



PRÉSENTATION LINUX

Maxime Tournier

SOMMAIRE

INFORMATION

1. Introduction
2. Distribution
3. Virtualisation

SYSTÈME & FONCTIONNEMENT

4. Terminal VS Shell
5. Système de Fichier
6. Commande de Base
7. Droits & Permissions
8. Plus de commandes
9. Droit d'accès

EXEMPLES D'UTILISATIONS

10. Processus VS Services
11. Script Bash
12. FTP / SSH (OpenSSH)
13. Apache (Web)

INTRODUCTION



OBJECTIFS

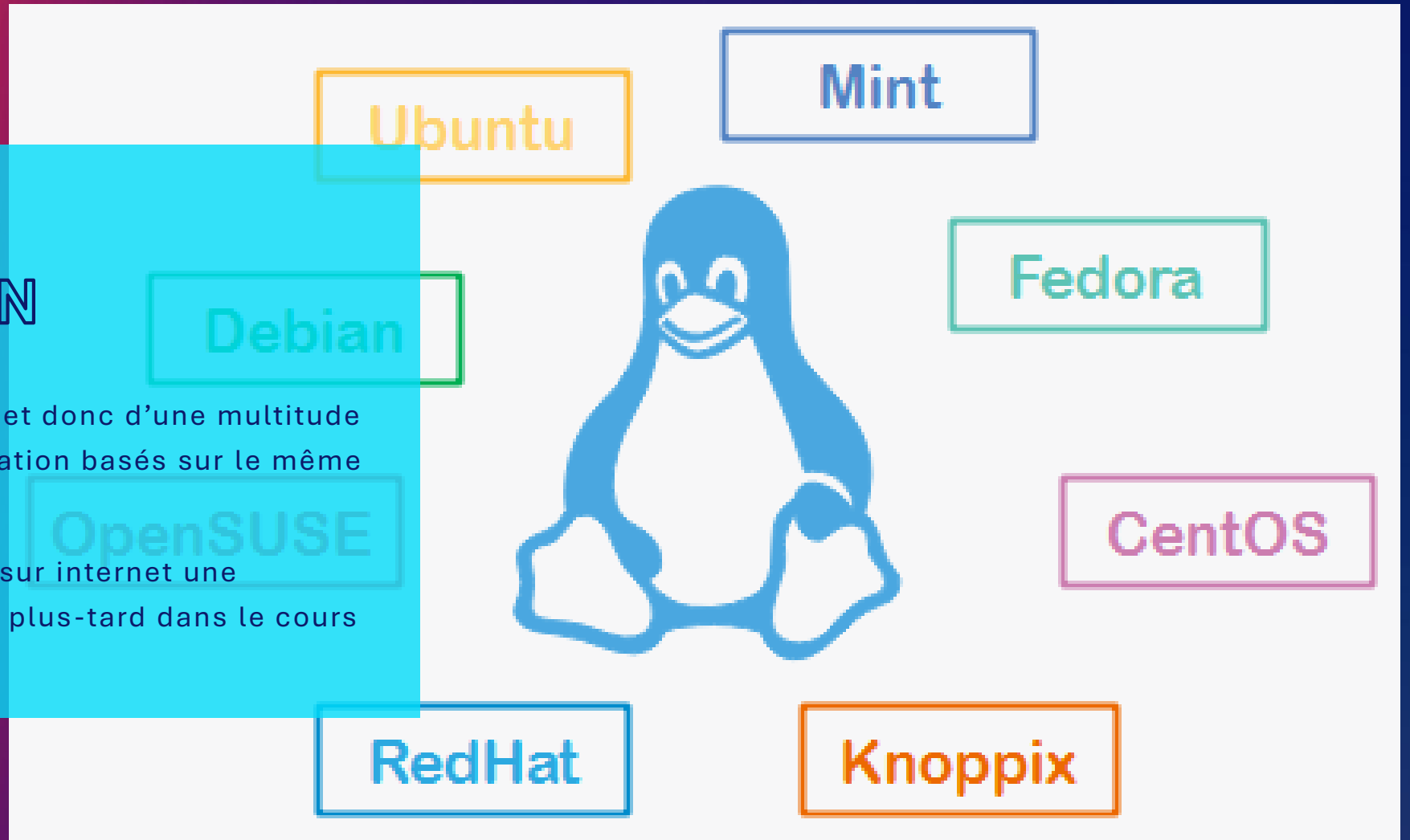
OBJECTIFS

- **Comprendre comment marche le monde du réseau**
- **Savoir utiliser une machine sous Linux**
- **Connaître précisément les avantages de Linux dans la tech**

DISTRIBUTION

Linux est formé de distributions et donc d'une multitude de différents systèmes d'exploitation basés sur le même noyau que celui de GNU/Linux

Activité : Nous allons chercher sur internet une distribution et nous la testerons plus-tard dans le cours



VIRTUALISATION





VIRTUALISATION

UTILISATION

Il existe plusieurs utilisations traditionnelles à la virtualisation

- Virtualisation dans un serveur dédié
- Virtualisation sur une machine client



VIRTUALISATION

PARAVIRTUALISATION

La Paravirtualisation est une manière de virtualisation

- VM pensée pour virtualisation
- Kernel modifié
- Drivers optimisés

VIRTUALISATION HYBRIDE

La Virtualisation Hybride est une manière de virtualisation

- Tout les OS compatibles
- Instructions CPU
- Meilleures performances



VIRTUALISATION

LOGICIELS

Différent logiciels de virtualisation

- VirtualBox (Oracle)
- VMware
- Proxmox VE
- WSL (Windows)
- Qemu



VIRTUALISATION

DIFFÉRENTES VM

Il existe plusieurs manières de lancer une VM

- OVA et OVF
- Installation Complète



VIRTUALISATION

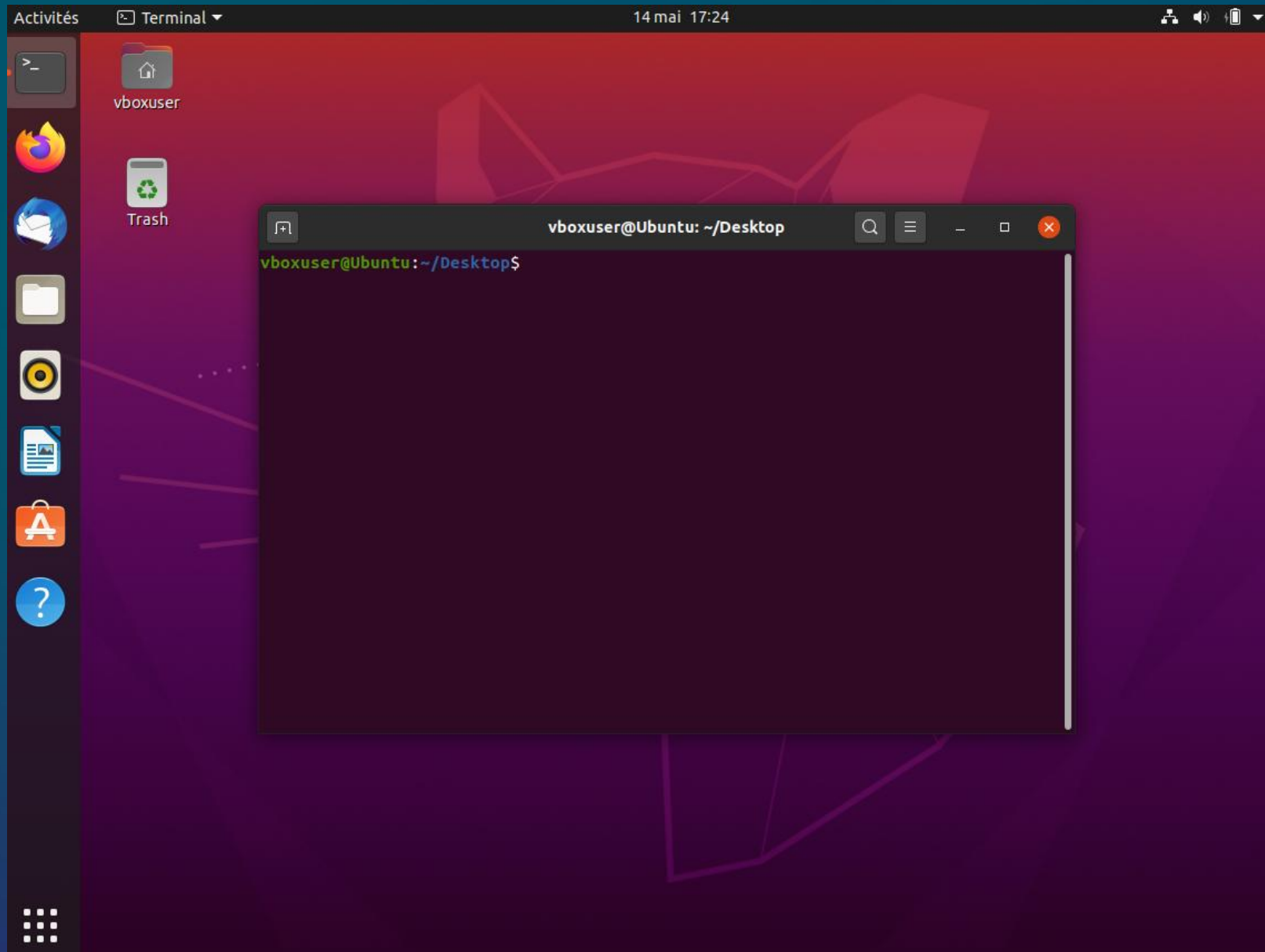
ACTIVITÉ

Nous allons installer VirtualBox et une VM fournies par ton formateur préféré pour prendre en main la distribution :

GNU/Linux Ubuntu

TERMINAL VS SHELL





IMPORTANT

TERMINAL

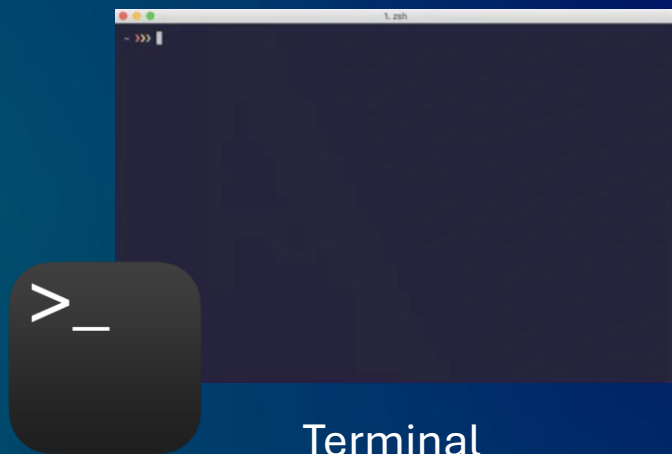
SHELL

**DEUX CHOSES BIEN
DIFFERENTES**

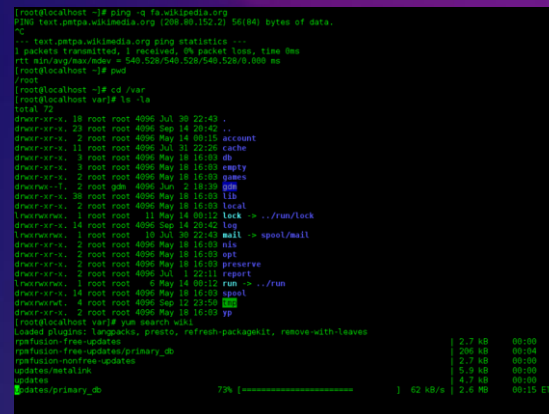
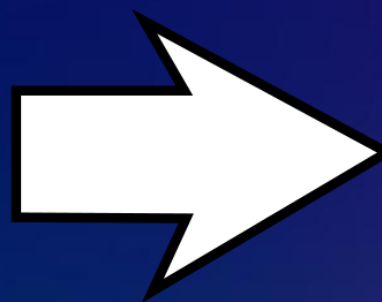


TERMINAL

SHELL



Terminal

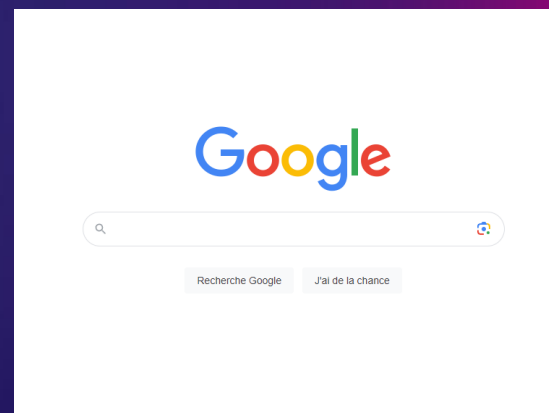
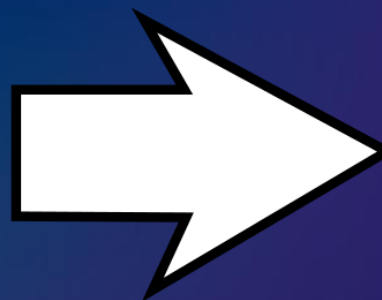


Console

Commande en :

Bash
PowerShell
Batch

Affiche



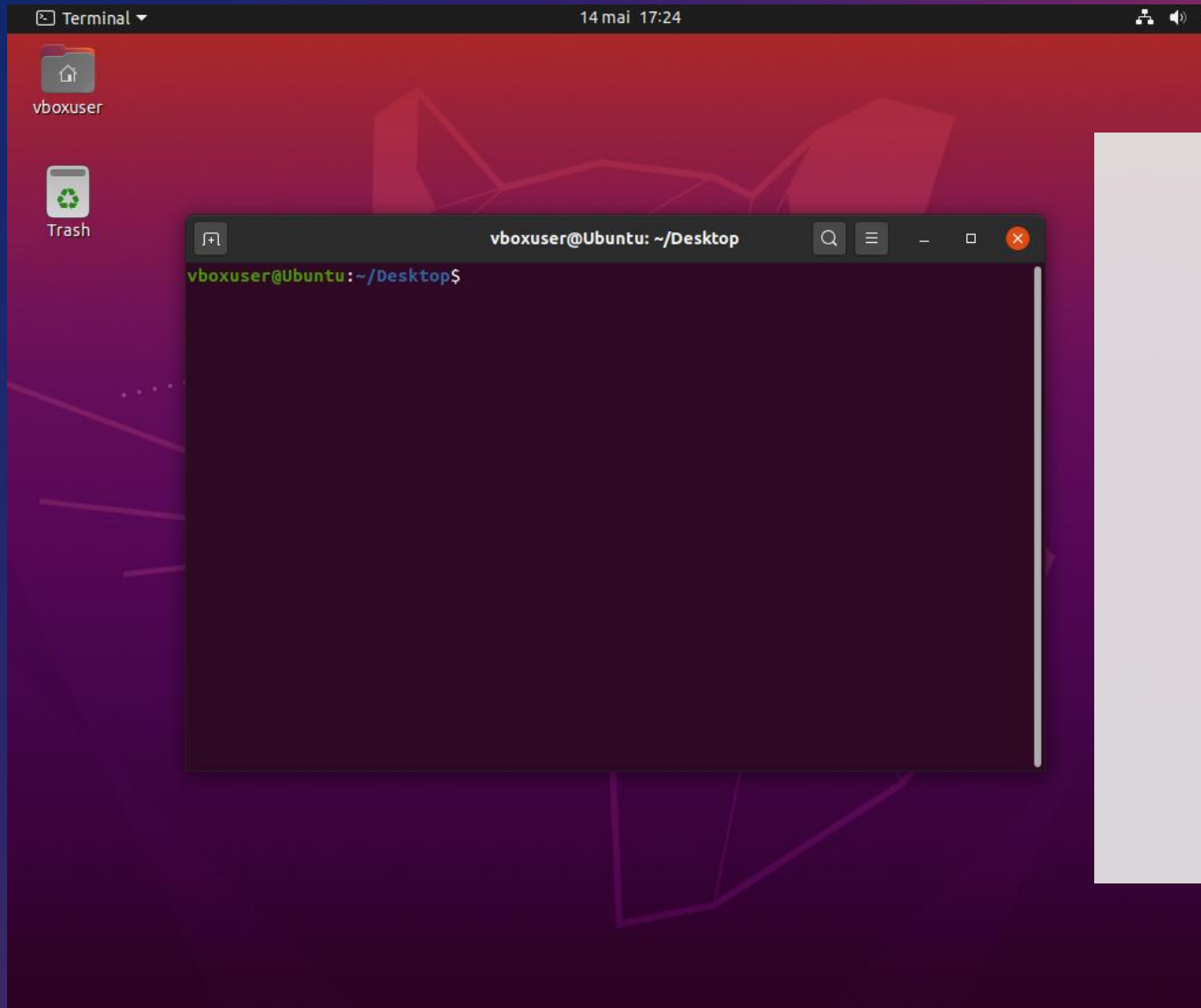
Site Web

Codé en :

