Introduction à Linux pour le poste de travail

Attribution - Partage dans les Mêmes Conditions :
http://creativecommons.org/licenses/by-sa/4.0/fr/

Table des matières

Objectifs 3

Introduction 4

I - Linux en deux mots 5

II - Exercice 7

III - Le terminal 8

IV - Exercice 9

V - Lancer des applications sous Linux 10

VI - Exercice 11

VII - Commandes de base sous Linux : cd, ls, mkdir, rm, find, cat, nano... 12

VIII - Découverte de la ligne de commande sous Linux 14

1. Exercice : man man 14
2. Exercice : ls 14
3. Exercice : cd pwd mkdir touch 15

Solutions des exercices 17

Glossaire 20

Contenus annexes 21

Objectifs

- Savoir survivre devant un PC sous Linux en utilisant le terminal

Introduction

Durée : 1h

I Linux en deux mots

- Linux est un *système d'exploitation* p.20 né en 1991.
- C'est **le premier système utilisé pour les serveur informatiques** (devant Windows et d'autres UNIX).
- C'est le troisième système utilisé pour les ordinateurs personnels (derrière Windows et MacOSX)
- C'est un système *libre* p.20 et gratuit.
- Il fait partie de la famille des Unix, à l'instar d'Android, MacOSX ou FreeBSD.

Distribution Linux — Az Définition

On appelle distribution Linux un ensemble de logiciels composé d'un système Linux et de logiciels complémentaires pré-installés et pré-paramétrés, typiquement : une interface graphique de gestion de fichiers, des suites bureautiques, des outils internet, des logiciels multimédias...

Exemples de distributions Linux — Exemple

- Debian
- Ubuntu
- Xubuntu
- Fedora
- ...

Environnement de bureau — Az Définition

Un environnement de bureau graphique permet de manipuler l'ordinateur à travers une interface graphique.

Chaque distribution Linux propose un ou plusieurs environnements de bureau.

Exemples d'environnement de bureau Linux — Exemple

- Gnome
- XFCE
- KDE
- ...

Linux ou GNU/Linux Complément

GNU/Linux est la réunion de deux parties, le projet GNU de Richard Stallman et le projet Linux de Linus Torvalds.

Le nom Linux désigne en général le système d'exploitation dont le nom complet est GNU/Linux. On utilisera l'un pour l'autre dans le cadre de ce cours.

Exercice

[solution n°1 p. 17]

A Windows **B** MATE **C** Unix **D** ArchLinux **E** MacOS **F** RedHat **G** Debian **H** Gnome **I** Ubuntu **J** XFCE **K** Linux

Distribution Linux	Environnement de Bureau	Système d'Exploitation Propriétaire	Famille de Système d'Exploitation

III Le terminal

Les distributions Linux comportent un mode graphique, pratique pour de nombreuses opérations.

Savoir utiliser un terminal en mode ligne de commande n'est donc pas indispensable, mais cela présente des avantages comme :

- connaître des opérations qui seront reproductibles sur toutes les distributions (voire sur d'autres Unix) ;
- savoir utiliser un serveur à distance (via SSH) ;
- savoir échanger ou reproduire des procédures (sans avoir besoin de refaire une succession de manipulations à la souris).

Fondamental

Dès lors qu'on se connecte à un système Linux, on peut ouvrir un terminal et faire de nombreuses opérations, telles que :

- lancer des applications,
- manipuler des fichiers,
- se connecter à un autre ordinateur.

Complément

Qu'est-ce que la console ? (cf. p.21)

La console sous Linux (cf. p.22)

Exercice

[solution n°2 p. 17]

Ouvrez un terminal et tapez l'instruction suivante :

