

# Linux débutant

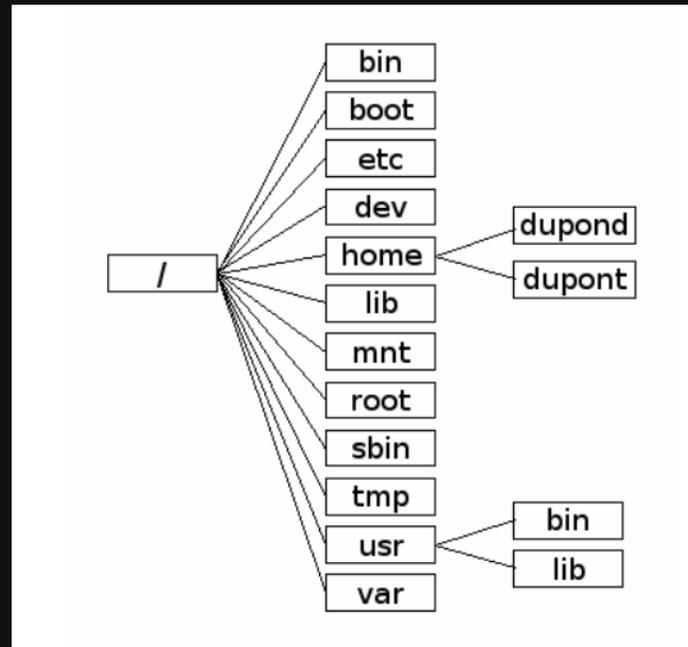
Thibault Ayanides

# Shell, terminal, console



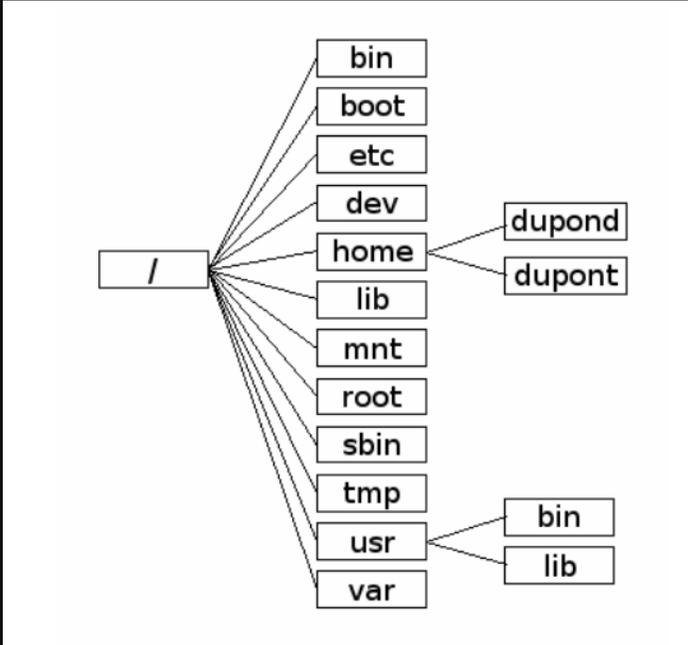
- Shell
  - `bash`, `csh`, `zsh`, `ksh`, `sh`
- Terminal
  - `gnome-terminal`, `mate-terminal`, `terminator`, `xterm`, `konsole`
- Console
  - `tty1`, `tty2`, ... (`CTRL+ATL+F1`, `CTRL+ATL+F2`)

# L'arborescence des fichiers



# Chemins relatifs, chemins absolus

Si vous êtes dans `home`



Chemin absolu `/home/dupond`

- Chemin relatifs `dupond`
- `.` répertoire où l'on est
- `..` répertoire parent

# Le prompt

- `utilisateur@nom-de-la-machine:~$`
- `~` signifie `/home/utilisateur`
- `$` signifie que l'utilisateur a des droits restreints