Html
CSS
JS



Navigateur Internet



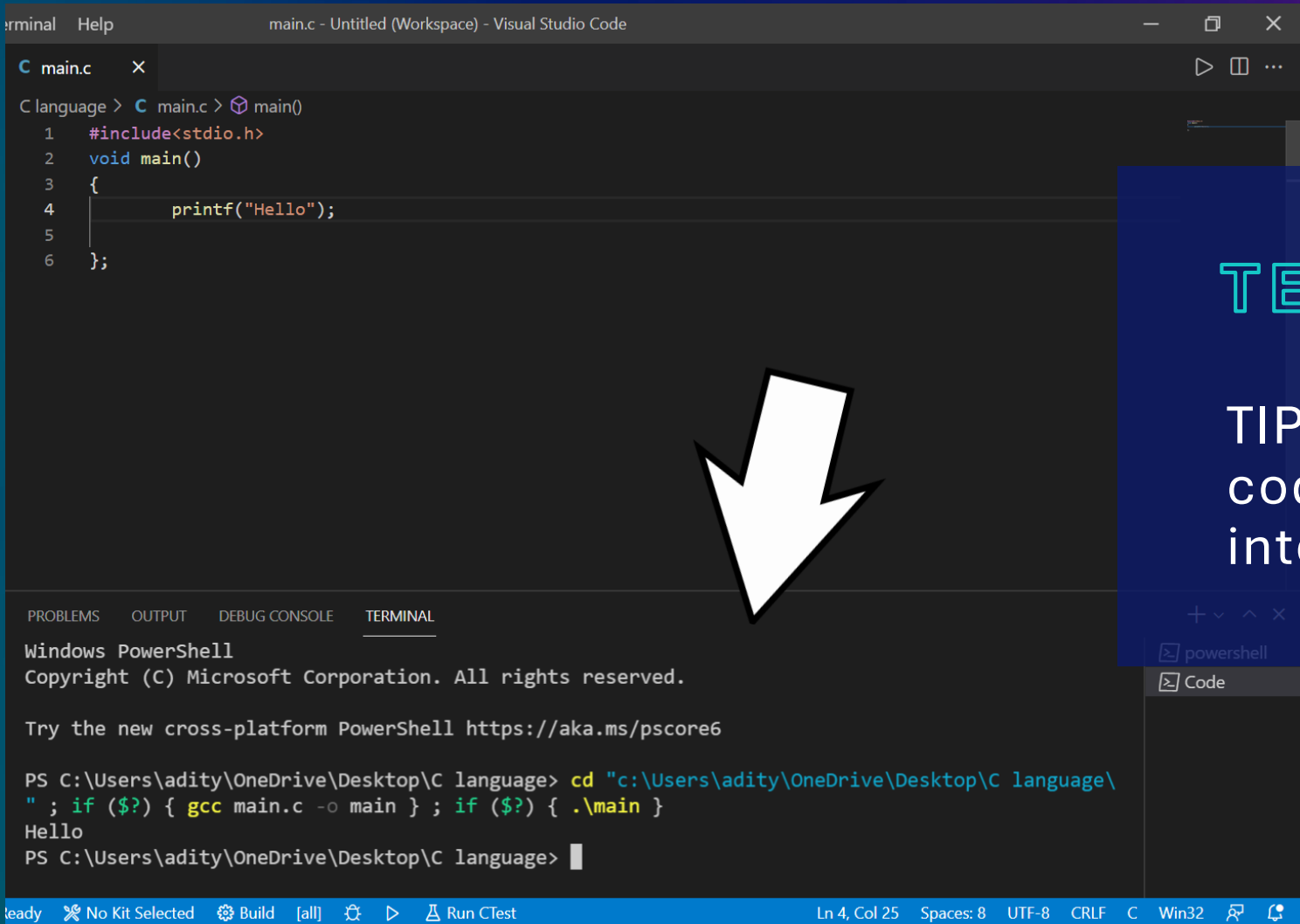
TERMINAL

Il s'agit simplement d'une application

Chaque système d'exploitation en possède au moins une de manière native

On peut toujours le changer en installant une deuxième application qui sert de terminal

Il permet donc d'afficher une console générée par un Shell



The screenshot shows the Visual Studio Code interface. The top editor pane displays a C program named `main.c` with the following code:

```
1 #include<stdio.h>
2 void main()
3 {
4     printf("Hello");
5 }
6 ;
```

A large white arrow points from the code editor to the terminal pane at the bottom. The terminal pane shows the execution of the program in a Windows PowerShell environment:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\adity\OneDrive\Desktop\C language> cd "c:\Users\adity\OneDrive\Desktop\C language\"
PS C:\Users\adity\OneDrive\Desktop\C language> gcc main.c -o main ; if ($?) { .\main }
Hello
PS C:\Users\adity\OneDrive\Desktop\C language>
```

The status bar at the bottom indicates the file is at line 4, column 25, with 8 spaces, UTF-8 encoding, CRLF line endings, and Win32 architecture.

TERMINAL

TIP : Beaucoup d'éditeurs de code possèdent un terminal intégré

SHELL EN DETAIL

ATTENTION !!!

SHELL



BASH

SHELL EN DETAIL

LES 3 SHELL LES PLUS CONNUS



BASH

- Très largement plus utilisé que les autres
- Natif à Linux
- Assez simple d'accès et très complet



BATCH

- Très peu connu et très peu utilisé
- Natif à Windows (ancêtre du MC-DOS)
- Très complexe à prendre en main



POWERSHELL

- Assez connu et le deuxième le plus utilisé
- Créé et pensé pour Windows (Dispo aussi sur Linux)
- Assez complexe mais extrêmement complet (surtout pour Windows)

SHELL EN DETAIL

À RETENIR

IL S'AGIT D'UN LANGUAGE DE
PROGRAMMATION

ILS SONT SOUVENT PENSÉS POUR UN
SEUL SYSTÈME D'EXPLOITATION

SHELL N'EST PAS ÉGAL À BASH !!

QU'EST CE QUE LE PROMPT ?

```
maxime@Tyrolium-Laptop:~/Desktop$
```

1

USER

L'utilisateur auquel
vous êtes connecté à
l'instant

2

MACHINE

Le nom de la machine
auquel vous êtes connecté
à l'instant

Tip : Il peut s'agir d'une IP

3

PATH

Il s'agit de l'endroit où
vous êtes situé dans
les fichiers de la
machine

EXEMPLE DE PROMPT

Tips : Les prompts sont personnalisable

```
maxime@Tyrolium-Laptop:~/Desktop$
```

Prompt par défaut de Bash

```
(maxime@Tyrolium-Laptop)-[~/Desktop]  
$ |
```

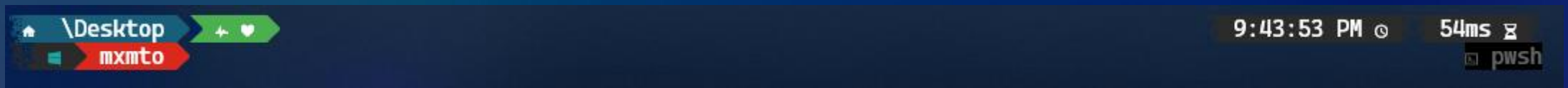
Prompt de Kali-Linux

```
PS C:\Users\mxmto\Desktop> |
```

Prompt par défaut de PowerShell

```
C:\Users\mxmto\Desktop>|
```

Prompt par défaut de Batch



Mon prompt sur PowerShell

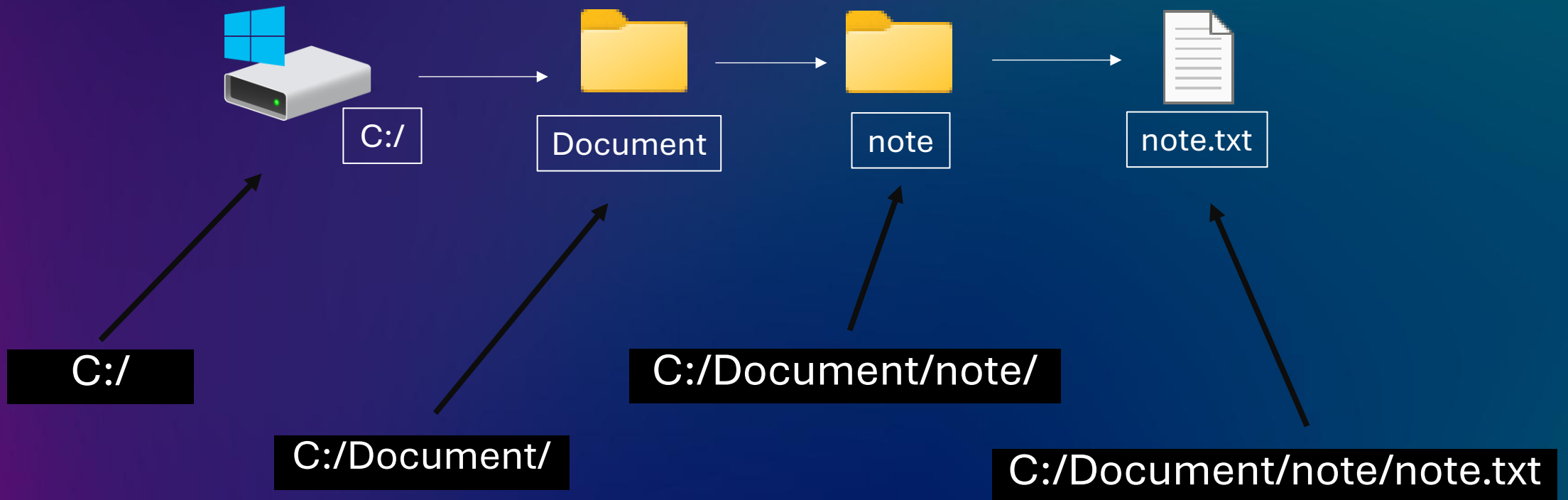
SYSTÈME DE FICHER



SYSTÈME DE FICHER

COMPRENDRE UN PATH (CHEMIN)

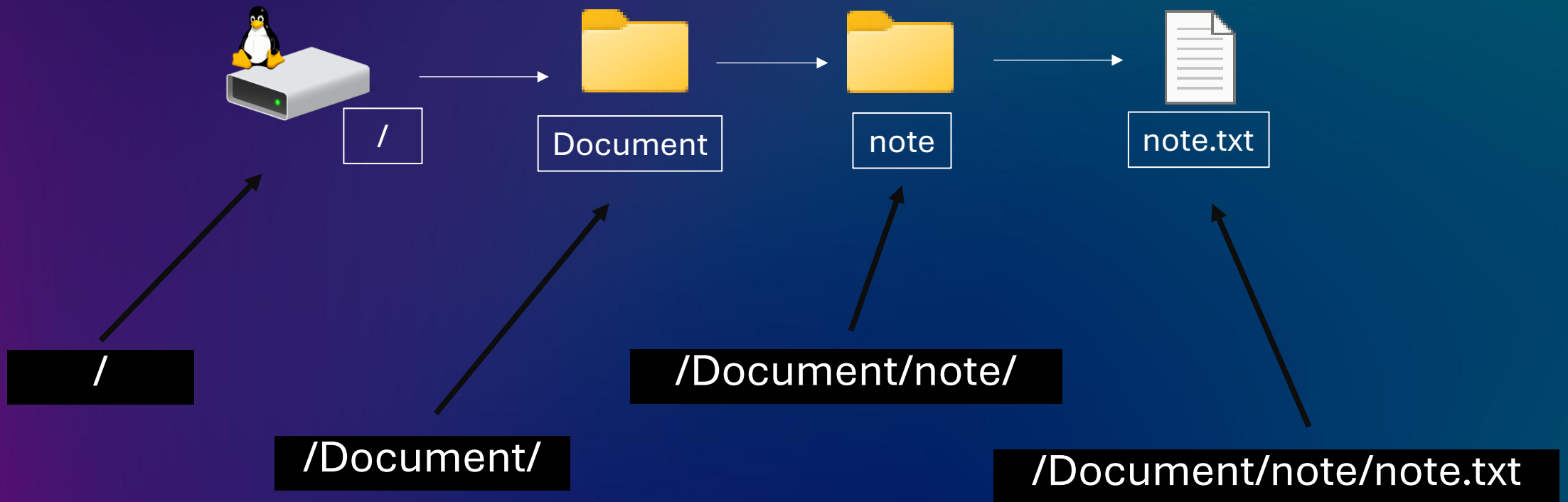
WINDOWS :



SYSTÈME DE FICHER

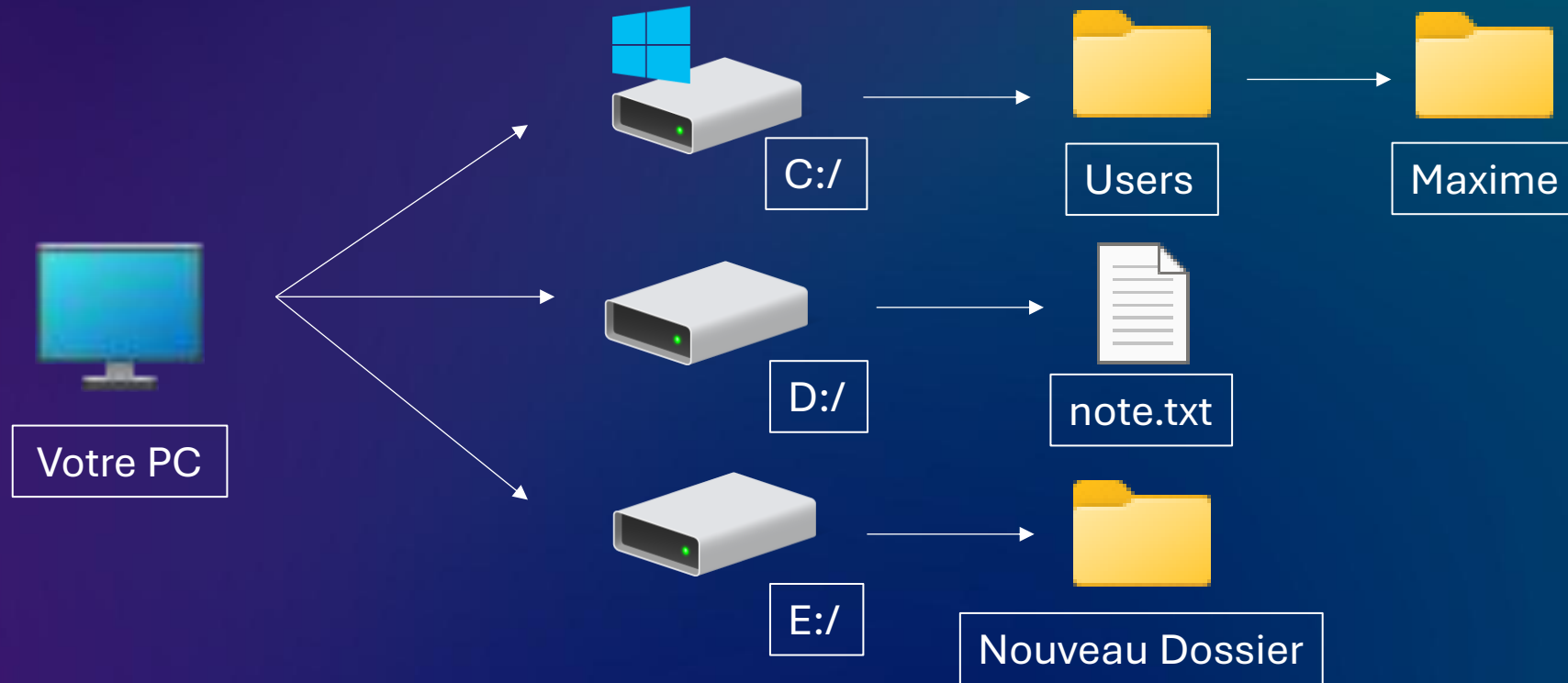
COMPRENDRE UN PATH (CHEMIN)

LINUX :



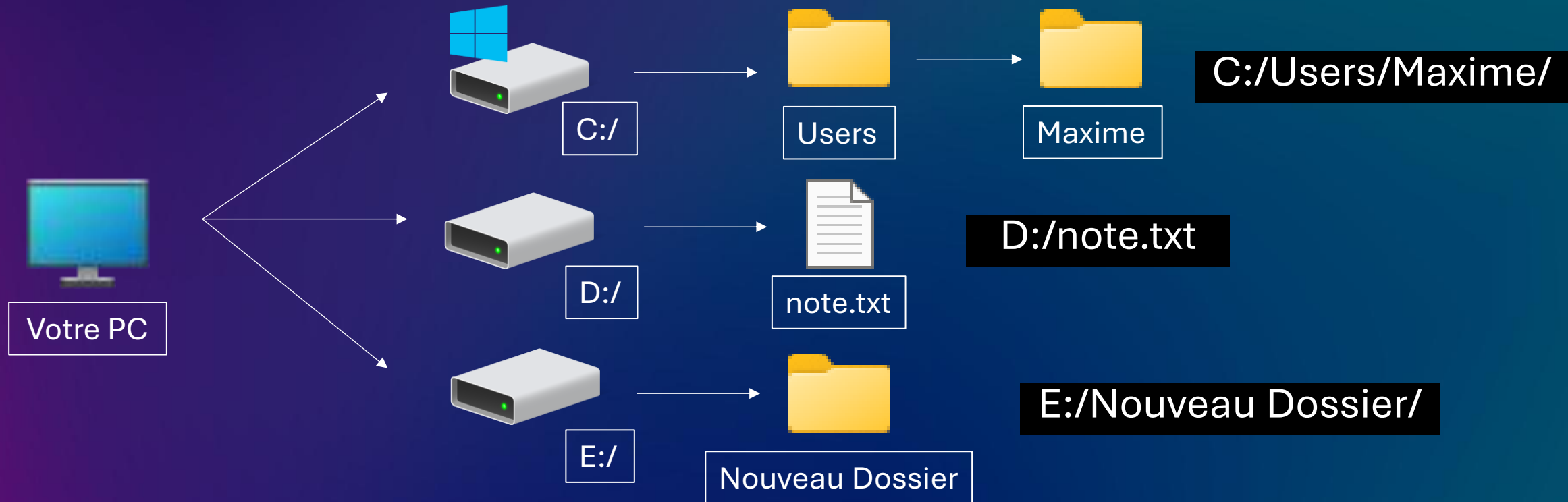
SYSTÈME DE FICHER

ARBORESCENCE WINDOWS



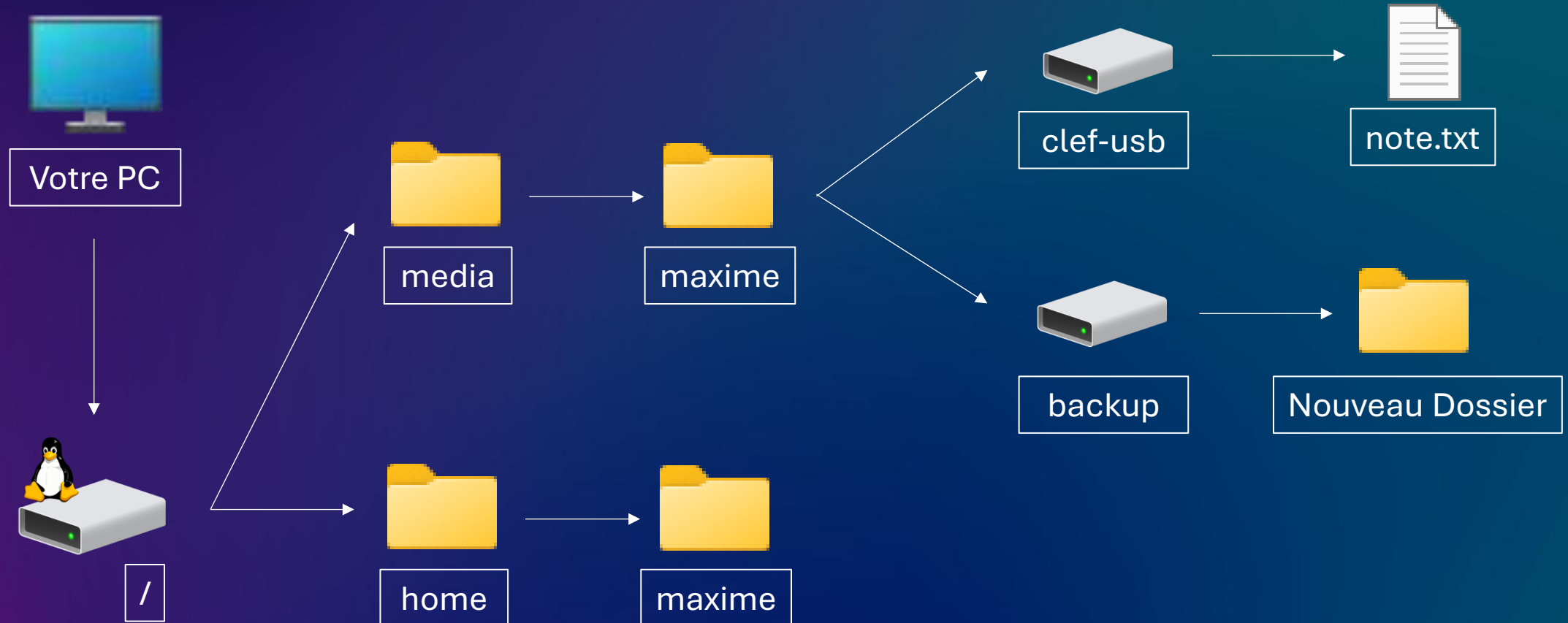
SYSTÈME DE FICHER

ARBORESCENCE WINDOWS



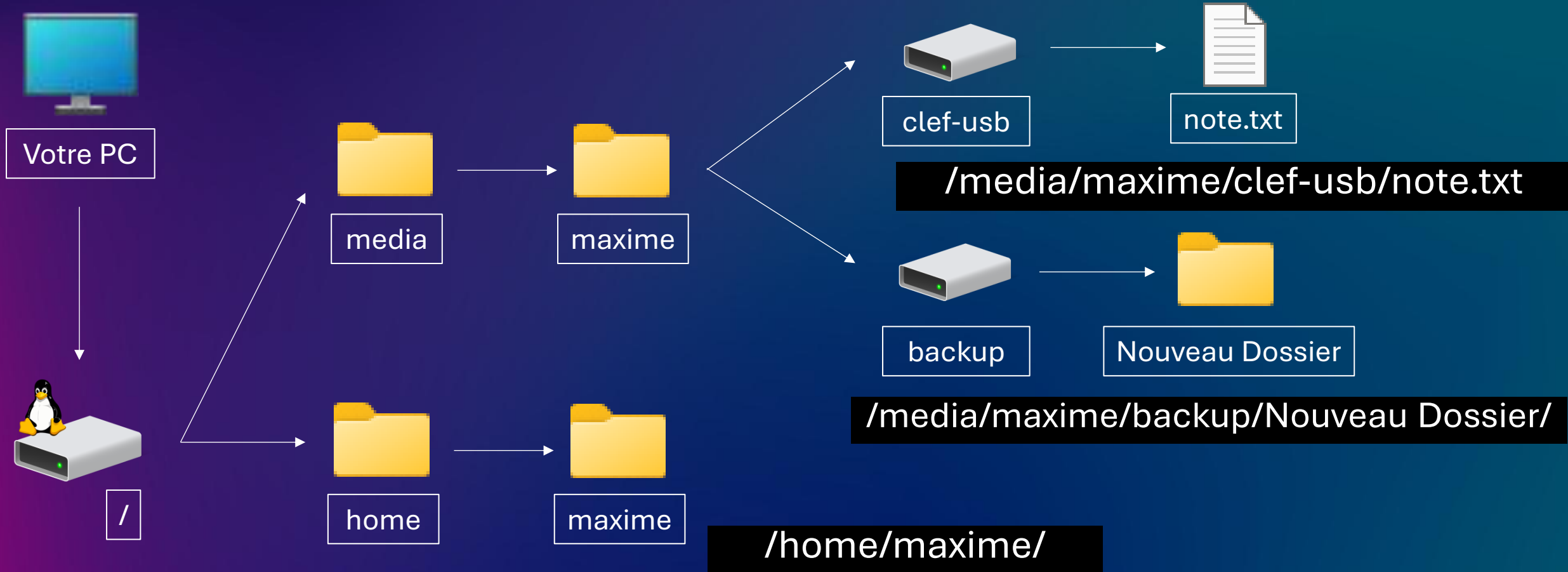
SYSTÈME DE FICHER

ARBORESCENCE LINUX



SYSTÈME DE FICHER

ARBORESCENCE LINUX



Un dossier utilisateur est le dossier qui appartient à chaque utilisateur, qui contient ses dossiers personnels.

Document – Image – Bureau – Téléchargement – Vidéo – Musique
(Document - Picture - Desktop - Download - Vidéo - Music)

WINDOWS :

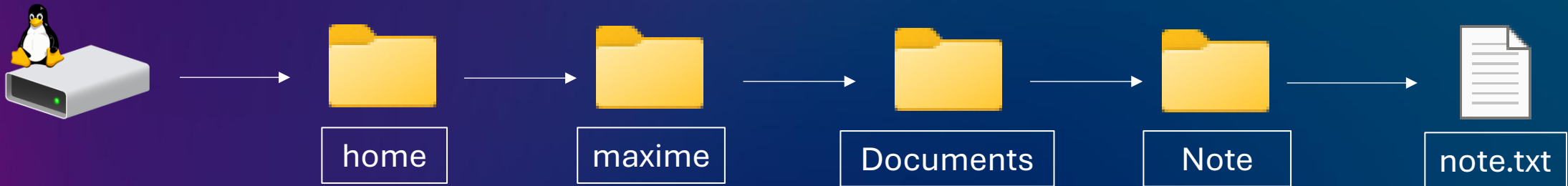
C:/Users/Maxime/

LINUX :

/home/maxime/

~

QUEL EST LE PATH (CHEMIN) ?

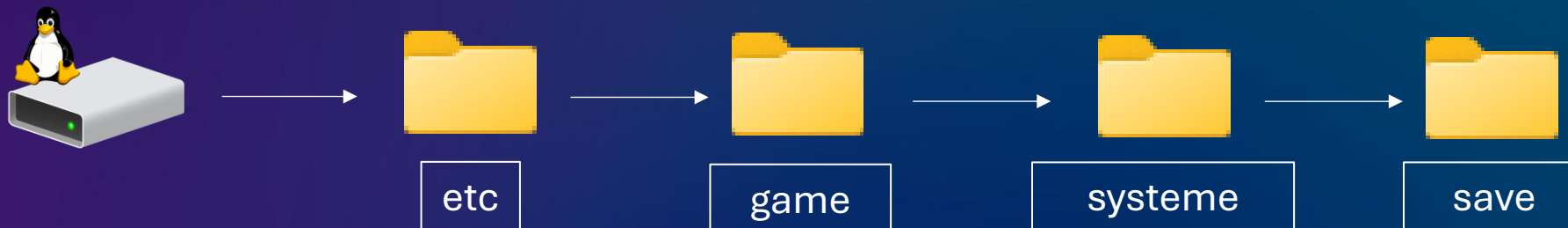


QUEL EST LE PATH (CHEMIN) ?

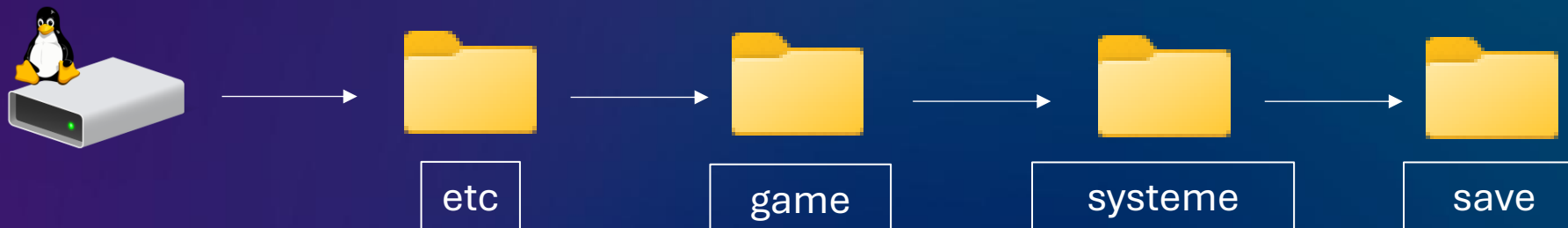


`/home/maxime/Documents/Note/note.txt`

QUEL EST LE PATH (CHEMIN) ?



QUEL EST LE PATH (CHEMIN) ?



`/etc/game/systeme/save/`

COMMANDE DE BASE



PRÉAMBULE

Une commande est un texte qu'on écrit dans le terminal afin d'exécuté une action

Toutes les commandes sont en réalité une application

COMMANDE DE BASE

PREMIÈRE COMMANDE

FONCTIONNEMENT

Toute commande est une application.
Ici pour la première commande on va simplement lancer Firefox

