

**ISSET de Sfax
Département Technologie de l'Informatique**

**Mastère Professionnel : DSIR :
Développement des Systèmes d'Information et des
Réseaux**

Administration Système Linux

Chapitre 1

Les systèmes de gestion des fichiers sous Linux

Par : Azer ZAIRI

Courriel : azer.zairi@gnet.tn

Introduction générale à Linux

- Linux a commencé en **1991** en tant qu'un petit projet de **Linus Torvalds**.
- Comme le système UNIX, Linux était aussi **écrit en langage C**.
- Il a rendu la source disponible gratuitement et d'autres se sont joints à lui pour façonner ce nouveau système d'exploitation.
- Avant ça, il y avait un autre projet opensource était lancé : le **projet GNU** (GNU, pas UNIX).
 - C'est un système d'exploitation libre créé en **1983** par **Richard Stallman**,
- En **1985** Richard Stallman a fondé la **FSF** (Free Software Foundation) dans le but de protéger légalement le projet GNU et de promouvoir les logiciels libres.
- GNU construisait son propre système d'exploitation avec les outils, tels que les compilateurs et les interfaces graphiques.
- La source étant disponible gratuitement, Linux a pu cibler ses outils et fournir un système complet.
- Autrement, la plupart des outils qui font partie du système Linux proviennent de ces outils GNU.

*Remarque : En **janvier 2004**, l'**UNESCO** élève le logiciel libre au rang de **patrimoine mondial de l'humanité** et confère à GNU la valeur symbolique de « Trésor du monde »*

Les distributions Linux (1)

- Puisque Linux est un logiciel open source **plusieurs communautés ont développé des distributions différentes.**
- Pratiquement, pour avoir un nouveau système Linux :
 - il suffit de prendre le Kernel de Linux
 - et quelques outils GNU,
 - et y ajouter des applications plus orientées utilisateur,
- Comme UNIX, il existe de nombreuses versions différentes des distributions.
- De nos jours, il existe des distributions qui se concentrent sur l'exécution des fonctions :
 - de serveurs,
 - d'ordinateurs de bureau (DeskTop)
 - Des appareils mobiles (Android,...)
 - ou même d'outils spécifiques à l'industrie comme les objets connectés, la conception électronique (Raspberry,...) ou le calcul statistique.

Les distributions Linux (2)

- Les **deux principaux** acteurs du marché sont **Red Hat** et **Debian**.
- La **différence** la plus visible est le **gestionnaire de paquets**, bien qu'il existe d'**autres différences** telles que l'**emplacement des fichiers et les scripts de configurations** des services systèmes.
- Les distribution Redhat respectent la norme **FHS** (Filesystem Hierarchy Standard) beaucoup plus que celles de Debian
- **Il existe des centaines de distributions Linux.**
- Principales distributions compatibles **RedHat** :
 - **RHEL** : Redhat Entreprise Linux : Version payante
 - **Fedora** : Distribué par Redhat
 - **CentOS** : C'est une recompilation de tous les paquetages RHEL et distribués gratuitement
 - **Scientific Linux**
 - **SUSE**
- Principales distributions compatibles Debian :
 - **Ubuntu** : développé par Canonical
 - **Kali Linux**
 - **Parrot Security**

Les versions serveurs et les versions bureaux

- Linux peut être distribué en version “**Server**” ou “**DeskTop**”
- La **version Serveur** :
 - **Est plus adaptée pour les solutions professionnelles des serveur**
 - Elle possède **seulement un environnement texte**, sans interface graphique
 - Mondialement, Linux est très performant dans l'exécution d'applications serveur
 - en raison de l'optimisation des ressources matérielles,
 - ce qui augmente sa fiabilité et son efficacité.
 - Sans oublier la sécurité qui est un facteur clé dans le choix d'une solution
- **La version bureau (Desktop) sont plus adaptée pour l'utilisation personnelle**, car elle intègre beaucoup d'applications de bureau : des jeux, des applications de bureautique et des outils créatifs.
- Un bureau Linux exécute un système appelé X Window, également connu sous le nom de X11.
- Le serveur Linux X11 est X.org, qui permet aux logiciels de fonctionner en mode graphique.
- L'interface graphique et les icônes sont gérées par un autre logiciel appelé gestionnaire de fenêtres ou environnement de bureau.
- Autrement, un poste de travail Linux en texte seul, devient un bureau graphique avec l'ajout de X-Windows et d'un environnement de bureau ou d'un gestionnaire de fenêtres.

