

# Introduction à Linux

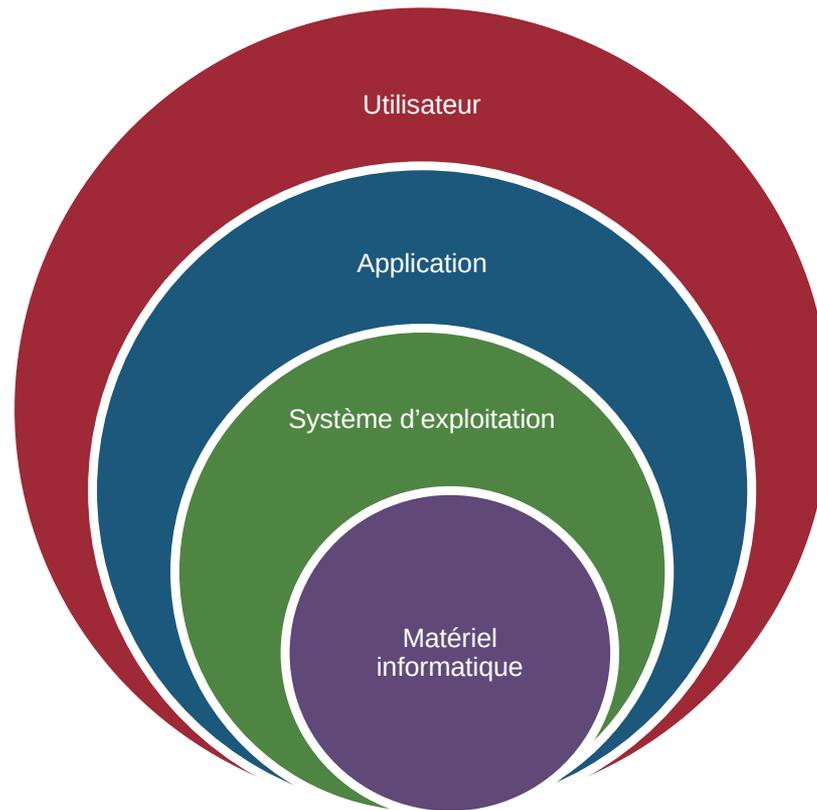
Cours n°1

Systemes d'exploitation - ESIEE Paris - 3R-IN3

Document original : Clément BOIN

Adaptation : Xavier HILAIRE

# Qu'est-ce qu'un système d'exploitation ?



# Définition

- Système d'exploitation (SE)
  - Operating System (OS), en anglais
- Le système d'exploitation est un intermédiaire (**interface**) entre les **applications** et le **matériel**.
- Chaque SE fonctionne avec **une gamme** particulière de machines.

# Typologie des systèmes d'exploitation (1/2)

- **multi-tâches** : permet l'exécution simultanée de plusieurs programmes.
  - Début dans les années 1960,
  - tous les systèmes d'exploitation contemporains le sont
- **multi-utilisateurs** : conçu pour être utilisé simultanément par plusieurs utilisateurs
  - Via un réseau le plus souvent
  - Utilisés pour des serveurs notamment
  - Sécurisation des connexions des utilisateurs

# Typologie des systèmes d'exploitation (2/2)

- **multi-processeurs** : conçu pour exploiter un ordinateur équipé de plusieurs processeurs.
  - plusieurs programmes sont exécutés simultanément par les différents processeurs.
- **temps réel** : garantit que les opérations seront effectuées en respectant des délais très stricts, et ce quelles que soient les conditions d'utilisation.
  - Utilisés notamment dans l'industrie, l'aéronautique

# Composition d'un SE (1/4)

- **Interface de programmation**

- API (Application Programming Interface)
- Mise à disposition de **fonctions** (dans des bibliothèques)
- Norme **POSIX** (famille UNIX) - IEEE 1003

- **Ordonnanceur**

- Contrôle de déroulement des programmes
- Exécution simultanée (**multi-tâche**)

- **Communication inter-processus**

- échanges d'informations entre les processus

# Composition d'un SE (2/4)

- **Gestion de la mémoire**

- **allocation** : réservation de la mémoire
- **protection** : la mémoire est utilisé uniquement par le programme qu'il l'a réservé
- **pagination** : décomposition en zones de taille fixe (pages)

- **Mémoire virtuelle (swap)**

- **Simuler la présence de mémoire centrale** en utilisant un autre type de mémoire (un disque dur par exemple)
- Exécuter plus de programmes que ce que la mémoire centrale peut contenir.

