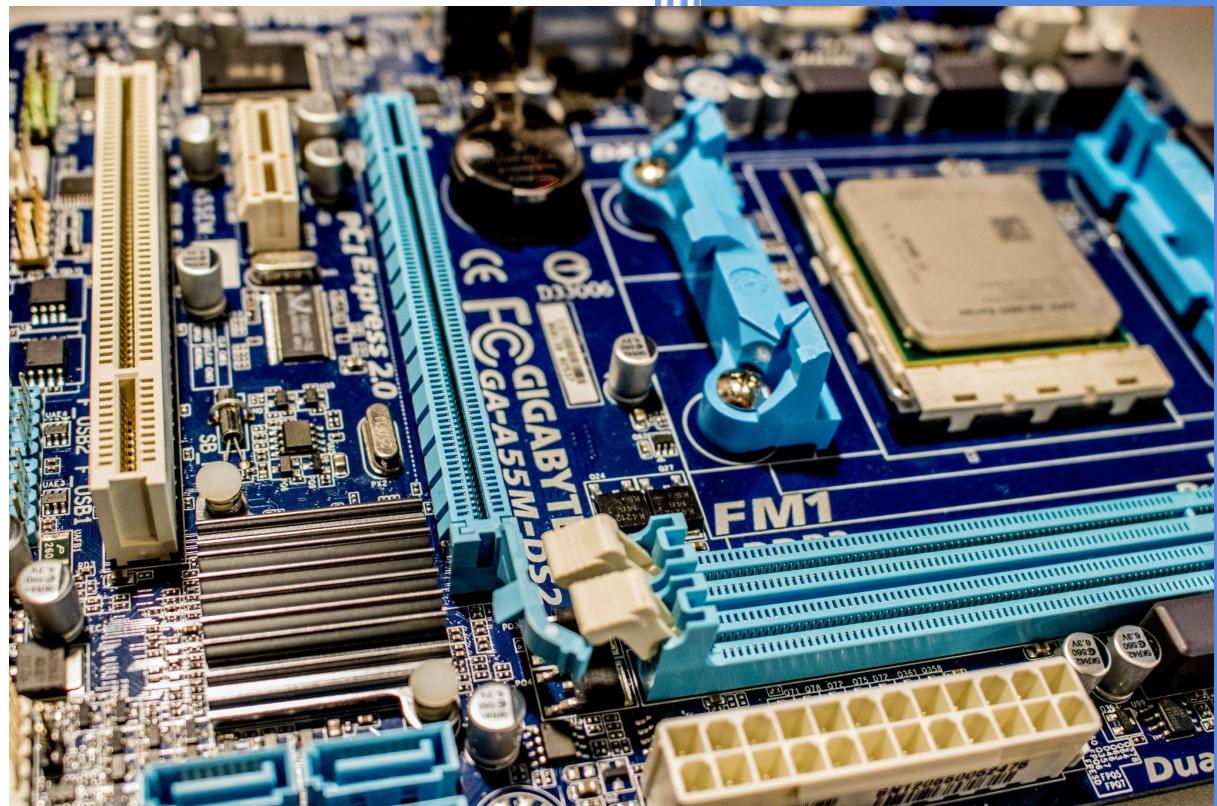


# Documentation Hardware



Projet Team START réalisé par : Jérôme Lacroix, Ibni-Yamine Chabane,  
Félix Hauger, Fériaile Bourega et Zahra AIT AOUDIA.

# SOMMAIRE

## Introduction

### 1. Les composants

- a. Carte mère
- b. La RAM
- c. Différentes mémoires
- d. Disque dur
- e. Carte réseau
- f. GPU
- g. CPU
- h. L'alimentation
- i. Compatibilité entre composants

### 2. Le système d'exploitation

- a. Qu'est-ce que le BIOS
- b. Qu'est-ce que l'UEFI
- c. Le POST
- d. Procédure de démarrage d'un ordinateur.
- e. Les différences entre le BIOS et UEFI

### 3. Installation d'un système d'exploitation

- a. Création d'une clé USB Bootable
- b. Installation de l'OS Debian 11

### 4. Debian

- a. Qui est le fondateur de Debian ?
- b. Qu'est-ce que Debian ?
- c. Qu'est-ce que l'open source ?

### 5. Introduction à Linux et aux lignes de commande

- a. Arborescence du système Linux
- b. Terminal, shell et commandes Linux

### 6. Élévation des privilèges des utilisateurs

- a. root et sudo
- b. La commande sudo visudo
- c. Le fichier sudoers
- d. Créer un utilisateur sudo

### 7. Installer un logiciel sur Debian avec le gestionnaire de paquets

- a. Préparatifs
- b. Les gestionnaires de paquets
- c. L'installation simple
- d. La méthode manuelle
- e. Alternatives à Google Chrome



## Introduction

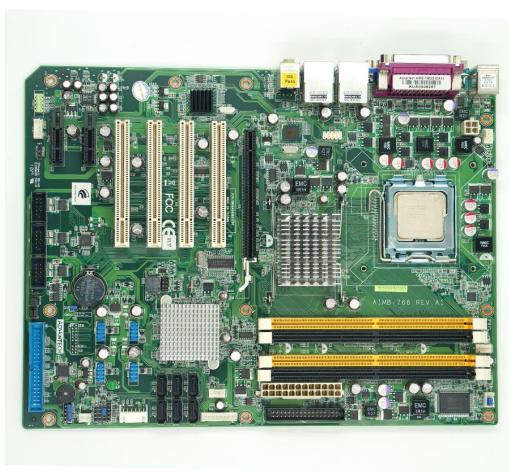
Un ordinateur est composé de pièces, appelées composants, qui permettent de le faire fonctionner. Celui-ci permet d'exécuter des programmes. Il est doté d'un ensemble de circuits électroniques. Pour communiquer avec l'ordinateur, le langage binaire est utilisé.

Cette documentation expose les différents composants que l'on peut trouver dans un ordinateur, et son fonctionnement.

## 1. Les composants

### a. Carte mère

[Quel est le rôle d'une carte mère dans un ordinateur ?](#)



La carte mère centralise la gestion de la RAM (ou mémoire à accès aléatoire), les lectures de disque dur et l'utilisation du processeur. Son rôle est également essentiel

## DOCUMENTATION HARDWARE

---

pour la détection et la compatibilité des cartes graphiques. La gestion du réseau et des ports USB se fait également par son intermédiaire. C'est sur celle-ci que sont branchés les composants.

Il existe plusieurs types de carte mère :



ATX



micro atx



mini-itx

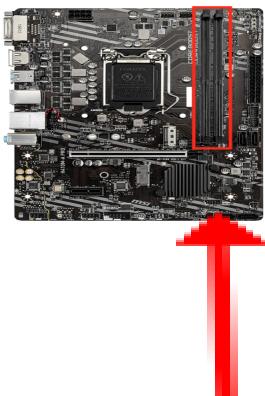
### b. La RAM

A quoi sert une barrette de ram?

Cette mémoire vive est tout simplement indispensable à votre ordinateur : c'est elle qui permet au processeur de stocker temporairement les données dont il a besoin pour lancer un programme. Le système d'exploitation (Windows, Mac OS...) lui-même est chargé sur la RAM.

Les différentes barrettes de ram

- DDR-SDRAM (Double Data Rate-Synchronous DRAM, mémoire vive dynamique synchronisée à double débit de données) ...
- SDRAM (Synchronous Dynamic RAM, mémoire vive dynamique synchrone) ...
- RDRAM (Rambus Direct RAM, mémoire vive dynamique Rambus)



### [Comment placer une barrette de ram](#)

Pour imbriquer la RAM correctement, **l'encoche de la barrette mémoire doit être alignée sur le détrompeur du slot**. Il suffit de regarder sur le module quel côté du connecteur est le plus court pour trouver le bon sens. L'encoche de la barrette mémoire est bien alignée par rapport au détrompeur.

Emplacement des barrettes

### [Si j'enlève les barrettes de ram, que se passe-t-il ?](#)

Sans barrette de RAM branchée le pc ne peut pas démarrer, et on peut entendre un bip signalant une erreur (voir manuel de la carte mère)

## c. Les différentes mémoires

Il existe différentes mémoires , une mémoire vive (RAM),et une mémoire morte (ROM).

### [À quoi sert la mémoire vive d'un ordinateur?](#)

Cette mémoire vive est tout simplement indispensable à votre ordinateur : c'est elle qui permet au processeur de stocker temporairement les données dont il a besoin pour lancer un programme. Le système d'exploitation (Windows, Mac OS...) lui-même est chargé sur la RAM.

### [Une mémoire morte \(ROM\) c'est quoi ?](#)

Également appelée mémoire morte ou mémoire en lecture seule, la ROM permet de stocker des données sur un disque dur. Elle a la particularité d'être non volatile. C'est-à-dire qu'elle n'efface pas le contenu si l'appareil en question (ordinateur, smartphone, tablette...) est mis hors tension

### d. Disque dur



#### Un ssd c'est quoi?

Un disque électronique SSD est la nouvelle génération d'appareils de stockage utilisés dans les ordinateurs. Ils remplacent les disques durs mécaniques traditionnels en utilisant des mémoires flash, nettement plus rapides.  
ces généralement utiliser pour le gaming(jeux video,et os plus rapide pour le démarrage)

#### Un hdd c'est quoi ?

Il permet de stocker des données importantes comme des documents, photos, etc.  
Ce genre de disque dur est généralement utilisé pour la bureautique.

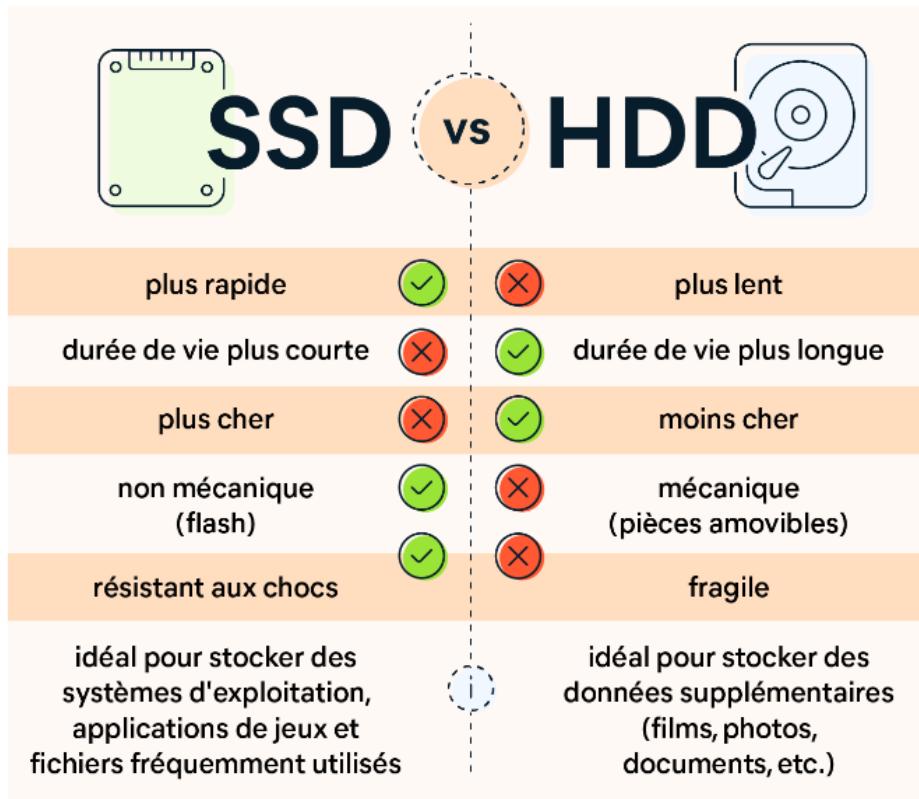


Câble de branchement du disque dur à l'alimentation



Câble de branchement du disque dure a la carte mère

Quelles sont les différences entre un ssd et un hdd ?



