

République algérienne démocratique et populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Dr. Moulay Tahar de Saïda

Faculté de Sciences

Département de Chimie



Polycopié de cours : Informatique pour la chimie

Filières : 3^{ième} année chimie physique

Réalisé par Mr : Hadji Djebar

2018-2019

Introduction

Dans ce polycopié de cours, je présente une partie du module de l'informatique pour la chimie. Il est destiné aux étudiants des troisièmes années de la licence chimie physique de la faculté des sciences. Il est conforme au programme du ministère de l'enseignement supérieur et de la recherche scientifique. Ces cours seront développés et complétés par des séries d'applications et même d'exercices détaillés, aussi on fait des travaux pratiques sous Linux utilisant les deux distributions Ubuntu et CentOS pour permettre une bonne assimilation des diverses notions des fichiers, des principaux répertoires de Linux, le mode Vi et les commandes principales et les plus utilisées du système et de l'éditeur de texte sous Linux Vi. Ce polycopié est le fruit de mes efforts, que ce document est personnel, et cite en référence toutes les sources utilisées.

**« Mon ordinateur, j'essaie de faire tout ce qu'il me dit mais lui ne fait rien de ce que je
veux. »**

Anne Roumanoff, Humoriste



Table des matières

Introduction.....	8
Chapitre 1 : Les systèmes d'exploitations.....	9
1.1 Introduction.....	9
1.2 Windows.....	10
1.2.1 Définitions.....	10
1.2.2 Caractéristiques de Windows.....	10
1.2.3 Historique des différentes versions de Windows.....	11
1.3 Linux	14
1.3.1 Définitions.....	15
1.3.2 Caractéristiques de Linux.....	16
1.3.3 Les répertoires principaux de Linux.....	18
1.3.4 Les distributions majeures de Linux.....	24
1.3.5 Notion de distribution.....	25
Chapitre 2 Les commandes principales de Linux.....	28
2.1 L'interface SHELL.....	28
2.2 Saisie et exécution d'une commande.....	29
2.3 Les commandes de base.....	30
2.3.1 Commandes de manipulation de répertoires.....	30
2.3.2 Commandes de manipulation de fichiers.....	32
2.3.3 Commandes de connexion et d'administration.....	39
2.3.4 Commandes d'information sur le matériel et le système.....	41
2.3.5 Commandes d'arrêt du serveur.....	41
2.4 Exemples d'application.....	43

Chapitre 3 : L'éditeur Vi.....	48
3.1 Intérêt de Vi.....	48
3.1.1 Introduction.....	48
3.2 Les deux modes Vi.....	49
3.2.1 Mode commande.....	50
3.2.2 Mode insertion.....	50
3.3 Exemples d'application.....	50
3.3.1 Sauvegarder un fichier et quitter.....	51
3.3.2 Autres commandes de Vi.....	51
Références.....	53

Liste des tableaux

2.1 Quelques options de la commande linux rm	35
2.2 Options de la commande shutdown	42
3.1 Commandes de base du mode Vi	51

Liste des figures

1.1 Fenêtre de Windows au démarrage du système.....	10
1.2 Les distributions les plus connus de Windows.....	13
1.3 Le fondateur de système d'exploitation Linux (Linus Benedict Torvalds).....	14
1.4 Les principaux répertoires de Linux.....	19
1.5 Les commandes principales dans le répertoire /bin.....	20
1.6 Les principaux répertoires de Linux visualisés en mode graphique.....	22
1.7 Les principaux répertoires de Linux visualisés avec le terminale.....	23
1.8 Les deux utilisateurs portés par le répertoire /home.	24
2.1 Exemple d'un terminal vide Linux.....	28
2.2 L'exécution de la commande Shutdown -r now.....	42
3.1 Exécution de la commande :wq.....	52
3.2 Exécution de la commande :w monfichier.....	52

Introduction

Dans ce cours, nous allons aborder quelques notions nouvelles en informatique et spécialement en système d'exploitation pour nos étudiants en troisième année licence chimie : option chimie physique, tel que les commandes Linux et le mode Vi. Dans cette polycopie je présente la première partie du module d'informatique en chimie. Ou je présente l'essentiel des notions des systèmes d'exploitation (Windows et Linux), ensuite, je base sur le système d'exploitation Linux, leurs répertoires principaux, et leurs commandes. Le dernier chapitre a été consacré sur l'un des éditeurs de base en environnement Linux qui est le mode Vi.

Chapitre 1 : Les systèmes d'exploitations

1.1 Introduction

Dans un ordinateur, le système d'exploitation est le logiciel le plus important, assurant la gestion de l'ordinateur et de ses périphériques d'entrées et de sorties.

Il regroupe un ensemble de programmes qui permettent l'utilisation de l'ordinateur et de ses périphériques (écran, imprimante, clavier, etc.). Sans système d'exploitation, l'ordinateur n'est pas capable de gérer toutes ces périphériques.

Il permet tout d'abord le démarrage de l'ordinateur et est indispensable à la mise en œuvre des autres programmes présents sur le PC. Il assure donc le fonctionnement général de la machine.

Système d'exploitation se traduit par « Operating System » en anglais. Il existe plusieurs types connus des systèmes d'exploitation comme Windows et Linux. Dans une machine :

- Le système d'exploitation montre l'interface avec les différents éléments d'entrées et de sortie matériels, il assure aussi le pilotage entre les programmes et matériels.
- Affecter les ressources de la machine aux programmes ; cela concerne en particulier le (ou les) processeur(s) ainsi que la mémoire centrale.
- Synchroniser l'exécution des programmes et assurer qu'il n'y ait pas d'incompatibilité entre eux.
- Gérer les opérations d'entrée-sortie : affichage à l'écran, saisi au clavier et à la souris, lecture-écriture sur des disques, impression, etc.
- Piloter l'accès aux fichiers.
- Permettre l'accès aux réseaux. La plupart des systèmes d'exploitation d'aujourd'hui, dont Windows est le plus connu dans l'informatique personnelle, sont capables de réaliser "presque" en même temps ces différentes actions : on les qualifie de multitâches car vous pouvez lancer une impression tandis que vous êtes en train de surfer avec votre navigateur Web.
- Enfin, un système d'exploitation digne de ce nom doit prendre en compte la sécurité, au minimum en protégeant la machine par un dispositif de mot de passe. S'il est plus évolué, il permettra de sauvegarder et de récupérer des données en cas de panne ou de défaillance.

1.2 Windows

1.2.1 Définition

Windows est le système d'exploitation le plus connu, ce système est développé par la société Microsoft et destiné aux ordinateurs compatibles PC. La caractéristique principale de Windows est une gestion cohérente, normalisée, à l'aide de symboles, menus et champs de dialogue graphiques que l'on active généralement par un clic de la souris. Il n'est donc plus nécessaire de saisir les commandes manuellement, comme c'était le cas avec Linux. Le nom "Windows" provient du fait que l'on utilise des fenêtres pour représenter la surface de travail entre l'utilisateur et la machine sur laquelle on exploite les programmes d'application et les ressources machine (Figure 1). Les tâches générales, telles que l'impression et la gestion des éléments du système (disque dur, carte graphique, etc.), sont gérées centralement par Windows et mises à la disposition de tous les programmes d'application.