Remarque : Pour des raisons de sécurité le superutilisateur root ne peut pas se connecter directement à l'environnement graphique. Car les processus en tâche de fond de X11 peuvent provoquer des failles, si elles sont exécutées en tant que root.

Les systèmes de fichiers sous Linux

Introduction

- Un **système de fichiers** (*file system* ou *filesystem*) ou **système de gestion de fichiers** (SGF) est une structure de données permettant de stocker les informations et de les organiser dans des fichiers sur des mémoires secondaires (disque dur, disquette, CD-ROM, clé USB, disques SSD, etc.).
- Une telle gestion des fichiers permet :
 - **de traiter,**
 - **de conserver des quantités importantes de données**
 - **ainsi que de les partager entre plusieurs programmes informatiques.**
- Un SGF offre à l'utilisateur une vue abstraite sur ses données et permet de les localiser à partir d'un chemin d'accès.

Fonctions d'un système de fichiers

- Le système de gestion des fichiers assure plusieurs fonctions :
 - **Manipulation des fichiers** : des opérations sont définies pour permettre la manipulation des fichiers par les programmes d'application, à savoir : créer/détruire des fichiers, insérer, supprimer et modifier un article dans un fichier.
 - **Allocation de la place sur mémoires secondaires** : les fichiers étant de taille différente et cette taille pouvant être dynamique, le SGF alloue à chaque fichier un nombre variable de blocs dans la mémoire secondaire.
 - **Localisation des fichiers** : il est nécessaire de pouvoir identifier et retrouver les données ; pour cela, chaque fichier possède un ensemble d'informations descriptives (nom, adresse...) regroupées dans un inode.
 - **Sécurité et contrôle des fichiers** : le SGF permet le partage des fichiers par différents programmes d'applications tout en assurant la sécurité et la confidentialité des données.

Restriction des noms

- Différentes méthodes permettent d'associer un nom de fichier à son contenu.
- Dans le cas des systèmes de fichier d'Unix (ou de Linux) ;
 - Les fichiers et les répertoires sont identifiés par un numéro unique le numéro **d'inode**.
 - Ce numéro permet d'accéder à une structure de données (inode) regroupant toutes les informations sur un fichier à l'exception du nom,
 - Notamment la protection d'accès en lecture, en écriture ou des listes de dates, ainsi que le moyen d'en retrouver le contenu.
 - Le nom est stocké dans le répertoire associé à un numéro d'inode.
 - Cette organisation présente l'avantage qu'un fichier unique sur disque peut être connu du système sous plusieurs noms.

Métadonnées

- Chaque fichier est décrit par des métadonnées (conservées dans l'inode sous Linux/Unix), alors que le contenu du fichier est écrit dans un ou plusieurs blocs du support de stockage, selon la taille du fichier.
- Les métadonnées les plus courantes sous UNIX sont :
 - droits d'accès en lecture, écriture et exécution selon l'utilisateur, le groupe, ou les autres ;
 - dates de dernier accès, de modification des métadonnées (inode), de modification des données (*block*) ;
 - propriétaire et groupe propriétaire du fichier ;
 - taille du fichier ;
 - nombre d'autres inodes (liens) pointant vers le fichier ;
 - nombre de blocs utilisés par le fichier ;
 - type de fichier : fichier simple, lien symbolique, répertoire, périphérique, etc.
- ***Remarque : Sur la plupart des systèmes Unix, la commande stat permet d'afficher l'intégralité du contenu de l'inode.***