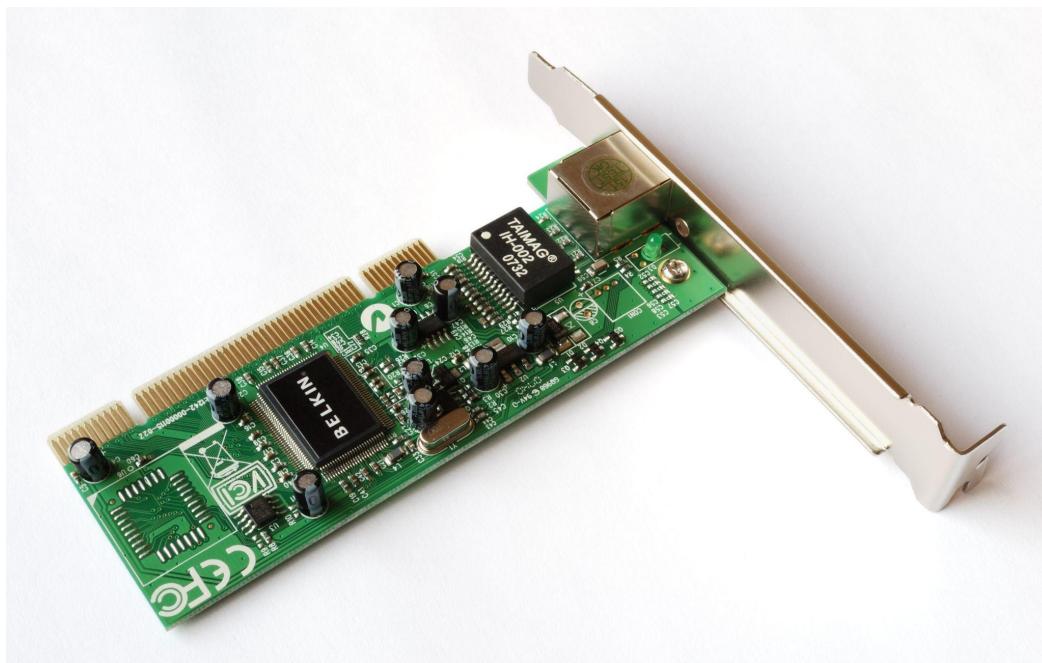
### e. Carte réseau

C'est quoi une carte réseau ?

La carte réseau d'un ordinateur permet de se connecter à internet et au réseau local : télévision, smartphone, tablette, console, imprimante réseaux, etc. Elle est installée sur la carte-mère et se connecte au réseau via un adaptateur USB-Ethernet ou un câble RJ45.

## DOCUMENTATION HARDWARE

---



### Où placer la carte réseau ?

La carte réseau va se positionner à l'arrière de votre ordinateur. Vous trouverez sur votre carte-mère, un slot PCI libre qui se présente sous la forme d'un emplacement rectangulaire muni ou non d'une protection anti-poussière. Retirez la protection anti-poussière si votre emplacement en possède une.

### f. CPU

#### C'est quoi un CPU ?

Le processeur est le cerveau de votre ordinateur. Il interprète et exécute les consignes reçues en langage binaire. Votre processeur est très utile pour le fonctionnement des programmes et l'affichage de votre écran.

#### Les différents CPU

Il existe deux types de processeurs, le processeur à usage général et le processeur à usage spécifique. Un processeur polyvalent peut être utilisé pour une grande variété d'applications, comme l'exécution d'un système d'exploitation et la réalisation de calculs dans des applications scientifiques et techniques.

#### Où brancher le processeur sur la carte mère ?

Elle se branche sur la carte mère. Il faut redresser le levier sur le côté du socket du processeur et mettre le CPU en veillant à ne pas mettre les doigts sous le cpu. C'est une pièce fragile.

### g. GPU

Il existe différents types de constructeur de gpu : **amd, nvidia et Intel**



#### C'est quoi un gpu ?

GPU est l'acronyme de Graphics Processing Unit (unité de traitement graphique en français). Communément appelée processeur graphique ou carte graphique, c'est une puce informatique qui permet d'optimiser l'affichage et le rendu des images 2D et 3D ainsi que des vidéos

#### Les différents gpu

Il existe deux versions de GPU : Le GPU intégré au PC, partage sa mémoire avec le processeur du CPU ; Le GPU dédié dispose de sa propre mémoire vidéo et est placé sur sa propre carte, pour un meilleur résultat.

#### Où brancher la carte graphique sur la carte mère ?

Elle se branche sur la carte mère sur une prise PCI. Ces prises ont les connecteurs les plus longs.

### Quelles sont les différences entre un cpu et gpu ?

Le CPU est le processeur central de l'ordinateur, qu'il se nomme Core 2 Duo ou Athlon 64. Son acronyme signifie unité de traitement principal. Le GPU désigne le processeur qui équipe la carte graphique, GPU étant le diminutif d'unité de traitement graphique

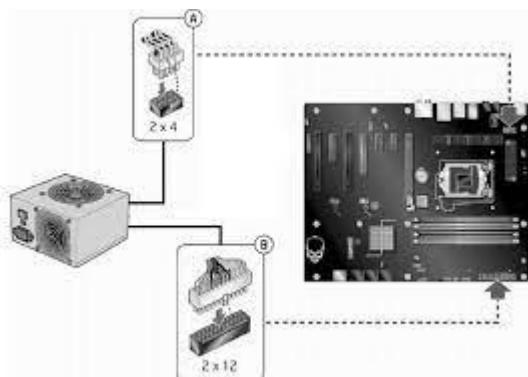
## h. L'alimentation

### Qu'est-ce que l'alimentation d'un PC ?

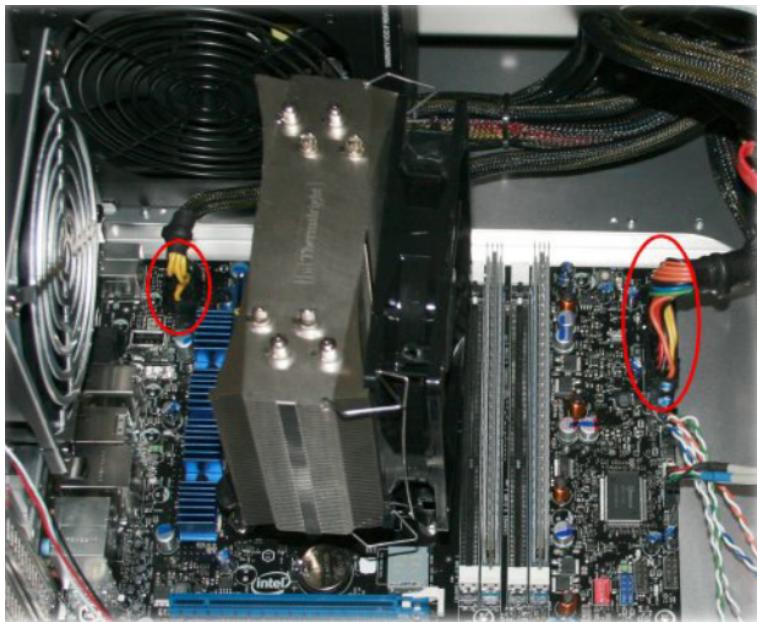
L'alimentation PC est un élément de grande importance pour votre PC. C'est ce qui permet de garantir le bon fonctionnement de l'ensemble des composants qui le constituent, notamment la carte graphique, le processeur ou encore le disque dur de votre ordinateur.

### Comment brancher l'alimentation du PC ?

L'installation de l'alimentation est très simple. Il suffit de dévisser son support de l'arrière du boîtier, fixer les 4 vis au bloc d'alimentation et remonter l'ensemble.



La carte mère a deux prises d'alimentation : une principale avec 24 broches et une secondaire avec 8 broches. Il faut bien entendu les brancher toutes les deux ! En raison de leur forme, il est impossible de procéder à un mauvais branchement.



### Pourquoi choisir une alimentation correcte pour son pc ?

Chaque composant du pc a besoin d'un certain montant de watts pour fonctionner correctement. Pour avoir une bonne alimentation on doit donc additionner le nombre de watts de tous les composants Si le nombre de watts n'est pas suffisant les composants seront mal alimentés en énergie et donc les composants ne fonctionnent pas en pleine capacité ou le pc ne s'allumera pas.

### **i. Compatibilité entre composants**

#### Quelle incompatibilité entre composants peut-on avoir ?

Les incompatibilités qu'on peut rencontrer sont:

- la carte mère n'est pas compatible avec le processeur  
alimentation pas assez puissante pour la configuration du pc
- les barrettes de ram pas compatible avec la carte mère

## 2. Le système d'exploitation

Quand un PC est assemblé correctement, il faut lui installer un système d'exploitation.

Le système d'exploitation (OS, pour "Operating System"), est le logiciel principal de l'ordinateur qui démarre après le BIOS. L'OS servira d'intermédiaire entre l'utilisateur et les programmes, et aussi du programme au matériel. Il gère les demandes de d'utilisation de ressources de l'ordinateur : les ressources de stockage, des mémoires (par exemple des accès à la mémoire vive, aux disques durs), ressources de calcul du processeur central, ressources de communication vers des périphériques (pour parfois demander des ressources de calcul au GPU par exemple ou tout autre carte d'extension) ou via le réseau – de la part des logiciels applicatifs. Le système d'exploitation gère les demandes ainsi que les ressources nécessaires évitant les interférences entre les logiciels.

Il existe plusieurs systèmes d'exploitation, très souvent livrés avec les ordinateurs (Windows, Mac OS, Unix, GNU/linux, etc) il existe aussi des OS pour les Smartphone (Android, iOS) le noyau de l'OS d'Android est basé sur le noyau de Linux de type Unix.

### a. Qu'est-ce que le BIOS ?

Le BIOS (Basic Input Output System) est un microprogramme avec de nombreuses fonctions, stockées dans la mémoire morte (ROM) de la carte mère d'un ordinateur, lui permettant d'effectuer des opérations élémentaires lors de sa mise sous tension et de charger le système d'exploitation.

Sur la carte mère, se trouve également une puce CMOS (Complementary Metal Oxide Semiconductor) qui stocke les paramètres personnalisés du BIOS (date/heure du système, ordre d'amorçage, configuration des disques, overclocking du CPU et la RAM, mot de passe d'accès à l'ordinateur, etc...) Cette fois, il faudra bien de l'électricité pour garder en mémoire ces informations. C'est une pile 3V (CR2032) qui permet d'alimenter la puce CMOS.

### b. Qu'est-ce que l'UEFI ?

L'UEFI "Unified Extensible Firmware" est un peu plus d'actualité puisqu'il remplace le BIOS depuis 2006. L'UEFI est comme le BIOS, un microprogramme qui vise à configurer la carte mère.

Il est stocké dans la mémoire flash de la carte mère et comporte de nombreux avantages comparés à son ancêtre, le BIOS.

L'UEFI utilise un schéma de partitionnement en GPT au lieu du MBR utilisé par le BIOS. C'est entre autres ceci qui permettra d'utiliser des disques système de plus de 9 Zettaoctets !

### c. Le POST

Le POST, abréviation de "Power On Self Test", est l'ensemble initial de tests de diagnostic effectués par l'ordinateur juste après sa mise sous tension, avec l'intention de vérifier les problèmes liés au matériel.

Comme son nom l'indique, c'est un processus qui se produit lorsque l'ordinateur est allumé.

Lorsque l'ordinateur s'allume pour la première fois, la première chose qui démarre réellement est le BIOS du système.

Le POST ne repose sur aucun système d'exploitation spécifique . En fait, il n'y a même pas besoin d'un système d'exploitation installé sur un disque dur pour que le POST s'exécute. C'est parce que le test est géré par le BIOS du système , pas de logiciel installé.

Les tests suivants sont effectués dans l'ordre sur les PC compatibles:

- Test du microprocesseur lui-même (test des registres en écriture et en lecture) et de la stabilité de l'alimentation.
- A la mise sous tension, le processeur est réinitialisé pour exécuter le programme BIOS situé sur une mémoire de type EPROM
- Test de la ROM contenant le programme de démarrage et le BIOS
- Initialisation du contrôleur de mémoire
- Test des 16 premiers K de mémoire RAM
- Initialisation du contrôleur d'interruption et des interruptions
- Test du temporisateur servant à compter le temps
- Test de l'adaptateur d'écran et affichage du curseur
- Test de l'ensemble de la mémoire RAM

- Test du clavier
- Vérification de la présence d'un lecteur de disquette ou d'un disque dur
- Test de l'imprimante et des ports de communication
- Comparaison des résultats obtenus lors des tests avec la configuration sauvegardée en mémoire CMOS
- et pour finir le «Bip» du haut-parleur

### d. La procédure de démarrage d'un ordinateur

Quand on appuie sur le bouton d'alimentation de l'ordinateur portable/de bureau. Le CPU démarre, mais a besoin de quelques instructions pour travailler. Étant donné que la mémoire principale est vide à ce stade, le processeur diffère le chargement des instructions de la puce du micrologiciel sur la carte mère et commence à exécuter les instructions.

Le code du micrologiciel effectue un test automatique de mise sous tension (POST), initialise le matériel restant, détecte les périphériques connectés (souris, clavier, clé USB, etc.) et vérifie si tous les périphériques connectés sont sains.