- **Pilotes**

- contiennent les instructions permettant d'utiliser certains **périphériques**

# Composition d'un SE (3/4)

- **Systeme de fichiers**

- file system en anglais
- **structure arborescente** dans laquelle sont stockés les données (**fichiers**)
- Exemples de dispositions : ext2fs, ext4fs (Linux), NTFS, FAT32 (Windows), HFS (Mac)

- **Réseau**

- niveaux 1 à 4 du modèle OSI pris en charge par le SE

- **Contrôle d'accès**

- identification des utilisateurs
- **droits d'accès** sur les fichiers

# Composition d'un SE (4/4)

- **Interface utilisateur (graphique ou texte)**
  - permet à l'utilisateur de dialoguer avec la machine via une image envoyée au matériel
  - Famille UNIX : X Window System
- **Logiciels utilitaires**
  - fournis avec un SE
  - applications
  - interpréteur de commande (scripting)
  - environnement de bureau
  - installation et mise à jour des logiciels
  - surveillance de l'utilisation
  - configuration, droits d'accès
  - ...

# Le noyau d'un OS

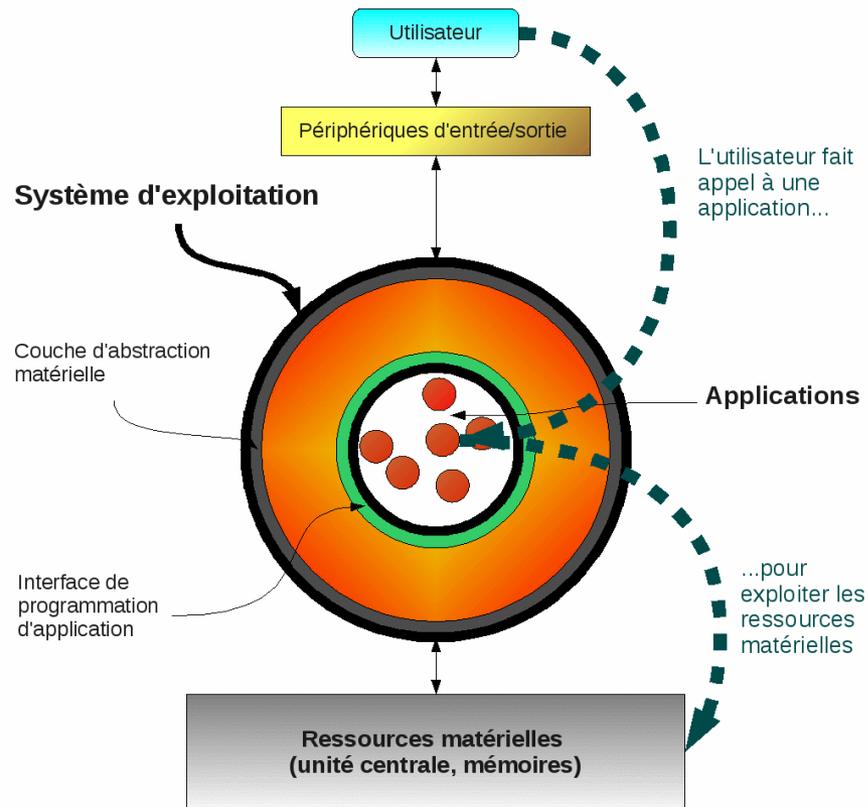
- Espace mémoire protégé  
+
- Ensemble de programmes qui forment la base minimale de l'OS
- Tout ce qui n'est pas un appel système fonctionnera dans l'espace utilisateur
- Des **choix de conception** sont fait :
  - **Exemple:** L'interface graphique
    - ▢ Sous Windows : intégré dans le système
    - ▢ Sous Linux : programme utilisateur (serveur X)

# Les types de noyau

- **Monolithique:**
  - tout est dans le noyau (système de fichiers, pilotes, etc)
  - *Exemples : Linux, FreeBSD*
- **Micro-noyau:**
  - seulement le strict minimum (ordonnanceur+mémoire virtuelle)
  - *Exemples : Minix, Mac OS X*
- **Hybride**
  - *Windows NT (noyau de Windows 10)*
- **Exo-noyau:** rien n'est protégé

# Les relations dans un SE moderne

(source: Wikipedia)

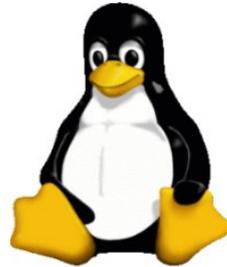


*Les relations entre utilisateur, applications, système d'exploitation et matériel*