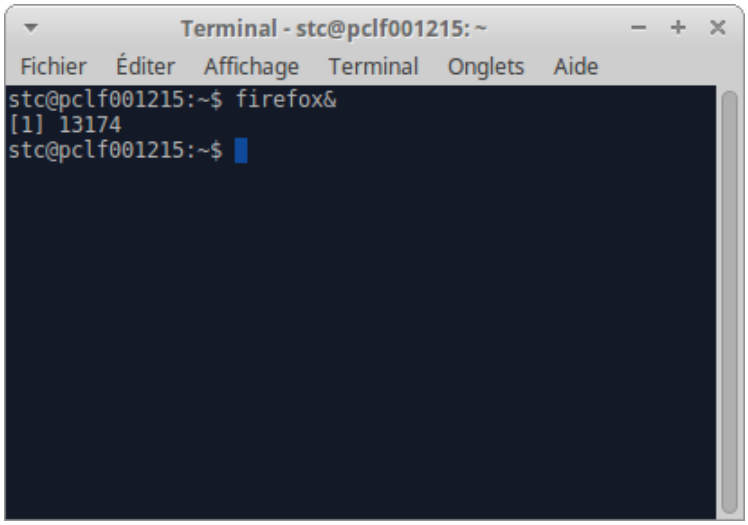
```
1 echo 42
```

V Lancer des applications sous Linux

Méthode

Le plus générique pour lancer une application sous Linux consiste à :

1. ouvrir un terminal,
2. écrire le nom de l'application, par exemple `firefox`,
3. saisir le caractère & et appuyer sur entrée.



La touche TAB

Conseil

En général, il n'est pas nécessaire de saisir l'intégralité du nom du programme : on saisit le début, par exemple `fir` et on appuie sur la touche TAB.

Dès que le système identifie l'application recherchée, il complète automatiquement avec les caractères manquants.

Exercice

[solution n°3 p. 17]

L'application graphique permettant de gérer les fichiers (gestionnaire de fichiers) de l'environnement graphique XFCE s'appelle `thunar`.

Quelle est la commande à entrer dans un terminal pour ouvrir une fenêtre du gestionnaire de fichier ?

VII Commandes de base sous Linux : cd, ls, mkdir, rm, find, cat, nano...

Le manuel

La commande `man` permet d'afficher le manuel d'une autre commande.

Par exemple `man ls` permet d'afficher les options de la commande `ls`, et `man man` est l'affichage du manuel du manuel.

Organisation des fichiers

Les fichiers sont organisés sous Linux dans des dossiers (ou répertoires) arborescents (il n'y a pas de notion de disques).

Le premier de ces dossiers, appelé racine, est `/`.

Des fils courants de racines sont :

- `/bin` qui contient des programmes du systèmes ;
- `/home` qui contient les données des utilisateurs ;
- `/tmp` qui contient des données volatiles accessibles à tous.

Gestion de fichiers

- `pwd` : savoir où je me situe dans l'arborescence
- `cd` : aller quelque part dans l'arborescence
  - `cd /home` : aller dans /home (déplacement absolu)
  - `cd me` : aller dans le répertoire me fils de mon répertoire courant (déplacement relatif)
  - `cd ..` : remonter dans l'arborescence (déplacement vers son père)
  - `cd ~` : permet de retourner dans son dossier initial (retour au domicile)
- `ls` : voir les fichiers et dossiers dans mon dossier actuel (sauf les fichiers cachés commençant par un .)
- `ls -al` : voir tous les fichiers et dossiers dans mon dossier actuel avec leurs informations associées (vue détaillée)
- `touch file` : créer un fichier file
- `rm file` : supprimer le fichier file dans mon dossier actuel
- `mkdir dir` : créer un nouveau dossier dir dans mon dossier actuel
- `rm *` : supprimer tous les fichiers de mon dossier actuel
- `rm -R dir` : supprimer le dossier dir dans mon dossier actuel
- `cat file` : afficher le contenu du fichier file
- `more` : afficher le contenu du fichier file en mode paginé
- `less` : afficher le contenu du fichier file en mode défilement

Édition de fichiers

- `nano file`
  - Éditeur dans le terminal (simple d'utilisation)
  - Les commandes sont indiquées en bas de l'éditeur
- `gedit file &`
  - Éditeur graphique
  - Utiliser une extension de fichier standard ou le menu `Affichage > Mode de coloration` pour obtenir une visualisation adaptée au type de fichier édité (par exemple .sql pour un fichier SQL)

Rechercher un fichier

- `find / -name '*test*'` permet de rechercher un fichier contenant la chaîne test sur tout le disque
- `find ~ -name '*test*'` permet de rechercher un fichier contenant la chaîne test dans son espace personnel

Complément

Agir en tant que root (su / sudo) (cf. p.24)

Installer des applications sous Linux (cf. p.25)

VIII Découverte de la ligne de commande sous Linux

Introduction

Cet exercice consiste à expérimenter quelques commandes de base sous Linux.