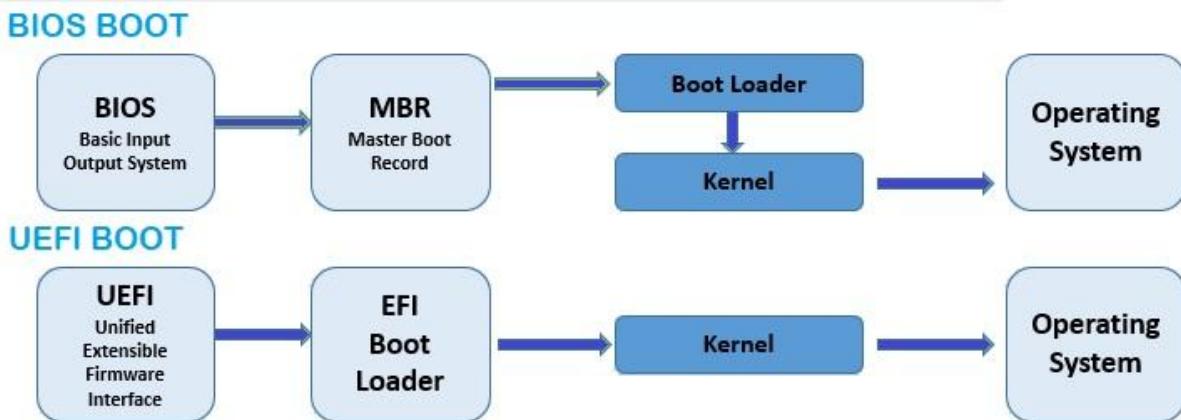
Le code du micrologiciel parcourt tous les périphériques de stockage et recherche un chargeur de démarrage (généralement situé dans le premier secteur d'un disque). Si le chargeur de démarrage est trouvé, le micrologiciel lui cède le contrôle de l'ordinateur.

Une fois que le chargeur de démarrage est chargé, son travail consiste à charger le reste du système d'exploitation. **GRUB** est l'un de ces chargeurs de démarrage capable de charger des systèmes d'exploitation de type Unix et également capable de charger en chaîne le système d'exploitation Windows.

Boot-loader n'est disponible que dans le premier secteur d'un disque, qui est de 512 octets. Compte tenu de la complexité des systèmes d'exploitation modernes, certains de ces chargeurs de démarrage ont tendance à effectuer un chargement en plusieurs étapes, où le chargeur de démarrage principal charge le chargeur de démarrage de deuxième étape dans un environnement qui n'est pas limité à 512 octets.

Après tout cela, et après l'initialisation de certains autres pilotes, l'interface utilisateur graphique (GUI) est chargée et l'écran de connexion s'affiche.

### e. Les différences entre le BIOS et UEFI



Le BIOS et l'UEFI sont identiques, mais complètement différents l'un de l'autre. Ils servent un objectif majeur : démarrer la machine et ils le font de différentes manières et avec différentes options.

avant tout voici quelques explications sur le MBR du BIOS et le GPT de UEFI :

Le MBR : Master Boot Record, est un secteur de démarrage situé au début d'un lecteur. Ce secteur contient un bootloader pour le système d'exploitation installé ainsi que des informations sur les partitions logiques. Le Bios charge le MBR. Ce sont les 512 premiers octets du disque. À partir des informations du MBR, il détermine l'emplacement du chargeur d'amorçage.

Si le disque de boot a plusieurs partitions, le BIOS lit le MBR du disque, puis le VBR de la partition (Volume Boot Record). À partir de ces informations, il peut déterminer l'emplacement du chargeur d'amorçage et le lancer.

Le GPT : GUID Partition Table". C'est une nouvelle norme qui permet de remplacer progressivement le vieillissant MBR. Elle est associée à UEFI qui lui, remplace le BIOS. On l'appelle « GUID Partition Table », car chaque partition sur votre disque dur a un identifiant unique (GUID = Globally Unique IDentifier).

le bootloader : Les données du système d'exploitation doivent être chargées directement dans la mémoire vive lors du démarrage de l'appareil. Cela est possible grâce à ce que l'on appelle un chargeur d'amorçage ou bootloader, qui fait office de programme de démarrage. Ainsi, le bootloader se lance généralement juste après le démarrage d'un appareil, quel que soit son support comme un disque dur, CD/DVD ou clé USB. Ce support va chercher dans le firmware de l'ordinateur (par ex. le BIOS) les informations qui concernent l'emplacement du chargeur d'amorçage. Ce processus est généralement désigné sous le nom d'« amorçage »

### Les différences entre BIOS et UEFI

UEFI s'exécute en mode 32 bits ou 64 bits, tandis que le BIOS s'exécute en mode 16 bits. L'UEFI est donc capable de fournir une interface graphique (navigation avec la souris) par opposition au BIOS qui permet la navigation uniquement à l'aide du clavier.

UEFI prend en charge des tailles de disque jusqu'à 9 zettaoctets, tandis que le BIOS ne prend en charge que 2,2 téraoctets. donc UEFI supporte plus de 2.2 TB HDD ou SSD. Le BIOS traditionnel prend en charge les petites partitions et les petits disques.

UEFI fonctionne en mode 32 bits ou 64 bits, et l'espace d'adressage est augmenté en fonction du BIOS, le processus de démarrage est beaucoup plus rapide. donc UEFI fournit un temps de démarrage plus rapide.

UEFI a un support de pilote discret, tandis que le BIOS a un support de disque stocké dans sa ROM, donc la mise à jour du micrologiciel du BIOS est un peu difficile.

UEFI offre une sécurité telle que Secure Boot, qui empêche l'ordinateur de démarrer à partir d'applications non autorisées/non signées. Cela aide à prévenir les rootkits, mais entrave également le double démarrage, car il traite les autres systèmes d'exploitation comme des applications non signées.

UEFI prend en charge le démarrage sécurisé, empêchant le PC d'être endommagé par des logiciels malveillants.

UEFI a un menu de réglage détaillé, plus utile que le BIOS traditionnel.

UEFI a le grand avantage de la table de partition GUID (GPT) qu'il utilise pour lancer les exécutables EFI, il n'a aucun problème pour détecter et lire les disques durs de grande taille.

I'UEFI présente de nombreux avantages par rapport au BIOS. Et c'est en raison de ces avantages que l'UEFI est considéré comme le successeur du BIOS. Cependant, l'UEFI n'est pas pris en charge par tous les ordinateurs ou périphériques. Pour utiliser le firmware UEFI, le matériel sur votre disque doit prendre en charge UEFI

### 3. Installation d'un système d'exploitation

Dans cette section, nous allons vous expliquer étape par étape l'installation du système d'exploitation Debian 11.

#### a. Création d'une clé USB Bootable

Tout d'abord il faut obtenir le Fichier ISO d'un système d'exploitation de votre choix. Pour créer une clé USB Bootable, ici nous utiliserons l'ISO de l'OS Debian.

Une clé USB Bootable est tout simplement une clé contenant un programme ou un système capable de s'exécuter au démarrage de l'ordinateur indépendamment du système d'exploitation installé. On peut aussi créer un CD bootable mais les supports CD sont moins utilisés de nos jours.

Un **fichier ISO** (souvent appelé image ISO), est un fichier d'archive qui contient une copie (ou image) identique des données trouvées sur un disque optique, comme un CD ou un DVD. Ils sont souvent utilisés pour sauvegarder des disques optiques ou pour distribuer de gros fichiers destinés à être gravés sur un disque optique.

Le nom ISO provient du nom du système de fichiers utilisé par les supports optiques, qui est habituellement **ISO 9660**. Vous pouvez considérer une image ISO comme une copie complète de tout ce qui est stocké sur un disque optique physique comme un CD, un DVD ou un disque Blu-ray, y compris le système de fichiers lui-même. Ils sont une copie secteur par secteur du disque, et aucune compression n'est utilisée.

Monter une image ISO vous permet de monter l'image ISO dans un lecteur de disque optique virtuel. Toutes vos applications traiteront l'image comme s'il s'agissait d'un disque physique.

## DOCUMENTATION HARDWARE

Il vous faut : une clé USB, Le fichier ISO de l'OS (Debian ici) et le logiciel [Rufus](#) sur un OS Windows.

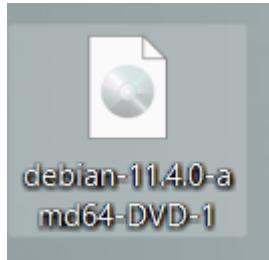
Rufus est un logiciel dédié au formatage et à la création de supports bootables USB. Il est multilingue et portable.

Il permettra le formatage de support USB de façon totalement personnalisée :

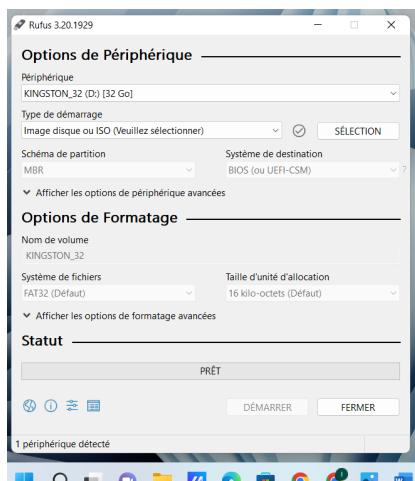
Type de partition (MBR pour ordinateur BIOS ou UEFI et MBR ou GPT pour UEFI)

Système de fichiers (FAT, FAT32, NTFS, UDF, exFAT)

Taille de cluster.

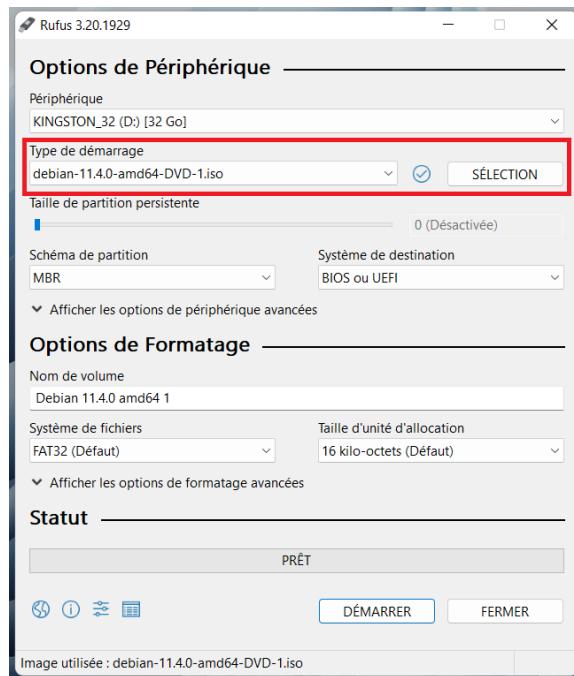


Tout d'abord, brancher la clé usb sur un PC avec un OS, et ouvrir le logiciel Rufus.

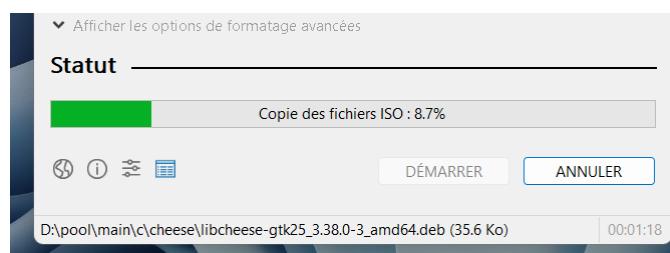


Ici on vérifie que notre clé usb est bien détectée par Rufus, puis on sélectionne Image disque ou ISO dans type de démarrage puis on clique sur "Sélection" pour choisir fichier ISO à charger sur Rufus.

## DOCUMENTATION HARDWARE



Dans la section Schéma de partition on laissera sur MBR, de cette façon on pourra boot sur notre Clé USB sur le BIOS ou sur l'UEFI.  
Puis Cliquez sur DÉMARRER, une fenêtre s'ouvrira pour nous demander quel mode d'Image, sélectionnez écrire en mode Fichier d'Image ISO (recommandé). et cliqué sur "OK" dans la prochaine fenêtre qui dit que tout sera reformatted dans la clé USB.



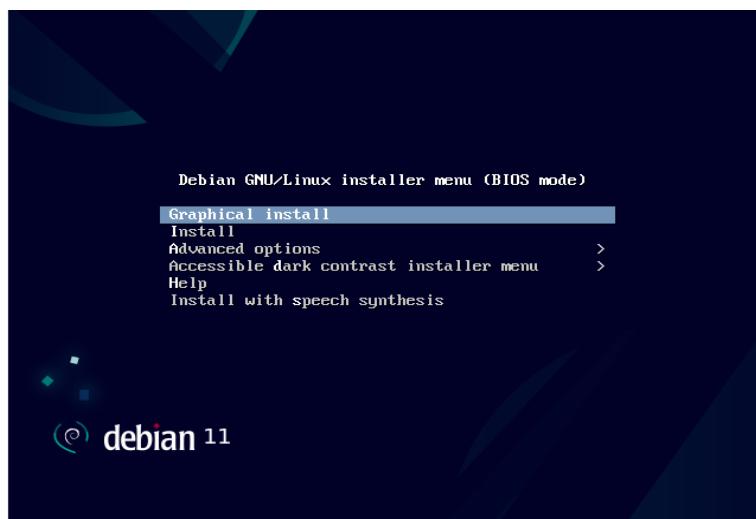
Quand la barre de progression arrive au maximum avec le statut "Prêt". La création de notre Clé USB Bootable est terminée, on peut fermer et éjecter la clé USB.

