

# TP Unix avancé

## Installation d'un Linux (dans une VM)

ENSIMAG

Novembre 2018

### 1 Installation d'un système d'exploitation

Le système d'exploitation est le logiciel qui contrôle le matériel et expose des abstractions de ce matériel, comme la notion de *fichier* ou de *processus*, à l'utilisateur. Le but est de factoriser et partager le code qui est nécessaire à tous les autres programmes.

Le but de ce TP est de vous faire installer Linux sur une machine pour mettre en lumière certains aspects fondamentaux.

#### 1.1 Variante pour ceux qui ont déjà installé Linux

Pour ce TP, nous allons installer GNU/Linux Debian. C'est notre compromis entre facilité d'installation et affichage des choix importants, pour le côté pédagogique.

Pour ceux qui auraient déjà installé Debian, vous êtes libres de faire ce TP avec les fichiers ISO d'autres versions de Linux ou d'autres systèmes (par ordre supposé de difficulté/geekitude) : Ubuntu (plus facile et joli), Fedora (avec la toute dernière version des logiciels à la mode, parfois même la suivante), FreeBSD, Arch, Gentoo, OpenBSD, Minix 3, Redox OS, Haiku, ou Hurd.

#### 1.2 Installation dans un simulateur ou installation sur une machine physique

Pour ce TP, en 1h30, nous utiliserons le simulateur de PC Qemu, avec son interface en ligne de commande (CLI). Il existe des interfaces graphiques, comme Aqemu, et d'autres simulateurs qui proposent une gestion plus directe, comme Virtualbox.

L'utilisation d'un simulateur permet de contrôler facilement le comportement de l'installation. Par contre cela oblige à définir, et donc à comprendre, la taille de la mémoire,

l'organisation du stockage, le réseau, etc. Il faut aussi configurer les relations réseaux entre le système d'exploitation hôte (host) qui s'exécute sur votre machine physique, et le système d'exploitation invité (guest) qui s'exécute dans le simulateur.

Sur une machine physique, l'installation du système lui-même n'est pas plus difficile, juste différente. Le matériel a été choisi par le constructeur avec des tailles raisonnables pour une utilisation courante. Par contre, il faut régler ou configurer les trois parties impliquées dans le démarrage (cf section 3). C'est ce point qui est parfois délicat, suivant ce que le constructeur a choisi comme défaut.

Pour une véritable installation sur votre machine perso., les *Bug Busters* sont là pour vous aider. Une affiche avec leur salle et les horaires est présente dans chaque salle.

## 2 Avant de commencer !

### 2.1 Vérification de la place en mémoire (RAM)

Notre simulateur de PC utilisera une partie de votre mémoire RAM de manière intensive. Il est donc important, sous peine de fort ralentissement, que le PC simulé ait une mémoire plus petite que la machine réelle.

On peut obtenir la taille de la RAM avec la commande

```
free -h
```

Dans la suite du TP, nous utiliserons 8 Gio. Si votre machine n'a que 4 Gio, ou un peu moins, vous ne lancerez votre PC simulé qu'avec 2 Gio. Pour cela il faudra remplacer dans la suite du sujet l'usage de l'option `-m 8G` par `-m 2G`.

### 2.2 Changer le PATH pour avoir une version récente de qemu

Pour ce TP, nous n'utiliserons pas l'une des deux versions de qemu déjà installé. Nous allons utiliser la version plus récente qui est dans le compte de `~jdpunix`. Pour cela nous allons modifier le PATH pour y ajouter le bon répertoire.

```
export PATH=/matieres/3MMUNIX/local/bin:$PATH
```

### 2.3 Vérification de la place disque sur les machines de l'Ensimag

Votre compte Ensimag a un quota maximal de 10 Gio. Pour faire ce TP nous allons utiliser un "disque" simulé de 100 Gio, il ne pourra donc pas être stocké sur votre compte. Heureusement, il existe sur chaque machine un espace disque local dans lequel vous pouvez créer des fichiers : `/tmp_data`.

Vérifiez qu'il y a suffisamment de place dans cet espace :

```
df -h /tmp_data
```

Cela devrait afficher des informations du style :

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda7	232G	1G	231G	1%	/tmp_data

L'important est que la place disponible (*Avail*) soit plus grande que les 100 Gio que nous allons utiliser.

**NB : pour les prochains qui passeront après vous pour faire ce TP, il faudra bien penser à effacer votre image à la fin du TP !**

### 2.3.1 Si il n'y a pas assez de place...

S'il reste plus de 25 Gio, vous pouvez faire ce TP avec l'image Debian (Gnome). Il faudra juste adapter la taille du disque, en prenant 25 Gio au lieu de 100 Gio.

S'il reste moins de 25 Gio, vous utiliserez l'image Debian (LXDE) qui demande moins de ressource disque pour son installation par défaut. Un disque de 3 Gio devrait suffire. Il vous faudra adapter la taille du disque, le nom de l'image iso, le schéma de partition.

Si, sur votre PC, vraiment il n'y a plus assez de place dans `/tmp_data`, vous utiliserez votre HOME avec une image de 3 Gio et l'image LXDE.