# Les grandes familles de systèmes d'exploitations

## Famille UNIX

- Distributions GNU/Linux
  - **Debian**
  - Ubuntu Server
  - Red Hat Enterprise Linux
  - CentOS
  - Fedora
  - ...
- Android
- Mac OS X / iOS
- FreeBSD, NetBSD, OpenBSD
  - Unix "libre"



android



## Famille Windows

- Windows 10
- Windows Server
  - Version actuelle : 2022



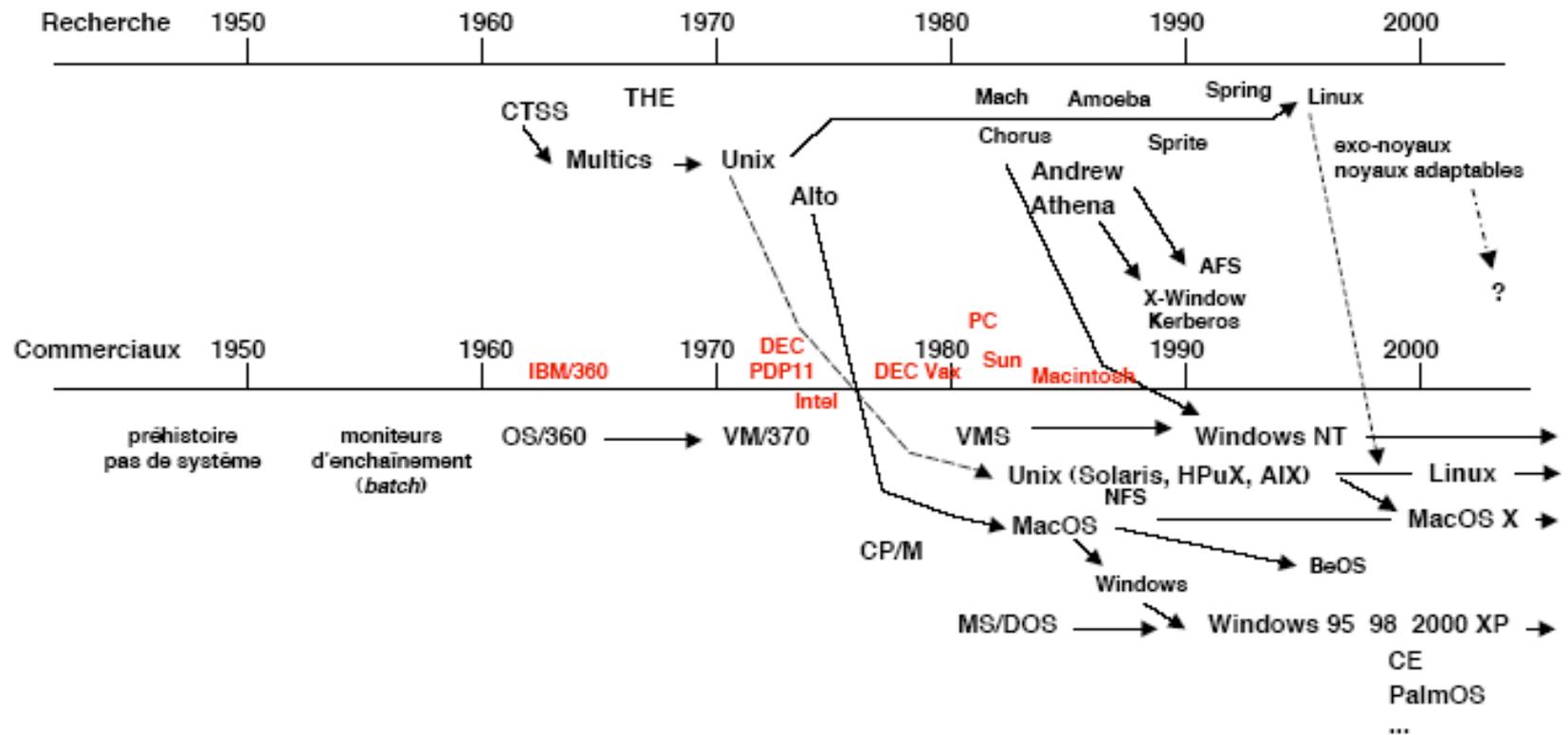
# Principaux systèmes d'exploitations

(source: Wikipedia)

Apple	Mac OS Classic	Système 5 · Système 6 · Système 7 · Mac OS 8 · Mac OS 9
	Dérivés de NeXTSTEP	NeXTSTEP · Rhapsody · Darwin · OS X · iOS
Dérivés de BeOS	BlueEyedOS · Haiku · ZETA	
DOS	DR-DOS · FreeDOS · MS-DOS · PC-DOS	
IBM	AIX · MVS · OS/2 · OS/360 · OS/390 · z/OS · OS/400	
Microsoft Windows	Basés sur MS-DOS	1.0 · 2.x · 3.x · 95 · 98 · Me
	Branche NT	NT 3.x · NT 4.0 · 2000 · XP · 2003 · Vista · 2008 · 7 · 2012 · 8 · 10
POSIX / Unix	BSD	FreeBSD · PC-BSD · GhostBSD · DragonFly BSD · FreeNAS · OpenBSD · NetBSD
	GNU/Hurd	Debian GNU/Hurd · Arch Hurd
	GNU/Linux (liste)	Arch · CentOS · Debian · Fedora · Mageia · openSUSE · PCLinuxOS · Puppy · Red Hat · Slackware · Ubuntu
	Autres dérivés	AIX · HP-UX · IRIX · LynxOS · Minix · QNX · Oracle Solaris · System V · Tru64 · UnixWare · ChorusOS · UNICOS · Raspbian
Dérivés d'AmigaOS	MorphOS · AROS	
D'importance historique	CP/M · CTSS · GCOS · Genera · ITS · Multics · Plan 9 · QDOS · RSTS · TENEX · TOPS-20 · TOS · VMS	
Mobile	Android · Bada · BlackBerry OS · Firefox OS · Open webOS · iOS · Sailfish OS · Tizen · Ubuntu Touch · Windows Phone	
Embarqués	pour capteur en réseau	Contiki · TinyOS
	pour carte à puce	Java Card · Multos
	Temps réel	eCos · FreeRTOS · Linux embarqué · LynxOS · MenuetOS · OS-9 · PikeOS · QNX · RTEMS · RTLinux · RTX · µC/OS-II · VxWorks
Autres systèmes	eyeOS · IOS (Cisco) · Inferno · MenuetOS · ReactOS	