```
$ firefox https://google.fr
```

↑ lance firefox directement sur google.fr

Activité : Exécuter la commande

COMMANDE DE BASE

SUDO (SWITCH USER DO)

FONCTIONNEMENT

SUDO est une application native d'Ubuntu qui fonctionne pour les permissions

On va fournir une commande après celle-ci pour changer d'utilisateur et donc obtenir des permissions

COMMANDE DE BASE

APT (ADVANCED PACKAGING TOOL)

FONCTIONNEMENT

APT est une application native d'Ubuntu qui fonctionne par paquet

On retrouve un url qui va lui-même chercher les applications à installer

COMMANDE DE BASE

APT (ADVANCED PACKAGING TOOL)

MISE À JOUR

Sur les machines Linux (Exemple Ubuntu)
il est important de mettre à jour le système

```
$ apt update
```

↑ met à jour les programmes installés

```
$ apt full-upgrade
```

↑ met à jour les paquets d'APT et les
dépendances nécessaires, de plus il peut
supprimer les dépendances inutiles

COMMANDE DE BASE

APT (ADVANCED PACKAGING TOOL)

COMMANDE

Les commandes fonctionnent avec des arguments

```
$ apt install nsnake
```

- APT : Est l'application demandée
- Install : Argument qui donne l'action à APT
- nSnake : Argument du Logiciel à installer

APT ET SUDO

ACTIVITÉ

Nous allons installer 2 nouvelles applications grâce à APT

```
$ sudo apt install dolphin
```

↑ Un explorateur de fichier

```
$ sudo apt install konsole
```

↑ Un Autre Terminal

+ Bonus

```
$ sudo apt install nsnake
```

↑ Snake dans un terminal

COMMANDE DE BASE

DOLPHIN & KONSOLE

Nous allons nous servir pour la suite du logiciel Dolphin et Konsole

\$ dolphin

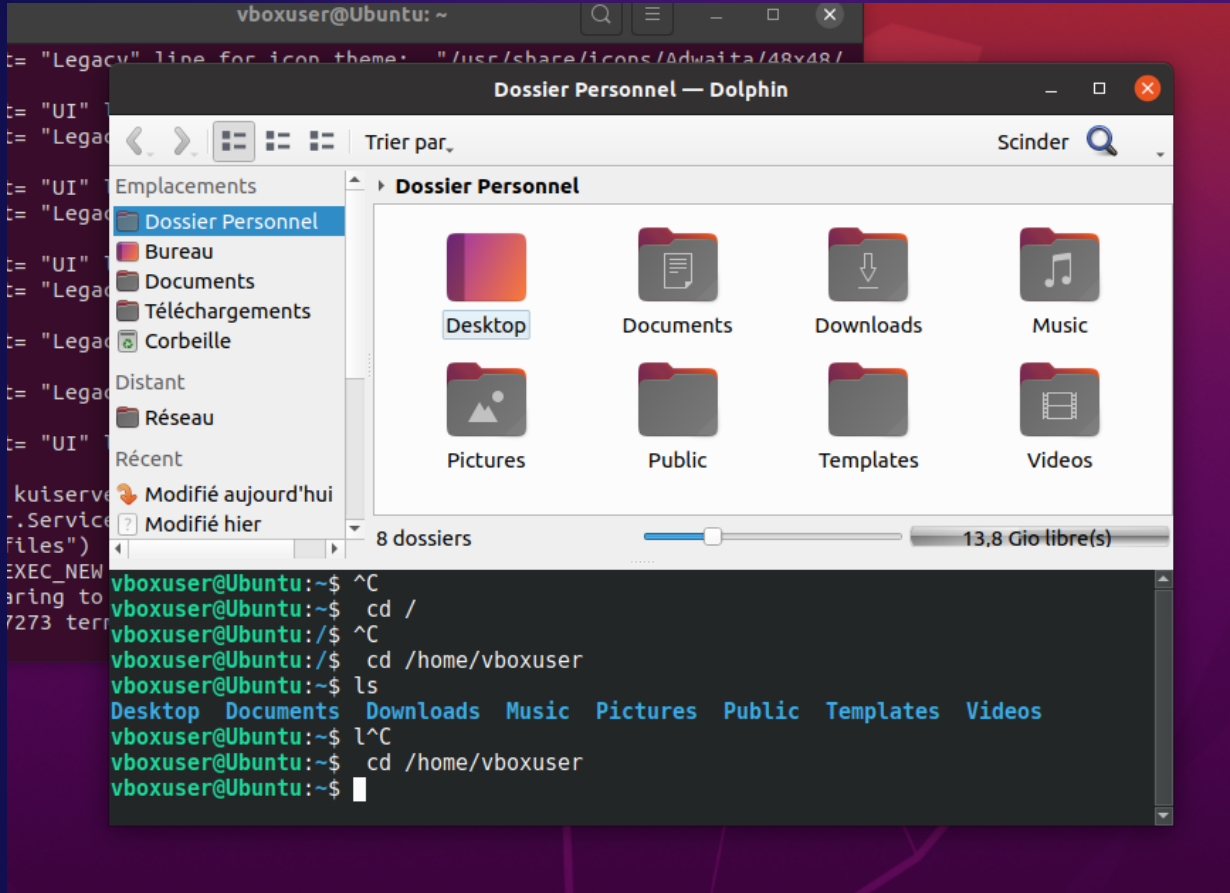
↑ Ouvre le nouvel explorateur de fichier

Puis appuyer sur la touche F4

+ **Bonus**

\$ nSnake

↑ Lance Snake





AFFICHAGE

1. **LS** est la commande pour afficher et rechercher un des éléments dans un dossier
2. **PWD** est une commande pour afficher votre position

```
$ ls [option]
```

- l Affichage long
- h Lecture humaine (Go, Mo)
- a Affiche les fichiers cachés

```
$ ls *.zip : Chercher tout les fichiers en .zip
```

```
$ pwd : Afficher votre position
```



POSITION

3. **CD** est une commande pour se balader concrètement dans les différents dossiers

```
$ cd Picture/
```

Vous venez d'ouvrir le dossier du nom de Picture

```
$ cd ..
```

Dans cet exemple vous venez de sortir du dossier actuel

Tip : Utiliser tab pour afficher la **Liste** des dossiers disponibles et il vous auto-complètera si vous commencez à écrire un dossier



COPIE & DÉPLACÉ

4. **CP** il copie un fichier
5. **MV** permet de déplacer et de renommer un fichier

```
$ cp [option] source destination
```

- r Récursif (dossier)
- f Supprime la destination
- p Préserve les propriétaires