Brancher la clé USB sur l'Ordinateur dans lequel vous voulez installer votre OS et allumez-le, le PC va démarrer sur la Clé USB pour l'installation de l'OS Debian. Il est possible que le PC ne démarre pas sur la Clé USB Bootable. Il faudra alors configurer le BIOS ou l'UEFI (au choix) pour lui faire démarrer le PC sur la clé USB, il y a deux solutions :

- Solution n°1 : accéder au menu de démarrage (Boot Menu) en appuyant sur une touche au démarrage du PC, et sélectionner la clé usb détectée dans la liste pour démarrer dessus, la touche peut varier selon le constructeur. Cette action nécessite d'ouvrir le Boot Menu à chaque fois qu'on veut démarrer sur quelque chose (lecteur CD/DVD, Clé USB, Disque dur), le boot ne sera pas automatique.
- Solution n°2 : accéder au “Setup Menu” du BIOS de L’UEFI en appuyant sur une touche au démarrage du PC, la touche varie aussi selon le constructeur. Dans l’onglet “Boot” se trouve “Boot Option Priorities”, dessus on va pouvoir changer l’ordre de boot, mettre la clé USB en tête de liste, sauvegarder et quitter. Ainsi le PC va démarrer sur la clé USB en premier au démarrage. à chaque démarrage si une clé usb bootable est détectée, le boot sur la clé sera automatique.

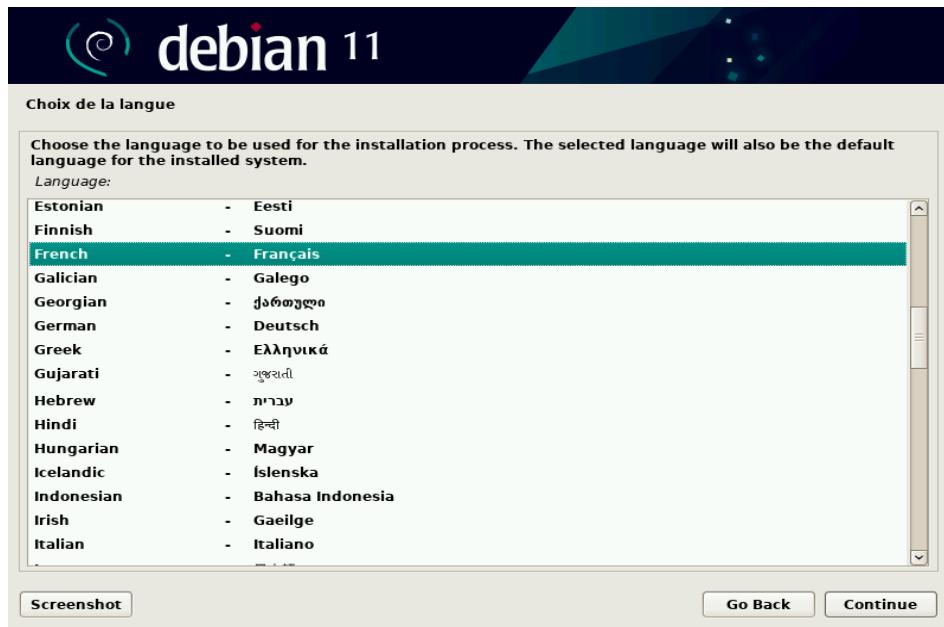
Désormais nous pouvons passer à l’installation de l’OS Debian 11 sur le PC.

### b. Installation de l’OS Debian 11

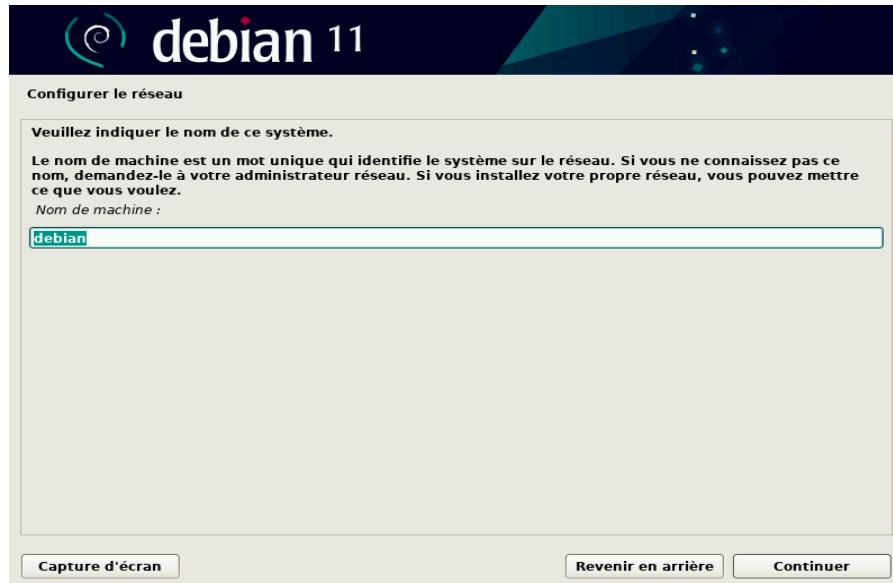


Voici l’écran d’installation de Debian 11, nous pouvons faire l’installation directement avec “Install”, cependant ce sera sans interface graphique et nous ne pourrons pas utiliser la souris. Nous allons donc choisir pour cette documentation, de faire **l’installation graphique** car nous voulons utiliser une souris.

## DOCUMENTATION HARDWARE

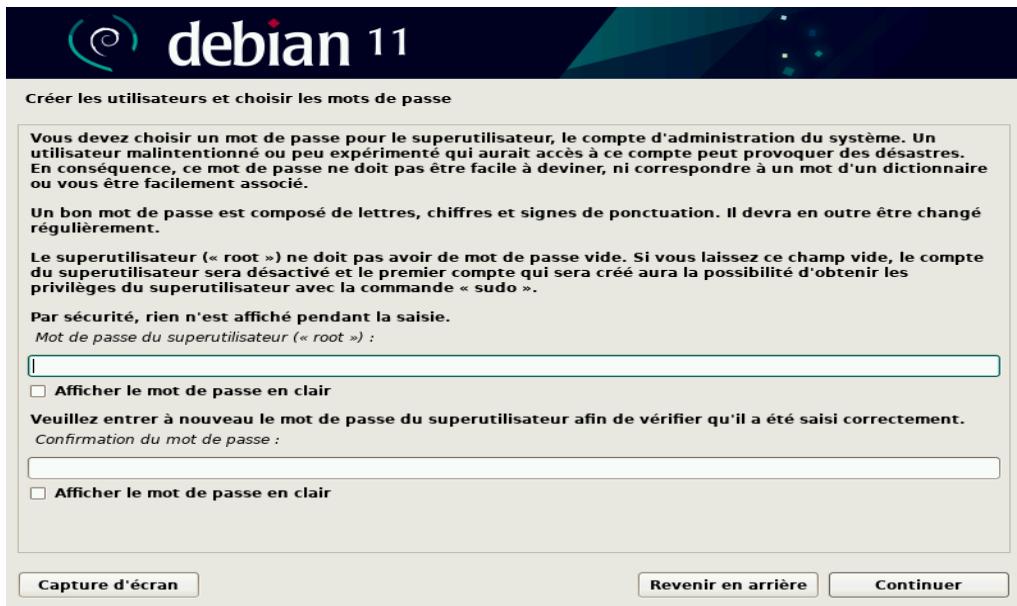


On choisit selon nos préférences le langage pour notre OS, la situation géographique, et la langue pour notre clavier. Nous allons tout mettre en français, notre clavier sera en "AZERTY".



Dans la configuration du réseau, on choisit le nom de votre machine, le nom choisi désignera notre machine quand il sera visible dans le réseau par d'autres machines. Nous avons laissé debian par choix. L'étape d'après est au sujet du domaine de notre machine, nous avons passé cette étape.

## DOCUMENTATION HARDWARE



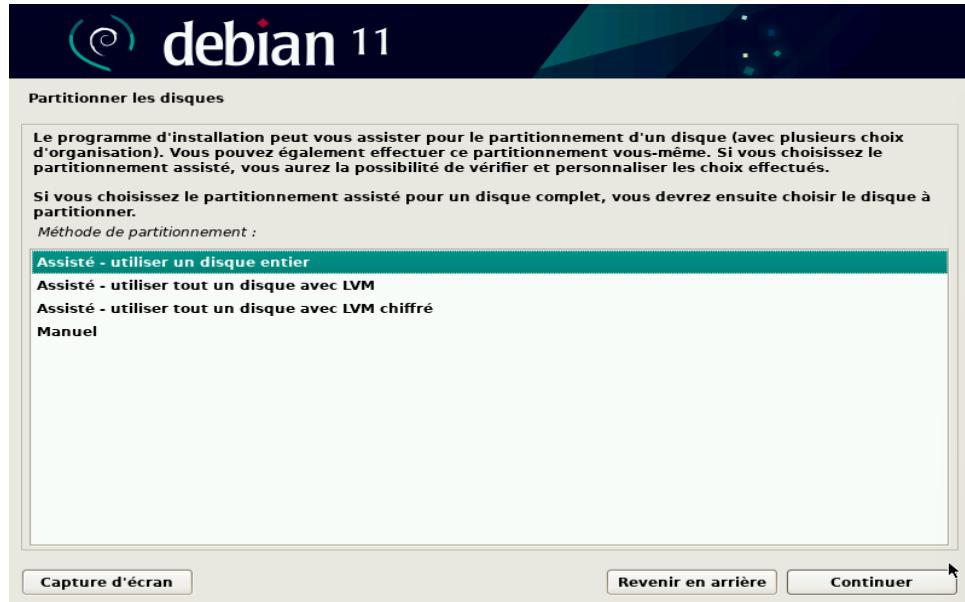
Dans cette étape c'est la création des utilisateurs, ici il faut choisir le Mot de passe de l'administrateur qui est le super utilisateur “root”, étape très importante car le super utilisateur à tous les droits sur le PC. ce mot de passe est très important et ne doit pas tomber entre de mauvaises mains, l'accès d'une personne extérieure et mal intentionnée pourrait provoquer des désastres.



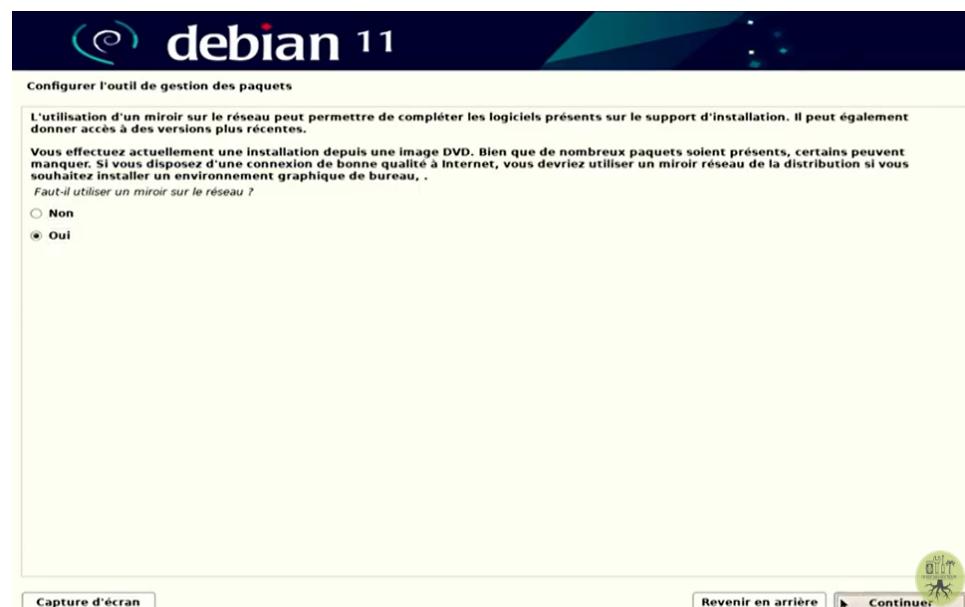
Ici on nous propose de créer un compte d'utilisateur différent du super utilisateur “root”, nous avons choisi “LaPlateforme” Comme nom pour ce compte, ce compte servira de compte courant du système.

l'étape d'après nous permet de choisir un identifiant “login”, nous avons choisi “toto” et un mot de passe a choisir pour le compte “LaPlateforme”.

