitation.



Une partition peut abriter par exemple un système de fichiers, ou un espace de swap, ou un espace disque dédié à une application. Dans un système Linux, les nominations de disques, de partitions et de fichiers spéciaux se réfèrent aux types de carte contrôleur d’interface :

* IDE ou EIDE ou ATA ou PATA : « hd » – disque n°1 : « a », disque n°2 : « b », etc.
* SCSI ou SATA ou USB : « sd » – disque n°1 : « a », disque n°2 : « b », etc.

Ainsi :

hda1 Disque EIDE/ATA n°1, partition 1 (primaire) /dev/hda1

hda2 Disque EIDE/ATA n°1, partition 2 (primaire) /dev/hda2

hdb1 Disque EIDE/ATA n°2, partition 1 (primaire) /dev/hdb1

Si l’ordinateur est déjà pourvu de deux disques durs EIDE/ATA, alors :

hdc Lecteur de CD-ROM EIDE/ATA /dev/hdc

sda1 Disque SCSI/SATA/USB n°1, partition 1 (primaire) /dev/sda1

sda5 Disque SCSI/SATA/USB n°1, partition 5 (logique) /dev/sda5

sdb1 Disque SCSI/SATA/USB n°2, partition 1 (primaire) /dev/sdb1

Système Linux : Nomination des lecteurs de tapes et fichiers spéciaux :

st0 Premier lecteur /dev/st0

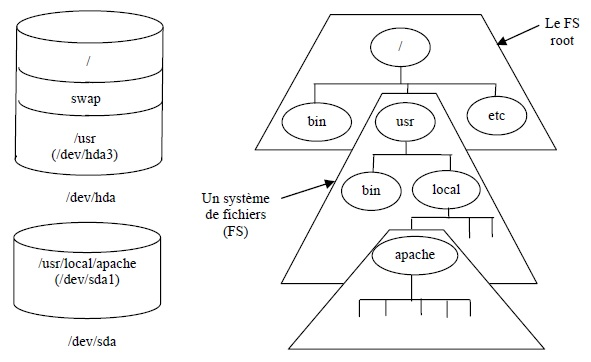
st1 Deuxième lecteur /dev/st1

Système Linux : Nomination des lecteurs de disquettes et fichiers spéciaux :

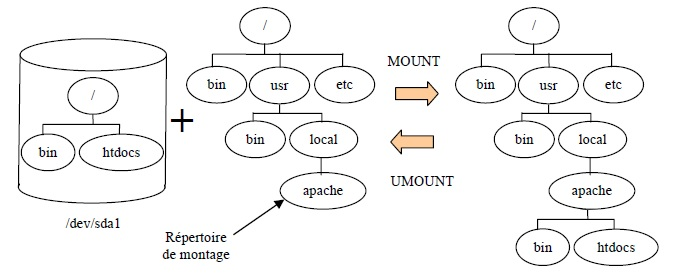
fd0 Premier lecteur /dev/fd0

fd1 Deuxième lecteur /dev/fd1

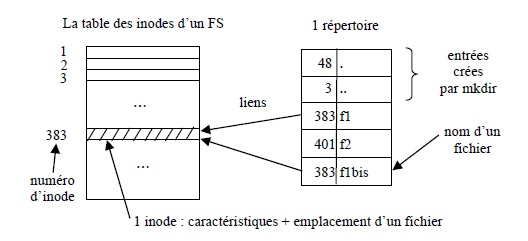
Dans un système Linux, tout est logiquement inséré dans une seule arborescence globale, la hiérarchie des fichiers (commençant à la racine notée « / »), comprenant éventuellement plusieurs systèmes de fichiers (FS).



Un système de fichiers (FS = File System) peut résider dans une partition sur un disque ou sur un autre support externe, ou dans la mémoire centrale, ou même dans un fichier ordinaire. Pour qu’un système de fichiers soit accessible, il faut qu’il soit activé, c’est-à-dire « monté » dans la hiérarchie des fichiers. Le montage de ce FS implique qu’à la racine de ce FS soit associé un répertoire « de montage ». Le « démontage » du FS signifie que cette association soit rompue. Le FS contenant la racine (root), lui, est monté automatiquement par le noyau de Linux au cours du démarrage du système. Les autres FS peuvent être montés automatiquement, ou à monter manuellement, selon la configuration. Les systèmes de fichiers des systèmes Windows peuvent être montés aussi, dans la hiérarchie des fichiers de Linux.