## 3 Démarrage d'un PC

Le démarrage et la gestion d'un ordinateur fait intervenir trois logiciels qui collaborent : l'UEFI/BIOS, le chargeur/amorce/bootstrap (« bootloader ») et le système d'exploitation. Les liens entre les trois ne sont vraiment visibles que lors des installations (ou certains bugs autour la mise en veille).

L'UEFI (souvent encore appelé BIOS) est le logiciel écrit « en dur » dans une mémoire permanente et qui est le premier code exécuté lorsque vous allumez la machine. Il est responsable de la détection et de l'allumage du matériel de base : taille de la mémoire, contrôle de l'alimentation électrique des périphériques, etc. L'UEFI/BIOS termine en trouvant un *chargeur* et en lui passant la main.

Démarrez le simulateur en tapant :

```
qemu-system-x86_64 -machine q35,accel=kvm
```

L'UEFI/BIOS exécuté par le simulateur devrait vous indiquer qu'il ne trouve rien à lancer (les messages « Boot failed ») ni sur un des disques ni sur le réseau simulé.

Le chargeur est le logiciel qui va copier le code du système d'exploitation dans la mémoire. Son code doit être à un endroit accessible par l'UEFI/BIOS : les 512 premiers octets d'un disque, ou dans une partition (morceau) d'un disque géré par UEFI.

Vous pouvez maintenant démarrer le simulateur en lui indiquant une image ISO (cdrom/dvdrom) d'une distribution Linux à utiliser comme contenu de son lecteur de cd/dvd. Le début de l'image ISO contient un chargeur.

```
qemu-system-x86_64 -machine q35,accel=kvm -cdrom  
~jdpunix/iso/debian-gnome.iso
```

Votre simulateur va maintenant vous offrir un écran d'accueil, celui du chargeur. Vous pouvez démarrer en "Live". Dans le menu du chargeur vous pouvez aussi choisir la langue (très utile pour le clavier!). Ensuite, le démarrage va paniquer (Kernel panic). Par défaut Qemu n'utilise que 128 Mio, ce qui est trop peu pour démarrer un Linux récent sans précaution.

### 3.1 Préparer un fichier creux comme disque

Avant de relancer le simulateur, nous allons préparer un fichier pour servir de stockage pérenne (disque, ssd) de 100 Gio. Il est possible de créer un fichier de 100 Gio mais qui restera vide tant qu'il ne sera pas utilisé.

Vous pouvez créer un fichier «creux» (`3mmunix_vmdisk.img`) dans le répertoire `/tmp_data` avec la commande

```
dd of=/tmp_data/3mmunix_vmdisk.img bs=100G seek=1 count=0
```

Il est possible de vérifier que votre disque est bien un fichier de taille 100 Gio creux, occupant pour l'instant 0 octet sur le disque.

```
ls -lh /tmp_data/3mmunix_vmdisk.img
-rw-r--r-- 1 gregory gregory 100G oct. 17 17:27 /tmp_data/3mmunix_vmdisk.img

du -h /tmp_data/3mmunix_vmdisk.img
0      /tmp_data/3mmunix_vmdisk.img
```

Démarrer le simulateur en lui indiquant d'utiliser 8 Gio de mémoire, l'image ISO comme cdrom et le fichier que nous venons de créer comme disque dur. Attention à bien lancer la commande complète, même si elle apparaît sur plusieurs lignes ci-dessous :

```
qemu-system-x86_64 -machine q35,accel=kvm -m 8G
-cdrom ~/jdpunix/iso/debian-gnome.iso
/tmp_data/3mmunix_vmdisk.img
```

Pour l'instant vous allez juste démarrer votre système en *Live* (“Debian Live with Localization”, puis “French”), sans commencer l'installation. Avant une installation sur une véritable machine, cela permet de vérifier que le minimum fonctionne (écran, clavier, souris, réseau). Et cela suffit pour faire déjà beaucoup de choses comme copier les données du disque pour faire un backup, ou surfer avec Firefox.

## 4 Installer Debian sur notre pseudo disque de 100Go

Arrêtez la machine virtuelle puis relancez-la. Cette fois vous allez choisir de lancer l'installateur : « Debian Installer ».

### 4.1 Langue et utilisateurs

Après avoir choisi la langue, vous devrez choisir le mot de passe de l'administrateur, ainsi que le login et le mot de passe de l'utilisateur standard. Vérifiez bien que la disposition de clavier est en azerty pour le choix du mot de passe...

### 4.2 Partition du disque

Le choix suivant est un peu délicat. Il faut choisir quelle place donner à Linux sur le disque. Avec notre disque vide, c'est particulièrement simple. Sur une machine physique

ayant déjà un système que l'on veut conserver, il faudra d'abord réduire les partitions de cet OS pour faire de la place. Si cet OS est Windows, il faut aussi parfois régler coté Windows des détails pour le démarrage coté UEFI ou chargeur de Windows. Vous devriez voir avec les Bugs Busters en cas de doute.