```
$ mv [option] source destination
```

- f Supprime la destination
- b crée un backup du fichier de destination s'il existe

Tip : Pour renommer il suffit de changer le nom du fichier dans la destination



CRÉATION & SUPPRESSION

6. **Mkdir** Créé un dossier

7. **Touch** Créé un fichier

8. **RM** supprimer des
fichiers/dossiers

\$ mkdir dossierbg : a créé un dossier « dossierbg »

\$ touch test.txt : a créé un fichier « test.txt »

\$ rm [option] lamor

-r, -R Récursif

-f Forcer la suppression

Vous venez de supprimé « lamor »

Tip : Pour un dossier on est obligé d'utiliser le récursif

MANIPULATION DE FICHIER

1. **Placez**-vous sur votre dossier utilisateur (~)
2. **Afficher** son contenu
3. **Créer** un dossier "CopieMoiPas"
4. **Placez**-vous dans ce nouveau dossier
5. **Créer** un fichier "Fichier.txt"
6. **Afficher** le contenu du dossier "CopieMoiPas"
7. Dans ce même dossier **copier** le fichier `/etc/passwd`
8. **Afficher** le contenu du dossier "CopieMoiPas"
9. **Renommer** le fichier copier en "JeNePossederPasDeMDP"
10. **Afficher** le contenu du dossier "CopieMoiPas"
11. **Supprimer** le fichier "Fichier.txt"
12. **Afficher** le contenu du dossier "CopieMoiPas"

COMMANDE DE BASE



AFFICHAGE

1. **LS** est la commande pour afficher et rechercher un des éléments dans un dossier

\$ ls [option]

- l Affichage long
- h Lecture humaine (Go, Mo)
- a Affiche les fichiers cachés

\$ ls *.zip : Chercher tout les fichiers en .zip

2. **PWD** est une commande pour afficher votre position

\$ pwd



POSITION

3. **CD** est une commande pour ce balader concrètement dans les différents dossiers

\$ cd dossier/

Vous venez d'ouvrir le dossier du nom de dossier ;)

\$ cd ..

Dans cet exemple vous venez de sortir du dossier actuelle

Tip : Utiliser tab pour afficher la liste des dossiers disponibles et il vous auto-complètera si vous commencez à écrire un dossier



COPIE & DÉPLACÉ

4. **CP** il copie un fichier

\$ cp [option] source destination

- r Récursif (dossier)
- f Supprime la destination
- p Préserve les propriétaires

5. **MV** permet de déplacer et de renommer un fichier

\$ mv [option] source destination

- f Supprime la destination
- b crée un backup du fichier de destination s'il existe

Tip : Pour renommé il suffit de changer le nom du fichier dans la destination



CRÉATION & SUPPRESSION

6. **Mkdir** Crée un dossier

\$ mkdir dossierbg : a créé un fichier « dossierbg »

7. **Touch** Crée un fichier

\$ touch test.txt : a créé un fichier « test.txt »

8. **RM** supprimer des fichiers/dossiers

\$ rm [option] lamor

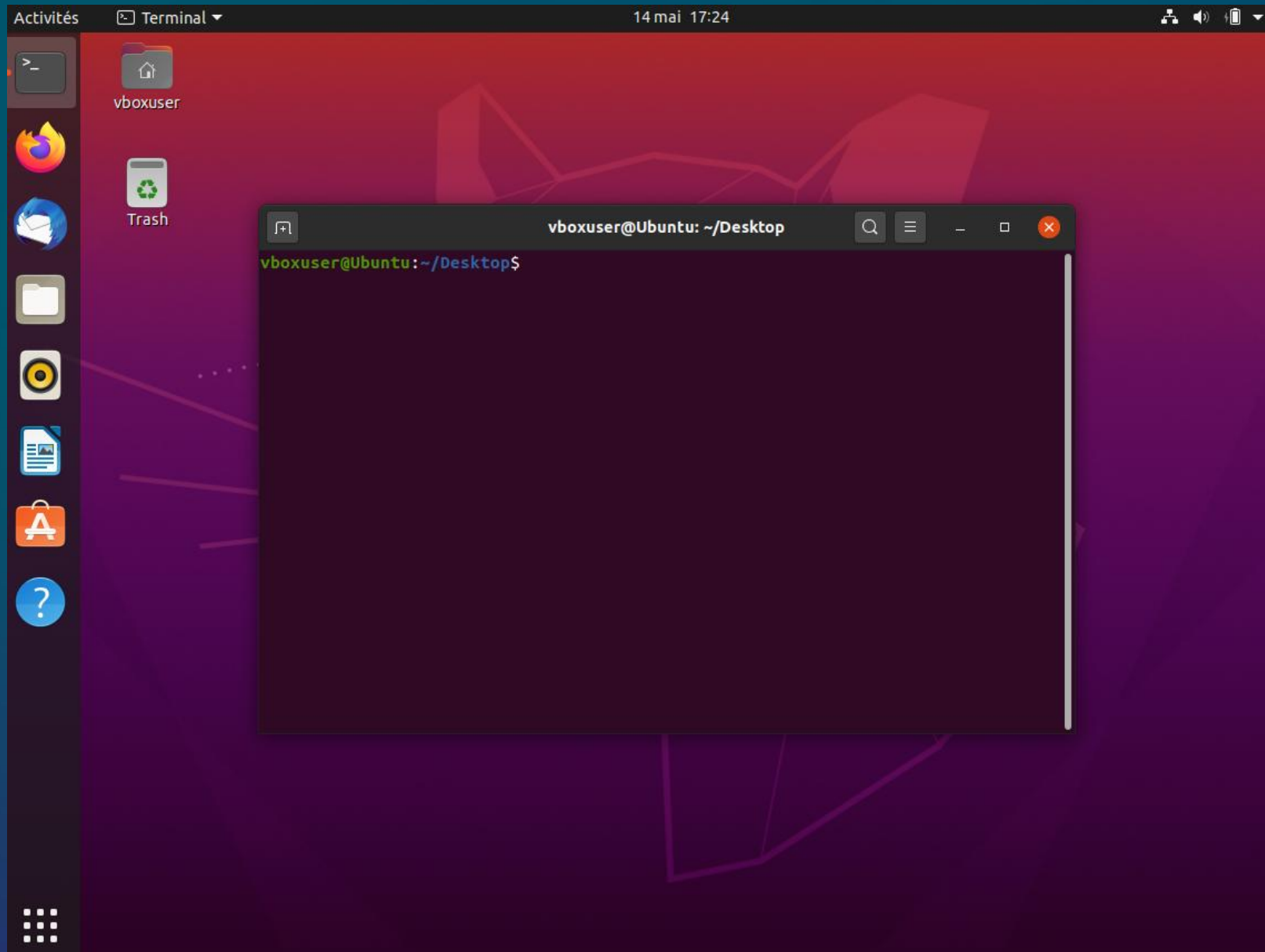
- r, -R Récursif
- f Forcer la suppression

Vous venez de supprimer « lamor »

Tip : Pour un dossier on est obligé d'utiliser le récursif

DROIT & PERMISSION





DROIT ET PERMISSION

PRÉAMBULE

LINUX EST COMPOSÉ
DE PLUSIEURS UTILISATEURS

CHAQUE PROCESSUS TOURNENT
AVEC UN UTILISATEUR ET UN GROUPE

CHAQUE FICHIERS APPARTIENT À
UN SEUL UTILISATEUR ET UN SEUL
GROUPE

IL EXISTE 2 TYPES D'UTILISATEUR



USER RÉEL

- A un nom d'utilisateur
- A un UID (Identifiant Système)
- Et d'autres propriétés personnelles
- Des permissions



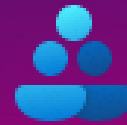
USER SYSTÈME

- Uniquement pour un processus
- Plage d'UID (Généré par le Système)
- Ne peut pas être utilisé par l'humain
- Des permissions

LE ROOT (ADMINISTREUR)

- Est un utilisateur réel
- Il a absolument tout les droits sur la machine
- A un UID qui est égal à 0
- Linux ne fourni pas toujours le mot de passe
- Il est très dangereux de se connecter au root
- Est seulement recommandé pour la création d'utilisateur et de groupe

IL EXISTE 1 SEUL TYPE DE GROUPE



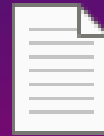
GROUP

- A un nom unique
- A un GID (Identifiant Système)
- Des utilisateurs (Réel ou non)
- Des permissions



/etc/passwd

CE FICHIER CONTIENT TOUTES LES
DONNÉES DES UTILISATEURS



/etc/shadow

CE FICHIER CONTIENT TOUT LES
MOTS DE PASSE DES UTILISATEURS
DE MANIÈRE CHIFFRÉ



/etc/group

CE FICHIER CONTIENT TOUTES LES
DONNÉES DES GROUPES

DROIT ET PERMISSION

SU (Switch User)

```
$ SU [USER]
```

- Permet de switch d'utilisateur
- Si [USER] est pas rempli il essayera avec le compte « Root »

COMMANDE UTILE

WHOAMI

```
$ WHOAMI [USER]
```

Permet d'afficher l'utilisateur

GROUPS

```
$ GROUPS [USER]
```

Permet d'afficher tout les groupes de l'utilisateur

**SUDO NE DONNE PAS DES PERMISSIONS IL EXECUTE
UNE COMMANDE EN ROOT**

WHOAMI

```
$ WHOAMI  
MAXIME
```

WHOAMI AVEC SUDO

```
$ SUDO WHOAMI  
ROOT
```

ON PEUT SE CONNECTER EN ROOT SANS LE MOT DE PASSE ROOT GRACE A SUDO

SU

```
$ SU
```

MOTS DE PASSE ROOT :

SU AVEC SUDO

```
$ SUDO SU
```

MOTS DE PASSE MAXIME :

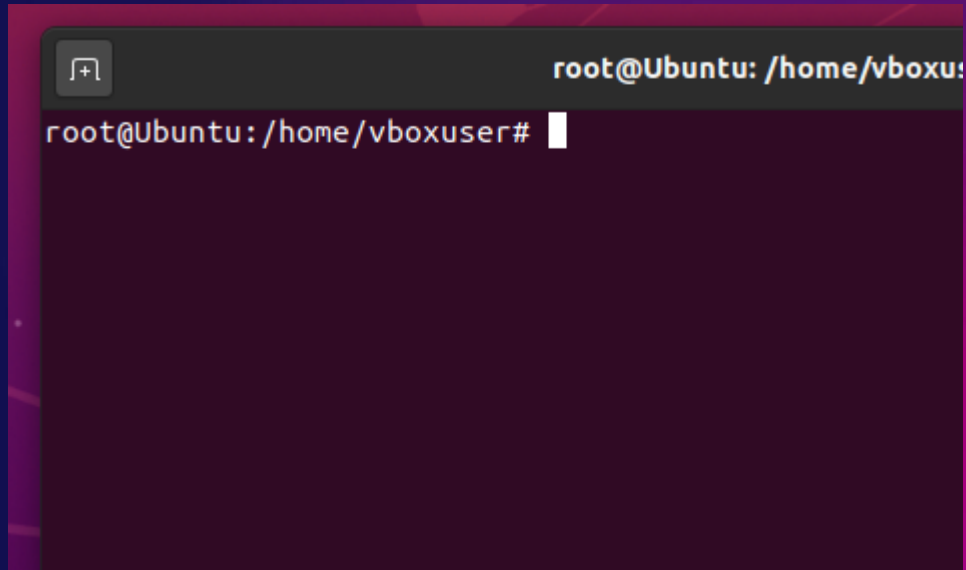


`/etc/sudoers`

CE FICHIER CONTIENT TOUT LES
UTILISATEURS ET LES GROUPES QUI
ON LA PERMISSIONS D'UTILISER
SUDO

DROIT ET PERMISSION

POUR LA SUITE !

A terminal window with a dark background. The title bar shows a window icon and the text 'root@Ubuntu: /home/vboxuser'. The terminal content shows the prompt 'root@Ubuntu: /home/vboxuser#' followed by a white cursor bar.

```
root@Ubuntu: /home/vboxuser#
```

IL FAUT ABSOLUMENT QUE VOUS
SOYEZ CONNECTER EN ROOT

AVEC LA COMMANDE SUIVANTE

```
$ SUDO SU  
[SUDO] MOT DE PASSE VBOXUSER :
```

DROIT ET PERMISSION

```
$ ADDUSER [NOUVEAU-USER]
ADDING USER `AURORE' ...
ADDING NEW GROUP `AURORE' (1001) ...
ADDING NEW USER `AURORE' (1001) WITH GROUP `AURORE' ...
CREATING HOME DIRECTORY `/HOME/AURORE' ...
COPYING FILES FROM `/ETC/SKEL' ...
ENTREZ LE NOUVEAU MOT DE PASSE UNIX : [MDP]
RETAPEZ LE NOUVEAU MOT DE PASSE UNIX : [MDP]
PASSWD : LE MOT DE PASSE A ÉTÉ MIS À JOUR AVEC SUCCÈS
CHANGING THE USER INFORMATION FOR AURORE
ENTER THE NEW VALUE, OR PRESS ENTER FOR THE DEFAULT
    FULL NAME []: AURORE DUPONT
    ROOM NUMBER []: 333, RUE DE LA GARE
    WORK PHONE []: (+33) 04 11 11 11 11
    HOME PHONE []: (+33) 06 11 11 11 11
    OTHER []: AURORE@EXEMPLE.COM
IS THE INFORMATION CORRECT? [Y/N] Y
```

GESTION USER

ADDUSER

- Permet d'ajouter un utilisateur
- Le [NOUVEAU-USER] est égal à AURORE dans l'exemple
- Il va vous demander un mot de passe obligatoire
- Tout le reste des informations ne sont pas obligatoire
- Il va créer un dossier utilisateur dans `/home/aurore`

DELUSER

```
$ DELUSER [USER]  
REMOVING USER `AURORE' ...  
WARNING: GROUP `AURORE' HAS NO MORE MEMBERS.  
DONE.
```

- Permet de supprimer un utilisateur
- Le [USER] est égal à AURORE dans l'exemple

POUR SE CONNECTER A UN NOUVEL UTILISATEUR ON
UTILISERA SU

SU

```
$ SU AURORE  
MOTS DE PASSE AURORE :
```

SU AVEC SUDO

```
$ SUDO SU AURORE  
MOTS DE PASSE MAXIME :
```

PASSWD

```
$ PASSWD [USER]  
NEW PASSWORD:  
RETYPE NEW PASSWORD:  
PASSWD: PASSWORD UPDATED SUCCESSFULLY
```

Permet de changer de mot de passe

ADDGROUP

```
$ ADDGROUP [NOUVEAU-GROUPE]  
ADDING GROUP `FAMILLE' (GID 1001) ...  
DONE.
```

- Permet de créer un groupe
- Le [NOUVEAU-GROUPE] est égal à FAMILLE dans l'exemple
- Le GID est unique à chaque groupe, dans l'exemple il s'agit de 1001

DELGROUP

```
$ DELGROUP [GROUP]  
REMOVING GROUP `FAMILLE' ...  
DONE.
```

- Permet de supprimer un groupe
- Le [GROUP] est égal à FAMILLE dans l'exemple

POUR AJOUTER UN UTILISATEUR A UN GROUPE

```
$ ADDUSER [USER] [GROUP]  
ADDING USER `AURORE' TO GROUP `FAMILLE' ...  
ADDING USER AURORE TO GROUP FAMILLE  
DONE.
```

- Permet d'ajouter un utilisateur à un groupe
- Le [USER] est égal à AURORE dans l'exemple
- Le [GROUP] est égal à FAMILLE dans l'exemple

MANIPULER LES GROUPES ET LES UTILISATEURS

1. Connectez-vous en **root** (Sans le mot de passe root)
2. Créer un groupe du nom de "**amis**"
3. Ajouter "**vboxuser**" dans le groupe "**amis**"
4. Vérifier les groupes de "**vboxuser**"
5. Créer un utilisateur qui s'appelle "**bff**"
6. Ajouter "**bff**" dans le groupe "**amis**"
7. Vérifier les groupes de "**bff**"
8. Connectez-vous à "**bff**"
9. Créer un fichier "**Friend.txt**" dans le dossier utilisateur de "**bff**"
10. Afficher le contenu du dossier

SAVOIR QUEL DOSSIER OU FICHER APPARTIENT A QUEL GROUPE OU QUEL USER

```
$ LS -L  
  
TOTAL 4  
-RW-R--R-- 1  ROOT FAMILLE  0 MAY 16 17:46 TEST.TXT  
DRWXR-XR-X 2 MAXIME FAMILLE 4096 MAY 16 17:40 DEKSTOP
```

User du Dossier/Fichier

Group du Dossier/Fichier

Nom du Dossier/Fichier

CHOWN

PERMET DE CHANGER
L'UTILISATEUR OU LE
GROUPE D'UN
FICHIER/DOSSIER

```
$ CHOWN [USER]:[GROUP] [FILE/DIR]
```

USER :

Il s'agit du nouveau USER du fichier/dossier

GROUP :

Il s'agit du nouveau GROUP du fichier/dossier

FILE/DIR :

Le fichier ou dossier que tu veux changer

EXEMPLE :

```
$ LS -L
DRWXR-XR-X 2 MAXIME FAMILLE 4096 MAY 16 17:40 DEKSTOP
$ CHOWN ROOT DESKTOP
$ LS -L
DRWXR-XR-X 2 ROOT FAMILLE 4096 MAY 16 17:40 DEKSTOP
```

MANIPULER LES GROUPES ET LES UTILISATEURS SUR UN DOSSIER

1. Connectez-vous en **root** (Sans le mot de passe root)
2. Placez-vous dans le dossier **/home/**
3. Faites un **affichage long** du contenu du dossier **/home/**
4. Changez le **groupe** et **l'utilisateur** du dossier **"bff"** pour le **groupe "amis"** et **l'utilisateur "vboxuser"**
5. Vérifier le changement du groupe
6. Connectez-vous sur **"vboxuser"**
7. Créer un fichier **NewFriend.txt** dans le dossier utilisateur de **"bff"**
8. Afficher le contenu du dossier utilisateur de **"bff"**

PLUS DE COMMANDE



ECHO

LA COMMANDE PERMET
D’AFFICHER DU TEXTE
DANS LA TERMINAL