Outre le secteur de démarrage (boot bloc), nécessaire pour initialiser le système de fichiers contenant la racine (root), l’exploitation de tout système de fichiers UNIX, et donc Linux, nécessite un « super-bloc », contenant toutes les informations générales que le système d’exploitation doit utiliser : nom, type, taille, table de gestion des blocs libres, table de gestion des inodes libres, etc. Ensuite, la table des « inodes », contenant un « nœud d’index » par fichier ou répertoire, c’est-à-dire l’enregistrement de toutes les informations permettant de l’exploiter (ses caractéristiques et son emplacement), le inode étant identifié par un numéro. Un FS est limité aussi bien en blocs qu’en inodes. Le nombre maximum d’inodes détermine donc le nombre maximum de fichiers (au sens large) du FS. Enfin, les répertoires, qui se résument à des tables de correspondance entre nom de fichier et numéro d’inode, dans le cas d’un FS UNIX ou Linux. Plusieurs types de FS Linux peuvent être créés et exploités, tels que ext2 (l’ancien standard non journalisé), ext3 et ext4 (journalisés), jfs, xfs, etc.



**Quelques commandes :**

SFDISK(8)

NOM

sfdisk - Gestionnaire de tables de partitions pour Linux

SYNOPSIS

sfdisk [options] périphérique

sfdisk -s [partition]

DESCRIPTION

sfdisk a quatre (principales) utilisations : donner la taille d'une partition, afficher

les partitions d'un périphérique, vérifier les partitions d'un périphérique et

– très dangereux – repartitionner un périphérique.

FDISK(8)

NOM

fdisk - Manipuler la table de partitions d'un disque

SYNOPSIS

fdisk [-uc] [-b taille\_du\_secteur] [-C cyls] [-H têtes] [-S sects] périphérique

fdisk -l [-u] [périphérique ...]

fdisk -s partition ...

fdisk -v

fdisk -h

DESCRIPTION

La première manière de lancer fdisk se fait à l'aide d'un programme piloté par un

menu qui permet la création et la manipulation des tables de partitions, y compris

les tables de partitions de type DOS, et les étiquettes de disques BSD ou SUN.

DF(1)

NOM

df - Indiquer l'espace occupé par les systèmes de fichiers

SYNOPSIS

df [OPTION]... [FICHIER]...

DESCRIPTION

Le programme df indique l'espace disque utilisé et disponible sur le système de

fichiers contenant chaque fichier donné en paramètre. Sans paramètre, l'espace

disponible sur tous les systèmes de fichiers montés sera affiché.

Les valeurs sont indiquées en unités de 1 K, à moins que la variable

d'environnement POSIXLY\_CORRECT ne soit positionnée, auquel cas la taille

des blocs sera de 512 octets. Si un fichier indiqué en paramètre est un périphérique

disque contenant un système de fichiers monté, df affiche l'espace disponible sur

ce système de fichiers plutôt que sur celui contenant le nœud du périphérique

(qui est toujours le système de fichiers racine).

MOUNT(8)

NOM

mount - Monter un système de fichiers

SYNOPSIS

mount [-lhV]

mount -a [-fFnrsvw] [-t vfstype] [-O liste\_options]

mount [-fnrsvw] [-o option[,option] ...] périphérique|rép

mount [-fnrsvw] [-t vfstype] [-o options] périphérique rép

DESCRIPTION

Tous les fichiers accessibles par un système UNIX sont insérés dans une

grande arborescence, la hiérarchie des fichiers, commençant à la racine /.

Ces fichiers peuvent résider sur différents périphériques. La commande mount

permet d'attacher un système de fichiers trouvé sur un périphérique quelconque

à la grande arborescence du système. À l'inverse, umount(8) le détachera à nouveau.

La forme standard de la commande mount est :

mount -t vfstype périphérique rép

Cela indique au noyau d'attacher le système de fichiers se trouvant sur le

périphérique mentionné (dont le type est vfstype) dans le répertoire rép indiqué.

Le contenu, le propriétaire et le mode d'accès précédents de rép deviennent invisibles

et, tant que le nouveau système de fichiers reste monté, le chemin rép représente

la racine du système de fichiers du périphérique.