Debian vous propose un choix par défaut (partitionnement assisté), qui est le plus simple et qui convient très bien dans notre cas. Dans les menus suivants, vous choisirez le mode avec la partition `/home` séparée. Cela permet d'installer facilement un autre Linux plus tard, sans perdre les données des utilisateurs qui sont stockées dans `/home`.

#### 4.2.1 Variante bonus : Utilisation de LVM, ou LVM chiffré

LVM ajoute un niveau d'indirection pour les partitions. Cela complexifie un peu les partitions avec la nécessité d'une partition `/boot` sans LVM pour le noyau et grub, mais permet plus tard de redimensionner les partitions ou d'ajouter un disque de manière transparente pour le système.

### 4.3 Miroir réseau

Les « miroirs réseau » sont des serveurs web qui contiennent l'ensemble des logiciels intégrés dans Debian. Pour que l'on puisse installer des logiciels une fois le système installé, il est donc nécessaire de choisir un miroir. Prenez un miroir en France pour que ce soit plus rapide, par exemple `ftp.rezopole.net`.

### 4.4 La fin de l'installation : le bootloader

La suite ne devrait pas vous poser de problème. La copie des données sur le disque peut prendre jusqu'à plusieurs minutes.

À la fin de l'installation, Debian vous demande sur quel disque vous souhaitez installer le bootloader. Il faut choisir le disque virtuel, à savoir `/dev/sda`. Si vous n'installez pas le bootloader (choix par défaut), alors votre machine virtuelle ne pourra pas démarrer !

## 5 Premier redémarrage

Après un redémarrage, vous obtenez à nouveau le menu de Grub, mais cette fois il démarre votre OS nouvellement installé.

Vous devriez obtenir le menu graphique GDM pour choisir votre utilisateur. Connectez-vous (la première connexion est longue et suivant la rapidité de votre machine physique, vous serez peut-être éjecté à la première connexion, dans ce cas recommencez). Vous devriez être sous GNOME, comme à l'Ensimag.

## 6 Installation d'autres logiciels

La grande force de Debian est son immense catalogue de logiciels (environ 40 000), bien testés et cohérents les uns avec les autres. Debian est une distribution communautaire, avec plus d'un millier de développeurs à travers le monde qui s'occupent de cette base de logiciels et de leur intégration dans Debian.

## 6.1 Su ou sudo

Au début, l'utilisateur de base n'a aucun droit. L'administrateur peut lui donner les droits d'utiliser sudo, pour faire ce que l'administrateur peut faire.

Pour cela, il faut se connecter comme administrateur dans un terminal et ajouter l'utilisateur au groupe sudo.

```
su - # pour devenir administrateur
adduser alice sudo # pour ajouter l'utilisatrice alice au group sudo
groups alice # devrait afficher plein de groupes, dont sudo
```

L'appartenance au groupe ne sera valide qu'à la prochaine connexion de l'utilisateur. Déconnectez-vous et reconnectez-vous.

Si sudo ne marche pas et que l'utilisateur est bien dans le groupe sudo, fermer et redémarrer la machine virtuelle puis reconnectez-vous.

Si vous êtes impatient, vous pouvez vous reconnecter comme utilisateur alice dans un terminal :

```
id # devrait afficher alice
su - alice # pour devenir alice, mais avec les nouveaux groupes
sudo ls /root # devrait maintenant fonctionner
```

## 6.2 Installer «sl» et «fortunes»

Pour mettre à jour la liste des logiciels disponibles :

```
sudo apt update
```

Bien qu'étant des programmes particulièrement indispensables, "sl" et "fortune" ne sont pas installés par défaut. Nous allons voir comment les installer.

Pour chercher dans la liste des logiciels :

```
apt search fortunes
```

Pour voir le descriptif détaillé d'un ou plusieurs logiciels :

```
apt show sl fortunes
```

Pour les installer :

```
sudo apt update
sudo apt install sl fortunes
sl
fortune
```

## 7 Faire des transferts de fichiers en utilisant SSH

Pour terminer ce TP, vous pouvez vous connecter au démon ssh de la machine hôte. Cela permet de faire facilement des copies de fichier entre les deux environnements.

## 7.1 Connection ssh

Par défaut, votre machine a comme adresse 10.0.2.15. La machine hôte est sur le même réseau à l'adresse 10.0.2.2. En utilisant votre login, vous devriez pouvoir vous connecter depuis un terminal.

```
ssh alice@10.0.2.2
```

Si tout fonctionne, vous pouvez configurer le gestionnaire de fichier nautilus, pour faire facilement des transferts de fichiers dans les deux sens.

## 8 En conclusion

Après ce TP, vous devriez être convaincu qu'installer Linux n'a rien de compliqué. C'est assez facile dans une machine virtuelle et très similaire sur une machine physique. Pour conserver Windows, suivant la façon avec laquelle il a été installé, vous aurez besoin d'un peu plus d'expérience pour obtenir une cohabitation harmonieuse facilement. Les Bug Busters sont là pour vous aider pour ça.

Pensez à effacer  
`/tmp_data/3mmunix_vmdisk.img` pour ne pas encombrer inutilement le disque du PC de votre TP !