1. Connectez vous à une machine Linux ;
2. Ouvrez un terminal.

1. Exercice : man man

[solution n°4 p. 18]

Exécutez la commande `man man`.

Exercice

Comment s'appelle la section qui résume la syntaxe d'une commande dans le manuel ?

Exercice

Quelle touche doit-on utiliser pour convoquer l'aide de *man* ?

Exercice

Quelle touche doit-on utiliser pour rechercher un mot dans un manuel *man* ?

Exercice

Quelle touche doit-on utiliser pour poursuivre une recherche **en avançant** dans un manuel *man* ?

Exercice

Quelle touche doit-on utiliser pour poursuivre une recherche **en arrière** dans un manuel *man* ?

Exercice

Quelle touche doit-on utiliser pour sortir de *man* ?

2. Exercice : ls

[solution n°5 p. 18]

Exercice

Quelle commande devez-vous exécuter pour visualiser le manuel de la commande `ls` ?

Exercice

En lisant le manuel, trouvez comment afficher le contenu du répertoire courant en mode *listing*.

3. Exercice : cd pwd mkdir touch ...

[solution n°6 p. 18]

L'exercice suivant consiste à manipuler le système de fichiers. On vous demande de saisir la commande nécessaire pour répondre à chaque question posée.

Exercice

Déplacez-vous dans le répertoire `tmp` situé à la racine du système de fichier.

Exercice

Vérifiez que vous êtes bien au bon endroit.

Exercice

Créez un répertoire *hello*.

Exercice

Déplacez-vous dans ce répertoire.

Exercice

Créez un fichier *world* dans ce répertoire.

Exercice

Affichez la liste des fichiers de ce répertoire en mode listing.

Exercice

Éditez le fichier *world* avec l'éditeur *nano*, et saisissez le texte de votre choix.

Exercice

Après être sorti de l'éditeur, affichez le contenu de votre fichier (avec *cat*, *more* ou *less*)

Exercice

Recherchez dans votre répertoire courant tous les fichiers dont le nom contient *orl*.

Exercice

Supprimez le fichier *world*.

Exercice

Retournez dans votre dossier personnel initial.

Exercice

Supprimez le dossier *hello* que vous avez créé dans */tmp*.

Solutions des exercices

Solution n°1

[exercice p. 7]

Distribution Linux	Environnement de Bureau	Système d'Exploitation Propriétaire	Famille de Système d'Exploitation
Debian	Gnome	Windows	Linux
RedHat	XFCE	MacOS	Unix
Ubuntu	MATE		
ArchLinux			

Solution n°2

[exercice p. 9]

Ouvrez un terminal et tapez l'instruction suivante :

```
echo 42
```

42

La commande `echo` sert à afficher quelque chose dans le terminal.

Solution n°3

[exercice p. 11]

L'application graphique permettant de gérer les fichiers (gestionnaire de fichiers) de l'environnement graphique XFCE s'appelle `thunar`.

Quelle est la commande à entrer dans un terminal pour ouvrir une fenêtre du gestionnaire de fichier ?

thunar &

La commande tunnar & (avec le &) permet de lancer une fenêtre du gestionnaire de fichier tout en conservant le terminal disponible pour de nouvelles opérations.

Solution n°4

[exercice p. 14]

Exercice

Comment s'appelle la section qui résume la syntaxe d'une commande dans le manuel ?

SYNOPSIS

Exercice

Quelle touche doit-on utiliser pour convoquer l'aide de *man* ?

h

Exercice

Quelle touche doit-on utiliser pour rechercher un mot dans un manuel *man* ?

/

Exercice

Quelle touche doit-on utiliser pour poursuivre une recherche **en avançant** dans un manuel *man* ?

n

Exercice

Quelle touche doit-on utiliser pour poursuivre une recherche **en arrière** dans un manuel *man* ?