```
$ ECHO [ARGUMENT] [STRING]
```

ARGUMENT :

-e = permet l’interprétation des
échappements antislash

STRING :

Simple texte

EXEMPLE :

```
$ ECHO "COUCOU"  
COUCOU  
$ ECHO -E "COUCOU\NTOI"  
COUCOU  
TOI
```

DATE

**PERMET D’AFFICHER
LA DATE**

```
$ DATE [ARGUMENT]
```

ARGUMENT :

+%Y = Année Uniquement
Etc...

EXEMPLE :

```
$ DATE  
DIM. 14 MAT 2023 16:42:38 CEST
```

MAN

**AFFICHE UN MANUEL
D'UTILISATION D'UNE
COMMANDE**

```
$ MAN [COMMANDE]
```

COMMANDE :

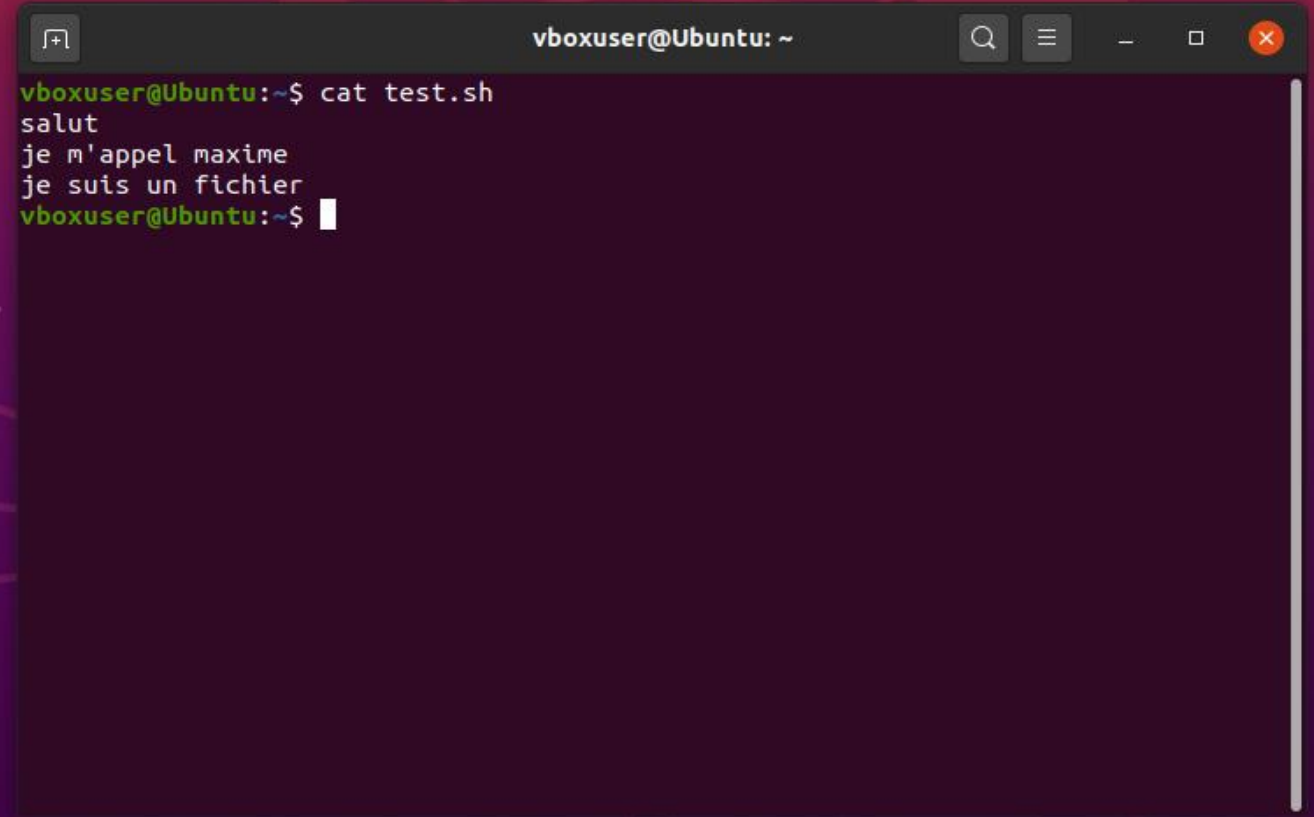
Commande Linux que tu cherches à comprendre

EXEMPLE :

```
$ MAN LS
NAME
    LS - LIST DIRECTORY CONTENTS
SYNOPSIS
    LS [OPTION]... [FILE]...
DESCRIPTION
...
```


PLUS DE COMMANDE

AFFICHAGE DE FICHIER

A terminal window titled 'vboxuser@Ubuntu: ~' with standard window controls. The terminal shows the command 'cat test.sh' being executed, followed by the output: 'salut', 'je m'appel maxime', and 'je suis un fichier'. The prompt 'vboxuser@Ubuntu:~\$' is visible at the end of the output.

```
vboxuser@Ubuntu:~$ cat test.sh
salut
je m'appel maxime
je suis un fichier
vboxuser@Ubuntu:~$
```

CAT

AFFICHER LE CONTENU D'UN FICHER

Recommander pour les
ficher avec peu de texte

```
$ CAT [FILE]
```

FILE :
Ficher Texte

EXEMPLE :

```
$ CAT NOTE.TXT  
- FINIR DE FAIRE LE COURS  
- FINIR DE CRÉER LE FICHER NOTE.TXT
```

LESS

AFFICHER LE CONTENU D'UN FICHER

Recommander pour les
fichiers avec beaucoup
de textes

```
$ LESS [FILE]
```

FILE :
Ficher Texte

Tips : Quitter LESS avec la
touche Q

EXEMPLE :

```
$ LESS NOTE.TXT  
- FINIR DE FAIRE LE COURS  
- FINIR DE CRÉER LE FICHER NOTE.TXT
```

MORE

AFFICHER LE CONTENU D'UN FICHIER

Recommander pour les
fichiers (grand ou petit)

```
$ MORE [FILE]
```

FILE :
Fichier Texte

EXEMPLE :

```
$ MORE NOTE.TXT  
- FINIR DE FAIRE LE COURS  
- FINIR DE CRÉER LE FICHIER NOTE.TXT
```

HEAD

AFFICHER LE DEBUT D'UN FICHIER

```
$ HEAD [ARGUMENT] [FILE]
```

ARGUMENT :

-n [num] = affiche le nombre de ligne à afficher

FILE :

Fichier Texte

EXEMPLE :

```
$ HEAD -N 1 NOTE.TXT  
- FINIR DE FAIRE LE COURS
```

TAIL

AFFICHER LA FIN D'UN FICHIER

```
$ TAIL [ARGUMENT] [FILE]
```

ARGUMENT :

-n [num] = affiche le nombre de ligne à afficher

FILE :

Fichier Texte

EXEMPLE :

```
$ TAIL -N 1 NOTE.TXT
```

- FINIR DE CRÉER LE FICHIER NOTE.TXT

SORT

TRIE LES LIGNES D'UN
FICHIER

```
$ SORT [ARGUMENT] [FILE]
```

ARGUMENT :

- b = ignore certaine ligne (espace, tab)
- f = non-sensible à la case

FILE :

Fichier Texte

EXEMPLE :

```
$ SORT -B NOTE.TXT  
- FINIR DE CRÉER LE FICHIER NOTE.TXT  
- FINIR DE FAIRE LE COURS
```

WC

CALCUL DES INFORMATIONS SUR UN FICHIER TEXTE

```
$ WC [ARGUMENT] [FILE]
```

ARGUMENT :

-l = Compte uniquement le nombre de **ligne**

-w = Compte uniquement le nombre de **mot**

-m = Compte uniquement le nombre de **caractère**

FILE :

Fichier Texte

EXEMPLE :

```
$ WC NOTE.TXT
```

```
2 13 64 NOTE.TXT
```


GREP

AFFICHE UNE RECHERCHE D'UN FICHIER

```
$ GREP [ARGUMENT] [RECHERCHE] [FILE]
```

ARGUMENT :

-i = non-sensible à la case

-n = Affiche le numéro de la ligne

-v = Inverse la commande (Affiche tout sauf la Recherche)

RECHERCHE :

String – Chaîne de caractère

FILE :

Fichier Texte

EXEMPLE :

```
$ GREP -N "FICHIER" NOTE.TXT
```

```
2:- FINIR DE CRÉER LE FICHIER NOTE.TXT
```

```
$ GREP -N -V "FICHIER" NOTE.TXT
```

```
1:- FINIR DE FAIRE LE COURS
```

TIPS

SI VOUS SAVEZ PAS COMMENT MARCHE
UNE COMMANDE

```
$ [COMMANDE] -H
```

```
$ [COMMANDE] -HELP
```

SUR LA GRANDE MAJORITÉ DES
COMMANDES, ELLES SONT
COMPOSÉE D'UN PETIT TUTO GRACE À
CES ARGUMENTS

PLUS DE COMMANDE

EDITIONS DE FICHER

```
vboxuser@Ubuntu: ~
GNU nano 4.8 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin

Lecture de 46 lignes
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^J Justifier  ^C Pos. cur.
^X Quitter   ^R Lire fich.^\ Remplacer  ^U Coller    ^T Orthograp.^_ Aller ligne
```

EDITEUR DE FICHIER DANS LE TERMINAL

NANO

- Très complet et simple d'utilisation
- Installer de base sur les distributions Debian et Ubuntu
- Recommandation de votre formateur ;)

VI

- Très simple et simple d'utilisation
- Installer de base sur absolument toute les distributions Linux
- Recommander pour les petits fichiers

VIM

- Très complet et compliqué d'utilisation
- A installer, installer de base sur aucun système
- Recommander pour les lourd fichiers textes

NANO

```
$ NANO [FILE]
```

FILE :

Fichier Texte existant ou non

**Nano s'occupera de le créer si il n'existe
pas**

Information Nano

- Nombre de ligne
- Message d'erreur

Action possible

- Quitter
- Sauvegarder
- Etc..

TIPS : ^ = Ctrl**TIPS** : Sauvegarder avec
CTRL + S**Fichier charger****Contenu du fichier**

```
GNU nano 4.8 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
```

Lecture de 46 lignes

^G Aide	^O Écrire	^W Chercher	^K Couper	^J Justifier	^C Pos. cur.
^X Quitter	^R Lire fich.	^_ Remplacer	^U Coller	^T Orthograp.	^_ Aller ligne

MANIPULER LES COMMANDES AVANCER ET LES EDITEURS

1. Connectez-vous a "vboxuser"
2. Placez-vous dans son dossier utilisateur
3. Créer un fichier du nom de "CVLinux.txt"
4. Editer le fichier grâce à nano
5. Ecrivez votre nom sur la première ligne
6. Ecrivez votre prénom sur la deuxième ligne
7. Sauvegarder le fichier et quitter nano
8. Vérifier le contenu du fichier grâce à la commande cat
9. Afficher seulement la première ligne du fichier "CVLinux.txt"
10. Afficher seulement la dernière ligne du fichier "CVLinux.txt"

MANIPULER LES COMMANDES AVANCER ET LES EDITEURS

1. Connectez-vous a "vboxuser"
2. Placez-vous dans son dossier utilisateur
3. Créer une copie de "CVLinux.txt" nommée "Backup_CVLinux.txt"
4. Créer un répertoire nommé "Recrutement"
5. Déplacer le fichier " Backup_CVLinux.txt" dans le répertoire "Recrutement"
6. Se déplacer dans le répertoire "Recrutement"
7. Renommer "Backup_CVLinux.txt" en "CVBackup.txt"
8. Ajouter votre adresse e-mail à la fin de "CVBackup.txt"
9. Ouvrez "CVBackup.txt" avec nano, ajoutez votre adresse e-mail à la fin, sauvegardez et quittez.
10. Compter le nombre de lignes dans "CVBackup.txt"
11. Rechercher votre prénom dans "CVBackup.txt" et afficher la ligne correspondante
12. Afficher les 3 premières lignes de "CVBackup.txt"

LES COMBOS DE COMMANDE

EXEMPLE

```
$ LS -HL | GREP "USB"
```

```
DRWXR-XR-X 2 ROOT ROOT 4.0K APR 21 04:07 IPP-USB  
-RW-R--R-- 1 ROOT ROOT 1.5K MAR 25 2022 USB_MODESWITCH.CONF  
DRWXR-XR-X 2 ROOT ROOT 4.0K SEP 6 2021 USB_MODESWITCH.D
```

LES SORTIES (RÉSULTATS DE COMMANDE)

CHAQUE COMMANDE UNE FOIS EXCUTÉ SORT UN TYPE D'INFORMATION
Il est en existe seulement 3

STDIN

ID de sorti = 0

Sorti Lecture

Qui va demander de remplir
quelque chose à l'utilisateur

STDOUT

ID de sorti = 1

Sorti d'affichage

Affiche dans la console une
réponse de type standard

STDERR

ID de sorti = 1

Sorti d'affichage

Affiche dans la console une réponse
de type Erreur

EXEMPLE

```
$ LS -HL | GREP "USB"
```



EXEMPLE

```
$ LS -HL | GREP "USB"
```

STDOUT



The diagram illustrates the flow of data in the command `$ LS -HL | GREP "USB"`. A white arrow originates from the word **STDOUT** and points to the `LS -HL` portion of the command, indicating that the output of the `ls` command is being piped into the `grep` command. Another white arrow originates from the `GREP "USB"` portion and points upwards, indicating the output of the `grep` command.

EXEMPLE

```
$ LS -HL | GREP "USB"
```

STDOUT



A white arrow originates from the word 'STDOUT' and points diagonally upwards and to the right, terminating at the 'LS' command in the terminal command line above.

STDIN



A white arrow originates from the word 'STDIN' and points diagonally upwards and to the left, terminating at the 'GREP' command in the terminal command line above.

EXEMPLE

```
$ LS -HL | GREP "USB"
```

```
DRWXR-XR-X 2 ROOT ROOT 4.0K APR 21 04:07 IPP-USB  
-RW-R--R-- 1 ROOT ROOT 1.5K MAR 25 2022 USB_MODESWITCH.CONF  
DRWXR-XR-X 2 ROOT ROOT 4.0K SEP 6 2021 USB_MODESWITCH.D
```



EXEMPLE

```
$ LS -HL | GREP "USB"
```

```
DRWXR-XR-X 2 ROOT ROOT 4.0K APR 21 04:07 IPP-USB  
-RW-R--R-- 1 ROOT ROOT 1.5K MAR 25 2022 USB_MODESWITCH.CONF  
DRWXR-XR-X 2 ROOT ROOT 4.0K SEP 6 2021 USB_MODESWITCH.D
```



STDOUT


```
$ CMD-A | CMD-B
```

EXEMPLE :

```
$ LS -HL | GREP "USB"
```

```
DRWXR-XR-X  2 ROOT ROOT  4.0K APR 21 04:07 IPP-USB
-RW-R--R--   1 ROOT ROOT  1.5K MAR 25  2022 USB_MODESWITCH.CONF
DRWXR-XR-X  2 ROOT ROOT  4.0K SEP  6  2021 USB_MODESWITCH.D
```

```
$ CMD-A > CMD-B
```

EXEMPLE :

```
$ LS -HL > FILE-LS
```

```
$ CAT FILE-LS
```

```
TOTAL 12
```

```
DRWXR-XR-X 2 MAXIME MAXIME 4096 MAY 13 19:40 BONJOUR
```

```
DRWXR-XR-X 2 MAXIME MAXIME 4096 MAY 14 21:37 DESKTOP
```

```
-RW-R--R-- 1 MAXIME MAXIME 0 MAY 16 17:08 COUCOU
```

```
-RW-R--R-- 1 MAXIME MAXIME 64 MAY 16 15:40 NOTE.TXT
```

Le ">" crée ou écrase le fichier demandé

```
$ CMD-A >> CMD-B
```

EXEMPLE :

```
$ ECHO "COUCOU" >> FILE-LS
```

```
$ CAT FILE-LS
```

```
TOTAL 12
```

```
DRWXR-XR-X 2 MAXIME MAXIME 4096 MAY 13 19:40 BONJOUR
```

```
DRWXR-XR-X 2 MAXIME MAXIME 4096 MAY 14 21:37 DESKTOP
```

```
-RW-R--R-- 1 MAXIME MAXIME 0 MAY 16 17:08 COUCOU
```

```
-RW-R--R-- 1 MAXIME MAXIME 64 MAY 16 15:40 NOTE.TXT
```

```
COUCOU
```

Le ">>" crée ou ajoute à la fin du fichier demandé

```
$ CMD-A | CMD-B > CMD-C
```

EXEMPLE :