Systèmes de fichiers non journalisés

- **ext et ext2** : *Extented FS version 2* (Linux, BSD)
- **exFAT** : *Extended File Allocation Table* (nouveau système de fichiers proposé par Microsoft pour remplacer la FAT sur les supports amovibles)
- **FAT** : *File Allocation Table* (DOS/Windows, Linux, BSD, OS/2, Mac OS X). Se décompose en plusieurs catégories :
 - FAT12 ; FAT16 ; FAT32 ; VFAT ; FATX : système de fichiers pour Xbox.
- **FFS** : *Fast File System* (BSD, Linux expérimental)
- **HFS** : *Hierarchical File System* (Mac OS, Mac OS X, Linux)
- **HPFS** : *High Performance FileSystem* (OS/2, Linux)
- **minix fs** (minix, Linux)
- **S5** (UNIX System V, Linux)
- **Unix File System** : (BSD, Linux en lecture seule)

Systèmes de fichiers journalisés

- **Ext3** : (*Extented FS*) - notamment pour l'ajout de la journalisation (Linux, BSD)
- **Ext4** : *Extented FS version 4* - notamment pour une capacité de 1 exa-octet et les *Extents* (Linux $\geq 2.6.28$)
- HFS+ (Mac OS X, Linux)
- JFS (AIX, OS/2, Linux)
- JFS2 AIX5
- LFS : (Linux)
- NSS : *Novell Storage Services* (Netware et Suse Linux)
- **NTFS** : *New Technology FileSystem* (Windows NT/2000/XP/Vista, Linux (écriture disponible grâce au pilote NTFS-3G), Mac OS X (écriture disponible grâce au pilote NTFS-3G))
- **ReiserFS** (Linux, BSD en lecture seule)
- **Reiser4** (Linux expérimental)
- Spufs : *Synergistic processing unit filesystem*
- UFS+ : *Unix FS + journal* (BSD, Linux en lecture seule)
- XFS (Irix, Linux, BSD en lecture seule)

Les inodes

- **inode (INdex nODE)** : *nœud d'index*
- **Les inodes sont des structures de données contenant des informations concernant les fichiers** stockés dans certains systèmes de fichiers (notamment de type Linux/Unix).
- **À chaque fichier correspond un numéro d'inode (*i-number*)** dans le système de fichiers dans lequel il réside, unique au périphérique sur lequel il est situé.
- Les **inodes contiennent notamment** les métadonnées des systèmes de fichiers, et en particulier celles concernant les **droits d'accès**.
- **Les inodes peuvent**, selon le système de fichiers, **contenir aussi des informations concernant le fichier**, tel que son créateur (ou **propriétaire**), son **type d'accès** (par exemple sous Unix : lecture, écriture et exécution), etc.
- **Les inodes sont créés lors de la création du système de fichiers.**
- La **quantité d'inodes** est généralement déterminée **lors du formatage** et dépendant de la taille de la partition.
- Elle indique **le nombre maximum de fichiers** que le système de fichiers peut contenir.

Les partitions

- Une partition est définie par :
 - son **type**
 - Son emplacement de **début**
 - Sa **taille** (ou son emplacement de fin)
- Une seule partition est activée au niveau du BIOS
- Cette activation indique où le BIOS doit aller chercher le noyau du système d'exploitation pour le démarrage
- Il existe 3 types de partitions :
 - **Les partitions principales** : Leur nombre est limité à quatre et elles supportent tous les types des systèmes des fichiers
 - **Les partitions étendues** : Elles ne peuvent contenir que des partitions logiques et ne peuvent pas avoir un système de fichiers. Elles sont incluses dans les partitions principales
 - **Les partitions logiques** : Elles sont contenues dans les partitions étendues. Elles ne sont pas limitées en nombre et acceptent tous les types des systèmes des fichiers.