# Historique sommaire et ancien !

## Les débuts des SE



# Installation d'un système

Quel système d'exploitation choisir ?

- Qui sont les utilisateurs ?
- Coûts directs (et indirects)
- Matériel (compatibilité)
- Fonctionnalités/Logiciels
- Objectifs à atteindre
- Logiciel libre ou propriétaire

# Installation d'un système

Quelle distribution choisir ?

- De nombreuses distributions Linux
  - [https://fr.wikipedia.org/wiki/Liste\\_des\\_distributions\\_GNU/Linux](https://fr.wikipedia.org/wiki/Liste_des_distributions_GNU/Linux)
- Qualité de la documentation en ligne
  - Communauté active
- Fréquence des mises à jours
- Debian ?

# Installation d'un système

## Méthode d'installation

- Quel support d'installation?
  - CD/DVD
  - Installation par internet avec une image minimale (netinst)
  - Clé USB
  - Live CD ou Live USB
  - PXE (par le réseau)
- Partitionnement des disques
  - Partitionnement classique (avec limitations)
  - LVM (Logical Volume Manager) : plus souple
  - RAID : sécurité des données
- Multi-boot avec d'autres systèmes
  - MBR/UEFI
- Choix des paquets
  - Choix de l'environnement de bureau (GNOME, LXDE, KDE, Cinnamon...)?
- Configurer le réseau et les services

# Linux

GNU/Linux pour les puristes

# Présentation de Linux

- Linux, appellation courante du système d'exploitation libre **GNU/Linux**,
  - implémentation libre d'un plus vieux système d'exploitation → **UNIX**
- Rencontre entre le mouvement du **logiciel libre** et le modèle de développement **collaboratif** et décentralisé via Internet.
- Alternative aux systèmes **propriétaires** Microsoft Windows et Apple Mac OS.
- Autres alternatives libres: **FreeBSD**, NetBSD et OpenBSD.

# Applications disponibles

- Tout type d'applications libres
- Certains logiciels propriétaires fonctionnent également sous Linux.
- Possibilité de faire fonctionner des applications Windows (programme **wine**)

# Historique de Linux

- **1969** Naissance du SE Unix dans les laboratoires Bell (AT&T).
- **1974** Diffusion de Unix dans les universités américaines.
- **1983** Versions commerciales d'Unix.
  
- **1991** Un étudiant finlandais, **Linus Torvalds**, écrit son propre noyau de système d'exploitation pour mieux comprendre le fonctionnement de son ordinateur. Il se base sur Unix mais réécrit tout depuis le début, publie son travail sur Internet (Licence GNU GPL en 1992)
  
- **1993** Première version de la distribution Debian.
  
- **1996** Le manchot *Tux* est utilisé comme symbole pour Linux. Lancement de l'environnement graphique KDE.
  
- **1997** Lancement de l'interface graphique Gnome.
  
- **1998** Début de la commercialisation massive de distributions payantes et de services payants autour de Linux.
  
- **2004**: Début du projet Ubuntu (basé sur Debian) par le millionnaire Sud-africain Mark Shuttleworth (société Canonical).
  
- **2009** : Microsoft contribue au noyau Linux

# Particularités/ Noyau

- Linux est un système d'exploitation :
  - **Multi-tâches** exécute plusieurs programmes à la fois.
  - **Multi-utilisateurs** plusieurs utilisateurs peuvent être actifs sur une même machine en même temps.
  - **Multi-plateformes**
- Versions du **noyau Linux** : Un numéro de version de Linux est noté ainsi : x.y.z
  - x → version de kernel
  - y → révision majeur
  - z → révision mineur
- Noyau développé par Linus Torvald (et la communauté) depuis 1991
- *Dernière version: version 5.8.10 (17 septembre 2020)*
- *Version utilisée par Debian 10.5 : 4.19.0 (octobre 2018)*
- Pour connaître la version utilisée: **uname -r**