N

Exercice

Quelle touche doit-on utiliser pour sortir de *man* ?

q

Solution n°5

[exercice p. 14]

Exercice

Quelle commande devez-vous exécuter pour visualiser le manuel de la commande `ls` ?

man ls

Exercice

En lisant le manuel, trouvez comment afficher le contenu du répertoire courant en mode *listing*.

ls -l

Solution n°6

[exercice p. 15]

Exercice

Déplacez-vous dans le répertoire `tmp` situé à la racine du système de fichier.

cd /tmp

Exercice

Vérifiez que vous êtes bien au bon endroit.

pwd

Exercice

Créez un répertoire *hello*.
mkdir hello

Exercice

Déplacez-vous dans ce répertoire.
cd hello

Exercice

Créez un fichier *world* dans ce répertoire.
touch world

Exercice

Affichez la liste des fichiers de ce répertoire en mode listing.
ls -l

Exercice

Éditez le fichier *world* avec l'éditeur *nano*, et saisissez le texte de votre choix.
nano world

Exercice

Après être sorti de l'éditeur, affichez le contenu de votre fichier (avec *cat*, *more* ou *less*)
cat world

Exercice

Recherchez dans votre répertoire courant tous les fichiers dont le nom contient *orl*.
find . -name '*orl*'

Exercice

Supprimez le fichier *world*.
rm world

Exercice

Retournez dans votre dossier personnel initial.
cd ~

Exercice

Supprimez le dossier *hello* que vous avez créé dans */tmp*.
rm -R /tmp/hello

Glossaire

Logiciel libre

Aujourd'hui, un logiciel est considéré comme libre, au sens de la *Free Software Foundation*, s'il confère à son utilisateur quatre libertés (numérotées de 0 à 3) :

- 0. la liberté d'exécuter le programme, pour tous les usages ;
- 1. la liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins ;
- 2. la liberté de redistribuer des copies du programme (ce qui implique la possibilité aussi bien de donner que de vendre des copies) ;
- 3. la liberté d'améliorer le programme et de distribuer ces améliorations au public, pour en faire profiter toute la communauté.

L'accès au code source est une condition d'exercice des libertés 1 et 3.

(Wikipédia ; fsf.org)

OS (Operating System, Système d'Exploitation)

Un ordinateur est une machine qui a besoin de programmes pour faire quelque chose. Lorsqu'un ordinateur démarre il exécute en général un programme interne, dit programme d'amorçage, puis le premier programme externe qu'il exécute est le **système d'exploitation**. C'est le système d'exploitation qui permet ensuite d'exécuter les applications (traitement de texte, navigateur web, client mail...).

Exemples de système d'exploitation :

- Windows
- MacOSX
- GNU/Linux
- Android
- iOS
- FreeBSD
- ...

Contenus annexes

1. Qu'est-ce que la console ?

Objectifs

- Comprendre ce qu'est la console
- Comprendre pourquoi on a inventé la console
- Comprendre l'intérêt de la console aujourd'hui

La console — Az Définition

La console est une interface humain-machine textuelle qui permet d'écrire des commandes à exécuter sur la machine et de visualiser le résultat de ces commandes.

```
kyane@europa:~$ ls
Bureau  code  Documents  Images  Logiciels  mobile  Musique  Nextcloud  obs-records  Téléchargements  Vidéos
kyane@europa:~$ cat /etc/
Display all 332 possibilities? (y or n)
kyane@europa:~$ cat /etc/ss
ssh/ ssl/
kyane@europa:~$ cat /etc/ssh/
moduli                  ssh_config.d/           sshd_config.d/          ssh_host_dsa_key.pub    ssh_host_ecdsa_key.pub  ssh_host_ed25519_key.pub  ssh_host_rsa_key.pub
ssh_config              sshd_config             ssh_host_dsa_key        ssh_host_ecdsa_key      ssh_host_ed25519_key    ssh_host_rsa_key
kyane@europa:~$ cat /etc/ssh/ssh_config

# This is the ssh client system-wide configuration file.  See
# ssh_config(5) for more information.  This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
#  1. command line options
#  2. user-specific file
#  3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options.  For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Include /etc/ssh/ssh_config.d/*.conf

Host *
#   ForwardAgent no
#   ForwardX11 no
#   ForwardX11Trusted yes
#   PasswordAuthentication yes
#   HostbasedAuthentication no
#   GSSAPIAuthentication no
#   GSSAPIDelegateCredentials no
#   GSSAPIKeyExchange no
#   GSSAPITrustDNS no
#   BatchMode no
#   CheckHostIP yes
#   AddressFamily any
#   ConnectTimeout 0
#   StrictHostKeyChecking ask
#   IdentityFile ~/.ssh/id_rsa
#   IdentityFile ~/.ssh/id_dsa
#   IdentityFile ~/.ssh/id_ecdsa
#   IdentityFile ~/.ssh/id_ed25519
#   Port 22
#   Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc
#   MACs hmac-md5,hmac-sha1,umac-64@openssh.com
#   EscapeChar ~
#   Tunnel no
#   TunnelDevice any:any
```