## DOCUMENTATION HARDWARE

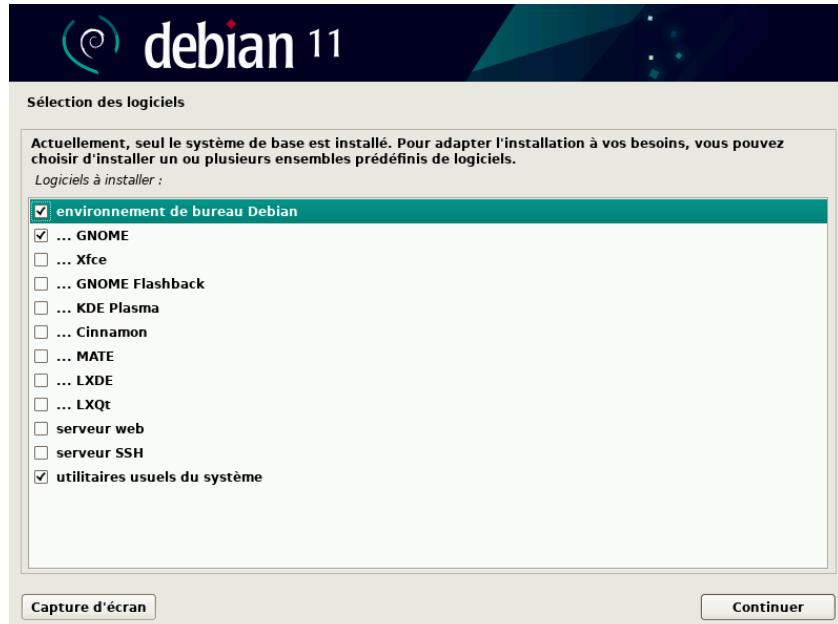


Étape “Partitionner les disques”, il faut choisir la méthode de partitionnement, la partie du disque dur ou résidera notre OS, ici on choisit d'utiliser un disque en entier. Dans l'étape suivante c'est la sélection du disque dans lequel on veut créer une partition **ATTENTION** le disque sera entièrement effacé, nous allons tous installer dans une seule partition dans l'étape schéma de partitionnement. Une fois validé, le formatage commence, et une fois le disque formaté, l'installation du système de base commence dans le disque dur.

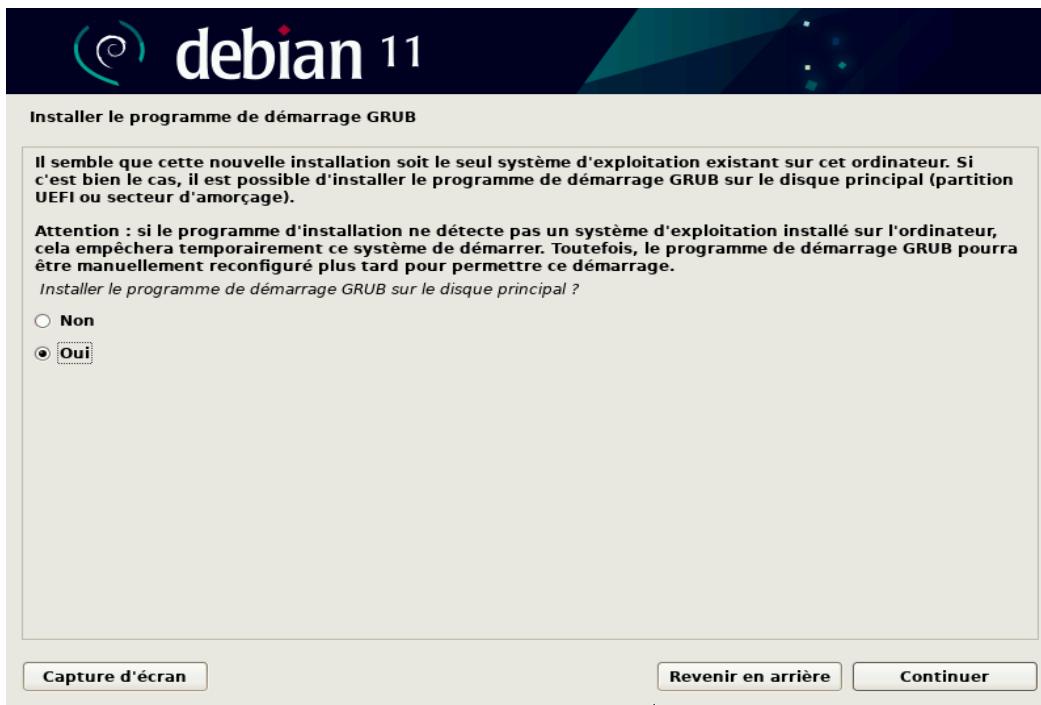


## DOCUMENTATION HARDWARE

Dans l'étape "Configurer l'outil de gestion des paquets" pour l'utilisation d'un miroir réseau, choisir "oui" est pratique car cela servira à télécharger des logiciels sur internet.

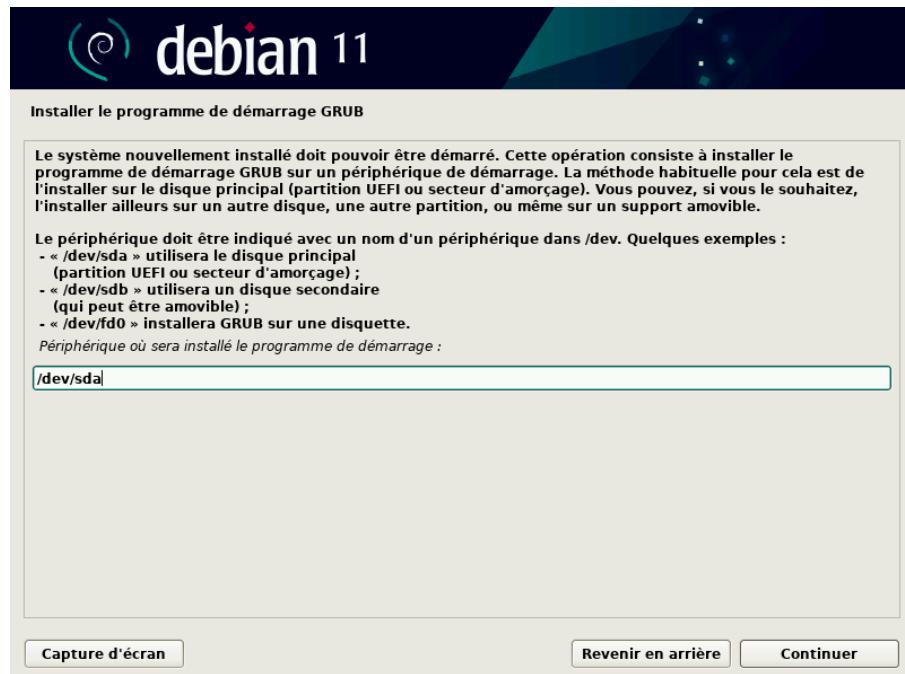


Ici nous gardons "GNOME" qui est l'interface graphique de GNU/Linux.

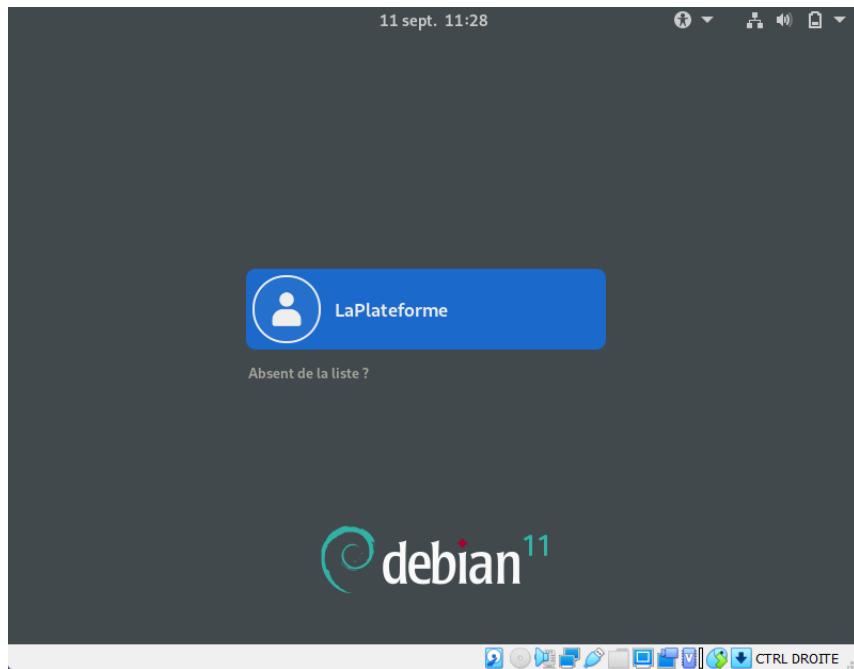


L'installation du programme de démarrage GRUB nous permettra de sélectionner notre OS au démarrage du PC.

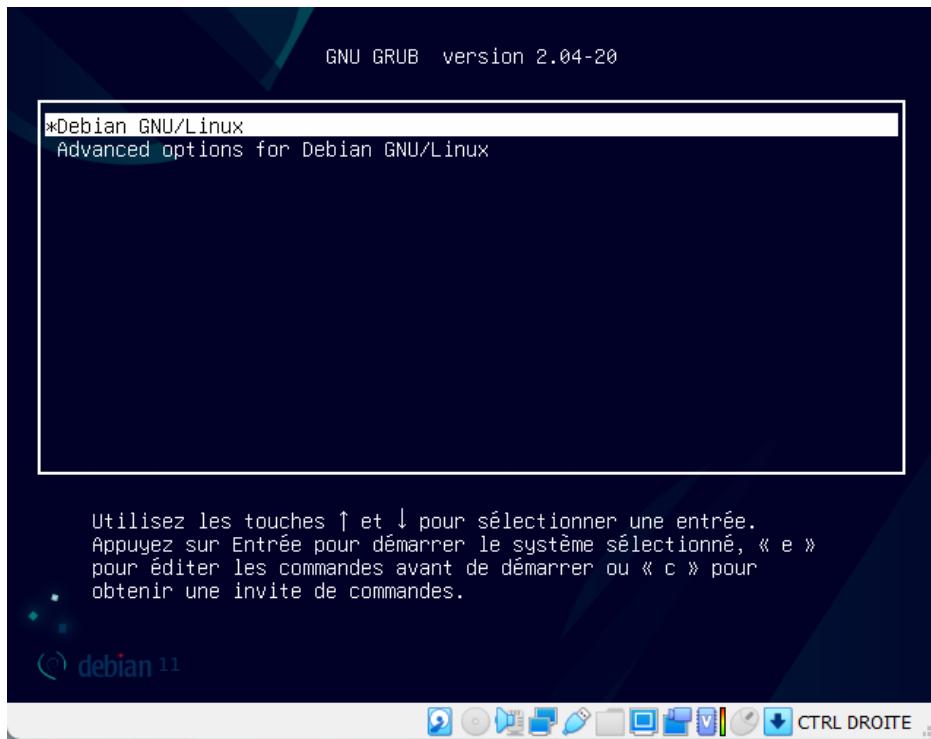
## DOCUMENTATION HARDWARE



On installe GRUB dans le disque principal avec le secteur d'amorçage.  
Après avoir validé l'installation se poursuivra jusqu'à la fin.



Voilà ! Nous avons installé Debian 11. Après avoir testé les mots de passe, on redémarre le PC pour voir si on démarre bien avec GRUB.



On peut voir que GRUB est bien installé, on peut soit charger notre OS ou accéder aux paramètres avancés de Debian GNU/Linux. Félicitations, votre OS est bien installé.

## 4. Debian

### a. Qui est le fondateur de Debian ?

**Ian Murdock**, né le 28 avril 1973 à Constance en Allemagne de l'Ouest, et mort le 28 décembre 2015 (à 42 ans) à San Francisco (États-Unis), est le fondateur du projet Debian, et de la distribution commerciale Progeny Debian. Il a été PDG du consortium *Free Standards Group*, et président du groupe de travail *Linux Standard Base*.

Il écrit le *Debian Manifesto* en 1993, alors qu'il est encore étudiant à l'université Purdue, où il obtient un *bachelor* en informatique en 1996.

Il se marie à Debra Lynn, dont la contraction du prénom avec le sien donne son nom à la distribution Debian.

Il fait déployer la première version de Debian le 15 septembre 1993 et la première version stable le 17 juin 1996.

### b. Qu'est ce que Debian ?

Debian est une organisation composée uniquement de bénévoles, dont le but est de développer le logiciel libre et de promouvoir les idéaux de la communauté du logiciel libre. Le projet Debian a démarré en 1993, quand Ian Murdock invita tous les développeurs de logiciels à participer à la création d'une distribution logicielle, complète et cohérente, basée sur le nouveau noyau Linux. Ce petit groupe d'enthousiastes, d'abord subventionné par la [Free Software Foundation](#), et influencé par la philosophie [GNU](#), a grandi pour devenir une organisation composée par environ 1000 *développeurs Debian*.