# Quelques commandes pour commencer ...

```
1 $ pwd
2 /home/tibo/tmp
3 $ ls
4 test1 test2 toto/
5 $ cd toto/
6 $ pwd
7 /home/tibo/tmp/toto
```

- `pwd` permet d'afficher le répertoire courant
- `ls` permet de lister les fichiers et les répertoires d'un dossier
- `cd` permet de changer de répertoire

# Options courtes, options longues

```
1 $ ls -a # option courte
2 . .. image.jpg test1 test2
3 $ ls --all # option longue
4 . .. image.jpg test1 test2
5 $ ls -l # option courte
6 total 712
7 -rw-r--r-- 1 tibo tibo 712178 oct. 14 2019 image.jpg
8 -rw-r--r-- 1 tibo tibo 0 mars 19 14:16 test1
9 -rw-r--r-- 1 tibo tibo 0 mars 19 14:16 test2
10 $ ls --list # option longue
11 total 712
12 -rw-r--r-- 1 tibo tibo 712178 oct. 14 2019 image.jpg
13 -rw-r--r-- 1 tibo tibo 0 mars 19 14:16 test1
14 -rw-r--r-- 1 tibo tibo 0 mars 19 14:16 test2
15 $ ls -lha # combiner plusieurs options
16 total 712
```

```
ls -l
```

```
-w r r 1 tibo tibo 807 Jul 4 10:45 .bash_history
```

- - pour un fichier, d pour un dossier, l pour un lien
- droits de l'utilisateur
- droits du groupe
- droits des autres
- nombre de fichiers qui pointent vers ce fichier
- utilisateur propriétaire
- groupe propriétaire
- poids du fichier en octet
- dernière date de modification
- nom du fichier

# Quelques raccourcis à connaître

- CTRL+ALT+T : ouvrir un terminal
- CTRL+L : effacer l'écran
- CTRL+C : interrompre l'exécution de la commande en cours
- CTRL+D : se déloguer de l'utilisateur courant (comme `exit`)
- CTRL+W : effacer le dernier mot
- TAB : autocomplétion
- MAJ+CTRL+C : copier
- MAJ+CTRL+V : coller

# L'opérateur glob \*

```
$ ls *.jpg  
image.jpg  
$ ls te*  
test1 test2
```

# RTFM !

```
1 $ man ls # manuel de ls
2 $ man man # même man a un manuel
3 $ ls --help # aide concise
4 Usage: ls [OPTION]... [FILE]...
5 List information about the FILEs (the current directory by default).
6 Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
7
8 Mandatory arguments to long options are mandatory for short options too.
9  -a, --all           do not ignore entries starting with .
10 [...]

```

# Gestion des fichiers (1)

- `mkdir` créé un répertoire vide
- `touch` créé un fichier vide

# À vous !

```
1 $ mkdir directory # création de dossier vide
2 $ ls
3 directory/
4 $ touch test1 # création de fichier vide
5 $ ls
6 directory/ test1
```

# Gestion des fichiers (2)

- `mv` déplace des fichiers
- `cp` copie des fichiers
- `rm` supprime des fichiers

# À vous !

```
1 $ mv test1 test2 # on déplace le fichier
2 $ ls
3 directory/ test2
4 $ cp test2 test1 # on copie le fichier
5 $ ls
6 directory/ test1 test2
7 $ rm test* # on supprime tous les test
8 $ ls
9 directory/
```

# Un éditeur de texte

- Nano ! Simple mais atteint vite ses limites
- Permet d'écrire dans un fichier déjà créé ou de créer un fichier et d'écrire dedans
- CTRL+O : sauvegarder les modifications
- CTRL+X : quitter

```
    :::
iLE88Dj. :jd88888Dj:
.LGitE888D.f86jjL8888E;
iE :8888Et. .G8888.
;i E888, :8888;
D888, :8888;
D888, :8888;
D888, :8888;
888W, :8888;
W88W, :8888;
W88W, :8888;
DGGD: :8888;
      :8888;
      :W888;
      :8888;
      E888i
      tW88D
```