```
$ LS -L | GREP -N "DESKTOP" > REQUEST.TXT
```

```
$ CAT REQUEST.TXT
```

```
3:DRWXR-XR-X 2 MAXIME MAXIME 4096 MAY 14 21:37 DESKTOP
```

MANIPULER LES COMMANDES DOUBLE

1. Connectez-vous a "vboxuser"
2. Placez-vous dans son dossier utilisateur
3. Ajouter le résultat de la commande `whoami` dans le fichier "CVLinux.txt" (En une seule commande et sans nano)
4. Vérifier le contenu du fichier grâce à la commande `cat`
5. Editer le fichier "CVLinux.txt" ajouter votre Age à la fin du fichier
6. Sauvegarder le fichier et quitter `nano`
7. Vérifier le contenu du fichier grâce à la commande `more`
8. Ecrivez le résultat de la commande `date` dans un nouveau fichier du nom de "GenialLaDate.txt" (En une seule commande et sans nano)
9. Vérifier le contenu du fichier grâce à la commande `cat`
10. Chercher votre Age dans le fichier "CVLinux.txt" et écrivez le dans un fichier que vous allez créer qui s'appellera "MonAge.txt" (En une seule commande et sans nano)
11. Vérifier le contenu du fichier grâce à la commande `more`
12. Faites un affichage long, sur cette affichage rechercher "Desktop" et ajouter la ligne au fichier "CVLinux.txt" (En une seule commande et sans nano)
13. Vérifier le contenu du fichier grâce à la commande `less`

DROIT D'ACCÈS



SAVOIR QUEL DOSSIER OU FICHER APPARTIENT A QUEL ACCES

\$ LS -L

TOTAL 4

-RW-R--R-- 1 ROOT FAMILLE 0 MAY 16 17:46 TEST.TXT

D**R**W**X**R-**X**R-**X** 2 **MAXIME** **FAMILLE** 4096 MAY 16 17:40 **DEKSTOP**

Type

D = Dossier
- = Fichier

les permissions pour
l'**utilisateur**
propriétaire du fichier

les permissions pour le
groupe propriétaire du
fichier

les permissions pour
les autre utilisateur

Nom du Dossier/Fichier

COMPRENDRE LES ACCES

```
DRWXR-XR-X 2 MAXIME FAMILLE 4096 MAY 16 17:40 DEKSTOP
```

R = Read, Lecture

W = Write, Ecriture

X = Execute, Exécuter

Dans cette exemple
L'utilisateur à le droit de
Read, Write, Excute

Dans cette exemple **le
groupe** à le droit de
Read, Excute

CONCLUSION

LINUX EST COMPOSÉ
DE 3 TYPE DE PERMISSION
READ WRITE EXECUTE

IL SON ATRIBUER A CHAQUE FICHER ET
DOSSIER SOUS FORME DE LETTRE

SI LA PERMSSION ATTRIBUER EST UN –
IL N'EST PAS AUTORISER

DROIT D'ACCÈS

CHMOD

PERMET DE CHANGER
LES DROIT D'ACCES
D'UN FICHIER/DOSSIER

```
$ CHMOD [OPTION] [FILE/DIR]
```

OPTION :

- Qui changer :
 - u (utilisateurs), g (groupes), o (autres)
- Quoi Faire :
 - + (ajouter), - (supprimer)
- Quoi Changer
 - r (read), w (write), x (execute)

FILE/DIR : Le fichier ou dossier que tu veux changer

EXEMPLE :

```
$ LS -L
-RW-RW-R-- 2 MAXIME FAMILLE 4096 MAY 16 17:40 TEST
$ CHMOD U+X TEST
$ LS -L
-RWXRW-R-- 2 MAXIME FAMILLE 4096 MAY 16 17:40 TEST
```

MANIPULER LES ACCES

1. Créer un utilisateur nommé shinji sans options particulières
2. Confirmer qu'il a bien été ajouté dans /etc/passwd (Grâce à la commande grep)
3. Créer un groupe nommé webmaster
4. Ajouter l'utilisateur shinji à ce groupe
5. Confirmer que le groupe a bien été ajouté avec une seule commande
6. Créer le groupe accounting
7. Créer 2 utilisateurs jimmy et sandra et ajouter les aux groupes accounting
8. Confirmer qu'il a bien été ajouté dans /etc/groups (Grâce à la commande grep)
9. Créer un fichier dans le dossier utilisateur de shinji qui s'appellera public.key
10. Changer les permissions du fichier public.key, Tout les utilisateur doit pouvoir écrire ce fichier
11. Constater en écrivant bonjour avec l'utilisateur vboxuser dans public.key (avec nano)
12. Maintenant vous voulez interdire les autres utilisateurs de pouvoir voir les fichiers du Dossier Utilisateur de vboxuser. Modifier les permission en fonction
13. Confirmer que sandra et jimmy ne peuvent pas voir le contenu de votre home
14. Supprimer l'utilisateur jimmy en supprimant aussi ses fichiers

PROCESSUS VS SERVICE



PROCESSUS

PRÉAMBULE

LINUX EST COMPOSÉ
DE PLUSIEURS PROCESSUS

CHAQUE PROCESSUS TOURNENT
AVEC UN UTILISATEUR ET UN GROUPE

CHAQUE PROCESSUS EST
APPLICATION, UNE DEPENCE, UN
MATERIALS OU L'OS

IL EXISTE 1 TYPES DE PROCESS



PROCESS

- S'agit d'Application, Materials ou OS
- Plage d'PID (Généré par le Système)
- D'un Nom (Non-unique)
- D'un chemin

PROCESSUS

HTOP

Gestionnaire des tâches de Linux

Installation

```
$ APT INSTALL HTOP
```

Tips : F10 pour quitté

Exemple :

```
$ HTOP
```


PROCESSUS

HTOP

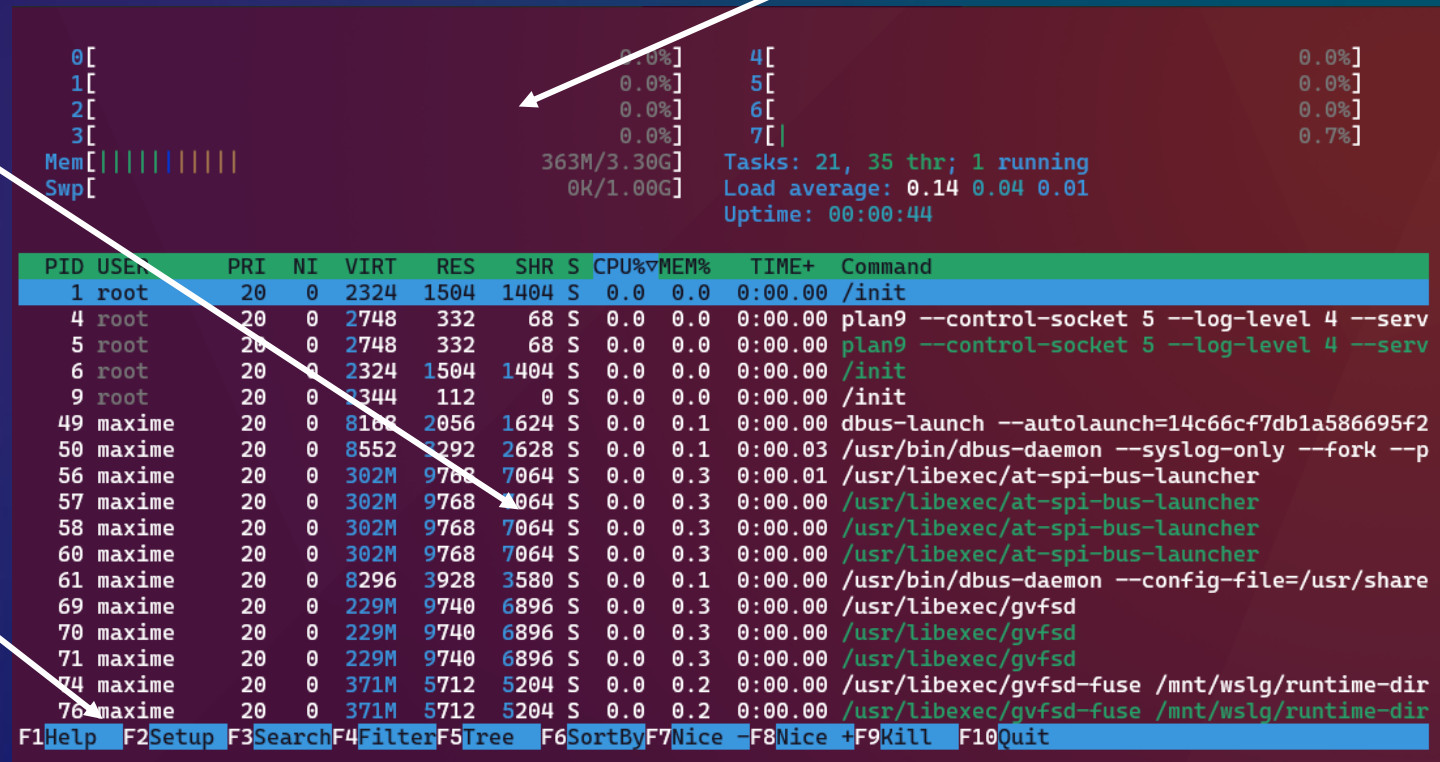
Information sur la stabilité
de votre machine

Tout les
processus

Action possible

- Quitter
- Help
- Etc..

TIPS : Quitter F10



The screenshot shows the htop interface. At the top, system statistics are displayed: CPU usage (0.0%), memory usage (363M/3.30G), and swap usage (0K/1.00G). Below this, a table lists running processes. The table has columns for PID, USER, PRI, NI, VIRT, RES, SHR, S, CPU%, MEM%, TIME+, and Command. The processes listed include /init, plan9, dbus-launch, and various instances of gvfsd and gvfsd-fuse. At the bottom, a menu bar shows keyboard shortcuts: F1Help, F2Setup, F3Search, F4Filter, F5Tree, F6SortBy, F7Nice, F8Nice, F9Kill, and F10Quit.

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	20	0	2324	1504	1404	S	0.0	0.0	0:00.00	/init
4	root	20	0	2748	332	68	S	0.0	0.0	0:00.00	plan9 --control-socket 5 --log-level 4 --serv
5	root	20	0	2748	332	68	S	0.0	0.0	0:00.00	plan9 --control-socket 5 --log-level 4 --serv
6	root	20	0	2324	1504	1404	S	0.0	0.0	0:00.00	/init
9	root	20	0	2344	112	0	S	0.0	0.0	0:00.00	/init
49	maxime	20	0	8168	2056	1624	S	0.0	0.1	0:00.00	dbus-launch --autolaunch=14c66cf7db1a586695f2
50	maxime	20	0	8552	2292	2628	S	0.0	0.1	0:00.03	/usr/bin/dbus-daemon --syslog-only --fork --p
56	maxime	20	0	302M	9768	7064	S	0.0	0.3	0:00.01	/usr/libexec/at-spi-bus-launcher
57	maxime	20	0	302M	9768	7064	S	0.0	0.3	0:00.00	/usr/libexec/at-spi-bus-launcher
58	maxime	20	0	302M	9768	7064	S	0.0	0.3	0:00.00	/usr/libexec/at-spi-bus-launcher
60	maxime	20	0	302M	9768	7064	S	0.0	0.3	0:00.00	/usr/libexec/at-spi-bus-launcher
61	maxime	20	0	8296	3928	3580	S	0.0	0.1	0:00.00	/usr/bin/dbus-daemon --config-file=/usr/share
69	maxime	20	0	229M	9740	6896	S	0.0	0.3	0:00.00	/usr/libexec/gvfsd
70	maxime	20	0	229M	9740	6896	S	0.0	0.3	0:00.00	/usr/libexec/gvfsd
71	maxime	20	0	229M	9740	6896	S	0.0	0.3	0:00.00	/usr/libexec/gvfsd
74	maxime	20	0	371M	5712	5204	S	0.0	0.2	0:00.00	/usr/libexec/gvfsd-fuse /mnt/wslg/runtime-dir
76	maxime	20	0	371M	5712	5204	S	0.0	0.2	0:00.00	/usr/libexec/gvfsd-fuse /mnt/wslg/runtime-dir

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit

PROCESSUS

KILL

**Commande qui permet
De tuer un processus**

```
$ KILL [PID]
```

ARGUMENT :

PID = Id d'un processus

```
$ KILL 4092
```

SERVICE

PRÉAMBULE

LINUX EST COMPOSÉ
DE PLUSIEURS SERVICE

CHAQUE SERVICE EST
UN PROCESSUS

UN SERVICE EST UN PROCESSUS QUI
CE LANCE AU DEMARRACHE DE LA
MACHINE

SERVICE

SERVICE

Permet de réaliser des actions sur les services

```
$ SERVICE [ARGUMENT] [NAME] [ACTION]
```

ARGUMENT :

`--status-all` = affiche tout les services de la machine

NAME :

Nom du service

ACTION :

`Start` = Permet de démarrer un service

`Stop` = Permet d'arrêter un service

`Restart` = Permet de redémarrer un service

EXEMPLE :

```
$ SERVICE APACHE2 RESTART
```

```
$ SERVICE --STATUS-ALL
```

```
[ - ] APPARMOR
```

```
[ ? ] APPORT
```

```
[ - ] AVAHI-DAEMON
```

```
[ - ] BLUETOOTH
```

```
[ - ] CONSOLE-SETUP.SH
```

SCRIPT BASH



SCRIPT BASH

PRÉAMBULE

TOUTES LES COMMANDES BASH PEUVE ÊTRE
ECRIT DANS UN UNIQUEME FICHIER;
ON APPEL SON UN SCRIPT

TOUT CES COMMANDES S'EXCUTERONS
D'UN COUP
ASSOCIER AVEC DES CONDITIONS OU DES
BOUCLE IL EST POSSIBLE DE CRÉER UN
PROGRAMME COMPLEXE

SCRIPT BASH

SH

**LANCE
LES FICHER
SCRIPT (BASH)**

```
$ SH [FILE]
```

FILE :
Fichier Script

EXEMPLE :

```
$ SH SCRIPT.SH  
BONJOUR
```

1. Créer un Script pour Afficher la Date et l'Heure

- Créez un fichier nommé showDateTime.sh.
- Écrivez un script qui utilise la commande date pour afficher la date et l'heure actuelles.
- Donnez au fichier les permissions nécessaires pour être exécuté par tout le monde
- Exécutez le script showDateTime.sh.

2. Script pour Lister les Fichiers du Répertoire Courant

- Créez un fichier nommé listFiles.sh.
- Dans ce script, utilisez la commande ls pour lister tous les fichiers et dossiers du répertoire courant.
- Rendez le script exécutable par seulement l'utilisateur et lancez-le.

SCRIPT BASH

ENCORE PLUS LOINS

Dans les scripts on vas pouvoir ajouter des éléments algorithmique

Notamment faire des variable,
des conditions,
Ou même des boucles
et demander des information à l'utilisateur

CRÉER DES SCRIPTS

1. Créer un Script pour Afficher le prénom de l'utilisateur

- Créez un fichier nommé myName.sh.
- Écrivez un script qui utilise la commande read et une variable pour afficher la prénom de l'utilisateur.
- Donnez au fichier les permissions nécessaires pour être exécuté par tout le monde
- Exécutez le script myName.sh.

1. Créer un Script pour savoir la politesse de l'utilisateur

- Créez un fichier nommé Bonjour.sh.
- Écrivez un script qui utilise la commande read, une variable et une conditions pour savoir si l'utilisateur dit bonjour
- Si oui dit lui bravo, sinon dit lui méchant
- Exécutez le script Bonjour.sh.

FTP / SSH (OPENSSH)



FTP / SSH (OPENSSSH)

UNE IP SAIT QUOI ?

UNE IP EST UN ADRESSE SUR INTERNET

IL EXISTE DEUX VERSION D'IP

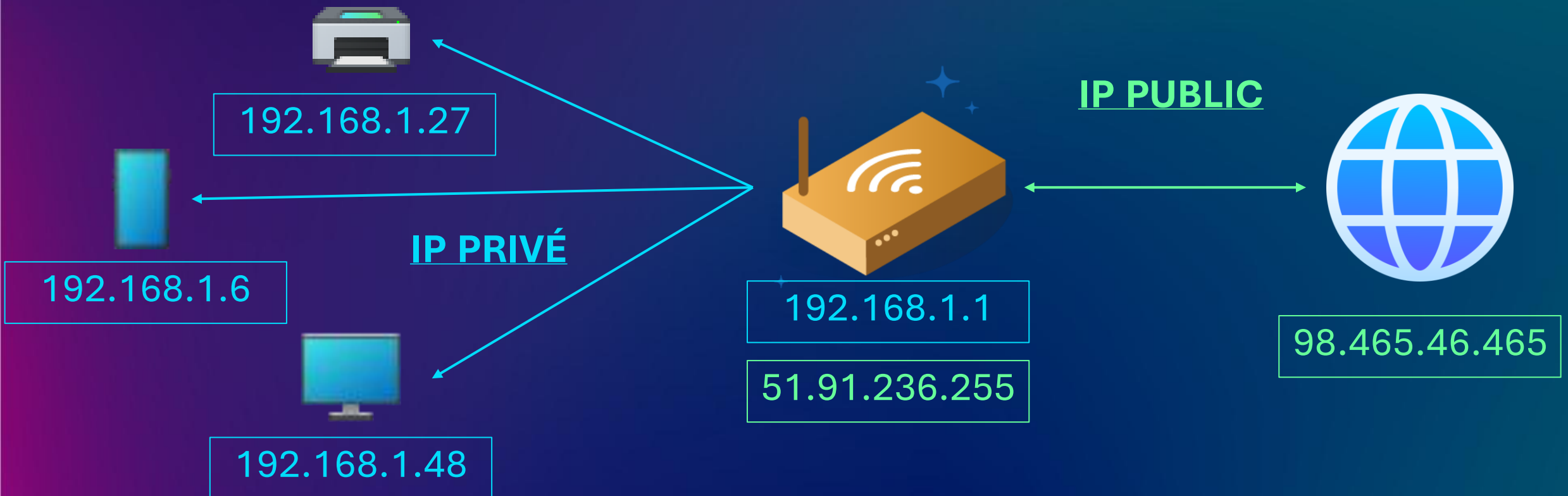
IPV4:

51.91.236.255

IPV6:

2001:0db8:3c4d:0015:0000:d234::3eee:0000

IL EXISTE 2 UTILISATIONS DE L'IP



TIPS POUR DIFFÉRENCIER UNE IP PRIVÉ D'UNE IP PUBLIC

UNE IP PUBLIC DÉMARRE TOUJOURS PAR CES CHIFFRES

IP PRIVÉ:

192.168.1.

TIPS POUR LES CONNEXIONS EN VERS SOIS MÊME



192.168.1.48

LOCALHOST

127.0.0.1

Un protocole est une manière d'envoyer un certain type de donnée

HTTP/HTTPS

- HyperText Transfer Protocol
 - Protocole Utilisé pour transmettre une page web
- HTTPS est chiffré avec le protocole SSL

FTP/SFTP

- File Transfer Protocol
- Protocole Utilisé pour transmettre des fichiers
- SFTP est chiffrer grâce au Secure Shell

SSH

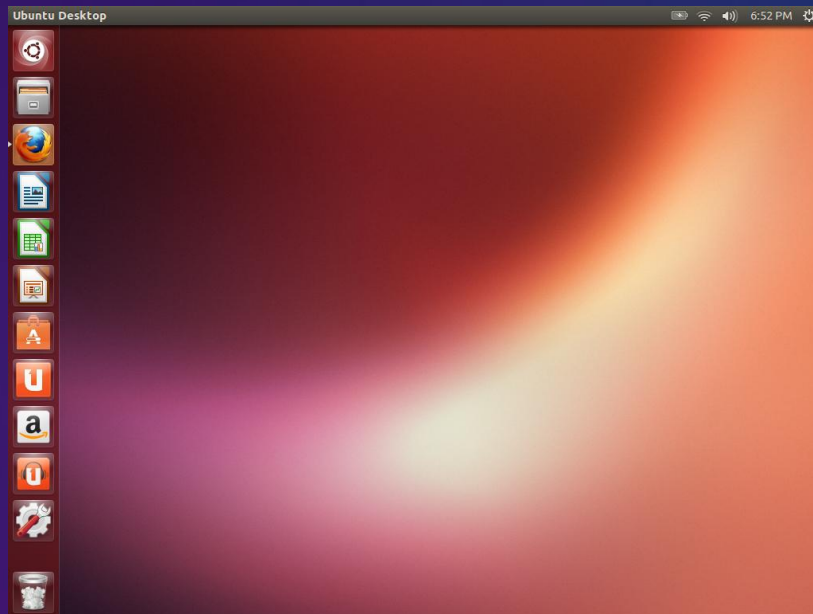
- Secure Shell
- Protocole Utilisé pour se connecter au terminal d'une autre machine

FTP / SSH (OPENSSSH)

DISTRIBUTION

IL EXISTE DEUX GRANDE CATEGORIE DE DISTRIBUTION LINUX

DESKTOP



SERVEUR