Les développeurs Debian s'impliquent dans de multiples activités, par exemple, l'administration des sites [web](#) et [FTP](#), la conception graphique, l'analyse juridique des licences logicielles, l'écriture de la documentation et, bien sûr, la maintenance des paquets logiciels.

Pour communiquer notre philosophie et attirer des développeurs qui adhèrent à nos principes, le projet Debian a publié un certain nombre de documents qui mettent en évidence nos valeurs et expliquent ce que signifie être un développeur Debian :

- Le [contrat social Debian](#) est la déclaration des engagements de Debian vis-à-vis de la communauté du logiciel libre. Quiconque est d'accord pour se conformer à ce contrat social peut devenir un [développeur Debian](#). Tout développeur Debian peut introduire de nouveaux logiciels dans Debian, à condition que ces paquets se conforment à nos critères de liberté et répondent à nos critères de qualité ;
- Les [directives Debian pour le logiciel libre](#) (*Debian Free Software Guidelines*, ou *DFSG*) sont une déclaration claire et concise des critères Debian en matière de logiciel libre. Ce document a une grande influence sur le mouvement pour le logiciel libre ; il est à la base de la définition de l'[Open Source](#) ;
- La [charte Debian](#) est une spécification détaillée des standards de qualité du projet Debian.

Les développeurs Debian participent aussi à d'autres projets : certains sont spécifiques à Debian, d'autres concernent tout ou partie de la communauté Linux. Voici quelques exemples :

- Le [standard pour l'organisation des systèmes de fichiers](#) (FHS) est un effort pour standardiser l'organisation du système de fichiers Linux. Le FHS permettra aux développeurs de logiciels de se concentrer sur la conception de programmes, sans avoir à se préoccuper de la façon dont le paquet sera installé dans les différentes distributions GNU/Linux ;
- [Debian Jr.](#) est un projet interne dont le but est de s'assurer que Debian a quelque chose à offrir à nos utilisateurs les plus jeunes.

### c. Qu'est-ce que l'open source ?

Le terme « open source » découle de l'expression « Open Source Software », abrégée en OSS. Ce type de logiciel propose un code source librement accessible. Les développeurs et utilisateurs peuvent le modifier et le partager. Comparés aux logiciels propriétaires, les produits open source sont plus flexibles et moins chers à l'achat. Un logiciel open source n'appartient ni à une seule personne, ni à une entreprise. Une grande communauté de développeurs en assure le support et **améliore continuellement le produit grâce aux contributions collectives**.

Dans la plupart des cas, les organisations et entreprises optent pour les projets open source afin de limiter les coûts de développement.

Ce modèle de production appliqué au développement de logiciels n'a rien de nouveau : en effet, l'open source est aussi vieux qu'Internet. Dans les années 1950 et 1960, les premiers protocoles de télécommunication en réseau furent développés dans un environnement ouvert. [L'ARPANET](#), pour **Advanced Research Projects Agency Network**, qui définit les bases de l'Internet moderne, reposait, comme les logiciels open source d'aujourd'hui, sur la collaboration et la révision par les pairs. Le mouvement des hackers dans les années 1970, puis celui du logiciel libre dans la décennie suivante, renforçèrent d'ailleurs l'idée de l'échange libre des connaissances en matière de développement de logiciels. Aujourd'hui, l'open source ne désigne pas seulement un « logiciel libre », mais aussi une **approche collaborative** dans le monde de l'informatique.

## 5. Introduction à Linux et aux lignes de commande

**Linux®** est un système d'exploitation Open Source. Un système d'exploitation est un logiciel qui gère directement les composants physiques du système ainsi que ses ressources, telles que le processeur, la mémoire et le stockage. Il représente l'interface entre les applications et le matériel.

Le système **Linux** possède notamment les avantages suivants : Le support des standards de l'internet, c'est-à-dire des protocoles TCP/IP, la famille de protocoles utilisée sur Internet. **Linux** est donc un moyen gratuit de créer un réseau local, de se connecter à Internet et de mettre en place un serveur.

Pour manipuler **Linux** on peut passer par la représentation graphique, comme on peut utiliser des lignes de commande dans le terminal.

- [Définition d'un répertoire](#)

Un répertoire (appelé également dossier ou folder en anglais) est un objet informatique pouvant contenir des fichiers. Imaginez une grande commode qui contient des tiroirs dans lesquels pourraient se trouver des fichiers ainsi que d'autres tiroirs.

### a. Arborescence du système Linux :

#### [Introduction :](#)

Notre système Linux possède des dossiers et des fichiers.

Chaque dossier a son utilité, et tout n'est pas stocké n'importe comment.

Dans cet article, je vais expliquer les notions de systèmes de fichiers, de racine, et de chemin.

#### [Le système de fichiers, racine, chemin](#)

Notre système de fichiers sous Linux est organisé. Chaque dossier à la racine de l'arborescence a une utilité particulière

On appelle cela une «arborescence» car schématiquement, nous partons de la racine du système de fichiers, nous parcourons cette arborescence en allant de dossiers en dossiers. (Un peu comme un arbre)

Il n'y a pas d'arbre sans racine ! Donc, tout chemin de fichiers dans le système Linux part de la racine. Cette racine appelée **root** est notée / (slash).

Cette racine est "à peu près" comparable au C:\ de Windows, histoire de faire un grossier parallèle.

A noter, quand on part de la racine puis qu'on parcourt les dossiers, l'ensemble de la localisation est appelé chemin.

Il existe 2 représentations d'un chemin :

- Chemin absolu, qui part de la racine : /home/adrien/Documents/Comptes
- Chemin relatif, qui part de l'endroit où on se situe. Exemple, si on se situe dans /home/adrien, un chemin relatif est Documents/Comptes.

### Définition de la session :

Une session est l'ensemble des applications, des paramètres et des ressources disponibles sur le bureau de l'utilisateur. La gestion des sessions représente l'ensemble des conventions et des protocoles permettant au Gestionnaire de sessions de sauvegarder et de restaurer la session d'un utilisateur.

Les utilisateurs ont des droits de super-utilisateurs ou des droits limités, et pour des utilisateurs à droit limité il leur faut le mot de passe de super-utilisateurs pour exécuter des tâches plus approfondies.

## b. Terminal, shell et commandes Linux

- Définition d'un terminal

Un **terminal** est un programme qui permet à l'utilisateur d'interagir avec le shell. Plusieurs émulateurs de terminaux existent sur Linux, comme gnome-terminal, konsole, xterm, rxvt, kvt, nxterm et eterm. Tous les shell peuvent exécuter des commandes situées dans un fichier.

- Définition de shell

Le **shell** (ou interface système en français) est un programme qui reçoit des commandes informatiques données par un utilisateur à partir de son clavier pour les envoyer au système d'exploitation qui se chargera de les exécuter.

***Il existe plusieurs commandes sur Debian comme :***

1. La commande **cd** est une commande pour se déplacer d'un répertoire à un autre. Elle nécessite soit le chemin d'accès complet, soit le nom du répertoire, selon le répertoire de travail dans lequel vous vous trouvez. Disons que vous êtes dans `/home/utilisateur/Documents` et que vous voulez aller dans Photos, un sous-répertoire de Documents. Pour ce faire, il vous suffit de taper la commande suivante : `cd Photos`.
2. La commande **ls** est utilisée pour visualiser le contenu d'un répertoire. Par défaut, cette commande affichera le contenu de votre répertoire de travail actuel.
3. La commande **pwd** pour trouver le chemin du répertoire de travail (dossier) dans lequel vous êtes actuellement. La commande retournera un chemin absolu (complet), qui est en fait un chemin de tous les répertoires qui commence par une barre oblique (`/`). Un exemple de chemin absolu est `/home/utilisateur`.
4. La commande **cat** (abréviation de concatenate) est l'une des commandes Linux les plus fréquemment utilisées. Elle est utilisée pour lister le contenu d'un fichier sur le résultat standard . Pour exécuter cette commande, tapez **cat** suivi du nom du fichier et de son extension. Par exemple : `cat fichier.txt`.
5. La commande **mv** L'utilisation principale de la commande **mv** est de déplacer des fichiers, bien qu'elle puisse également être utilisée pour renommer des fichiers. Les arguments de mv sont similaires à ceux de la commande cp.

## DOCUMENTATION HARDWARE

---

Vous devez taper mv, le nom du fichier et le répertoire de destination. Par exemple : `mv fichier.txt /home/utilisateur/Documents`. Pour renommer les fichiers, la commande Linux est `mv ancien_nom.ext nouveau_nom.ext`.

6. La commande `mkdir` pour créer un nouveau répertoire – si vous tapez `mkdir Music`, cela créera un répertoire appelé `Music`. Il existe également des commandes `mkdir` supplémentaires : Pour générer un nouveau répertoire à l'intérieur d'un autre répertoire, utilisez cette commande de base de Linux `mkdir Music/Nouveau` utiliser l'option `p` (parents) pour créer un répertoire entre deux répertoires existants. Par exemple, `mkdir -p Musique/2020/Nouveau` créera le nouveau répertoire « **2020** ».
7. La commande `cp` pour copier les fichiers du répertoire actuel dans un autre répertoire. Par exemple, la commande `cp scenery.jpg /home/utilisateur/Photos` créera une copie de `scenery.jpg` (de votre répertoire actuel) dans le répertoire `Photo`.
8. La commande `touch` vous permet de créer un nouveau fichier vierge via la **ligne de commande Linux**.  
Par exemple, entrez `touch /home/username/Documents/Web.html` pour créer un fichier HTML intitulé `Web` dans le répertoire `Documents`.
9. La commande `gedit` permet d'éditer un fichier avec l'éditeur de texte.
10. La commande `rm` est utilisée pour supprimer les fichiers.
11. La commande `rmdir` est utilisée pour supprimer les répertoires et leur contenu. Si vous voulez seulement supprimer le répertoire – comme alternative à `rmdir` – utilisez `rm -r`.

Note : Soyez très prudent avec cette commande et vérifiez à nouveau dans quel répertoire vous vous trouvez. Cela effacera tout et il n'y aura pas d'annulation.

12. La commande `man` est utilisée pour afficher le manuel de linux, Confus quant à la fonction de certaines commandes Linux ? Ne vous inquiétez pas, vous

pouvez facilement apprendre à les utiliser directement depuis le shell de Linux en utilisant la commande **man**. Par exemple, en entrant la commande **man tail**, vous verrez les instructions manuelles de la commande **tail**.

13. La commande **gpasswd** permet d'ajouter ou de supprimer un utilisateur d'un groupe. Avec l'option **--add** , gpasswd ajoute un utilisateur à un groupe. Avec l'option **--delete** , gpasswd supprime un utilisateur d'un groupe. On peut ajouter plusieurs utilisateurs d'une seule ligne de commande en séparant chaque utilisateur par une virgule.
14. La commande **chown** modifie le groupe du fichier et le lien symbolique n'est pas modifié. Pour changer le propriétaire d'un lien symbolique, il faut utiliser l'option **-h**. Cette fois, c'est bien le groupe du lien symbolique qui est modifié.
15. La commande **chmod** permet de modifier les permissions aux différents types d'accès (lecture, écriture, exécution) des fichiers (et répertoires) indépendamment pour le propriétaire, le groupe ou les autres utilisateurs.
16. La commande **su** (**s**ubstitute **u**ser ou **s**witch **u**ser) est une commande Unix permettant d'exécuter un interpréteur de commandes en changeant d'identifiant de groupe et de User identifier. Sans argument, la commande utilise les UID 0 et le GID 0, c'est-à-dire ceux du compte utilisateur root.
17. La commande **sudo** (Parfois considérée comme l'abréviation de **S**uper-**u**ser **d**o) est un programme dont l'objectif est de permettre à l'administrateur du système d'autoriser certains utilisateurs à exécuter des commandes en tant que superutilisateur (ou qu'un autre utilisateur).
18. La commande **visudo** pour est une commande spécifique pour modifier le fichier **sudoers** avec l'éditeur de texte de la console. Il est obligatoirement précédé de **sudo** . Il empêche plusieurs utilisateurs de modifier simultanément sudoers, vérifie son intégrité et la présence d'erreurs de syntaxe.
19. La commande **nano** est un éditeur de texte, en console ou en terminal, très utile pour modifier les fichiers de configuration de votre système. Qui dit éditeur de texte dit : programme qui permet essentiellement de modifier des fichiers, donc des textes bruts sans mise en forme (gras, italique, souligné...)

20. La commande `wget` permet de télécharger des fichiers sur le Web. Elle supporte les protocoles HTTP, HTTPS et FTP ainsi que le téléchargement au travers des proxies HTTP.

## 6. Élévation des privilèges des utilisateurs

Dans cette section, nous allons vous expliquer comment élever les droits des utilisateurs. De base, un utilisateur sur Debian a accès uniquement à ses propres fichiers. En effet, dans la mesure où potentiellement plusieurs comptes utilisateurs partagent un système installé, n'importe quel utilisateur ne devrait pas pouvoir accéder à n'importe quel fichier.