Organisation des partitions sous Linux (1)

- Tous les périphériques ont des fichiers descripteurs sous le répertoire **/dev**.
- Les **disques** de type **IDE** ont des **descripteurs** commençant par **hd**
- Les **disques** de type **SATA** (Serial Advanced Technology Attachment) ont des descripteurs commençant par **sd**
- Une **lettre supplémentaire** est ajoutée au descripteur **pour désigner le périphérique**.
- Il y a généralement **deux contrôleurs IDE** en standard dans les cartes mères.
- **Chaque contrôleur supporte deux périphériques** : en total quatre périphériques de supports de stockage ou d'archivage (Disque dur, lecteur CD/DVD, lecteur ZIP,...)

	Primaire	Secondaire
Maître	a	c
Esclave	b	d

Désignation des périphériques IDE

Organisation des partitions sous Linux (2)

- Les périphériques de stockage utilisant les bus séries, tels que les disques SATA ou USB sont désignés en fonction de leur position dans la chaîne : sda, sdb, sdc, sdd,...
- De même, les périphériques SCSI (Small Computer System Interface) sont désignés en fonction de leur position dans la chaîne SCSI : sca, scb, scc, scd,...
- Chaque partition logique aura un numéro incrémental

	Primaire	Secondaire
ID0	a	c
ID1	b	d

Désignation des périphériques SATA et SCSI

- **Exemples**

- *Le périphérique maître sur le contrôleur IDE primaire est : **hda***
- *La troisième partition du périphérique esclave sur le contrôleur secondaire est : **hdd3***
- *La deuxième partition du deuxième disque SATA est : **sdb2***

Organisation des partitions sous Linux (3)

- Les mémoires de stockage électronique ont aussi des désignations spécifiques.
- Les disques **SSD : Solid-State Drive** : Sont des mémoires flash permettant le stockage de données.
- Les disques SSD peuvent être connectés aux bus SATA ou NVME
 - Pour ceux connectés au bus SATA, ils auront les mêmes désignations que ceux des disques SATA classiques : sda, sdb,...
 - Les disques NVME : Non-Volatile Memory Express :
 - C'est une spécification d'interface permettant à un ordinateur d'accéder à un périphérique de stockage permanent à travers un bus PCI Express.
 - Il s'agit de donner l'accès de façon performante à la mémoire flash sans passer par le protocole disque SCSI et donc avec un temps d'accès réduit et avec le minimum des ressources du processeur
 - Les disques NVMe ont les drivers de la forme: /dev/nvme0n1
 - Les partitions auront des numéros successifs de la forme pN, où N est le numéro de la partition.
- **Exemples**
 - *La première partition du premier disque SSD-NVME aura le driver : /dev/nvme0n1p1*

- Cette commande permet de créer, configurer ou afficher les informations sur les partitions

- **Exemple**

```
root@localhost# fdisk -l /dev/sda
```

```
Disk /dev/sda: 60.0 GB, 60060155904 bytes
```

```
255 heads, 63 sectors/track, 7301 cylinders
```

```
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Device Boot Start End Blocks Id System
```

```
/dev/sda1 * 1 4 32098+ 83 Linux
```

```
/dev/sda2 5 7299 58597087+ 5 Extended
```

```
/dev/sda5 5 1098 8787523+ 83 Linux
```

```
/dev/sda6 1221 3652 19535008+ 83 Linux
```

```
/dev/sda7 3653 7299 29294496 83 Linux
```

```
/dev/sda8 1099 1220 979933+ 82 Linux swap
```

```
Partition table entries are not in disk order
```

■ Les opérations principales de fdisk sont :

- p créer une partition
 1. partition primaire (1-4), étendue (1 max) ou logique (dans la partition étendue)
 2. numéro de partition (si primaire ou étendue)
 3. cylindre de début de la partition (RETURN pour valeur proposée)
 4. dimension de la partition (valeur absolue en cylindres ou tailles comme 16M, 5000M ou 5G, etc)
- d supprimer une partition (numéro)
- t changer le type de la partition (numéro, type ; l pour liste)
- w pour inscrire les changements
- q pour quitter

La commande fdisk

(3)

Exemple

```
[root@localhost]# fdisk /dev/sda
```

Command (m for help): p

Disk /dev/sda: 60.0 GB, 60060155904 bytes

255 heads, 63 sectors/track, 7301 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sda1	*	1	4	32098+	83	Linux
-----------	---	---	---	--------	----	-------