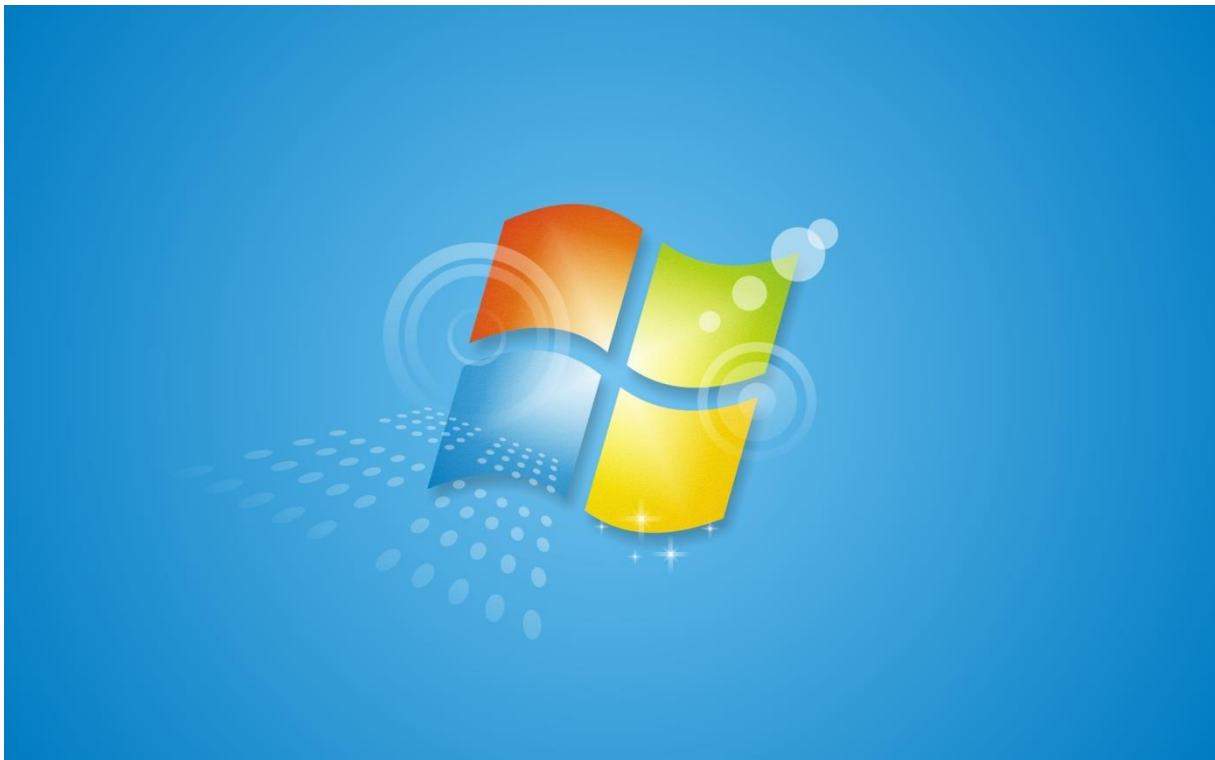


Fig. 1.1 Fenêtre de Windows au démarrage du système

1.2.2 Caractéristiques de Windows [1]

Le système d'exploitation Windows porte plusieurs propriétés et des avantages par rapport aux autres systèmes d'exploitation, on ne peut pas citer l'ensemble de ces propriétés. Parmi ces caractéristiques les plus importantes, on peut citer :

- On peut faire l'installation de Windows sur plusieurs plates-formes (Power PC, Intel, Mips, Alpha)
- Avec Windows l'installation et la configuration de réseaux est facile pour les débutants.
- La présence des interfaces graphiques conviviales et facile d'utiliser pour l'ensemble des distributions Windows.
- Lors de l'utilisation de Windows, on parle la simplicité d'utilisation. Même les débutants peuvent utiliser Windows directement. Par contre Linux, il impérativement d'apprendre quelque commande de base pour une meilleur utilisation de celui-ci.
- On peut dire que le système d'exploitation Windows est système sécurisé après une installation complète d'un antivirus fiable, aussi il faut bien noter qu'il faut la mise à jour automatique de cet antivirus si on veut obtenir une meilleure sécurité.
- On peut exécuter des applications DOS sous Windows.
- De même comme Linux, Windows est système d'exploitation multiutilisateur et multitâches.

Parmi les point faibles de Windows, on peut cite les points suivantes :

- Manque d'un langage de commande, d'où impossibilité d'écrire des scripts complexes.
- Manque de sécurités pour l'ensemble des fichiers par rapport à Linux, où on peut sécuriser nos fichiers utilisant des commandes spéciales comme le cas avec la commande chmod.
- Il est obligatoire d'installer un antivirus pour protéger la machine, par contre dans le cas de Linux, l'installation de l'antivirus n'est pas obligatoire.

1.2.3 Historique des différentes versions de Windows :

Dans ce tableau on rassemble l'ensemble des distributions les plus connus de Windows, la date de sortie et le nom de la distribution Windows

Date de sortie : Version (Nom de code)

Novembre 1985 : Windows 1.0

Décembre 1987 : Windows 2

Décembre 1987 : Windows 2.10 pour 286

Décembre 1987 : Windows 2.10 pour 386

Windows 2.11

Mai 1990 : Windows 3.0

Avril 1992 : Windows 3.1 (Janus)

Octobre 1992 : Windows for Workgroups 3.1 (Sparta)

Août 1993 : Windows NT 3.1

Novembre 1993 : Windows for Workgroups 3.11 (Snowball)

Windows for Workgroups 3.2 (en chinois uniquement)

Septembre 1994 : Windows NT 3.5 (Daytona)

Juin 1995 : Windows NT 3.51

Août 1995 : Windows 95 (Chicago)

Août 1996 : Windows NT 4.0

Novembre 1996 : Windows CE 1

Novembre 1997 : Windows CE 2

Juin 1998 : Windows 98 (Memphis)

Juillet 1998 : Windows CE 2.1

1999 : Windows CE 3

Mai 1999 : Windows 98 SE

2000 : Windows CE .NET

Février 2000 : Windows 2000 (Cairo)

Août 2000 : Windows 2000 SP1

Septembre 2000: **Windows** ME (Millennium Edition)

Mai 2001 : Windows 2000 SP2

2001 : **Pocket** PC 2000

Octobre 2001 : **Windows** XP (Whistler, fusion du projet grand public Neptune et du projet professionnel Odyssey)

2002 : Pocket PC 2002

Juillet 2002 : Windows 2000 SP3

Septembre 2002 : Windows XP SP1

2003 : Windows Mobile 2003

Mai 2003 : Windows Server 2003

Juillet 2003 : Windows 2000 SP4

2004 : Windows Mobile 2003 SE

Août 2004 : Windows XP Starter Edition

Août 2004 : Windows XP SP2

2005 : Windows Mobile 5.0 (Magneto)

Avril 2005 : Windows XP Home Edition N (Windows XP Edition familiale)

Windows XP 64 : Avril 2005

Août 2005 : Windows XP Media Center Edition

Novembre 2006 pour les entreprises et Janvier 2007 pour les particuliers : Windows Vista (Longhorn)

2008 : Windows XP SP3

2008 - Windows Vista SP1

Avec la distribution de Windows Vista, cette version est différemment des versions précédentes, ce qui lui permet d'offrir, selon Microsoft, une meilleure protection contre les virus informatique, et la plupart des programmes malveillants.

2008 - Windows Vista R2

2009 : Windows 7 (Vienna (Blackcomb))

Parmi les points forts de cette distribution de Windows est la sécurité. Les mécanismes des protections de fichier ont très peu évolué, les innovations de sécurité dans Windows 7 sont à peine différentes de celles de Vista.



Fig . 1.2 Les distributions les plus connus de Windows.

1.3 Linux

Les premiers démarches de l'appariation de Linux a été faite en 1989, par Andrew S. Tannenbaum. Andrew S. Tannenbaum trouve que le système proposé sur son ordinateur muni d'un processeur de la famille i386 ne lui convient pas tout à fait. Il s'attèle à une tâche ardue puisqu'il se met à écrire son propre système. Après une année, il sort la version 1.1 de son système qu'il appela Minix. C'est un système d'exploitation à vocation éducative, accessible pour un étudiant. Mais très rapidement Andrew tannenbaum s'essouffle ; il n'a pas que ça à faire et puis il tout seul.

La première version de Linux est apparait en En 1991 par le jeune finlandais Linus Torvalds (le créateur de Linux). Linus Torvalds est un étudiant en informatique à l'université d'Helsinki. Il utilise la version Minix sur son ordinateur tout neuf. Il est convaincu que l'informatique doit être pour l'utilisateur un moyen de gagner du temps, un outil et non une



« **Linus Benedict Torvalds** : né le 28 décembre 1969 à Helsinki en Finlande, est un *informaticien américano-finlandais* . En 1991 (à 22 ans) Linux crée le *noyau Linux* dont il continue de diriger le *développement*. Linux est considéré comme le « *dictateur bienveillant à vie* » ».

Fig.1.3 Le fondateur de système d'exploitation Linux (Linus Benedict Torvalds) contrainte.

Linus Benedict Torvalds est basé sur la distribution Minix, il se fonde sur le système d'exploitation Minix. Après leur installation sur son ordinateur doté d'un processeur 386

s'avère trop limité pour lui. Donc, Linus Torvalds trouve que tous les systèmes (y compris Minix) auxquels il a accès pour sa machine personnelle sont soit trop coûteux, soit trop peu efficaces.

Au mois d'août 1991 est né Linux, L'Unix de Linus dans sa version 0.0.1. Cette version n'était même pas exécutable. L'archive ne contenait que les rudiments des sources du noyau et il fallait compiler Linux sous Minix pour l'utiliser un tant soit peu.

Le 5 octobre 1991, Linux devient autonome lors de la sortie officielle de la version 0.0.2. Celle-ci permet de faire tourner quelques applications GNU (logiciels libres) essentielles comme le compilateur gcc ou le shell bash. Mais Linus Torvalds réalise très bien que réitérer l'expérience d'Andrew Tannenbaum sur Minix n'est pas fiable. Il se convainc qu'il faut répartir la tâche de développement, trouver des collaborateurs et étendre ce projet à un cercle le plus large possible. Pourquoi pas la planète ? A cette époque où la guerre froide vient de se terminer, l'Internet commence à envahir les universités et amorce l'explosion que nous connaissons. Linus Torvalds en profite pour lancer un appel qui est resté célèbre, sur le forum comp.os.minix et décide de mettre le code source sous licence GPL de telle sorte que tout le monde peut alors participer au développement de Linux.