### a. root et sudo

Par défaut, seul l'utilisateur **root**, la racine à la base de Debian, possède les privilèges nécessaires pour le paramétrier. Cependant, utiliser root en permanence n'est pas une bonne pratique, pour des raisons que nous expliquerons plus bas.

Le package **sudo**, permettant d'élever les droits des utilisateurs, est installé de base sur les derniers systèmes Debian. Si vous tentez d'utiliser la commande sudo avec un utilisateur sans les privilèges nécessaires,, la console renverra le code suivant (remplacer `foo` par le nom de l'utilisateur) :

```
foo is not in the sudoers file. This incident will be  
reported.
```

Traduction : l'utilisateur qui a tapé cette commande **sudo** n'est pas dans le fichier **sudoers**. Allons corriger ça !

### b. La commande sudo visudo

C'est le fichier **sudoers** qui va nous permettre de modifier les droits des groupes et des utilisateurs.

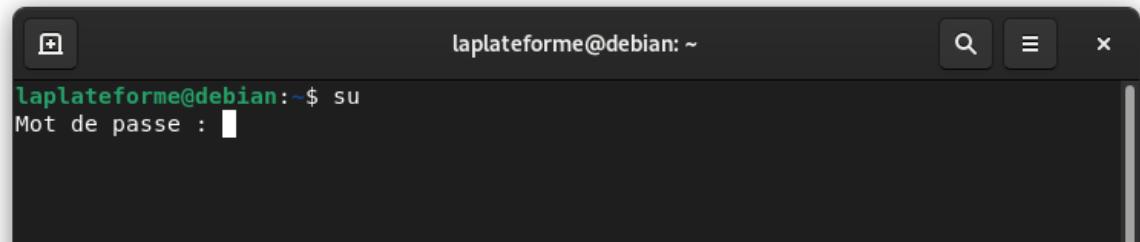
Il existe une commande spécifique pour ouvrir et modifier le fichier **sudoers**, la commande **visudo**. Elle est obligatoirement précédée de **sudo**. Mais rappelez-vous, nous ne pouvons pas encore utiliser la commande sudo.

La solution ?

Nous allons nous connecter au root avec la commande **su**.

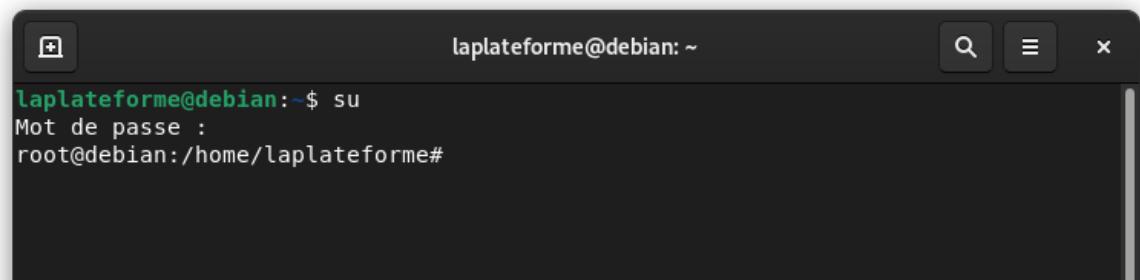
La commande **su** (switch user) tente de nous connecter à root si on ne lui ajoute aucune option.

Tapez la commande su dans votre Terminal. On vous demande alors le mot de passe de root :



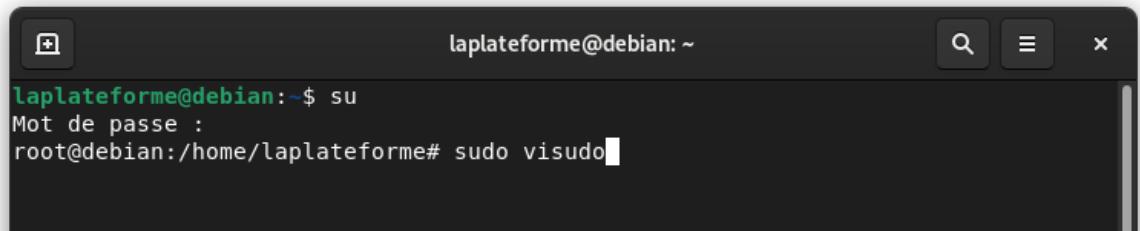
A screenshot of a terminal window titled "laplateforme@debian: ~". The user has typed "laplateforme@debian:~\$ su" followed by "Mot de passe :". The password field is empty and represented by a black redaction box.

Après avoir entré le mot de passe, vous êtes maintenant connecté en tant que root et la commande **sudo** est disponible.



A screenshot of a terminal window titled "laplateforme@debian: ~". The user has typed "laplateforme@debian:~\$ su" followed by "Mot de passe :". After entering the password, the terminal shows "root@debian:/home/laplateforme#", indicating successful root login.

Maintenant, nous allons ouvrir le fichier **sudoers**. Pour cela, il faut utiliser la commande spécifique **sudo visudo**. Elle va ouvrir le fichier sudoers situé par défaut dans le répertoire **/etc/**.



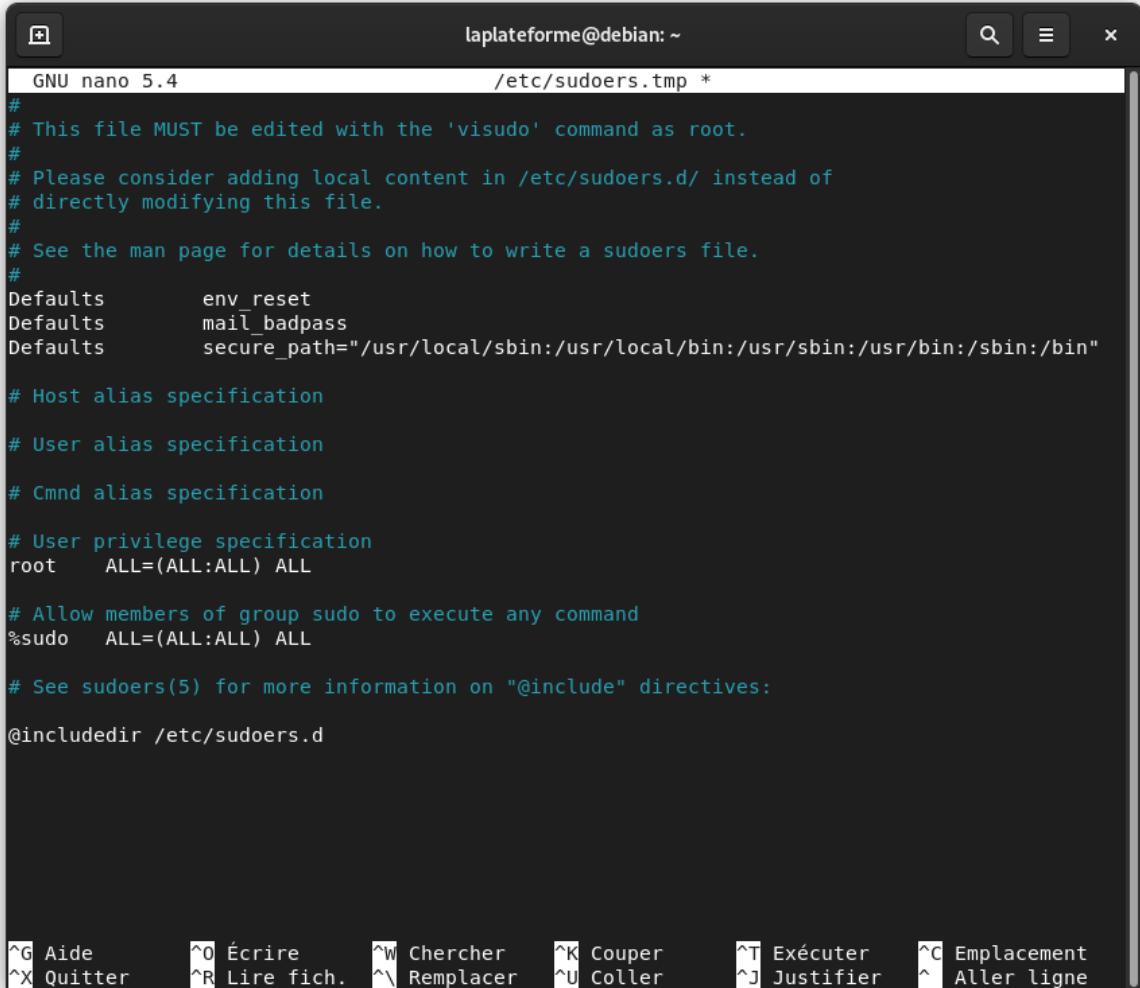
A screenshot of a terminal window titled "laplateforme@debian: ~". The window shows a root shell with the command "root@debian:/home/laplateforme# sudo visudo" entered and ready to be executed.

Passons maintenant à l'étape suivante.

### c. Le fichier sudoers

Vous êtes maintenant dans le fichier **sudoers**. Observons un peu son contenu.

## DOCUMENTATION HARDWARE



The screenshot shows a terminal window titled "laplateforme@debian: ~". The window contains the content of the /etc/sudoers file, which is a configuration file for the sudo command. The file includes sections for Defaults, Host alias specification, User alias specification, Cmnd alias specification, and User privilege specification. It also contains a section for members of the sudo group and a note about including other files. The terminal has a dark theme and includes standard keyboard shortcuts at the bottom.

```
GNU nano 5.4          /etc/sudoers.tmp *

#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root      ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo    ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:
@includedir /etc/sudoers.d
```

^G Aide ^O Écrire ^W Chercher ^K Couper ^T Exécuter ^C Emplacement  
^X Quitter ^R Lire fich. ^V Remplacer ^U Coller ^J Justifier ^L Aller ligne

La syntaxe est la suivante :

```
identifiant  ALL = (user)  /chemin/complet/commande,/chemin/complet/autrecommande
%groupe      ALL = (user)  /chemin/complet/commande,!/chemin/complet/autrecommande
```

- **identifiant** représente un identifiant utilisateur du système. Un seul identifiant doit être précisé par ligne ;
- **%groupe** désigne un groupe d'utilisateurs du système. Le nom du groupe doit donc être précédé d'un symbole de pourcentage (%). Un seul groupe doit être précisé par ligne ;
- **ALL** désigne la ou les machines dans lesquelles les commandes suivantes sont autorisées ou refusées pour cet utilisateur ou ce groupe d'utilisateurs. Le mot-clé **ALL** désigne l'ensemble des machines de votre parc informatique. Dans le cadre d'une utilisation à domicile, laisser **ALL** n'est pas un

inconvénient. Dans un grand parc d'entreprise, de meilleures stratégies sont à prévoir ;