```
chroot: can't execute 'glib-compile-schemas': No such file or directory
Using CD-ROM mount point /cdrom/
Identifying... [92043204952947830abda80987b766a7-2]
Scanning disc for index files...
Found 2 package indexes, 0 source indexes, 0 translation indexes and 1 signatures
Found label 'Ubuntu-Server 18.04 LTS _Bionic Beaver_ - Beta amd64 (20180404)'
This disc is called:
'Ubuntu-Server 18.04 LTS _Bionic Beaver_ - Beta amd64 (20180404)'
Copying package lists... [ TIME ] Timed out waiting for device dev-disk-by\x2duuid-00c629d6\x2d06ab\x2d4dfd\x2db21e\x2dc3186f3410
sd.device.
[DEPEND] Dependency failed for /subiquity_config.
[ OK ] Started Uncomplicated firewall.
[ OK ] Started Create list of required static device nodes for the current kernel.
Starting Create Static Device Nodes in /dev...
[ OK ] Mounted POSIX Message Queue File System.
[ OK ] Mounted Kernel Debug File System.
[ OK ] Started Remount Root and Kernel File Systems.
Starting Load/Save Random Seed...
[ OK ] Mounted Huge Pages File System.
[ OK ] Started Load/Save Random Seed.
[ OK ] Started Create Static Device Nodes in /dev.
Starting udev Kernel Device Manager...
[ OK ] Started Journal Service.
Starting Flush Journal to Persistent Storage...
[ OK ] Started udev Kernel Device Manager.
[ OK ] Started Load Kernel Modules.
Mounting Kernel Configuration File System...
Mounting FUSE Control File System...
Starting Apply Kernel Variables...
[ OK ] Started Flush Journal to Persistent Storage.
[ OK ] Mounted FUSE Control File System.
[ OK ] Mounted Kernel Configuration File System.
[ OK ] Started LVM2 metadata daemon.
[ OK ] Started udev Coldplug all Devices.
[ OK ] Started Monitoring of LVM2 mirrors, snapshots etc. using dmeventd or progress polling.
[ OK ] Started Apply Kernel Variables.
```


SSH

**PERMET DE SE
CONNECTER AU TERMINAL
D'UNE AUTRE MACHINE**

```
$ SSH [USER]@[HOSTNAME]
```

ARGUMENT :

USER = Utilisateur de la machine du serveur

HOSTNAME = IP du serveur auquel vous souhaitez vous connectez

EXEMPLE :

```
$ SSH MAXIME@51.89.468.15
```

MOT DE PASSE :

**OpenSSH est l'application qui permet
au SSH de fonctionner coté serveur**

Installation de OpenSSH

```
$ APT-GET INSTALL OPENSSSH-SERVER
```

Une fois installé tu pourras te connecter à cette machine
grâce à la commande SSH

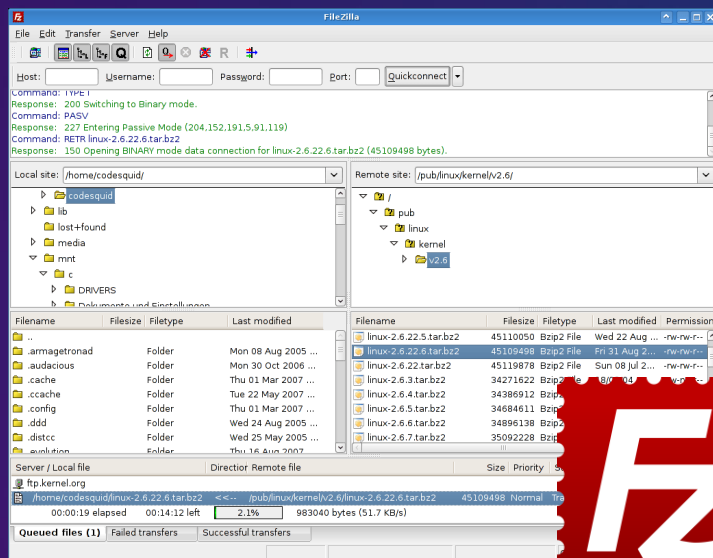
FTP / SSH (OPENSSSH)

FTP/SFTP

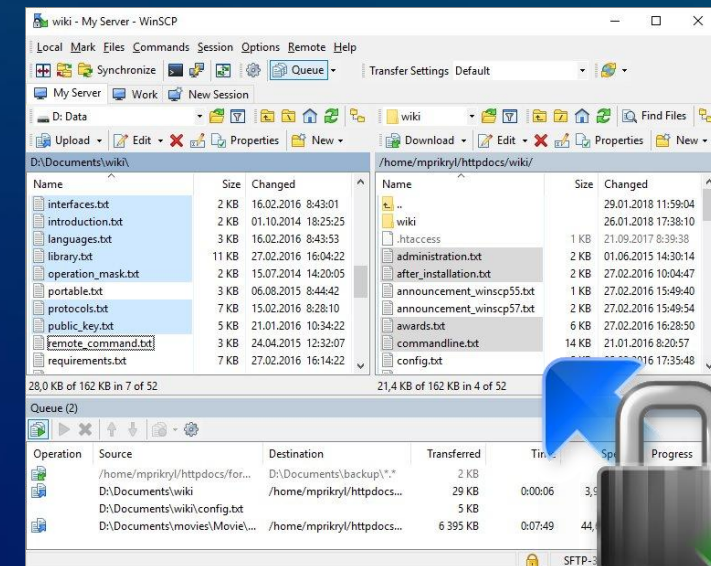
POUR VOUS CONNECTEZ EN FTP IL VOUS FAUT UN CLIENT FTP

Voici les 2 plus connus :

FILEZILLA



WINSCP



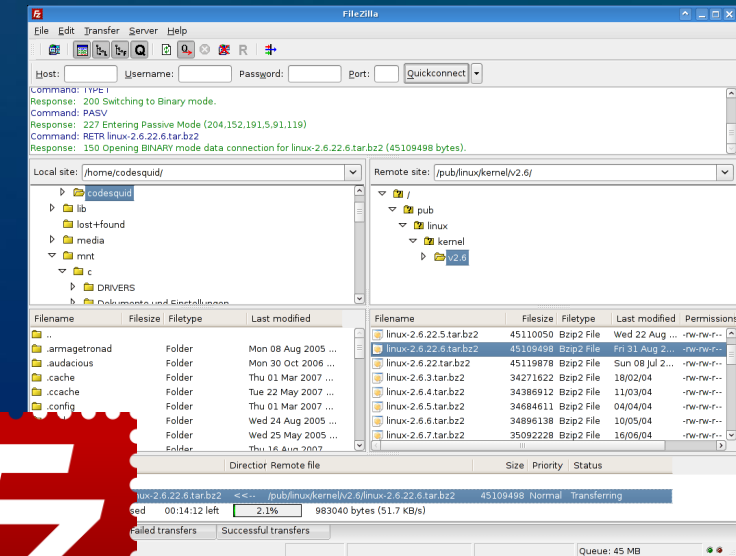
FTP / SSH (OPENSSSH)

FTP/SFTP

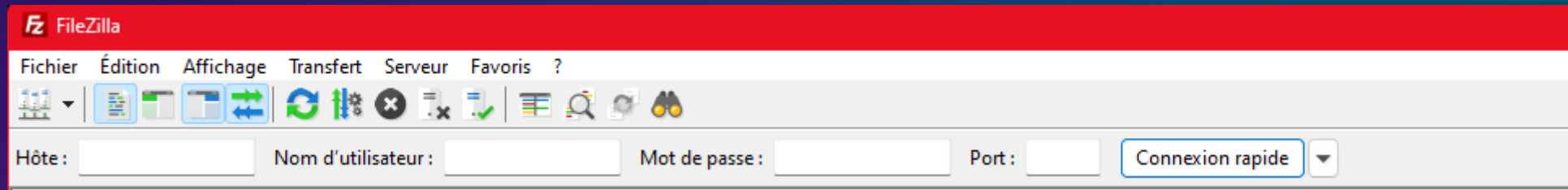
NOUS ALLONS INSTALLER FILEZILLA SUR NOTRE PC

FILEZILLA CLIENT

<https://filezilla-project.org>



COMPRENDRE UN CLIENT FTP



Hôte = l'ip de la machine auquel on veut se connecter
l'ip démarre soit par **sftp://** ou **ftp://**

Nom d'utilisateur = Utilisateur de la machine du serveur

Mots de passe = Mot de passe de l'utilisateur

Port = Port par défaut **21** pour le **FTP** & **22** pour le **SFTP**

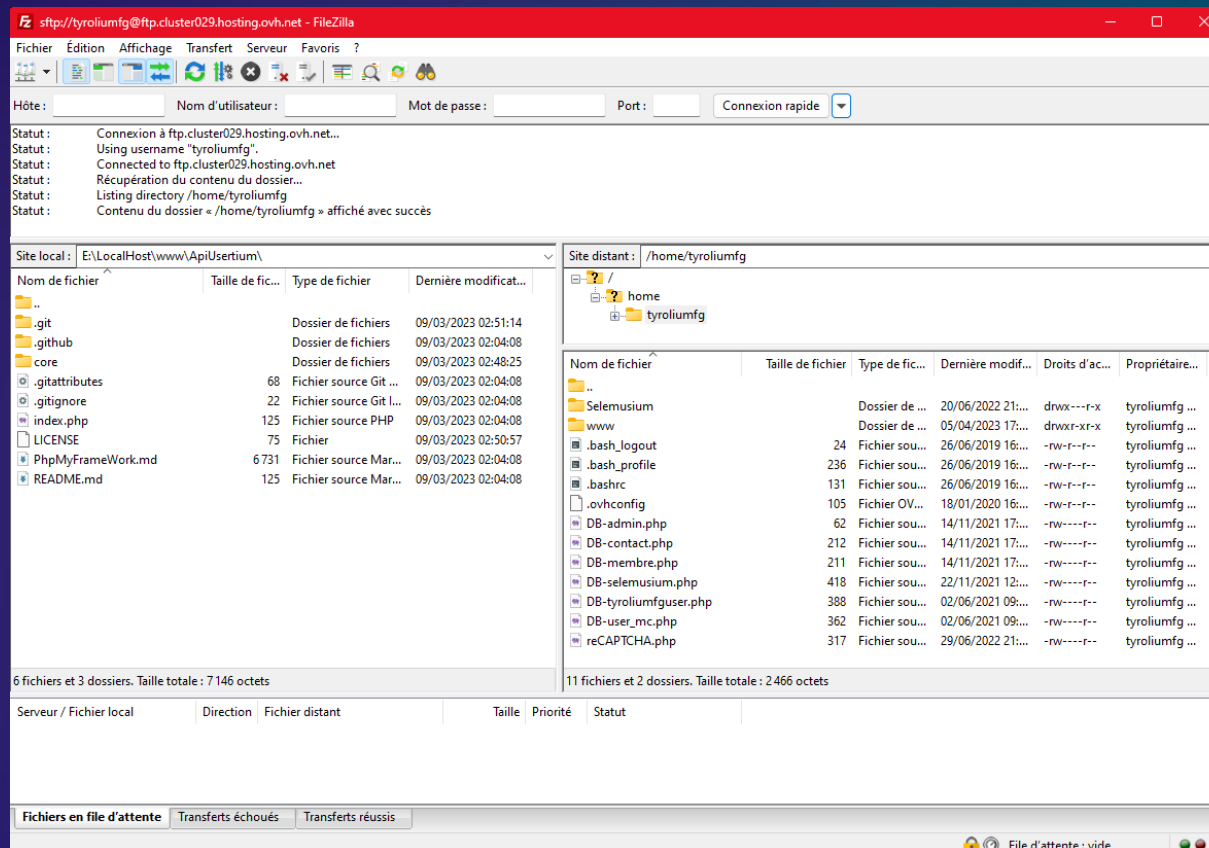
FTP / SSH (OPENSSSH)

CLIENT FTP



Votre PC

Log des données



Console
de Connexion



Le Serveur

Étape 1 :

Je vais vous demandez de vous connectez à mon serveur en SSH à fin de créer votre propre utilisateur

Le nom de l'utilisateur sera votre prénom

Information du Serveur