# Notion de distribution

- Une distribution Linux est un **ensemble cohérent de logiciels** formant un système d'exploitation composé :
  - d'un **noyau Linux**
  - et d'**applications** diverses destinées aux utilisateurs (navigateur, suite bureautique, lecteur vidéo...).
- Une distribution peut aussi être associée à des **prestations de service** (parfois payantes) et de la documentation.
- Distributions grand public:
  - **Ubuntu** (commerciale orientée grand public, dynamique)
  - Fedora (logiciels fréquemment mis à jours)
  - openSUSE (grand public et professionnels)
  - Mageia (issue de Mandriva, éditée par un association française)
  - Linux mint (grand public)
- Distributions larges
  - **Debian** (très utilisée sur les serveurs)
  - ArchLinux (dernières versions des logiciels)
  - Gentoo (pour les experts)
  - RedHat (distribution commerciale)
  - Slackware (la plus ancienne)

# Le système d'exploitation Debian

# Généralités sur Debian

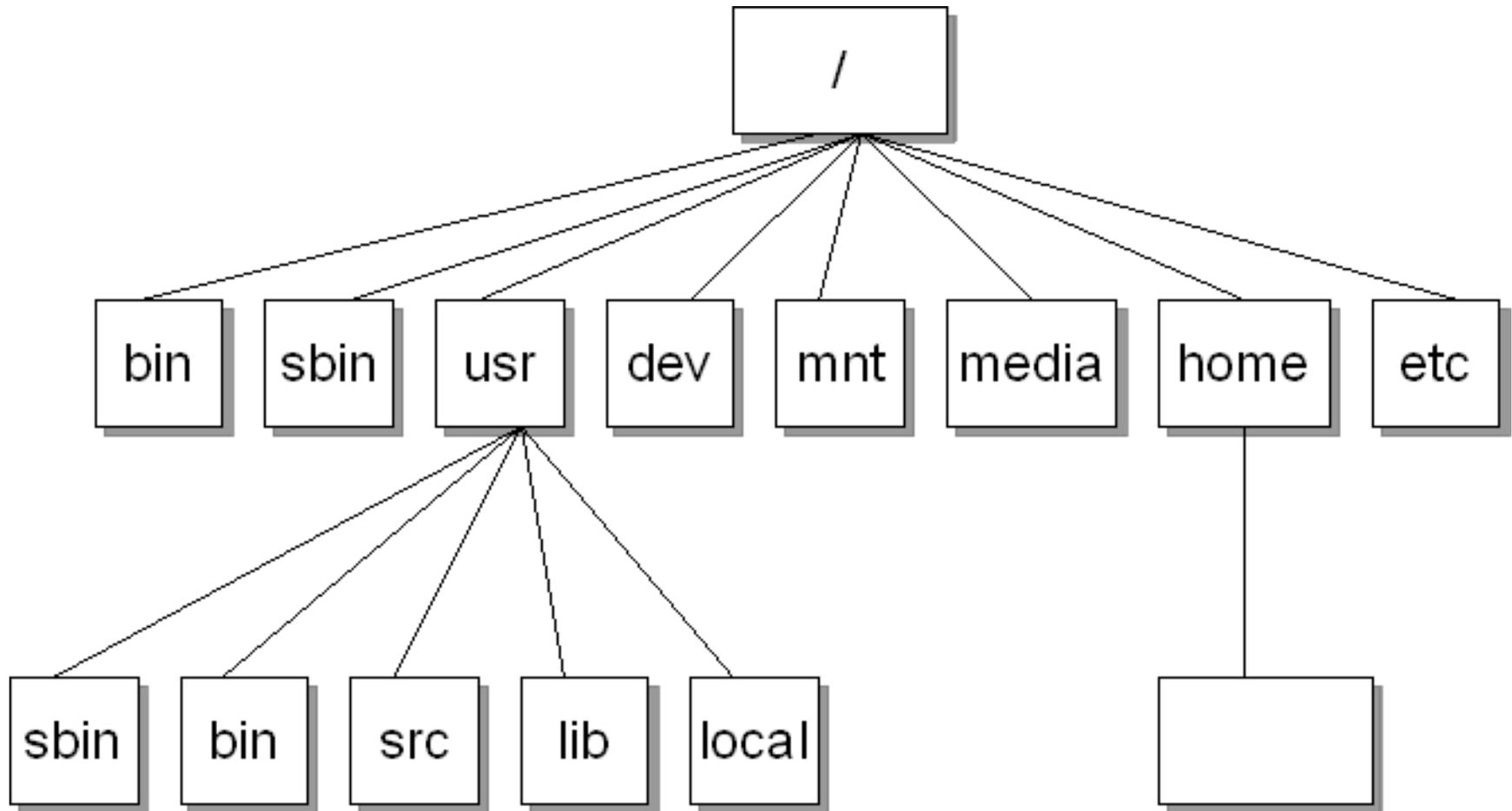
- Système d'exploitation "universel"
  - depuis 1997
- Disponible sur de nombreuses architectures
- Noyau Linux
  - Mais des projets avec d'autres noyaux (Hurd, BSD)
- Entièrement composé de logiciels libres par défaut
- Version utilisée à l'ESIEE en 2023 : 11 (Bullseyes)
- La commande **lsb\_release -a** donne la version de la distribution.