1.3.1 Définition

Sur la première page des livraisons régulières de la revue « Linux Magazine France », on peut trouver une bande rectangulaire portant en titre la question « Qu'est-ce que Linux », avec en bas la réponse qui suit :

« Linux est un système d'exploitation complet, multiutilisateur et multitâche, L'installation du système peut être faite sur plusieurs plates-formes (Sparc, Intel, Apple, DEC, etc.).

Linux est conforme à la norme POSIX et peut fonctionner sans problème avec d'autres systèmes d'exploitation comme Apple, Microsoft et Novell.

Parmi les points positifs de Linux par rapport aux autres systèmes d'exploitations est un système libre. Il peut être copié et redistribué gratuitement. Le code source est également disponible pour tous de la même manière. »

1.3.2 Caractéristiques de Linux

La sécurité

Pour n'importe quel système d'exploitation, la sécurité est nécessaire pour protéger les données des utilisateurs mais aussi pour éviter de servir de point de départ à une attaque sur internet (réseau en générale). Par ce que la majorité d'attaques par des virus sont référencier au réseau internet. En effet la protection de la machine (micro-ordinateur, pc portables... etc) prend une importance grandissante tout particulièrement pour les machines et spécialement des systèmes d'exploitation qui sont directement reliés au réseau internet. Aussi, a cause du développement rapide des machines et d'outil informatique, et la propagation rapide des programme malveillants (virus) qui apparaissent à l'air de l'internet.

Dépendant de leur stratégie de fonctionnement, Linux offre une très bonne garantie de sécurité. D'abord il est conçu dès le départ pour fonctionner en réseau. Où leur mode de développement est basé sur la démarche et les règlements de ce système pour bien fonctionner sur le réseau. Chaque erreur découverte est directement corrigée par le système, et n'attend pas la publication d'un correctif par l'éditeur original du logiciel. Le système lui-même est conçu dès le début pour corriger ces erreurs. C'est pour cette raison et d'autres, on parle et confirme la sécurité et même fiabilité de Linux, cette caractéristique est discutée dans le point suivante.

Fiabilité et stabilité

La fiabilité de Linux est bien marquée lors de leur utilisation, en effet, un effort particulier a été fait lors du développement de Linux pour s'assurer que le système soit fiable et stable :

- Les applications sont cloisonnées et ne peuvent faire planter le système entier ;
 - Les bogues découverts (ou on parle les erreurs) sont corrigés très rapidement et automatiquement, une fois une erreur est signalée, le système lui-même corrige cette erreur.
- Lors de consultation du répertoire /tmp, on trouve que ce répertoire est toujours presque vide, par contre au Windows, le répertoire des fichiers temporaires est toujours encombré

Simplicité

Lorsqu'on parle la simplicité de Linux, ici on parle sur les gens expérimentés dans Linux. Ce sont des utilisateurs à des expériences pour des longs temps de travail et des essaies avec ce

système. De grands progrès ont été réalisés ces dernières années pour rendre le système accessible à tous.

L'abondance de la documentation et aussi des pages web détaillées sur Linux (leurs caractéristiques, leurs commandes, leurs répertoires, et leurs applications) permet à tous de résoudre la plupart des problèmes les plus courants ;

La grande similarité de principes entre la plupart des composants du système facilite la tâche de compréhension des bases du système. Ce n'est pas facile d'apprendre Linux en quelques jours ou quelque mois, par contre, il faut pour tirer pleinement profit du système, de passer un peu de temps pour comprendre ces bases

Avec une bonne configuration, et aussi la présence du mode graphique comme celle de Windows pour les dernières distributions de Linux (à partir de la distribution RedHat), ce dernier se révèle plus simple que la plupart des autres systèmes d'exploitation dans le cadre de la maintenance quotidienne. D'où un gain de temps appréciable.

Sa conception multi-utilisateur comme Windows aussi rend possible et même facilite la mise en place d'un environnement distinct pour chacune des personnes et des utilisateurs amenées à utiliser la machine sous Linux. Cet environnement peut être personnalisé, empêchant un utilisateur d'accéder aux données des autres. Dans ce dernier point on parle la fiabilité et au même temps la sécurité de Linux. C'est d'ailleurs par défaut le cas.

Il existe malheureusement certains handicaps spécifiques qui ont jusqu'à présent freiné la généralisation de Linux comme plate-forme universelle de travail comme Windows par exemple

Manque de visibilité, en effet, des machines préinstallées avec Linux vont être prochainement commercialisées, mais ce n'était pas le cas jusqu'à présent avec Linux comme le cas de Windows où la présence des machines directement installées et bien configurées avec le système d'exploitation Windows. Linux est juste concerné par des gens où leur utilisation est obligatoire dans quelques cas.

L'abondance du système d'exploitation Windows, ou leur position et leur situation de quasi-monopole sur le créneau des machines de travail rendent l'implantation très difficile d'un système nouveau (Linux) par exemple, même ce système est fortement innovant face à Windows. Et donc, il est impérativement à les utilisateurs doivent changer leurs habitudes

et/ou convertir leurs données. Des formations spécifiques et des réadaptations doivent être obligatoires pour de nombreuses catégories de personnels pour rendre le système Linux accessible à tous.

D'après les caractéristiques les plus connues de Linux ; un système très puissant, multitâches, multi-utilisateurs, multiplateformes, muni de nombreux dispositifs de protection. La philosophie générale de fonctionnement est donc assez différente des systèmes grand public les plus répandus comme Windows, ce qui peut dérouter les novices. Il en résulte nécessairement certaines contraintes d'utilisation. Ces contraintes augmentent les difficultés d'usage inhérentes de Linux, pour l'utilisateur final.

Aussi la présence des interfaces graphiques moins gourmandes en ressources et restent tout à fait utilisables sur des machines anciennes possédant peu de mémoire vive (semblant être un minimum pour pouvoir utiliser sans trop de peine le système Windows). On note aussi, dans Linux la manque des applications dans le secteur éducatif ou ludique rendent Linux un système peut-être sombre pour les débutants.

1.3.3 Les répertoires principaux de Linux [2]

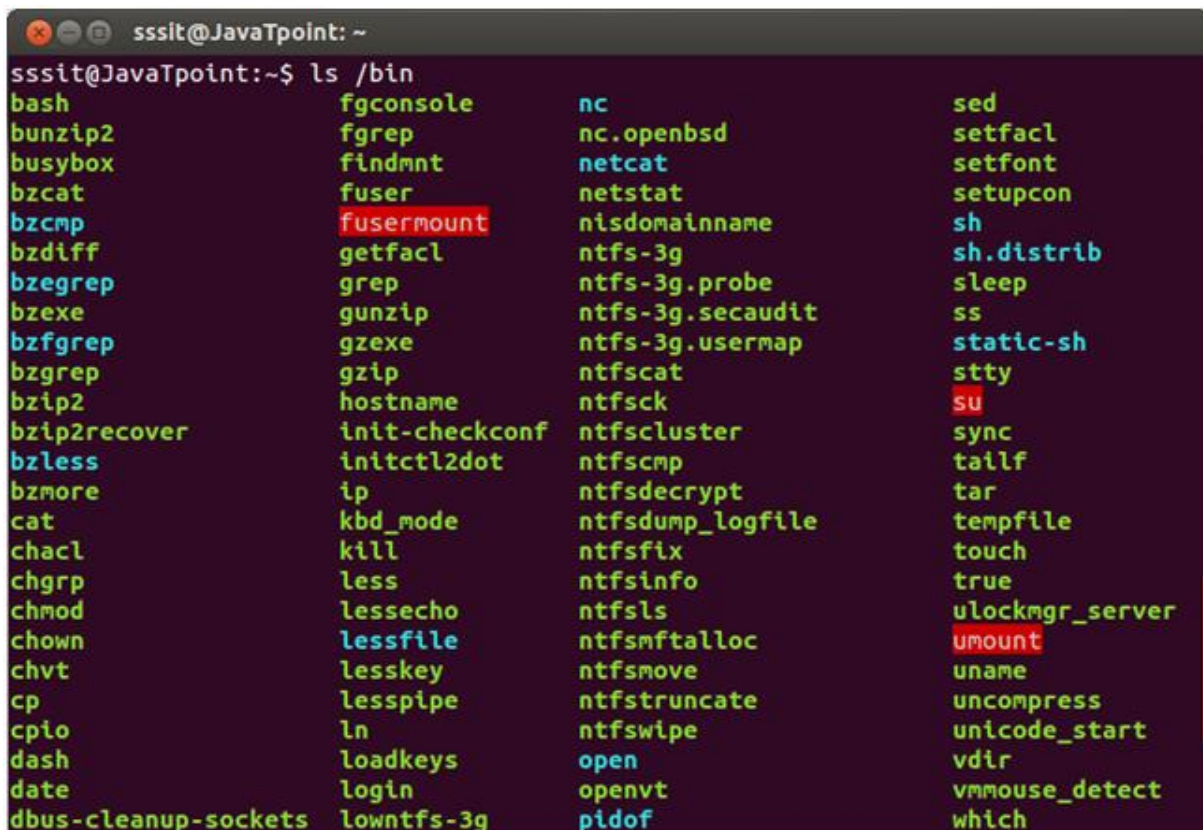
La structure des répertoires principaux de Linux

Il existe plusieurs répertoires principaux dans Linux, parmi ces répertoires, on note celle-ci qui dépendent de la racine nommée / qui sont :

/, /bin, /sbin, /usr, /home, /dev, /etc, /boot, /var, /lib, ... etc.