/dev/sda2		5	7299	58597087+	5	Extended
-----------	--	---	------	-----------	---	----------

/dev/sda5		5	1098	8787523+	83	Linux
-----------	--	---	------	----------	----	-------

/dev/sda6		1221	3652	19535008+	83	Linux
-----------	--	------	------	-----------	----	-------

/dev/sda7		3653	7299	29294496	83	Linux
-----------	--	------	------	----------	----	-------

/dev/sda8		1099	1220	979933+	82	Linux swap
-----------	--	------	------	---------	----	------------

- **Linux utilise deux types de partitions :**
 - Linux native : Elle contient l'arborescence du système de fichier et le noyau (Kernel) de Linux.
 - Exchange Linux (swap) : C'est la mémoire d'échange sous Linux.
 - **Remarque :**
 - *La mémoire vive et l'espace d'échange constituent ensemble la mémoire virtuelle du système.*
 - *Généralement la mémoire swap est le double de la mémoire vive*

Norme de la hiérarchie des systèmes de fichiers

- Parmi les normes prises en charge par la Foundation Linux on peut citer le **Filesystem Hierarchy Standard (FHS)**.
- Cette norme spécifie un ensemble de règles et de lignes directrices qu'il est recommandé de suivre, pour organiser les répertoires et leurs contenus.
- Cependant, **ces directives peuvent être contournées**, soit **par** des **distributions** entières, soit par des **administrateurs** sur des machines individuelles.
- La norme **FHS classe chaque répertoire système de deux manières** :
 - Un répertoire peut être catégorisé comme **partageable ou non**, ce qui signifie si le répertoire peut être partagé sur un réseau et utilisé par plusieurs machines.
 - Le répertoire est placé dans une **catégorie** :
 - de fichiers **statiques** (le contenu du fichier ne changera pas)
 - ou de fichiers **variables** (le contenu du fichier peut changer).

Remarque: Les distributions compatibles **Redhat respectent le standard FHS**, beaucoup plus que les distributions compatibles Debian

Arborescence des fichiers sous Linux (1)

- La racine du système des fichiers est : /
- Les principaux répertoires sont :
 - **/boot** : contient principalement le fichier binaire du noyau ainsi que les ressources nécessaires à son lancement au démarrage
 - **/dev** : contient les fichiers des périphériques de la machine ainsi que les fichiers spéciaux
 - **/etc** : répertoire très important où sont stockés tous les fichiers de configuration du système et des différents démons.
 - **/home** : répertoire où sont stockés par défaut les répertoires personnels des utilisateurs du système
 - **/proc** : contient les informations nécessaires au noyau. C'est une arborescence virtuelle généralement en lecture seul, sauf /proc/sys
 - **/tmp** : permet aux applications et aux utilisateurs d'avoir un espace d'échange où ils peuvent stocker leurs fichiers temporaires. Il est effacé à chaque redémarrage de la machine

Arborescence des fichiers sous Linux (2)

- **/usr** : contient les fichiers nécessaires aux applications, la documentation, les manuels, les fichiers sources ainsi que des librairies généralement statiques
- **/usr/local** : arborescence qui sert à installer les logiciels supplémentaires
- **/var** : contient les fichiers de log des différents démons, ainsi que les spools de mail, impression,...
- **/bin et /sbin** : contiennent l'ensemble des fichiers binaires indispensables au démarrage de la machine et les commandes essentielles d'administration
- **/lib et /usr/lib** : contiennent les librairies nécessaires aux commandes

Types des systèmes des fichiers

(1)

- Les principaux types des systèmes des fichiers supportés par Linux et leurs commandes de création sont résumés dans le tableau suivant

Système de fichiers	Commande de création
ext2	mke2fs mkfs.ext2 mkfs -t ext2
ext3	mke2fs -j mkfs.ext3 mkfs -t ext3
ext4	mkfs.ext4 mkfs -t ext4
reiserfs	mkreiserfs
xfs	mkfs.xfs mkfs -t xfs
vfat	mkfs.vfat mkfs -t vfat