# Le système de fichiers

- Utilise ext4fs comme **système de fichier**
  - Extended file system
  - Depuis 2008, successeur de ext3
- Système de fichiers **journalisé**
  - Enregistrement des modifications dans un journal avant de les faire sur les fichiers.
  - Possibilité de récupération
  - *Autre exemple de SF journalisé : NTFS*
  - *Non journalisé : FAT32, exFAT*

# Arborescence du système de fichiers



# Arborescence du système de fichiers

Répertoire	description
/	Répertoire "racine", point d'entrée du système de fichiers
/boot	Répertoire contenant le noyau Linux et l'amorceur
/bin	Répertoire contenant les exécutables de base, comme par exemple cp, mv, ls, etc.
/dev	Répertoire contenant des fichiers spéciaux nommés <i>devices</i> qui permettent le lien avec les périphériques de la machine
/etc	Répertoire contenant les fichiers de configuration du système
/home	Répertoire contenant les fichiers personnels des utilisateurs (un sous-répertoire par utilisateur)
/lib	Répertoire contenant les bibliothèques et les modules du noyau (/lib/modules)
/media	Répertoire contenant les « points de montage » des médias usuels : CD, DVD, disquette, clef USB
/root	Répertoire personnel de l'administrateur
/sbin	Répertoire contenant les exécutables destinés à l'administration du système
/tmp	Répertoire contenant des fichiers temporaires utilisés par certains programmes
/usr	Répertoire contenant les exécutables des programmes (/usr/bin et /usr/sbin), la documentation (/usr/doc), et les programmes pour le serveur graphique (/usr/X11R6).
/var	Répertoire contenant les fichiers qui servent à la maintenance du système (les fichiers de journaux notamment dans /var/log)

# Le shell

- Donne une interface en **ligne de commande**
- Disponible même sur un système minimal
- De nombreuses variantes et versions :
  - UNIX : sh, csh, tcsh, **bash** ,**dash**...
  - Windows : cmd, powershell
- *bash* est le shell interactif par défaut pour les utilisateurs Debian
  - *dash* est comme interpréteur de commande pour exécuter les scripts (plus rapide)

# Les commandes de gestion des fichiers

- Le dossier \$HOME
- cd : changer de répertoire courant
- pwd : afficher le répertoire courant
- ls : lister les fichiers
  - caractères joker \* et ?
  - options -liR
- Chemin relatif et absolu
- Dossiers biens connus : / ... ~
- Fichiers cachés
- cp : copier des fichiers
  - cp -R pour un répertoire
- mkdir : créer des répertoires
- rm : supprimer
  - rmdir
  - rm -r (répertoires)
  - rm -i (demande de validation)
- mv : renommer
- ln et ln -s : créer un lien

# Autres commandes

- cat file , more file , less file : voir le contenu
- vi file , nano file , emacs file : éditer les fichiers textes
  - vi toujours disponible mais nécessite un apprentissage :q
- file *file* : connaître le type du fichier
- man : aide sur la commande
- locate : chercher un fichier
- Flèche du haut et du bas : historique
  - commande : history

# La gestion des paquets

## Commande apt

- La commande apt (apt-get / aptitude)
  - `apt install nom_du_paquet`
  - `apt purge nom_du_paquet`
  - `apt remove nom_du_paquet`
  - `update` : mise à jour des dépôts
  - `upgrade` : mise à jour des paquets
  - `dist-upgrade` : mise à jour des paquets et des dépendances

# Les dépôts

- **main** -> paquets libres
- **non-free** -> paquets non libres
- **contrib** -> paquets libres mais ayant des dépendances en dehors de main
- **backports** -> dernières versions encore en développement
- Fichier contenant les sources des paquets : `/etc/apt/sources.list`
  - dépôts distants
  - éventuel

```
deb http://deb.debian.org/debian/ buster main
deb-src http://deb.debian.org/debian/ buster main

deb http://security.debian.org/debian-security buster/updates main
deb-src http://security.debian.org/debian-security buster/updates main

# buster-updates, previously known as 'volatile'
deb http://deb.debian.org/debian/ buster-updates main
deb-src http://deb.debian.org/debian/ buster-updates main
```

# Manipulation de texte

Une grande partie des commandes shell consiste en de la manipulation basique de chaîne de caractères.