Les principaux répertoires dans Linux sont bien organisés comme une unique arborescence. Il y a une partition notée / comme il est signalé précédemment nommée la racine. On Windows on note c:\. D'autres partitions, par exemple le cas des DVD ou des usb, etc., peuvent être "montés" dans des répertoires. Par exemple, sur la machine que j'utilise en ce moment, le contenu du CD-ROM est accessible dans le répertoire /mnt/cdrom. Une fois les différents systèmes de fichiers montés, leur utilisation est transparente, sauf dans quelques cas particuliers : impossibilité d'écrire sur un CD-ROM... Bien souvent, sur les systèmes importants, les utilisateurs ne savent même pas dans quelle machine se trouve le disque dur qu'ils utilisent.

/bin : Contient les commandes de base.



```
sssit@JavaTpoint: ~  
sssit@JavaTpoint:~$ ls /bin  
bash fgconsole nc sed  
bunzip2 fgrep nc.openbsd setfacl  
busybox findmnt netcat setfont  
bzip2 fuser netstat setupcon  
bzip2 fusermount nisdomainname sh  
bzdiff getfacl ntfs-3g sh.distrib  
bzegrep grep ntfs-3g.probe sleep  
bzexe gunzip ntfs-3g.secaudit ss  
bzfgrep gzexe ntfs-3g.usermap static-sh  
bzgrep gzip ntfsck stty  
bzip2 hostname ntfscluster su  
bzip2recover init-checkconf ntfsdump_logfile sync  
bzless initctl2dot ntfsfix talft  
bzip2more ip ntfsdecrypt tar  
cat kbd_mode ntfsdump_logfile tempfile  
chacl kill ntfsfix touch  
chgrp less ntfsinfo true  
chmod lessecho ntfsinfo ulockmgr_server  
chown lessfile ntfsinfo umount  
chvt lesskey ntfsmove uname  
cp lesspipe ntfsmove uncompress  
cpio ln ntfsmove unicode_start  
dash loadkeys open vdir  
date login openvt vmouse_detect  
dbus-cleanup-sockets lowntfs-3g pidof which
```

Fig. 1.5 Les commandes principales dans le répertoire /bin

/boot : Contient les informations nécessaires au démarrage de la machine.

/dev : Contient les fichiers spéciaux correspondant aux périphériques.

/etc : Contient la plupart des fichiers de configuration.

/home : Contient les répertoires personnels des utilisateurs. Par exemple, l'utilisateur Med a généralement pour répertoire /home/Med.

Chaque utilisateur du système aura un répertoire dans /home portant son nom. Ce répertoire /home porte les répertoires de travail des utilisateurs Linux.

/lib : Contient les principales bibliothèques partagées (équivalent des DLL de Windows).

/mnt : Les répertoires utilisés pour monter temporairement un système de fichiers (disquette, CD-ROM...).

/opt : C'est là qu'on installe les logiciels commerciaux.

/proc : Un répertoire, dont les fichiers contiennent des infos sur l'état du système et des processus en cours d'exécution.

/root : Le répertoire de l'administrateur système (le répertoire principal de Linux).

/sbin : Contient les commandes de base nécessaires à l'administration système (vérification et réparation des disques, mise en place du réseau...). Ces commandes sont exécutées par l'administrateur de la machine et pas les autres utilisateurs.

/tmp : Contient les fichiers temporaires. Ce répertoire est généralement ne porte pas plusieurs fichiers temporaires comme dans le cas de Windows, ou on trouve ce répertoire encombrés de fichiers temporaires. On parle de la fiabilité du système d'exploitation Linux par rapport au Windows. En effet, dans la vaste majorité des distributions Linux ce répertoire, est nettoyé à chaque redémarrage.

/usr : Les logiciels installés avec le système.

/usr/X11R6 : Xwindows.

/usr/bin : Contient les exécutable. Les programmes installés par le système d'exploitation se trouvent dans /usr/bin. Les programmes installés manuellement par les utilisateurs se trouvent dans /usr/local.

/usr/dict : Les dictionnaires (pour les correcteurs d'orthographe).

/usr/doc : La doc.

/usr/etc : Contient des fichiers de config.

/usr/games : Les jeux.

/usr/include : Contient les fichiers d'en-tête pour la programmation.

/usr/info : Contient la doc au format GNU info.

/usr/lib : Les DLL non vitales.

/usr/man : Le manuel en ligne. Les fichiers sont compressés.

/usr/sbin : principalement les serveurs réseau.

/usr/share : des fichiers de données.

/usr/src : les sources de certains logiciels, principalement le noyau de Linux.

/var : Contient des données fréquemment réécrites.

/var/run : Principalement des infos sur les serveurs en fonctionnement.

/var/log : Le journal du système.

/var/lib : Des bases de données, des fichiers de config...

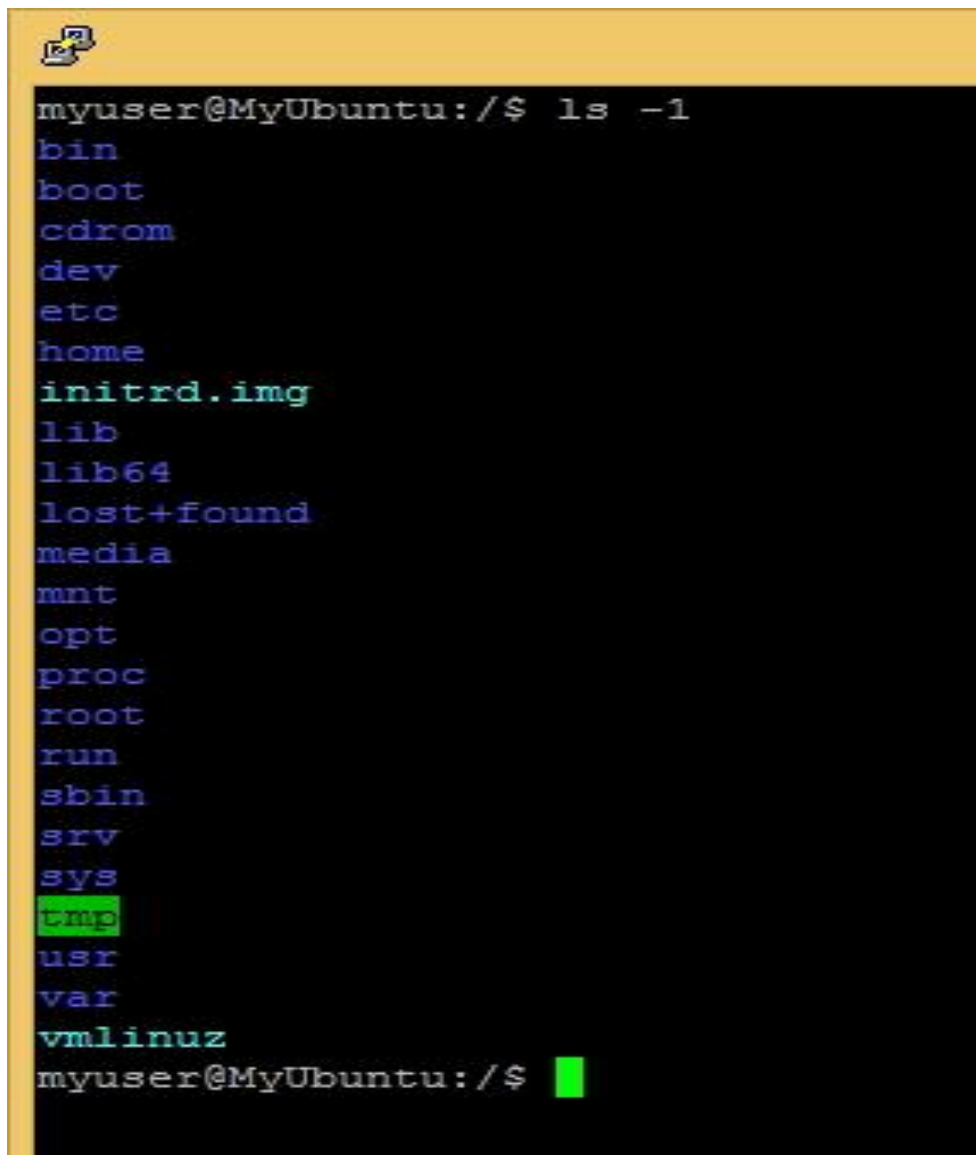
/var/tmp : Des fichiers temporaires.