UMOUNT(8)

NOM

umount - Démonter des systèmes de fichiers

SYNOPSIS

umount [-hV]

umount -a [-dflnrv] [-t vfstype] [-O options]

umount [-dflnrv] {rép|périphérique} ...

DESCRIPTION

La commande umount détache les systèmes de fichiers mentionnés de la hiérarchie

des fichiers. Un système de fichiers est indiqué en donnant le nom du répertoire

dans lequel il a été monté. Fournir le périphérique spécial sur lequel réside le système

de fichiers peut également fonctionner, mais est obsolète, car cela échouera si

ce périphérique a été monté dans plus d'un répertoire. Un système de fichiers ne peut être

démonté quand il est « occupé », par exemple quand il y a des fichiers ouverts dessus.

LSOF(8)

NAME

lsof - list open files

SYNOPSIS

lsof [ -?abChKlnNOPRtUvVX ] [ -A A ] [ -c c ] [ +c c ] [ +|-d d ] [ +|-D D ] [ +|-e s ]

[ +|-f [cfgGn] ] [ -F [f] ] [ -g [s] ] [ -i [i] ] [ -k k ] [ +|-L [l] ] [ +|-m m ] [ +|-M ] [ -o [o] ]

[ -p s ] [ +|-r [t[m<fmt>]] ] [ -s [p:s] ] [ -S [t] ] [ -T [t] ] [ -u s ] [ +|-w ] [ -x [fl] ] [ -z [z] ]

[ -Z [Z] ] [ -- ] [names]

DESCRIPTION

lsof revision 4.86 lists on its standard output file information about files opened

by processes for the following UNIX dialects :

Apple Darwin 9 and Mac OS X 10.[567]

FreeBSD 4.9 and 6.4 for x86-based systems

FreeBSD 8.2, 9.0 and 10.0 for AMD64-based systems

Linux 2.1.72 and above for x86-based systems

Solaris 9, 10 and 11

KILL(1)

NOM

kill - Envoyer un signal à un processus

SYNOPSIS

kill [options] PID [...]

DESCRIPTION

Le signal par défaut de kill est TERM. Les options -l et -L affichent les

signaux disponibles. Les signaux particulièrement utiles sont HUP, INT, KILL,

STOP, CONT et 0. Les signaux peuvent être indiqués de 3 façons différentes :

-9 ou -SIGKILL ou -KILL. Une valeur de PID négative peut être utilisée pour

sélectionner des groupes de processus tout entiers ; regardez la colonne PGID de la

sortie de la commande ps. Un PID de valeur -1 est particulier : il indique tous

les processus sauf le processus kill lui-même et init.

EJECT(1)

NAME

eject - eject removable media

SYNOPSIS

eject -h

eject [-vnrsfmqp] [<name>]

eject [-vn] -d

eject [-vn] -a on|off|1|0 [<name>]

eject [-vn] -c slot [<name>]

eject [-vn] -i on|off|1|0 [<name>]

eject [-vn] -t [<name>]

eject [-vn] -T [<name>]

eject [-vn] -x <speed> [<name>]

eject [-vn] -X [<name>]

eject -V

DESCRIPTION

eject allows removable media (typically a CD-ROM, floppy disk, tape, or JAZ or

ZIP disk) to be ejected under software control. The command can also control

some multi-disc CD-ROM changers, the auto-eject feature supported by some

devices, and close the disc tray of some CD-ROM drives. The device corresponding

to <name> is ejected. The name can be a device file or mount point, either a full path

or with the leading "/dev", "/media" or "/mnt" omitted. If no name is specified, the default

name "cdrom" is used. If the device is currently mounted, it is unmounted before ejecting.