- **echo** : Afficher un message
- **less, more** : afficher le contenu d'un fichier texte
- **wc** : statistique et comptage
- **cut** : découpage et extraction
- **grep** : recherche dans une chaîne
- **sort** : tri
- **uniq** : suppression des doublons consécutifs
- **tee**: sauvegarder un log dans un fichier
- **sed et awk** : multi-fonctions (plein)
- **cat** : concaténation de fichiers
- Et bien d'autres : `expr, head, tail, join, uniq, tr, nl, strings...`

# Connaître le manuel

Utilisation du manuel incontournable !

- `man man` : Le manuel de man
- `man <commande>` : Le manuel de la commande
- `man -k <mot clé>` : Chercher un mot clé dans tous les manuels
- `man -f <commande>` : Manuel simplifié (équivalent à `whatis`)

Gestion des utilisateurs et des droits

# L'organisation des utilisateurs sous linux

- Système multi-utilisateur.
- Tout le monde ne peut pas tout faire :
  - Excepté l'administrateur (root)
  - Faire partie des « sudoers » (groupe sudo) permet a un utilisateur de lancer des commandes en tant qu'administrateur
- Il faut donc donner des privilèges à certains utilisateurs et en priver d'autres.

# L'organisation des utilisateurs sous linux

- Un utilisateur n'est rien d'autre qu'un entier appelé **UID (User ID)**
- Un groupe est aussi un entier **GID (Group ID)**
- Tout utilisateur appartient au moins à un groupe, appelé **groupe primaire**
- Il peut aussi appartenir à d'autres groupes, dits **groupes secondaires**
- Possibilité de basculer d'un groupe secondaire en lançant un nouveau shell grâce à la commande **newgrp**

# L'organisation des utilisateurs sous linux

- Dès ses débuts, UNIX stockait les informations des utilisateurs dans le fichier `/etc/passwd`
- Et celles concernant les groupes dans `/etc/group`
- Solution par défaut toujours utilisée !
- Il existe aussi des fichiers `/etc/shadow` et `/etc/gshadow` qui reprennent ces mêmes informations et les complètent par les mots de passe chiffrés
- A la différence de `/etc/passwd` et `/etc/group`, ils ne sont lisibles par personne
- Solution par défaut toujours utilisée !
- Mais il en existe d'autres, notamment via **PAM** (**Pluggable Authentication Modules**, permet l'authentification LDAP, Kerberos, SSH, ...)

# Extrait de fichier /etc/passwd

```
sshd:x:74:74:Privilege-separated  
SSH:/usr/share/empty.sshd:/sbin/nologin  
chrony:x:994:992::/var/lib/chrony:/sbin/nologin  
dnsmasq:x:993:991:Dnsmasq DHCP and DNS  
server:/var/lib/dnsmasq:/sbin/nologin  
tcpdump:x:72:72:::/sbin/nologin  
systemd-coredump:x:989:989:systemd Core  
Dumper:/usr/sbin/nologin  
systemd-timesync:x:988:988:systemd Time  
Synchronization:/usr/sbin/nologin  
hilairex:x:1000:1000:Xavier HILAIRE:/home/hilairex:/bin/bash
```

login

UID

GID

Nom complet

Shell par défaut

groupe primaire

Répertoire  
"home"

Mot de passe

# Extrait de fichier /etc/group

wheel:x:10:hilairex

gluster:x:973:

qemu:x:107:hilairex

saslauth:x:76:

libvirt:x:972:

postgres:x:26:

hilairex:x:1000:

L'utilisateur hilairex est aussi membre du groupe wheel (utilisateurs avec privilèges)

L'utilisateur hilairex est aussi membre de qemu



Le groupe hilairex a pour GID 1000. Ne déclare pas que l'utilisateur hilairex est dans ce groupe (fait indépendamment dans /etc/passwd, cf. diapo précédente; ne pas confondre les deux)

# Les droits

- Tout fichier possède obligatoirement un propriétaire (UID) , un groupe (GID), et des droits dès sa création :

```
[hilairex@pc5352a ~]$ ls -l visu*
```

```
-rw-r-xr--. 1 hilairex vboxusers 55039 30 nov. 18:26 visupca.sh
```



L'utilisateur  
propriétaire



Le groupe  
propriétaire



Le fichier



Les droits :

- utilisateur (U) = rw- (lecture r + écriture w)
- groupe (G) = r-x (lecture r, et exécution)
- autres (O) = r-- (lecture r)

# Les droits

- Un seul des ensembles de droits UGO (User, Group, Others) est utilisé pour déterminer les droits d'accès au fichier
- Le système les examine dans l'ordre UGO et une seule fois :
  - S'il y a concordance entre l'UID du fichier et l'UID du programme qui demande l'accès → on applique les droits U
  - Sinon, s'il y a concordance de GID → on applique les droits G
  - Sinon, on applique les droits O
- Les droits sont non cumulatifs. Dans l'exemple précédent, même si hilairex est propriétaire du fichier et appartient à vboxusers, il n'aura pas le droit d'exécution. A l'inverse quelqu'un qui est du group vboxusers sans être hilairex aura ce droit.