Les travaux d'impression seront dans /var/spool.



Fig. 1.6 Les principaux répertoires de Linux visualisés en mode graphique.

Le répertoire racine / le symbole barre oblique (/) porte les plus importants répertoires dans linux, ce répertoire est le niveau le plus élevé du système de fichiers. Dans la figure suivante (Figure 7), on affiche le contenu du répertoire racine dans un terminal :

A terminal window with a yellow title bar and a black background. The prompt is 'myuser@MyUbuntu:/\$'. The command 'ls -l' has been executed, displaying a list of files and directories in the root directory. The output is as follows:

```
myuser@MyUbuntu:/$ ls -l
bin
boot
cdrom
dev
etc
home
initrd.img
lib
lib64
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
vmlinuz
myuser@MyUbuntu:/$
```

Fig. 1.7 Les principaux répertoires de Linux visualisés en mode texte.

Le répertoire /home porte les répertoires personnels pour chaque utilisateur dans linux, on peut afficher le contenu de ce répertoire avec les commandes suivantes :

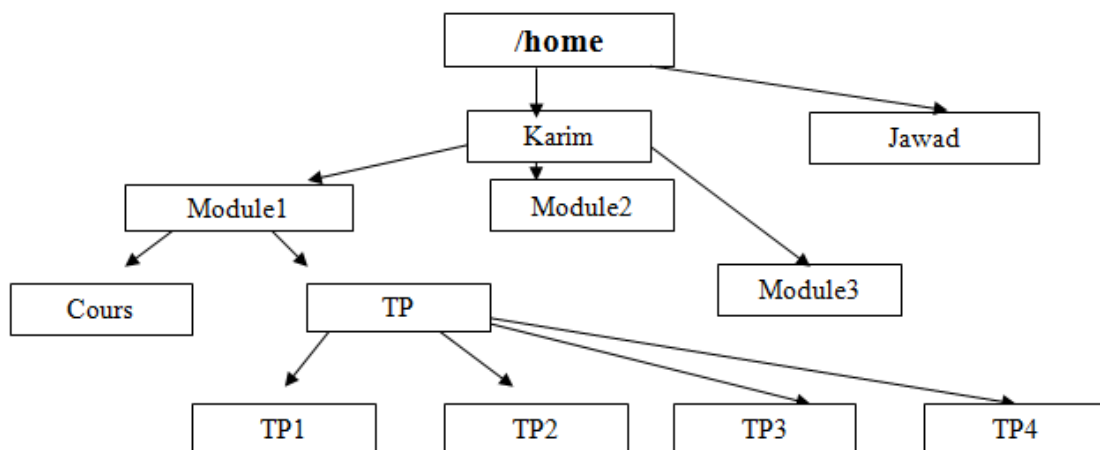


Fig. 1.8 Les deux utilisateurs portés par le répertoire /home.

Dans cet exemple le répertoire /home porte les documents pour les deux utilisateurs Karim et Jawad, l'utilisateur Karim (est considéré comme un enseignant universitaire) a trois répertoires nommés Module 1, Module 2 et Module 3, et même, les répertoires Module 1 et Module 2 portent les deux répertoires Cours et TD ... etc. On peut afficher le contenu du répertoire /home en détail utilisant la commande ls avec leurs options.

Si on prend l'exemple du répertoire /bin dans Linux, ce répertoire porte l'ensemble des commandes Linux les plus importants. Ces commandes sont des fichiers exécutables. Ils sont directement exécutables par le système. Tous les utilisateurs peuvent utiliser les commandes de ce répertoire par l'intermédiaire d'une console, par contre dans le cas du répertoire /sbin, ces commandes sont exécutées par l'administrateur du système.

1.3.4 Les distributions majeures Linux

L'abondance des applications disponibles et multiplicité des programmes ayant les mêmes objectifs c'est-à-dire disponibles pour répondre à un même besoin, pose un problème de choix pour l'utilisateur.

L'expérience l'a montré depuis longtemps : compte tenu du nombre impressionnant d'applications que l'on peut être amené à installer sur une machine, il faut absolument disposer d'un outil d'installation et de gestion de ces applications ; d'où la nécessité d'un tel outil d'installation ;

Enfin, faciliter la prise en main pour les débutants (mais aussi pour de nombreux utilisateurs non débutants) doit être une préoccupation des éditeurs et de la communauté. D'où l'importance des manuels d'installation, et autres outils de configuration du système.

1.3.5 Notion de distribution

Les distributions sont des ensembles logiciels contenant le système Linux, une interface graphique - contrairement à Ms-Windows, l'interface graphique est dissociée du reste du système - et un ensemble de programmes. En général elles disposent aussi d'un outil de gestion des applications installées et d'autres outils de configuration du système, plus orientés débutant que les méthodes de configuration traditionnelle qui s'adressent principalement aux administrateurs.

Plusieurs éditeurs de logiciels ont décidé de distribuer de tels ensembles afin de faciliter le travail de choix.

Dans la mesure où les logiciels libres peuvent être distribués à volonté, il est relativement facile de créer une nouvelle distribution, en reprenant le travail réalisé par autrui et en l'améliorant. Résultat : il existe un grand nombre de distributions dont certaines ne répondent qu'à un besoin très pointu et spécifique. Nous passons ici en revue les distributions majeures les plus diffusées.

La distribution Slackware

C'est l'ancien des distributions Linux. Pionnier dans ce domaine, la société Slackware commercialisa une distribution Linux contenue sur une centaine de disquettes (le cédérom n'étant pas aussi répandu que de nos jours).

Compte tenu de son âge, cette distribution ne disposait pas d'interfaces conviviales de configuration du système et de ce fait s'adresse à des utilisateurs expérimentés.

De plus sa méthode de gestion des paquets s'appuyé sur les fichiers archives compressés (tar.gz) est très sommaire.

La distribution Redhat

Considérée par certains comme la référence professionnelle, l'apparition de cette distribution a constitué un événement marquant dans le monde Linux.

Elle intègre un système complexe de gestion d'applications qui permet la mise à jour d'une application tout en conservant les fichiers de configuration et qui gère les dépendances.

Une dépendance est le fait qu'une application ait besoin d'une autre, ou d'une bibliothèque pour fonctionner. Ce système -Redhat Package Manager (rpm) -peut donc installer automatique tous les éléments logiciels requis pour faire fonctionner une application.

La distribution Suse

Diffusée par la société allemande SuSE créée en 1992, la première version de cette distribution sortira en 1993 sur disquettes et était basée sur la distribution Slackware. Depuis elle a beaucoup évolué. Techniquement très proche de la Redhat, elle est basée sur le même type de packages : RPM.

La distribution Debian

La distribution Debian est la seule distribution non commerciale et totalement libre et gratuite. Gérée par une association de plus de 400 volontaires bénévoles dont la majorité sont des développeurs, elle est aussi la plus complète : elle propose un très grand nombre d'applications/d'outils regroupés en plus de 2500 packages.

Son objectif est triple :

Fournir une distribution entièrement libre ;

Développer un outil de gestion d'application puissant ;

N'intégrer que des applications stables.

Points faibles de cette distribution :

Le mode graphique n'est pas privilégié : de c fait son installation et sa configuration sont relativement plus difficile ;

Les nouvelles fonctionnalités sont tardivement diffusées.

La distribution Corel Linux

Cette distribution est l'œuvre de l'éditeur canadien Corel (qui édite aussi Corel Wordperfect).

Elle est basée sur une Debian, pour laquelle une programme d'installation graphique et une amélioration de KDE ont été développés, de même qu'un nouveau gestionnaire de fichiers - Corel File Manager - qui permet par exemple d'accéder au réseau Windows par Samba sans passer par les outils complexes de configuration. La gestion de l'impression a aussi été facilitée.

La distribution Corel Linux est destinée avant tout à la bureautique, pas pour les serveurs ou les stations de travail plus professionnelles.

La distribution Mandriva

La distribution Linux-Mandriva anciennement Mandrake est une distribution récente, à l'origine basée sur La RedHat. Elle a été la première (avec sa version 5.x) à intégrer l'interface graphique KDE, conviviale et simple d'emploi, ce que refusait de faire Redhat pour des problèmes de licences aujourd'hui résolus. Un effort tout particulier a été fait au niveau de la traduction, pour la rendre utilisable par le plus grand nombre.

A partir de la version 6 Linux-Mandriva a pris son indépendance, et n'est plus basée sur la Redhat.