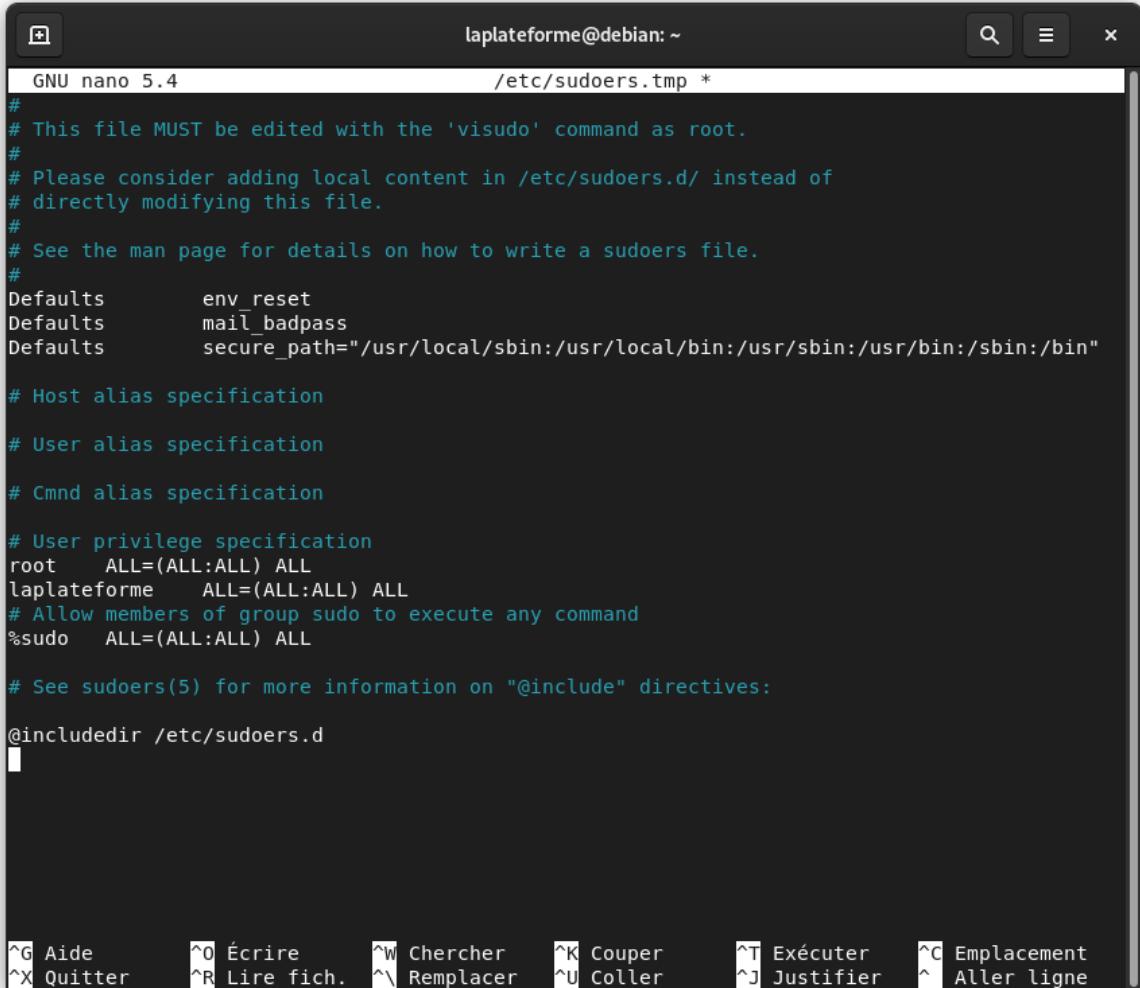
- `user` (entre parenthèses) désigne l'utilisateur dont on prend les droits (peut valoir ALL pour tous)
- `commande` et `autrecommande` représentent des commandes pouvant être exécutées par l'utilisateur ou le groupe d'utilisateurs désigné en début de ligne.
  - Les commandes précédées d'un point d'exclamation (!) sont refusées, alors que celles sans point d'exclamation sont autorisées ;
  - Les commandes multiples sont séparées par une virgule, sans espace ;
  - Les commandes doivent être entrées de manière exacte. Pour cette raison, préférez saisir des chemins absous vers des commandes plutôt que des chemins relatifs (par exemple, `/usr/sbin/update-manager` plutôt que `update-manager`). Pour connaître le chemin absolu d'une commande ou d'un utilitaire, saisissez dans un terminal `which commande`, ou `whereis commande` en remplaçant `commande` par la commande en question.

La ligne de root lui permet d'effectuer dans toutes les machines pour tous les utilisateurs de tous les groupes, toutes les commandes. On peut voir qu'il existe déjà un groupe %sudo avec les mêmes permissions que root. On pourrait donc taper la commande `sudo gpasswd -add [nomutilisateur] sudo`, pour ajouter notre utilisateur au groupe sudo ; il obtiendrait ainsi les droits de ce groupe.

Cependant, puisque nous sommes déjà dans le fichier **sudoers**, nous allons effectuer nos modifications dedans.

Ajoutez la ligne `[nomutilisateur] ALL=(ALL:ALL) ALL` en dessous de root :

## DOCUMENTATION HARDWARE



```
GNU nano 5.4          /etc/sudoers.tmp *

#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
laplateforme  ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
```

Quittez avec **Ctrl+x**, sauvegardez.

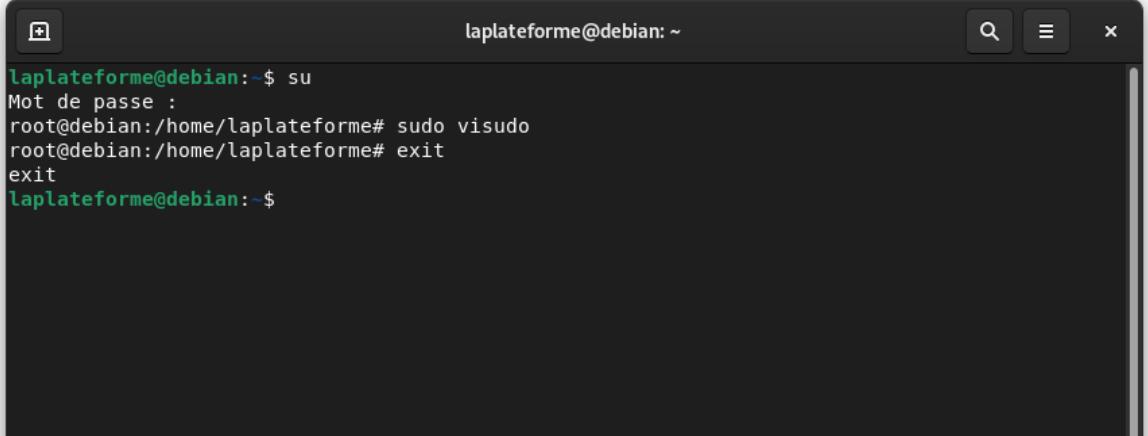
Voilà ! Vous pouvez maintenant utiliser la commande **sudo**, avec les mêmes droits que **root**.

Pour vérifier que nos changements ont bien été effectués, nous allons effectuer une commande sudo avec notre utilisateur.

Déconnectez-vous de root avec la commande **exit**.

## DOCUMENTATION HARDWARE

---



```
laplateforme@debian:~$ su
Mot de passe :
root@debian:/home/laplateforme# sudo visudo
root@debian:/home/laplateforme# exit
exit
laplateforme@debian:~$
```

Relancez la commande `sudo visudo sudoers`, tapez le mot de passe.



```
laplateforme@debian:~$ su
Mot de passe :
root@debian:/home/laplateforme# sudo visudo
root@debian:/home/laplateforme# exit
exit
laplateforme@debian:~$ sudo visudo
[sudo] Mot de passe de laplateforme : █
```

## DOCUMENTATION HARDWARE

---

Voilà ! Vous pouvez maintenant quitter.

The screenshot shows a terminal window titled "laplateforme@debian: ~". The window title bar also displays "GNU nano 5.4" and the file path "/etc/sudoers.tmp". The main content area of the terminal shows the configuration of the sudoers file. It includes standard sections like Defaults, Host alias specification, User alias specification, Cmnd alias specification, and User privilege specification. It also includes specific entries for the root user and the "laplateforme" user, both granting them full privileges (ALL). A note at the bottom of the file indicates that members of the sudo group can execute any command. The bottom status bar of the terminal shows the message "[ Lecture de 27 lignes ]" (Reading 27 lines) and a series of keyboard shortcuts for nano editor commands: ^G Aide, ^X Quitter, ^O Écrire, ^R Lire fich., ^W Chercher, ^Y Remplacer, ^K Couper, ^U Coller, ^T Exécuter, ^J Justifier, ^C Emplacement, ^L Aller ligne.

```
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root      ALL=(ALL:ALL) ALL
laplateforme  ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo    ALL=(ALL:ALL) ALL
%teamstart ALL=(ALL:ALL) ALL
# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
```

### d. Créer un utilisateur sudo

Un utilisateur qui est déjà membre du groupe sudo peut créer un autre utilisateur (par exemple foo) comme membre du groupe sudo à partir de la ligne de commande : `:sudo adduser foo -G sudo`.

# 7. Installer un logiciel sur Debian avec le gestionnaire de paquets

Maintenant que vous possédez les droits du superutilisateur, vous pouvez installer des packages à la racine de Debian.

Dans cette section nous allons vous apprendre à installer un logiciel sur Debian, en utilisant le gestionnaire de paquets. À la fin de cette section, nous aurons installé la dernière version stable du navigateur Web Google Chrome.

La dernière version de Debian est la 11.4, dite **bullseye**. C'est celle que nous utilisons.

Ici, pas de fichier **.exe** comme sur Windows : pour installer un logiciel sur Debian, il faut obtenir le paquet debian ayant l'extension **.deb**.

## a. Préparatifs

Tout d'abord, Debian doit pouvoir chercher les paquets depuis le dépôt distant. Pour ce faire, il faut ajouter les adresses sources sur le fichier **sources.list** dont le chemin complet est **/etc/apt/sources.list**.

Tapez dans votre console `sudo nano /etc/apt/sources.list`

Pour Debian 11.4, il vous faut remplacer le contenu de **sources.list** par les lignes suivantes :

```
# Debian Bullseye, dépôt principal + paquets non libres
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye main contrib non-free

# Debian Bullseye, mises à jour de sécurité + paquets non libres
deb http://deb.debian.org/debian-security/ bullseye-security main contrib
non-free
deb-src http://deb.debian.org/debian-security/ bullseye-security main contrib
non-free

# Debian Bullseye, mises à jour "volatile" + paquets non libres
deb http://deb.debian.org/debian/ bullseye-updates main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye-updates main contrib non-free
```

Note : les lignes commençant par # sont commentées, et ne sont pas interprétées comme du code.

Maintenant, Debian va pouvoir chercher les paquets et mises à jour demandés aux adresses url indiquées.

### b. Les gestionnaires de paquets

Pour gérer les packages sur Debian, il existe deux commandes principales :

- **apt** (pour **advanced package tool**)
- **dpkg** (pour **Debian package**)

**dpkg** est un gestionnaire dit “de bas niveau”, c'est-à-dire plus proche de la machine que le gestionnaire **apt**, quant à lui plus “user-friendly”. Celui-ci exécute **dpkg** et d'autres applications en arrière-plan, et permet d'automatiser la gestion des dépendances.

Nous allons vous présenter deux méthodes d'installation.

### c. L'installation simple

Ouvrez le **Terminal**.

Ajoutez le dépôt Google dans le fichier des sources :

```
sudo sh -c 'echo "deb https://dl.google.com/linux/chrome/deb/stable main" > /etc/apt/sources.list.d/google-chrome.list'
```

La commande précédente utilise **sh**, l'interpréteur de commande standard pour le système. L'option **-c** (**noclobber** = ne pas écraser) n'écrase pas les fichiers déjà existants. Ensuite, **echo** écrit la source dans le fichier **google-chrome.list** situé dans le répertoire **sources.list.d**.

Mettez à jour les listes :

```
sudo apt update
```

Installez la dernière version de Google Chrome disponible :

```
sudo apt install google-chrome-stable
```

### d. La méthode manuelle

Nous devons obtenir le fichier **.deb**. Téléchargez la dernière version stable [ici](#) ou sur le site de Google.

OU

Récupérez le fichier via la commande **wget** ci-dessous.

```
wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
```

Installez le paquet téléchargé avec **apt** :

```
sudo apt install ./google-chrome-stable_current_amd64.deb
```

Par défaut, **apt** va chercher des fichiers à partir des sources indiquées dans **sources.list** et **sources.list.d**, il faut donc lui indiquer le chemin du fichier **.deb**, voilà pourquoi nous utilisons **./** qui indique le répertoire actuel.

Et voilà ! Google Chrome est maintenant installé sur votre ordinateur.

### e. Alternatives à Google Chrome

Chrome est un navigateur Web populaire, développé par une très grande entreprise. Cependant, il a un inconvénient : ce n'est pas un logiciel Open Source.

Mais pourquoi est-ce un inconvénient, me demanderiez-vous ? Google Chrome est un logiciel propriétaire, ce qui signifie qu'une partie ou la totalité de son code est accessible uniquement par Google et ceux à qui Google donne l'autorisation.

## DOCUMENTATION HARDWARE

---

À titre comparatif, un logiciel propriétaire est comme une boîte noire : on ne peut pas voir le code du logiciel, ni savoir ce qu'ils font des données envoyées par votre ordinateur. Autre inconvénient, en cas de faille de sécurité, si une personne mal intentionnée obtient le code source du logiciel, seuls eux et les propriétaires y auront accès et pourront vérifier la présence de failles.

À l'inverse, les logiciels libres "montrent patte blanche" et tout le monde peut vérifier le contenu du code, et chaque commit est contrôlé et validé avant d'être ajouté au code source.

Ci-dessous nous allons vous présenter quelques exemples de navigateurs Open Source, tous disponibles sur et Linux.

### [Brave](#)



Le navigateur [Brave](#) est un navigateur Open Source basé sur Chromium. Ils mettent en avant leur politique de protection de la vie privée et de sécurité. Il possède un bloqueur publicitaire intégré, et bloque les traceurs Web. Tout cela a pour conséquence de diminuer votre empreinte numérique et de charger plus rapidement les pages Web. Brave supporte les extensions de Chrome.

### [Chromium](#)



[Chromium](#) est un logiciel libre dont Google a démarré le projet. Google Chrome se base sur Chromium en ajoutant des fonctionnalités supplémentaires, ce dernier est donc plus léger.

### [Mozilla Firefox](#)



Si vous avez installé Debian, [Firefox](#) est peut-être déjà installé sur votre OS. C'est est un navigateur populaire, complet et très personnalisable, avec de nombreux thèmes et extensions. Il a influencé le développement du Web et de navigateurs Web comme Google Chrome et c'est sans doute grâce à Firefox que le Web et les navigateurs Web sont ceux qu'ils sont aujourd'hui.