- La syntaxe générale de création des systèmes de fichiers est :

mkfs -t <Type de fichier> <Partition>

- Il est aussi possible de transformer une partition ext2 en ext3 avec la commande tune2fs

tune2fs -j /dev/sda1

Exemple : Formatage de la partition sda1 avec création d'un système de fichiers de type ext3

- Les trois commandes suivantes sont équivalentes :
 - mkfs.ext3 /dev/sda1
 - mkfs -t ext3 /dev/sda1
 - mke2fs -j /dev/sda1 # Création de journal spécifique

Contrôle de l'intégrité du système de fichiers et réparation

- Si le système de fichier est endommagé ou corrompu, l'utilitaire **fsck** est utilisé pour **vérifier** et **corriger** le système

- La **syntaxe générale** de la commande est :

fsck -t <Type du système de fichiers> <Partition>

fsck.<Type du système de fichiers> <Partition>

- Il y a aussi des commandes équivalentes pour chaque type de système de fichiers (fsck.ext3,...)
- L'**option -i** permet à l'utilitaire fsck de corriger les problèmes trouvés
- Une **vérification** de toutes les partitions est réalisée **au démarrage du système** par le lancement **automatique** de la commande **fsck**
- La commande **e2fsck** est équivalente à **fsck -t ext2**
- En cas de **grand problème** il est utile d'utiliser l'utilitaire **debugfs**

Montage et démontage d'un système de fichiers (1)

- Pour pouvoir **utiliser un système de fichiers**, il doit **être monté** sur un **point de montage** de l'arborescence Linux
- Son contenu est alors accessible comme un simple répertoire
- Le système Linux réalise diverses tâches de vérification afin de s'assurer du fonctionnement correct.
- La **commande mount** permet de monter le système de fichiers
- Elle a **deux arguments** :
 - Le premier est le **fichier spécial** correspondant à la partition contenant le système de fichiers
 - Le second est le **répertoire de montage** (point de montage)
- Au cas où Linux ne parvient pas à le déterminer automatiquement, il peut être nécessaire de spécifier le type de fichier avec **l'option -t**
- La commande **umount** permet de démonter le système de fichiers
- **Exemple**
 - `mount /dev/sdb1 /mnt/flashdisque`
 - `umount /mnt/flashdisque` (ou `umount /dev/sdb1`)

Montage et démontage d'un système de fichiers (2)

- La commande **df** permet de connaître le taux d'utilisation de toutes les partitions montées du système.
- L'option **-h** (human) facilite la lecture en utilisant des unités de taille plus commodes (Mo, Go, To,...)
- La commande **du** (disk usage) permet de connaître l'espace occupé par une arborescence.
- L'option **-s** permet d'afficher le total des partitions
- L'option **-k** permet d'afficher le total en kilo-octets
- Exemple
 - `du -ks /usr/local`

Les droits sur les fichiers et les répertoires

- Linux permet de spécifier les droits dont disposent les utilisateurs sur un fichier ou un répertoire par la commande `chmod`
- On distingue trois catégories d'utilisateurs :
 - u : Le propriétaire (user)
 - g : Le groupe
 - o : Les autres (others)
- On distingue aussi trois types de droits :
 - r : lecture (read)
 - w : écriture (write)
 - x : exécution

Exemple

```
ls -l
-rw-rw-r-- 1 azer azer 98067 oct. 3 2022 310084408_2922980734675161_5021932786144836849_n.png
-rw-rw-r-- 1 azer azer 74722 déc. 23 2021 800px-Noyau_monolithique_modulaire.svg.png
-rwxrwxrwx 1 azer azer 61909 sept. 22 13:42 'Chapitre 1-SF.odp'
-rwxrwxrwx 1 azer azer 94933 sept. 29 19:44 'Chapitre 2 - Configuration réseau.odp'
-rwxrwxrwx 1 azer azer 121002 déc. 2 2022 'Chapitre 3 - Apache-NFS-Samba.odp'
-rwxrwxrwx 1 azer azer 183629 nov. 24 2022 'Chapitre 4 - DNS.odp'
```