La distribution Linux-Mandriva offre de nombreux packages supplémentaires par rapport à la Redhat ainsi qu'une installation et une utilisation faciles.

Pour cela, elle est dotée d'outils graphiques pour toutes les étapes de l'installation (programme DrakX), pour la gestion des différents programmes (programme RpmDake), et pour l'installation de nouveaux périphériques (DrakConf).

Tous ces outils sont intégrés dans le gestionnaire de fenêtres KDE, une interface graphique intuitive, avec une traduction en français et une personnalisation du bureau KDE.

La distribution Linux-Mandriva utilise bien sûr le système de paquets RPM développé par Redhat. Mais elle ajoute une interface graphique de recherche de paquet, pour une utilisation encore plus facile, intégrée au programme RpmDrake.

Distribution française à l'origine, Linux-Mandrake est désormais polyglotte. La documentation de la plupart des applications courantes est disponible dans de nombreuses langues différentes. La procédure d'installation est disponible en plus de 50 langues à l'heure actuelle.

En conclusion, on peut dire que Linux-Mandriva est une réelle amélioration de la Redhat avec des outils d'installation et de configuration très agréables à utiliser, une distribution complète et de grande qualité qui peut convenir à la majorité des utilisateurs aussi bien débutants que professionnels. Elle est souvent recommandée et distribuée par de nombreux magazines (Programmez!, Mirror, Planète Linux, PC info, etc ...).

Chapitre 2. Les commandes principales de Linux [3]

2.1 L'interface SHELL

L'utilisateur utilise le Shell pour connecté avec Linux en mode text. Le Shell veut dire *Coquille*, qui entoure le noyau. C'est nommé aussi Terminal. C'est un interpréteur de commandes qu'on utilise pour écrire et exécuter des commandes ou exécuter une suite de commandes. L'utilisateur discute avec le Shell, qui discute avec le noyau, qui à son tour discute avec le matériel. Originellement le Shell est utilisé sous Unix, il s'est répandu depuis avec différentes versions, la forme la plus simple est sh.

La figure suivante montre un exemple d'un terminal vide Linux :

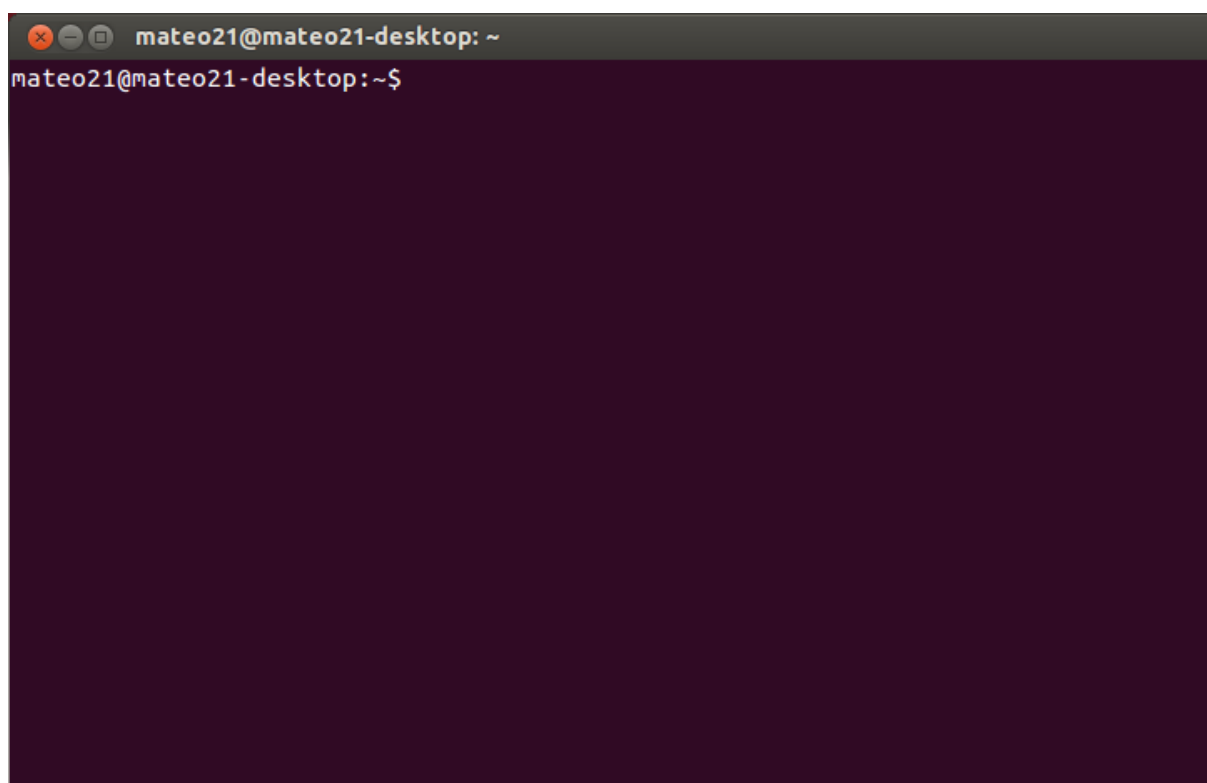


Fig. 2.1 Exemple d'un terminal vide Linux

Le Shell est constitué de petits Shells spécifiques à chaque travail, chacun ayant un fichier de configuration stocké dans le répertoire /home de chaque utilisateur, lequel peut les modifier à sa convenance :

.profile: exécuté automatiquement lors d'un Shell de connexion (à chaque login), quel que soit le Shell

Fichiers de configuration spécifiques au Shell **Bash** :

.bash_profile: exécuté automatiquement lors d'un Shell de connexion (à chaque login), C'est par exemple le cas lors de votre première connexion au démarrage de votre machine.

Un login Shell est le premier processus qui est exécuté avec l'ID de votre utilisateur. Dans ce contexte, c'est le fichier bash_profile qui est exécuté. Le fichier .bashrc est exécuté lors du lancement d'un non-login Shell, par exemple lorsque vous ouvrez un nouveau Shell dans votre machine alors que vous y êtes déjà authentifié, typiquement un nouveau terminal sur votre bureau.

.bash_logout : chargé lors de la fermeture du Shell (à la déconnexion)

.bash_history : c'est un fichier text contenant l'historique des commandes tapées.

2.2 Saisie et exécution d'une commande

La saisie à l'invite de commandes, ou au Shell, ou au Terminal, d'une commande Linux est de la forme suivante : la commande [options] [arguments]

La commande est dans la plupart des cas en minuscules. Les crochets [] indiquent le caractère optionnel. Les options sont en général précédées d'un " - " ou deux " -- ", elles servent à modifier le comportement d'une commande. Les arguments sont les paramètres sur lesquels s'applique la commande.

Dans Linux, il faut bien écrit les caractères soit en majuscules ou en minuscules pour les commandes, parce que Linux fait la distinction entre les deux cas (majuscules et minuscules) pour les commandes, noms de fichiers, répertoires, c'est parfois agaçant et source d'erreurs pour les débutants.

Par exemple :

- 1- Lorsqu'on écrit un nom de fichier Med.txt, MED.txt et med.txt sont des noms différents.
- 2- Lorsqu'on écrit une commande (ex : ls) minuscule et (ex : Ls) L en majuscule et s en minuscule sont deux commandes différents, et donc chaque commande après l'exécution à une réponse très différente par rapport à l'autre.

A l'invite de commandes, ou au Shell, ou au Terminal, l'exécution d'une commande Linux est



lancée après validation par la touche "Entrée". Avant validation vous pouvez corriger votre saisie en utilisant les touches "Flèche Gauche", "Flèche Droite", "Effacement Arrière" et



"Suppr" pour modifier le text

2.3 Les commandes de base [4-6] Pour connaître le type d'une commande, La commande `type` (voir ci-après) permet de savoir de quel type est une commande.

2.3.1 Commandes de manipulation de répertoires

ls

La commande `ls` permet de lister le contenu de répertoires, et même, permet d'obtenir beaucoup d'informations sur les répertoires.

Syntaxe : `ls [options] [répertoire1] [répertoire2]`

Cette commande a un très grand nombre d'options par rapport aux autres commandes Linux les plus connus pouvant être combinées.

`ls` liste les entrées du répertoire courant.

`ls -R` liste récursivement tous les répertoires.

`ls -a` liste tous (penser au mot "all") les fichiers du répertoire, y compris les fichiers cachés. (Les fichiers précédés par point (.) et deux points (..)) Cette option est très utile lorsque l'on se trouve dans son répertoire personnel car il contient les fichiers de configuration de l'utilisateur dont les noms commencent généralement par un point et seule l'option `-a` permet de détecter leur existence.

`ls -l` liste les entrées et affiche toutes les informations.

`ls -F` liste les entrées et ajoute / derrière les noms de répertoires.