Une console

Historique — Complément

Aux débuts de l'informatique, les premiers ordinateurs étaient beaucoup moins puissants que ceux que l'on utilise aujourd'hui. Il était nécessaire de pouvoir piloter ces ordinateurs à l'aide d'une interface simple et peu gourmande en ressources, utiliser une interface graphique avec des millions de pixels de couleurs et un pointeur de souris n'était pas imaginable à l'époque.

C'est pourquoi la console a été inventée : une interface entièrement textuelle et en noir et blanc.

Utilité de la console

Méthode

On pourrait penser que, maintenant que les ordinateurs savent largement faire tourner une interface graphique, la console ne sert plus à rien.

C'est en réalité tout le contraire, la console s'avère utile dans de très nombreuses situations :

- pour administrer un serveur à distance, qui n'a pas d'écran branché à lui
- pour réaliser des tâches d'administration, par exemple qui relèvent du système d'exploitation, que n'ont pas une interface graphique dédiée
- pour réaliser des opérations courantes qui sont parfois plus simple à faire qu'en utilisant l'interface graphique.

Pour ce dernier point, il faut savoir que la console permet d'utiliser plusieurs centaines ou milliers de commandes différentes et de les combiner entres elles. Les possibilités sont donc très importantes, et il est impossible de permettre de réaliser toutes les opérations avec une interface graphique aboutie.

La console, plus rapide que l'interface graphique

Exemple

Prenons l'exemple d'une opération classique qui est beaucoup plus simple à réaliser en console qu'à l'aide d'une interface graphique : compter le nombre d'image JPG qui se trouve dans un dossier, avec d'autres fichiers.

Avec l'interface graphique il faudra ouvrir son explorateur de fichier, se rendre dans le dossier en question, compter manuellement tout les fichiers en JPG (éventuellement en les triant par types), vérifier qu'on a pas fait une erreur, etc. C'est long et fastidieux.

Dans la console, ça se fait en une ligne de texte :

```
1 $ ls -l | grep -i ".jpg" | wc -l
2 386
```

Bien entendu pour le moment cela peut sembler incompréhensible tant que l'on ne connaît pas les commandes en question, mais ce sont ici en réalité des commandes très basiques (la première qui liste les fichiers, la seconde qui filtre les JPG, la troisième qui compte), que vous connaîtrez bientôt par cœur.

À retenir

La console existe historiquement car les premiers ordinateurs ne pouvaient pas avoir d'interface graphique complexe. Cette interface humain-machine s'est avérée tellement performante que l'on continue de l'utiliser au quotidien aujourd'hui. Ce n'est pas un outil réservé à une élite ou aux personnes qui "connaissent l'informatique", et elle s'avère très pratique pour réaliser certaines des tâches classiques sur un ordinateur.

2. La console sous Linux

Objectifs

- Savoir ouvrir une console sur sa machine
- Savoir lire un prompt

Ouvrir un terminal

Sur Ubuntu, on utilise l'application "Terminal" pour ouvrir une console (le terme `terminal` est équivalent à `console`). Une fenêtre s'ouvre, nous somme dans la console.

Comme prévu, c'est minimaliste : du texte, un rectangle indiquant que l'on peut écrire au clavier, et rien de plus.

Le prompt

Az Définition

La première chose que l'on constate est ce qui s'appelle le prompt. C'est le texte qui se trouve à gauche de de la zone pour entrer la commande, et qui indique le contexte dans lequel on se trouve.

Le prompt se compose de en 3 parties principales :

- le nom d'utilisateur
- le nom de la machine
- le dossier dans lequel on se trouve

Le nom d'utilisateur et le nom de la machine sont séparés par le caractère `@`, le caractère `:` permet de séparer le nom de la machine et le nom du dossier.