- **Il existe trois droits spéciaux :**
 - **suid** : set user ID
 - **sgid** : set group ID
 - **sticky bit**
- **Ils peuvent être positionnés par la commande `chmod`**
- **SUID**
 - Lorsque le bit **suid** est positionné, une commande se lancera avec l'uid de son propriétaire, ce qui permet d'acquérir ses droits durant l'exécution de la commande
 - Ce bit est positionné par l'option **s** de la commande **chmod**, ou le premier bit (à gauche) de ce groupe de droit
 - **Exemple**
`chmod 4777 /bin/cat`
`chmod u+s /bin/cat`

■ SGID

- Il a le même principe pour les fichiers exécutables, mais il donne des droits pour le groupe
- `suid` peut aussi être attribués à un dossier : dans ce cas tout fichier créé dans un dossier portant le `sgid` aura comme groupe propriétaire le groupe du dossier
- Ce bit est positionné par l'option **s** de la commande **chmod**, ou par le deuxième bit de ce groupe de droits

■ Exemple

- `chmod 2755 /home/etudiant`
- `chmod g+s /home/etudiant`

■ sticky bit

- Si le **sticky bit** est placé sur un dossier, seul le possesseur d'un fichier pourra le renommer ou le supprimer, mais tous les utilisateurs pourront y avoir accès.
- C'est le cas des répertoires **/tmp** ou **/var/mail**
- Ce bit est positionné par l'option **t** de la commande **chmod**, **ou** le troisième bit de ce groupe de droits

■ Exemple

- `chmod 1666 /home/etudiant/partage`
- `chmod o+t /home/etudiant/partage`

Modifier le propriétaire et le groupe sur les fichiers et les répertoires

- **Linux permet de spécifier le propriétaire d'un fichier ou d'un répertoire par la commande `chown`**
- **Exemple**
 - `chown etudiant /home/linux/notes.txt`
- **Linux permet de spécifier le groupe d'un fichier ou d'un répertoire par la commande `chgrp`**
- **Exemple**
 - `chgrp /groupe_etudiants /home/linux/examen.txt`

Les quotas

- L'attribution des quotas dans un système de fichiers est un outil qui permet de maîtriser l'utilisation de l'espace disque par les utilisateurs
- Les quotas ne sont pas toujours possibles avec le système des fichiers reiserfs
- Pour activer les quotas il faut passer par plusieurs étapes
 - Éditer le fichier **/etc/fstab** et rajouter **usrquota** dans les options de montage
 - Remonter la partition sur laquelle on définit les quotas
 - Vérifier les quotas avec la commande **quotacheck**
 - Éditer les quotas pour chaque utilisateur avec **edquota**
 - **edquota -u <utilisateur>**
 - Activer la gestion des quotas par la commande **quotaon**
 - Afficher un résumé des informations sur les quotas définis par la commande **repquota**

Recherche des fichiers

- La recherche dans l'arborescence du système de fichier peut se faire grâce à des utilitaires tel que :
 - **find**, **locate**, **which**, **whereis**, **whatis** et **apropos**
 - **find** est la plus ancienne commande de recherche d'Unix
 - **locate** est identique à **find**, mais elle utilise une base de données
 - On peut mettre à jour la base des données par la commande **updatedb**
 - La commande **slocate** est la version sécurisée de **locate**, donc elle n'affiche pas les fichiers auxquels l'utilisateur n'aurait pas accès
 - **which** est utilisée pour trouver l'emplacement d'un programme
 - **whereis** est similaire à **which**, mais elle peut aussi chercher dans les pages de manuel et les codes sources
 - **whatis** cherche des commandes dans l'intégralité des systèmes de fichiers comme **which**, mais elle utilise une base de données qui contient une courte description ainsi que des mots clefs
 - La base des données est créée en utilisant la commande **makewhatis**
 - **apropos** utilise la même base de données que **whatis** et donne plus d'informations