`ls --color` liste les entrées avec une couleur différente suivant le type (fichier exécutable, text, répertoire, liens ...)

`ls -t` trie par date, c'est-à-dire en les classant du récent au plus ancien (penser à *time*).

`ls -x` trie par extension.

ls -r trie par ordre alphabétique inverse (penser à *reverse*). Cette option a la particularité d'inverser l'effet de toutes les autres options de tris requises. ls -tr affichera ainsi les fichiers par date (-t) en commençant par les plus anciens pour finir par les plus récents.

cd

La commande cd permet le positionnement sur un répertoire

Syntaxe : cd [répertoire]

Pour cette commande, il y a des options fréquentes :

cd / positionnement à la racine

cd .. positionnement sur le répertoire parent

cd /usr/X11/bin positionnement depuis la racine sur le répertoire /usr/X11/bin

cd ../lib positionnement à partir du répertoire parent sur le répertoire /lib.

cd cd~ ou cd \$HOME positionnement depuis n'importe où sur le répertoire personnel.

cd - positionnement sur le répertoire précédent.

mkdir

La commande mkdir permet la création de répertoires. Chaque répertoire contient deux entrées : le répertoire lui-même (.) et le répertoire parent (..).

Syntaxe : mkdir répertoire1 [répertoire2] [répertoireN]

mkdir med créé le répertoire med dans le répertoire courant

mkdir ../med créé le répertoire med dans le répertoire parent.

mkdir /root/docs/med créé le répertoire med dans le répertoire /root/docs

On peut créer plusieurs répertoires au même temps dans le même répertoire, il suffit de taper la commande mkdir plus les répertoires séparés avec un espace.

mkdir rép1 rép2 rép3 rép4

rmdir

La commande rmdir permet la suppression de répertoires vides.

Ça syntaxe : rmdir répertoire1 [répertoire2] [répertoire]

Pour supprimer un répertoire non vide voir la commande rm avec leurs options.

man rm permet de connaître tous les options de la commande rm.

rm -r rép1 permet la suppression du répertoire non vide rep1

2.3.2 Commandes de manipulation de fichiers

Les trois commandes `cat`, `more` et `less` permettent de lire un fichier text. La différence entre ces trois commandes est la méthode pour lire le fichier, soit page par page, soit ligne par ligne ou même avec le pourcentage % du fichier entier qui apparaît à la fin du terminal.

cat

Syntaxe : `cat fichier fichier1 [fichier2]`

Exemple d'affichage d'un fichier :

```
[root@linuxserveur /root]# cat exemple.txt
...1234567 cat commande
000014872morecommande
89547txt less commande
```

more ou less

Avec ces deux commandes, on peut afficher le contenu d'un fichier page par page, ou bien ligne par ligne utilisant la touche entrée, page par page avec "Espace", on quitte l'affichage avec "q". Des fois le pourcentage % apparaît sur la dernière ligne du terminal, ce dernier indique la position actuelle par rapport à la totalité du fichier.

Exemple :

```
[root@linuxserveur linux_logo-1.11]# more README
/*-----*\
LINUX_LOGO 2.12 -- Shows a Logo With some System Info – 15 Novembre 2004
by Vince Weaver (weave@eng.umd.edu, http://www.glue.umd.edu/~weave )
http://www.vince.weaver.org/
SUPPORTS Linux (Intel, m68k, Alpha, Sparc) and some non-Linux OS's
\*-----*/
...
The program itself has grown more "feature-full" as people around the
--More--(10%)
```

Dans cet exemple, on remarque que le pourcentage est atteint plus de 10%. C'est-à-dire, si le fichier porte 100 pages, on dépasse l'affichage de plus de 10 pages au total du fichier.

N.B. Pour savoir si un fichier peut être affiché voir la commande `file`

cp

La commande cp sert à copier des fichiers.

Syntaxe : cp [options] [rép1/]file1 [rép2/]file2

cp [options] file1 [fichier2] ... [fichier] répertoire

Avec la première syntaxe le fichier1 du répertoire1 est copié sous le nom fichier2 dans le répertoire2. Notons que répertoire1 et 2 peuvent être le même répertoire. Avec la seconde syntaxe les fichiers sont copiés dans le répertoire de destination à partir du répertoire en cours.

Exemples :

Dans l'exemple ci-dessous on utilise les deux commandes le fichier profile est copié du répertoire courant /etc vers le répertoire /root.

```
[root@linuxserveur /root]# cd /etc
```

```
[root@linuxserveur /etc]# cp profile /root
```

Dans l'exemple ci-dessous le fichier profile est copié du répertoire courant /etc vers le répertoire /root et renommé

Attention : si vous effectuez une copie d'un fichier sur un autre qui existe déjà, le contenu de ce dernier sera remplacé par celui de l'autre ! (écrasé)

Par exemple pour faire une copie d'un fichier du répertoire où l'on se trouve nommé « linux-cour » en un fichier « linux-cour2 » :

```
cp linux-cour linux-cour2
```

```
cp linux-cour /tmp
```

Pour lui donner un autre nom :

```
cp file1 /tmp/file2
```

Avec l'option -i, on peut vérifier l'existence d'un fichier de même nom ;

```
cp -i linux-cour/tmp/linux-cour2
```

Il existe plusieurs options pour cette commande, on peut le trouver avec la commande man cp, ou avec la commande cp -help

file

La commande file permet d'obtenir la nature du ou des fichiers spécifiés.

Avec cette commande, on peut connaître le type d'un fichier quelconque, doc, ps, pdf, txt, .. etc.

Syntaxe : file fichier1 [fichier2] [fichierN]

file courcommandelinux.doc

courcommandelinux.doc : Word document

mv

La commande mv (penser à *move*) permet de déplacer un fichier soit à le renommer.

Syntaxe : mv [répertoire1/]file1 [répertoire2/]file2

mv file1 [file2] ... [filen] répertoire

Dans la première forme si les répertoires sources et cibles sont les mêmes le fichier est simplement renommé. Dans l'exemple ci-dessous

le fichier file1 est renommé en file2.

```
root@linuxserveur /mnt]# ls
```

```
wordfile file1
```

```
root@linuxserveur /mnt]# cd /
```

```
root@linuxserveur /]# mv /root/ file1 /root/ file2
```

```
root@linuxserveur /]# cd /root
```

```
root@linuxserveur /]# ls
```

```
wordfile file2
```

Exemple:

mv file1 file1 écrase le fichier existant avec le fichier copié (dit « **source** »).

mv personnel mon-répertoire : renomme le répertoire « personnel » en « mon-répertoire ».

mv perso /home/delcros/mon-répertoire : déplace le fichier « perso » dans le répertoire /home/delcros/mon-répertoire

Donc : Si le ou les fichiers cibles existent déjà, ils seront écrasés purement et simplement.

rm

La commande rm est une commande parmi les commandes les plus dangereux en Linux. rm sert à supprimer des fichiers et des répertoires. Il existe des options pour cette commande.

Parmi ces options : r permet la suppression d'un répertoire même non vide

L'option -i demande confirmation avant suppression. Cette commande est très dangereuse, c'est pourquoi en général un alias est créé, forçant la demande de confirmation.

L'option -r permet la suppression des répertoires même non vide.

Une option très dangereux lors de leur application, c'est le cas de l'option -f, avec cette option, on peut supprimer les fichiers dans lesquels il est interdit d'écrire ainsi que les répertoires, sans demander une confirmation (cette commande à utiliser avec précaution...)

Tab. 2.1 Quelques options de la commande linux rm

Commande	Description
rm -r	suppression d'un répertoire même non vide
rm -i	suppression d'un répertoire même non vide
rm -f	supprime les fichiers dans lesquels il est interdit d'écrire
rm --	supprime les fichiers cachés
rm -/	supprime les fichiers cachés
rm -rf	Suppression des répertoires et son contenu, sans confirmation
rm -R	suppression enlèvement récursif

Droits d'accès (rwx)

Dans un système Linux quelconque (Ubuntu par exemple), tous les fichiers ont des droits d'accès qui permettent ou empêchent aux autres utilisateurs la visualisation, la modification ou l'exécution. Seul l'administrateur qui s'appelle le super-utilisateur (« root ») a la possibilité d'accéder à n'importe quel fichier du système. Chaque fichier a des restrictions d'accès, des restrictions d'utilisateurs et une association propriétaire/groupe.

Les fichiers sont gardés par ces trois couches de droits, par ordre d'importance :

- Utilisateur : s'applique à l'utilisateur propriétaire du fichier
- Groupe : s'applique au groupe associé au fichier
- Autres : s'applique à tous les autres utilisateurs

Dans chacun de ces trois ensembles se trouvent les droits d'accès réels. La manière dont les droits s'appliquent, qui diffère selon qu'il s'agit d'un fichier ou d'un répertoire, est décrite ci-dessous :

- Lecture : ou le contenu du répertoire peut être visualisé
- Ecriture : les fichiers peuvent être modifiés ou supprimés. C'est-à-dire le contenu du répertoire peut être modifié
- Exécution : les fichiers exécutables peuvent être lancés en tant que programme

Dans ce cas, on utilise la commande `chmod`.