Exemple

Dans notre cas le prompt est :

```
kyane@europa:~$
```

Ce qui signifie donc que :

- l'utilisateur est `kyane`
- la machine se nomme `europa`
- le dossier dans lequel on se trouve est ~ (c'est un raccourci pour dire que l'on est dans le répertoire de l'utilisateur)

Le symbole `$` à la fin signifie simplement que l'utilisateur est "normal", par opposition à un super-utilisateur (ou administrateur) symbolisé par le caractère #.

Une autre manière d'accéder à la console

Complément

En réalité, ouvrir l'application "Terminal" dans notre environnement graphique n'est pas la manière originelle d'utiliser la console. En effet, même si nous avons une interface textuelle, cela reste un outil qui fonctionne en mode graphique.

Linux (et en fait la plupart des OS basés sur Unix) propose aussi d'accéder à une "vraie" console, dans le sens où c'est uniquement une interface texte, en dehors de l'interface graphique. Pour cela il faut utiliser les touches `CTRL + Alt + F1` (on peut utiliser aussi F2, F3, etc jusqu'à F6), qui bascule notre écran de l'interface graphique à une console. Pour revenir en arrière, on fait `CTRL + Alt + F7`.

Dans la pratique on utilise pas cette console sur un ordinateur de bureau, cela s'avère pratique pour des serveurs par exemple (qui n'ont souvent pas de logiciels pour gérer une interface graphique).

À retenir

Sur Ubuntu, on peut ouvrir très simplement une console avec l'application "Terminal". On se retrouve ainsi avec une interface textuelle dans la laquelle on pourra entrer des commandes. Mais avant cela, il est important de savoir déchiffrer le prompt, ce petit bout de texte dans le terminal qui nous permet de connaître le contexte dans lequel on se trouve.

3. Agir en tant que root (su / sudo)

Un système Linux est généralement utilisé par plusieurs utilisateurs (ou *user*) identifiés par un compte.

Tous les systèmes ont en commun le premier de ces utilisateurs : *root*, l'administrateur du système qui possède tous les droits.

su — Syntaxe

Pour se connecter en tant qu'utilisateur *root* (et pouvoir faire les opérations qui lui sont réservées) : `su root`.

- Le mot de passe de l'utilisateur *root* est demandé par le système.

sudo — Syntaxe

Pour exécuter une commande en tant que *root* sans changer d'*user* : `sudo` *`commande`*.

- Le mot de passe de l'utilisateur lançant `sudo` est demandé par le système.
- Seuls certains utilisateurs appelés *sudoers* ont le droit d'exécuter la commande `sudo`.
- Pour ajouter un utilisateur dans la liste des *sudoers*, exécuté en tant que *root* (ou via un autre utilisateur *sudoers*) :
  - `adduser user sudo` (en tant que *root*)
  - `sudo adduser user sudo` (avec un utilisateur déjà membre des *sudoers*)

Debian (sudo) — Attention

Sous Debian la commande `sudo` n'est pas installée par défaut.

Ubuntu (su) Attention

Sous Ubuntu l'utilisateur *root* n'est pas actif et il n'est donc pas possible d'exécuter la commande `su root`. On peut en revanche utiliser `sudo`, l'utilisateur créé à l'installation est membre des *sudoers*.

Pour activer le compte *root*, il faut lui attribuer un mot de passe avec la commande : `sudo passwd root`.

Agir en tant qu'un autre utilisateur Complément

Pour agir en tant qu'utilisateur *user*, exécuter :

`su user`, ou `su - user` pour adopter l'environnement de *user*.

4. Installation d'applications sous Debian et Ubuntu (apt-get)

Pour installer une application, exécuter la commande suivante en tant que *root* ou via *sudo* :

`sudo apt-get install` *`applications`*

Installer virtualbox, lftp, cups-pdf, pdftk, dia, gimp et git Exemple

```
1 sudo apt-get install virtualbox lftp cups-pdf pdftk dia gimp git
```

Maintenir le système à jour Complément

Pour mettre le système à jour, exécuter les deux commandes suivantes en tant que *root* ou via *sudo* :

1. `sudo apt-get update`
2. `sudo apt-get upgrade`