DD(1)

NOM

dd - Convertir et copier un fichier

SYNOPSIS

dd [OPÉRANDE]...

dd OPTION

DESCRIPTION

Copier un fichier en le convertissant et le formatant selon les opérandes.

bs=OCTETS

lire et écrire jusqu'à OCTETS octets à la fois

cbs=OCTETS

convertir OCTETS octets à la fois

conv=CONVS

convertir le fichier selon la liste de symboles séparés par des virgules

count=BLOCS

copier seulement BLOCS blocs d'entrée

ibs=OCTETS

lire jusqu'à OCTETS octets à la fois (par défaut : 512)

if=FICHIER

lire le FICHIER au lieu de l'entrée standard

iflag=FANIONS

lire selon la liste de symboles séparés par des virgules

obs=OCTETS

écrire OCTETS octets à la fois (par défaut : 512)

of=FICHIER

écrire dans le FICHIER plutôt que sur la sortie standard

oflag=FANIONS

écrire selon la liste de symboles séparés par des virgules

seek=BLOCS

sauter BLOCS blocs de taille « obs » en début de sortie

skip=BLOCS

sauter BLOCS blocs de taille « ibs » en début d'entrée

status=noxfer

supprimer les statistiques de transfert

MKFS(8)

NOM

mkfs - Créer un système de fichiers Linux

SYNOPSIS

mkfs [options] [-t type options-du-système-de-fichiers ] périphérique [ taille ]

DESCRIPTION

mkfs est utilisé pour créer un système de fichiers Linux sur un périphérique,

généralement une partition d'un disque dur. Le paramètre périphérique est soit le nom du

périphérique (par exemple : /dev/hda1, /dev/sdb2), soit un fichier normal qui peut

contenir le système de fichiers. Le paramètre taille est le nombre de blocs à utiliser

pour le système de fichiers. Le code de retour renvoyé par mkfs est 0 en cas de succès

et 1 en cas d'échec. En fait, mkfs est une simple interface pour les divers constructeurs

de systèmes de fichiers disponibles sous Linux.

OPTIONS

-t, --type type

Indiquer le type de système de fichiers à créer. S'il n'est pas indiqué, le système de

fichiers (ext2) est utilisé par défaut.

FSCK(8)

NOM

fsck - Vérifier et réparer un système de fichiers Linux

SYNOPSIS

fsck [-lsAVRTMNP] [-C [descripteur]] [-t type\_sf] [système\_de\_fichiers ...]

[--] [options\_spécifiques\_sf]

DESCRIPTION

fsck est utilisé pour vérifier et éventuellement réparer un ou plusieurs systèmes

de fichiers Linux. Si aucun système de fichiers n'est précisé sur la ligne de commande

et que l'option -A n'est pas indiquée, fsck vérifiera les systèmes de fichiers présents

dans /etc/fstab.

**Un fichier :**

FSTAB(5)

NOM

fstab - Informations statiques sur les systèmes de fichiers

SYNOPSIS

/etc/fstab

DESCRIPTION

Le fichier fstab contient des informations décrivant les différents systèmes de fichiers.

fstab est uniquement lu par les programmes, jamais écrit. Cette responsabilité est laissée

à l'administrateur, qui doit créer et maintenir lui-même ce fichier. Chaque système

de fichiers est décrit sur une ligne contenant des champs.

Le premier champ (fs\_spec)

Ce champ décrit le périphérique bloc ou le système de fichiers distant à monter.

Le deuxième champ (fs\_file)

Ce champ indique le point de montage du système de fichier.

Le troisième champ (fs\_vfstype)

Ce champ décrit le type de système de fichiers.

Le quatrième champ (fs\_mntops)

Ce champ définit les options de montage associées au système de fichiers.

Le cinquième champ (fs\_freq)

Ce champ est utilisé par la commande dump(8) pour déterminer quels sont

les systèmes de fichiers à décharger.

Le sixième champ (fs\_passno)

Ce champ est utilisé par le programme fsck(8) pour déterminer l'ordre

de vérification.

**LABORATOIRE – Gestion des disques**

**Dans la distribution Debian GNU/Linux :**

« ~/.profile » (ou « ~/.bash\_profile », selon les versions) est un fichier de script qui est exécuté automatiquement en début de session par le shell qui la contrôle.

Connectez-vous dans un terminal sous le compte « root », c’est-à-dire le superutilisateur (Password du root)

Affichez le manuel de la commande sfdisk

man sfdisk

Affichez la liste des différents disques avec leurs tailles respectives

sfdisk -s

Affichez le manuel de la commande fdisk

man fdisk

Affichez l’ensemble des différentes partitions en affichage long

fdisk -l

Listez les partitions d’un disque particulier en affichage long

fdisk -l /dev/sda

Affichez le manuel de la commande df

man df

Listez les FS (File Systems) montés et, pour chacun d’eux, le type, les tailles étant dans un format lisible par un humain

df -Th

Affichez à quel FS appartient le fichier : « /home/user/.profile »

df /home/user/.profile

Remarque : « /home/user/.profile » ou « /home/user/.bash\_profile », selon les versions !