# Afficher des fichiers et du texte

- `cat` affiche le contenu d'un fichier
- `echo` affiche des variables et des chaînes de caractères
- `grep` cherche un pattern dans un fichier

# À vous !

```
$ nano test
$ # on édite le fichier test
$ cat test
orange
pomme
poire
banane
$ grep po coucou
pomme
poire
```

# Les groupes et les utilisateurs

- **Utilisateur** : login, mot de passe, groupe(s)
- **Groupes** : regroupe des utilisateurs
- famille, parent

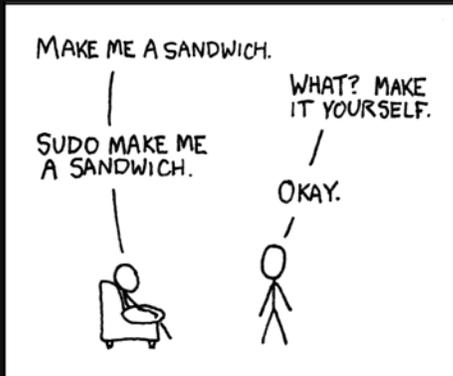
# Les droits (1)

- `chmod (u/g/o) (+/-) (r/w/x) fichier`
- change les droits d'un fichier (ou d'un répertoire avec `-r`)
- `u/g/o` : quelle colonne changer ?
- `+/-` : ajouter ou retirer des droits
- `r/w/x` : quels droits ajouter ?

# Les droits (2)

- `chmod XXX fichier` change tous les droits d'un fichier d'un coup
- `x` est un chiffre entre 0 et 7
- Le premier représente l'utilisateur, le deuxième, le groupe, et le troisième les autres
  - 4 pour `r`
  - 2 pour `w`
  - 1 pour `x`
- L'option `-R` permet de changer les droits récursivement

# Je s'appelle Root ?



- Tous les droits sur tous les fichiers
- À n'utiliser qu'en cas de nécessité
- Le \$ devient # pour montrer que vous avez tous les droits
- `su` permet de se connecter en tant qu'un autre utilisateur (demande le mot de passe `root`)
- `sudo` commande permet d'exécuter une commande en `root` (demande votre mot de passe)
- `sudo su` permet de passer `root` avec votre mot de passe

# Les pipes et les redirections

- Gérer les entrées/sorties des commandes
- Le pipe | permet de rediriger la sortie d'une commande dans l'entrée d'une autre commande
- L'opérateur > permet de rediriger la sortie d'une commande dans un fichier
- L'opérateur >> permet d'ajouter la sortie d'une commande à la fin d'un fichier

# À vous

```
$ cd ~
$ # la sortie de la commande1 devient l'entrée de la commande2
$ ls -alR | grep Do
drwxr-xr-x  2 tibo tibo    4096 janv. 18 11:28 Documents
drwxr-xr-x  6 tibo tibo    4096 mars  18 15:14 Downloads
$ ls > log
$ # on affiche le contenu de log
$ cat log
```

# L'installation de logiciels

Les logiciels sont regroupés dans des **dépôts**.

On les appelle **paquets**, car ils gèrent également chacun des **dépendances**.

# Il était une fois apt ...

- `apt install <paquet>` installe un paquet
- `apt search <paquet>` cherche un paquet
- `apt remove <paquet>` supprime un paquet
- `apt update` met à jour le cache des paquets que l'on peut mettre à jour
- `apt upgrade` met à jour les paquets (faire `update` avant)

# À vous

```
$ # nécessite les droits root
$ apt update
[...]
$ apt install cowsay
[...]
$ cowsay
[...]
$ apt remove cowsay
[...]
```

# Random

- Vim
- cowsay, fortune

# Questions ?