```
hostname : vps204.tyrolium.fr  
user : ft10  
password : p6rzKgCLE
```

Étape 2 :

Après ça je vous demanderais de vous connectez en SFTP avec FileZilla

Et de mettre un fichier (CV, un gif, une photo) dans votre dossier utilisateur

APACHE (WEB)



APACHE (WEB)

PRÉAMBULE

APACHE EST L'APPLICATION
QUI PERMET A UN SYSTÈME LINUX
D'HEBERGER UN SITE WEB

APACHE POUR FONCTIONNER A 100% A
BESOIN DE PHP ET D'UN MOTEUR SQL

COMMENT MARCHE LES URLS SUR UN SITE WEB (AVEC APACHE)

SUR LE SERVEUR



`/var/www/html/`

`/var/www/html/dossier/image.png`

POUR LES UTILISATEURS

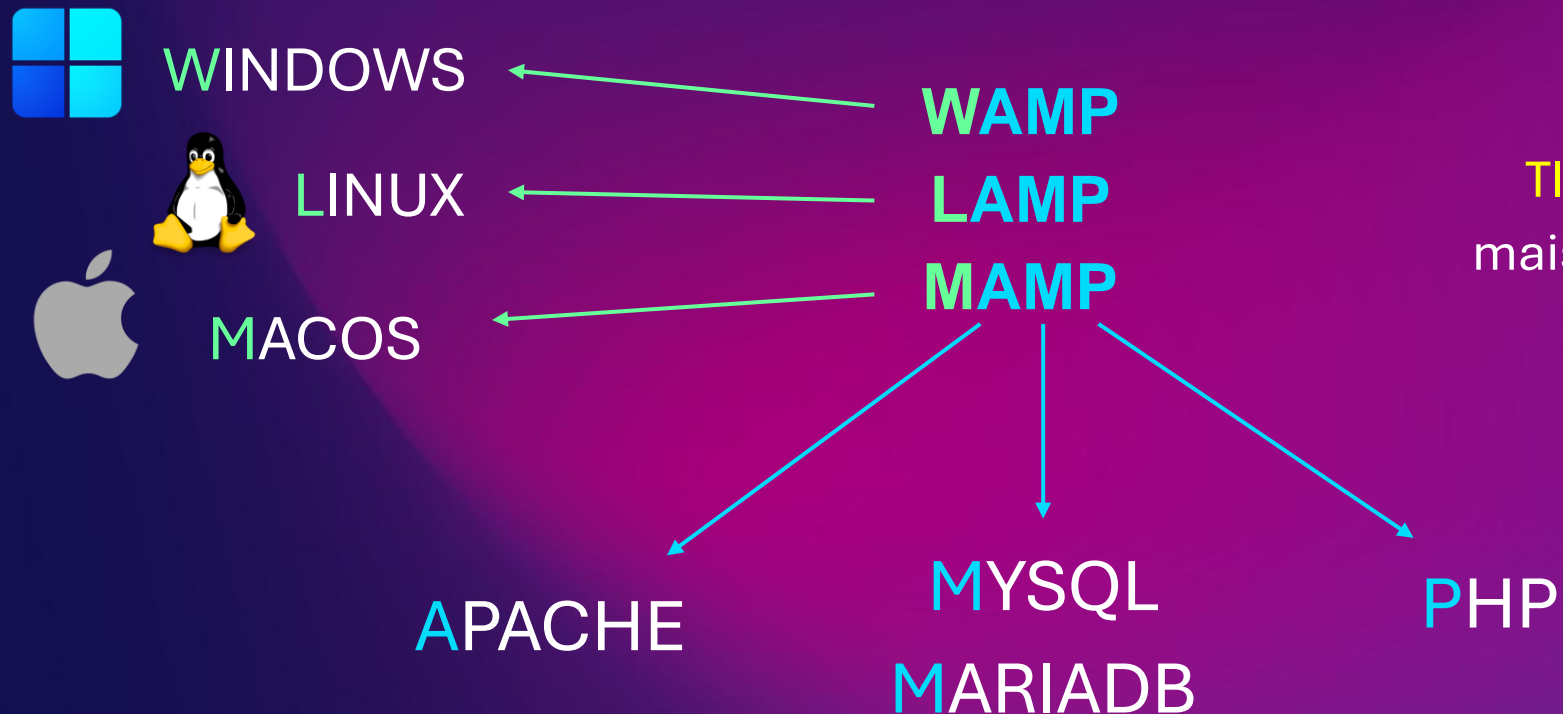


`https://google.fr/`

`https://google.fr/dossier/image.png`



POUR QUE APACHE MARCHE AU MIEUX
IL EXISTE DES PACKS POUR LES DEVELOPPEURS



TIPS: Il existe d'autre pack
mais ils sont plus anecdotique

INSTALLATION DE LAMP

NOUS ALLONS INSTALLER CETTE COMMANDE QUI PERMET
D'INSTALLER LAMP (APACHE, MYSQL, PHP)

```
$ SUDO APT INSTALL APACHE2 PHP LIBAPACHE2-MOD-PHP MYSQL-SERVER PHP-MYSQL
```

LA COMMANDE RISQUE DE PRENDRE
UN PETIT MOMENT

APACHE (WEB)

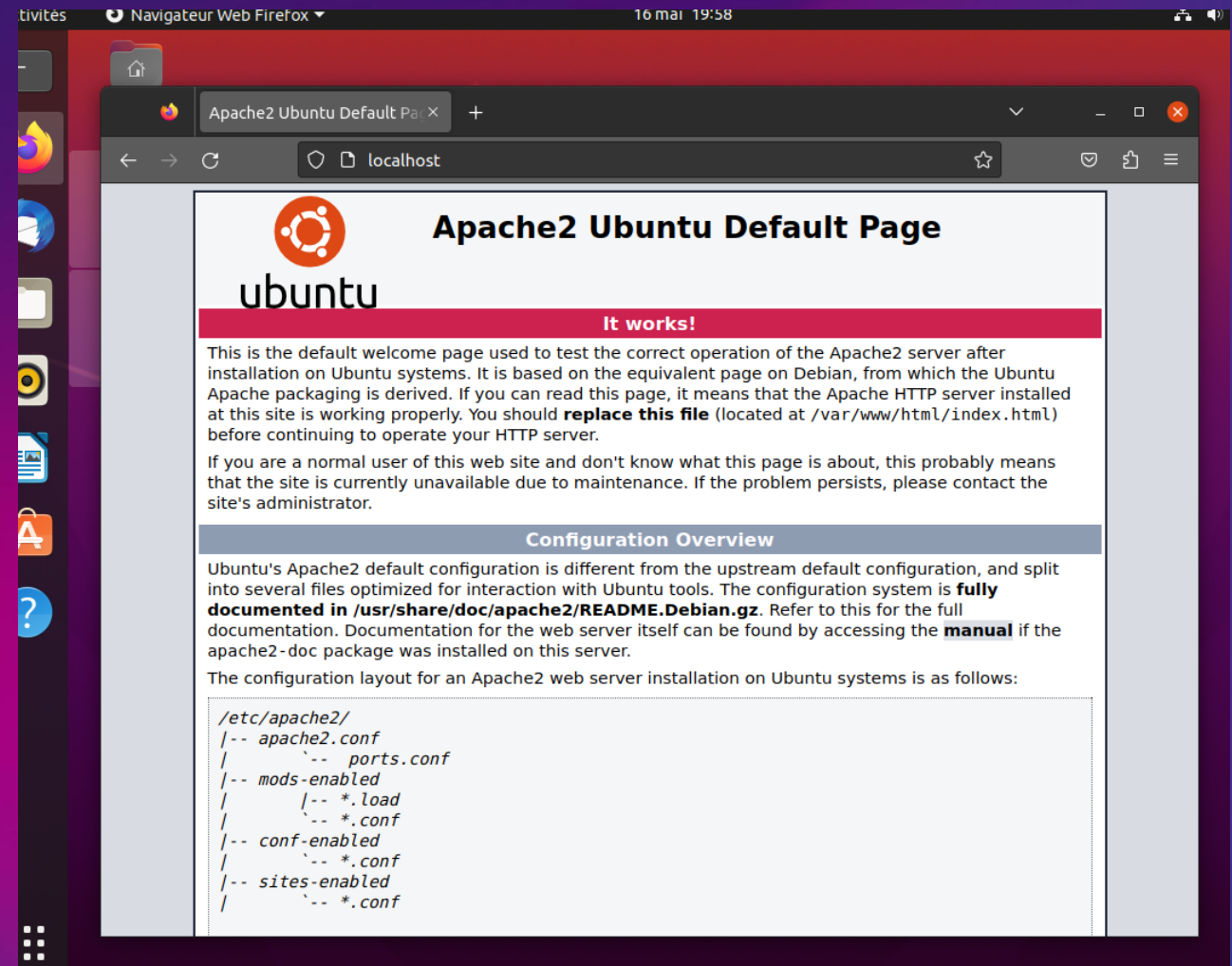
PACK APACHE

INSTALLATION DE LAMP

VERIFIER SI LAMP MARCHE BIEN

ALLEZ SUR FIREFOX

http://localhost



APACHE (WEB)

PACK APACHE

INSTALLATION DE LAMP

OÙ SE TROUVE VOTRE LOCALHOST

LOCALHOST

/VAR/WWW/HTML/

LOCALHOST/DOSSIER/

/VAR/WWW/HTML/DOSSIER/

LOCALHOST/TEST.TXT

/VAR/WWW/HTML/TEST.TXT

UTILISATION DE LAMP

1. Allez dans le dossier `/var/www/html/`
2. Changer **les permissions** du dossier `html/` et faite que le dossier appartienne à `vboxuser`
3. Supprimer le fichier `index.html` dans `html/`
4. Créer **votre propre dossier** dans `html/` qui s'appellera par votre **prénom**
5. Une fois fait, vous créez dans votre dossier un fichier `bonjour.txt`
6. Utiliser `nano` pour écrire "**Salut**" dans le fichier `bonjour.txt`
7. Lancer **Firefox** et rendez-vous sur votre fichier `bonjour.txt`
8. Puis en utilisant `>>` dans votre commande, vous allez écrire le contenu de la commande `date` dans votre fichier `bonjour.txt` en utilisant une seule ligne de commande uniquement

APACHE (WEB)

ACTIVITÉ GROUPE

Étape 1 :

Récupéré votre dossier projet
Sur le GitHub

Télécharger l'eu
(ou git clone)

Information du Serveur

```
hostname : vps204.tyrolium.fr  
user : ft10  
password : p6rzKgCLE
```

Étape 2 :

Après ça je vous demanderais de
vous connectez en SFTP avec
FileZilla

Et de mettre

`/var/www/html/[VotrePrénom]`

PS: Oublier pas d'installer ce qu'il
faut

EVAL



BONUS



NOM DE DOMAINE



NOM DE DOMAINE

.com

.net

.org

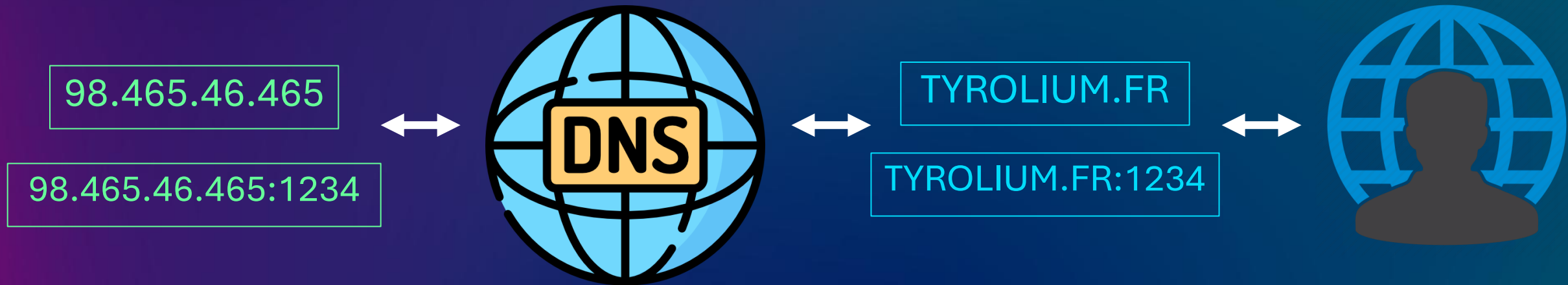
.eu

.us

.info

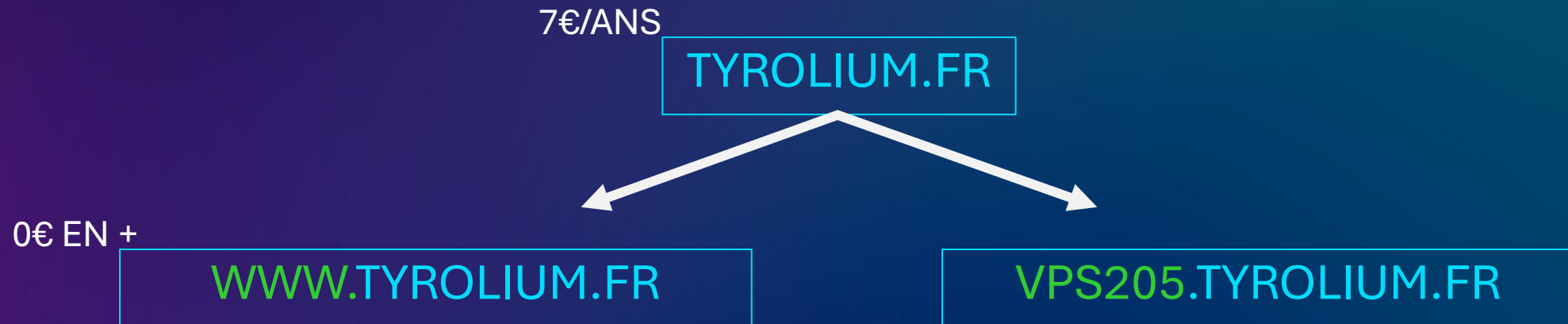
NOM DE DOMAINE

FONCTIONNEMENT DU NOM DE DOMAINE



NOM DE DOMAINE

IL EXISTE LES SOUS DOMAINE



UN NOM DOMAINE DOIT ÊTRE ACHETER TOUTE LES ANNER
LE SOUS-DOMAIN EST CONFIGURABLE UNE FOIS LE DOMAINE ACHETER

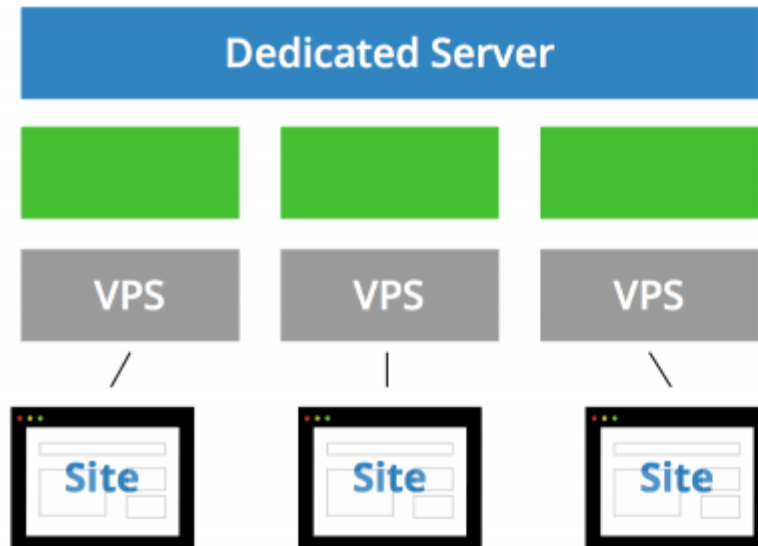
GLOBAL



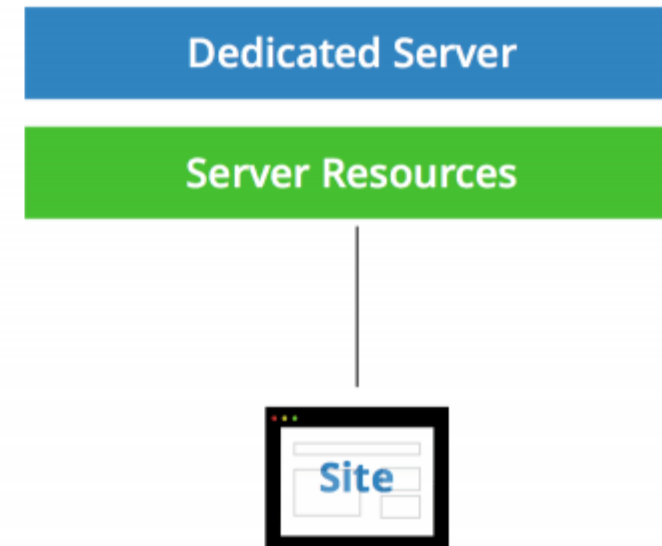
BONUS

TYPE DE SERVEUR

VPS Hosting



Dedicated Server Hosting



SETUP DE PHPMYADMIN

Commande d'installation de PhpMyAdmin

```
$ SUDO APT-GET INSTALL PHPMYADMIN
```

Connecter PhpMyAdmin à Apache

```
$ SUDO MV /ETC/PHPMYADMIN/APACHE.CONF /ETC/APACHE2/CONF-ENABLED/APACHE.CONF
```

Redémarré MySql et Apache

```
$ SUDO SERVICE MYSQL RESTART
```

```
$ SUDO SERVICE APACHE2 RESTART
```

POUR FAIRE FONCTIONNER UN PROGRAMME JAVA

```
$ JAVA -JAR SERVEUR.JAR
```

- Permet de lancer un programme java
- Le **-Jar** est égal à **Au type de fichier** dans l'exemple
- Le **Fichier** est égal au **Lien du fichier** dans l'exemple

BONUS

SCREEN

Créer un terminal virtuel

```
$ SCREEN [ARGUMENT] [NAME]
```

ARGUMENT :

-ls = Afficher tout les screens actifs

-S = Créer un screen

-x = Se connecter a un screen

FILE :

Nom du screen

Tips : Quitter un screen avec la
CTRL + A puis D

EXEMPLE :

Tips : Fermer un screen `$ exit`

```
$ SCREEN -LS
```

```
THERE IS A SCREEN ON:
```

```
16872.MC (05/25/23 09:49:28) (DETACHED)
```

```
1 SOCKET IN /RUN/SCREEN/S-MAXIME.
```

```
$ SCREEN -S MINECRAFT
```

```
$ SCREEN -X MINECRAFT
```

MANIPULER LES GROUPES ET LES UTILISATEURS

1. Connectez-vous en **root** (Sans le mot de passe root)
2. Créer un groupe du nom de "**PE9**"
3. Ajouter "**vboxuser**" dans le groupe "**PE9**"
4. Vérifier les groupes de "**vboxuser**"
5. Créer un utilisateur qui s'appel "**Kevin**"
6. Ajouter "**Kevin**" dans le groupe "**PE9**"
7. Vérifier les groupes de "**Kevin**"
8. Connectez-vous à "**Kevin**"
9. Créer un fichier "**PasCopain.txt**" dans le dossier utilisateur de "**Kevin**"
10. Afficher le contenu du dossier

MANIPULER LES GROUPES ET LES UTILISATEURS SUR UN DOSSIER

1. Connectez-vous en **root** (Sans le mot de passe root)
2. Placez-vous dans le dossier **/home/**
3. Faites un **affichage long** du contenu du dossier **/home/**
4. Changez le **groupe** et **l'utilisateur** du dossier "**Kevin**" pour le **groupe** "**PE9**" et **l'utilisateur** "**vboxuser**"
5. Vérifier le changement du groupe
6. Connectez-vous sur "**vboxuser**"
7. Créer un fichier **NewPasCopain.txt** dans le dossier utilisateur de "**Kevin**"
8. Afficher le contenu du dossier utilisateur de "**Kevin**"

FIN