Mettez un cdrom dans le lecteur, puis affichez où il a été monté automatiquement

df

Affichez le manuel des commandes mount et umount

man mount

man umount

Démontez le cdrom puis remontez-le dans : « /mnt/cdrom »

umount /dev/cdrom

mkdir -p /mnt/cdrom

mount -t iso9660 -o ro /dev/cdrom /mnt/cdrom

« | » symbolise un « pipe », c’est-à-dire un « tube de communication » entre deux processus.

Visualisez

df | grep /mnt

Activez un nouveau shell

bash

Chargez le cdrom et listez son contenu

cd /mnt/cdrom

ls

Essayez de démonter le cdrom

umount /mnt/cdrom

La tentative échoue car le shell a ouvert un fichier (en l’occurrence un répertoire) sur le cdrom.

Affichez le manuel des commandes lsof, kill et eject

man lsof

man kill

man eject

Listez les processus qui accèdent au cdrom, avec leur pid

lsof | grep /mnt/cdrom

Tuez le processus « bash » qui y accède

kill -9 <pid du bash>

Ejectez le cdrom

eject /mnt/cdrom

Affichez le manuel des commandes dd et mkfs

man dd

man mkfs

Créez un fichier de 128 Méga (dans lequel vous créerez ensuite un FS ext3)

dd if=/dev/zero of=/root/fs\_ext3.img bs=1M count=128

La commande dd copie un fichier bloc par bloc.

A « if » est affecté le fichier source, par défaut l’entrée standard (STDIN).

A « of » est affecté le fichier destination, par défaut la sortie standard (STDOUT).

A « bs » est affectée la taille des blocs à copier.

A « count » est affecté le nombre de blocs à copier.

Visualisez les caractéristiques du fichier de 128 Méga que vous avez créé

ls -lh /root/fs\_ext3.img

Formatez le fichier que vous avez créé en ext3

mkfs -F -j /root/fs\_ext3.img

-F force le formatage du fichier (car la commande ne s’applique par défaut qu’à des partitions).

-j demande la journalisation (donc : un FS ext3, et non ext2).

Un exemple d’application de mkfs à une partition : mkfs -t ext3 /dev/hdb7

Montez le FS dans : « /mnt/disk »

mkdir /mnt/disk

mount -t ext3 -o loop /root/fs\_ext3.img /mnt/disk

Listez les FS (File Systems) montés et, pour chacun d’eux, le type, les tailles étant dans un format lisible par un humain

df -Th

En redirigeant cal et date, créez des fichiers dans le FS monté

cal > /mnt/disk/fichcal

date > /mnt/disk/fichdate

Créez un répertoire dans le FS monté et créez-y un fichier en redirigeant uname

mkdir /mnt/disk/rep

uname -a > /mnt/disk/rep/fichinfosys

Visualisez le nombre de blocs, puis d’inodes libres dans le FS

df

df -i

Affichez le manuel de la commande fsck

man fsck

Essayez de vérifier le FS

fsck /root/fs\_ext3.img

n

Un FS doit être démonté pour pouvoir être vérifié.

Démontez le FS, puis vérifiez-le

umount /mnt/disk

fsck /root/fs\_ext3.img

Vérifiez le FS en profondeur

fsck -f /root/fs\_ext3.img

-f force une vérification en profondeur.

Affichez la page de manuel concernant le fichier fstab

man 5 fstab

Sauvegardez le fichier fstab actuel, en le nommant : « fstab.old »

cp /etc/fstab /etc/fstab.old

Montez automatiquement, à chaque démarrage du système, le FS créé, en éditant fstab et en y ajoutant une ligne pour ce FS

gedit /etc/fstab

/root/fs\_ext3.img /mnt/disk ext3 defaults,loop 0 0

Redémarrez le système

reboot

Après le redémarrage, listez les FS (File Systems) montés et, pour chacun d’eux, le type, les tailles étant dans un format lisible par un humain

df -Th

Déconnectez-vous