# Autre exemple de droits

		droits concernant le <u>propriétaire</u> du fichier			droits concernant les autres membres du même groupe que le <u>groupe propriétaire</u> du fichier			droits concernant tous les <u>autres</u> utilisateurs		
\$ ls -l	type de fichier	u(ser)			g(roup)			o(ther)		
-rwxrw-r--	unfichier	r	w	x	r	w	-	r	-	-
	indique un fichier ordinaire	Le propriétaire peut <u>lire</u> , <u>écrire</u> et <u>exécuter</u> ce fichier			Les membres du même groupe propriétaire peuvent <u>lire</u> et <u>écrire</u> , mais pas <u>exécuter</u> ce fichier			Les autres utilisateurs peuvent seulement <u>lire</u> ce fichier		

# Les bits spéciaux

- En plus des 3 ensembles de bits UGO, il existe un ensemble supplémentaire de **bits spéciaux** :
  - **SetUID** : utilisé avec le droit d'exécution, aura pour effet de lancer l'exécutable sous l'UID de son propriétaire
  - **SetGID** : même chose avec le GID du propriétaire
  - **Sticky** : utilisé avec un répertoire sur lequel le droit d'écriture est accordé à tout le monde, interdit les utilisateurs non propriétaires d'un fichier de supprimer ce fichier.

# Les bits spéciaux : SetUID + SetGID

- Exemple d'utilisation de SetUID : la commande `passwd` → utilisée pour changer de mot de passe
- Doit modifier le fichier `/etc/passwd`
- Mais l'utilisateur n'a pas de droit d'écriture sur ce fichier (heureusement !)

```
[hilairex@pc5352a ~]$ ls -l /etc/passwd
-rw-r--r--. 1 root root 2889 15 févr. 17:44
/etc/passwd
[hilairex@pc5352a ~]$
```

# Les bits spéciaux : SetUID + SetGID

- Solution : root, propriétaire de /usr/bin/passwd, positionne le bit SetUID :

```
[hilairex@pc5352a ~]$ ls -l /usr/bin/passwd  
-rwsr-xr-x. 1 root root 32744 21 janv. 2022  
/usr/bin/passwd  
[hilairex@pc5352a ~]$
```

- La commande passwd s'exécutera en tant que root et pourra ainsi modifier /etc/passwd

# Les bits spéciaux : sticky

- Exemple typique d'utilisation de sticky : /tmp

```
[hilairex@pc5352a ~]$ ls -l / | grep tmp
```

```
drwxrwxrwt 26 root root 880 3 avril 11:47 tmp
```

```
[hilairex@pc5352a ~]$
```

- Tout le monde peut créer ou supprimer des fichiers dans ce répertoire
- Mais personne ne peut supprimer les fichiers des autres

# Les droits sur les fichiers

- Pour voir les droits affectés à un fichier on peut utiliser les commandes `ls -l` ou `stat`
- **chmod** : changer les droits d'un fichier
- Deux façons d'exprimer les droits : littérale et octale
- Littérale : `rwsr_xr_`
  - r : lecture
  - w : écriture
  - x : exécution (pour un fichier) ou accès (pour un répertoire) = 1
  - s et S : SetUID et SetGID
  - t : sticky
  - Exemple (ci-dessus) : `u=rws, g=rx, o=r`

# Les droits sur les fichiers

- Octale : rwsr\_xr\_\_
  - $r=4, w=2, x=1 \rightarrow$  s'applique pour chacun des ensembles de bits de UGO, et donne 3 chiffres en base octale
  - Setuid= 4, setgid=2, sticky=1  $\rightarrow$  forme un chiffre supplémentaire qui doit être placé devant les chiffres UGO
- Exemple : rwsr-xr-- = 4754 car :
  - SetUID positionné  $\rightarrow$  4 frontal
  - $rwX = 4+2+1 = 7$  ;  $r-x = 4+1 = 5$  ;  $r-- = 4$
- Autre exemple : rwxrwxrwt = 1777 car :
  - Sticky positionné  $\rightarrow$  1 frontal
  - $rwX = 7$ , trois fois de suite

# Commandes utilisateur

- **who** : qui est connecté à la machine
- **passwd** : changement de mot de passe
- **adduser**, **useradd**, **deluser** et **userdel**
  - En mode administrateur
  - **adduser** et **deluser** sont des versions améliorées de **useradd** et **userdel**

# Gestion des utilisateurs et des groupes

- **addgroup, usermod** : gestion des groupes
- **chown, chgrp** : Gestion des propriétaires d'un fichier
- Les utilisateurs et les groupes du système sont stockés dans le fichier **/etc/group**
- Les informations sur le dossier home de l'utilisateur et son shell par défaut sont dans **/etc/passwd**