Cette commande sert à changer les permissions de lecture d'écriture et d'exécution d'un fichier ou d'un dossier, aussi, il permet de modifier les droits d'un fichier quelconque. Sa syntaxe se présente comme suit :

`chmod fichier`

Exemples

Voici un exemple simple d'application de la commande `chmod`

`chmod u+x mon_fichier`

ajoute le droit d'exécution à `mon_fichier` en ce qui concerne l'utilisateur. Par ailleurs,

`chmod a-rx mon_fichier`

les droits de *lecture* (*read*) et d'*exécution* (*execute*) pour *tous* (*all*), c'est-à-dire l'utilisateur, le groupe et les autres.

Remarque importante : Il existe plusieurs options dans ce cas. Il bien connaître le rôle de chaque option avant l'exécution de la commande `chmod`

Pour savoir quel droit a un fichier, un dossier ou autre, on cible un répertoire qui porte le fichier dans un premier temps. L'étape suivante c'est que l'on utilise la commande `ls -la` dans le dossier père de votre cible.

Exemple : Quels sont les droits des dossiers et fichiers qui se trouvent dans `/boot/`

```
[Ubuntu@localhost ~]$ cd /boot/
```

```
[Ubuntu @localhost boot]$ ls -la
total 7070
drwxr-xr-x  3 root root   4096 sep  8 11:41 .
drwxr-xr-x 23 root root   4096 sep 13 06:39 ..
-rw-r--r--  1 root root 56991 jun  3 01:09 config-2.6.10-1.1347_FC4
-rw-r--r--  1 root root 56927 mar 20 02:45 config-2.6.12-1.13x47_FC4
drwxr-xr-x  2 root root   4096 sep  8 17:42 grub
-rw-r--r--  1 root root 1107313 oct  8 12:26 initrd-2.6.11-1.1369_FC4.img
-rw-r--r--  1 root root 1109676 oct  8 17:42 initrd-2.6.12-1.1447_FC4.img
-rw-r--r--  1 root root 764778 jun  3 05:03 System.map-2.6.11-1.1119_FC4
-rw-r--r--  1 root root 767665 août 27 02:45 System.map-2.6.12-1.1447_FC4
-rw-r--r--  1 root root 1639539 jun  3 05:03 vmlinuz-2.6.11-1.1119_FC4
-rw-r--r--  1 root root 1641264 mar 27 02:45 vmlinuz-2.6.12-1.1429_FC4
[Ubuntu@localhost boot]$
```

L'option ou le caractère **d** n'indique pas un droit, mais il indique que le fichier est un répertoire: (**d**irectory = répertoire). Pour les fichiers, archives... aucune lettre n'est placée devant les permissions (rwx (readwriteexecution)). Pour les liens, il y aura un **l** (comme lien)

1er groupe : -rwx----- Propriétaire

2ème groupe: ----rwx--- Groupe

3ème groupe : -----rwx Public (autres utilisateurs)

- **r** = read (lire)
- **w** = write (écrire)
- **x** = execute (executer)

Les permissions avec les chiffres

Les quarts chiffres les plus important pour décrire et connaître ces droit de lecture , écriture et exécution sont :

Les chiffres sont les suivants :

"4" pour le droit de lecture (**r**ead)

"2" pour le droit d'écriture (**w**rite)

"1" pour le droit d'exécution (execute)

Et ce n'est pas tout, il faut faire un peu de mathématiques. Il suffit d'additionner les chiffres entre eux, exemple pour pouvoir avoir le droit de :

- **lecture** et celui d'**écriture** : vous additionner : $2 + 4 = 6$,
- Pour avoir tout les droits (exécution, lecture, écriture) : $1 + 2 + 4 = 7$

Puis, il faut comprendre que :

- Le premier chiffre correspond aux autres utilisateurs,
- Le deuxième chiffre correspond au groupe,
- Le troisième correspond à l'utilisateur.

Exemple: si on veut que l'utilisateur obtienne tout les droits ($1 + 2 + 4=7$), que le groupe ait juste le droit de lire (4) et que le reste des utilisateurs n'aient aucun droit (0): le nombre sera 740, donc, dans le terminal, que l'on s'appelle l'invite du Shell, vous taperez donc :

```
chmod 740 fichier
```

Si on veut l'appliquer ces droits à un répertoire, on ajoutera **-R** à la commande, exemple :

```
chmod -R 750 repertoire
```

Les permissions avec les lettres

Vous devrez ajouter soit read, write ou execution (sont en anglais)

r pour read, donc le droit de lire votre cible.

w pour write, donc le droit d'écrire sur la cible.

x comme execution, donc le droit d'exécuter votre cible.

Il ne faut pas oublier que la commande chmod est une commande dangereuse dans quelques cas, puisqu'elle s'exécute généralement en root puisque l'utilisateur root a tout les droits.

2.3.3 Commandes de connexion et d'administration [7]

who

Cette commande permet de savoir qui est connecté à votre machine. Avec l'option « -q » le nombre d'utilisateurs connectés est fourni :

```
[colling@linuxserveur /]$ who -q  
root root colling  
# users=3
```

Sinon la commande fournit, entre autres, l'identité et le type de connexion. Les commentaires (#) ont été ajoutés.

```
[colling@linuxserveur /]$ who  
root tty1 Mar 18 11:42 # root depuis une console.  
root tty1 Mar 18 11:46 (:0.0) # root via xterm.  
colling tty0 Mar 18 13:34 (172.28.47.2) # colling via telnet.
```

whoami

La commande whoami ou on dit : qui je suis actuellement ?

Cette commande affiche le nom d'utilisateur associé à l'U-ID effectif en cours. C'est à dire vous donne votre identité. Cette commande très utile en phase d'essais, par exemple, lorsque vous travaillez sur plusieurs comptes.

```
[etudiant@linuxserveur /]$ whoami  
etudiant
```

passwd

passwd cette commande permet à l'utilisateur de changer son mot de passe. Comme on peut le constater dans la séquence reproduite ci-dessous,

Linux effectue un contrôle syntaxique du mot de passe.

Exemple

```
[etudiant@linuxIG med]$ passwd  
Changing password for etudiant  
(current) UNIX password:.....  
New UNIX password:.....  
BAD PASSWORD: it's WAY too short
```

```
New UNIX password:.....  
BAD PASSWORD: is too similiar to the old one  
New UNIX password:.....  
Retype new UNIX password:.....  
passwd: all authentication tokens updated successfully
```

Avec cette commande passwd, l'administrateur peut changer le mot de passe d'un utilisateur

```
[root@linuxIG /root]# passwd phil
```

```
New UNIX password:
```

```
Retype new UNIX password:
```

```
passwd: all authentication tokens updated successfully
```

```
[root@linuxIG /root]#
```

Ici on remarque que l'administrateur utilise le répertoire root, ce qui normal avec l'administrateur, parce que seul l'administrateur du système peut modifier le mot de passe de lui-même, et même, le mot de passe des autres utilisateurs.

Dans les fichiers /etc/passwd, /etc/shadow et /etc/group, on trouve l'ensemble des informations y apparaissent en clair mais naturellement les mots de passe sont cryptés.

su

La commande su permet à tout utilisateur de passer en mode Super Utilisateur. Avec l'option "- " les fichiers profile sont pris en compte.

Il suffit d'entrée le mot de passe super utilisateur ou le mot de passe de l'administrateur système.

Exemple

```
password:
```

```
Last login: Wed Jan 27 14:45:49 from 172.28.47.2
```

```
[etudiant@linuxserveur etudiant]$ su -
```

```
Password:
```

```
[root@linuxserveur /root]#
```

Donc on passe en mode super utilisateur, on remarque la présence du répertoire /root, seul l'administrateur système peut modifier cette mot de passe.

2.3.4 Commandes d'information sur le matériel et le système [8, 9]

uname

La commande uname fournit un certain nombre d'informations système.

Syntaxe : `uname [options]`

Parmi les options de cette commande :

`uname -s` affiche le nom du système d'exploitation (OS).

`uname -r` affiche le numéro de version du système d'exploitation (OSV).

`uname -a` affiche les informations affichés par le commandes ci-dessus et d'autres.

```
[root@linuxserveur home]# uname -sra
```

```
Linux linuxserveur.ac-nancy-metz.fr 2.0.36
```

df

La commande df (disk free) sert à affiche l'espace libre sur les systèmes de fichiers montés.

Exemple :

```
[root@linuxserveur /root]# df
```

File system	1024-blocks	Used	Available	Capacity	Mounted on
/dev/hda1	3678932	743120	2785212	13%	/
/dev/fd0	1390	14	1304	1%	/mnt/disque

2.3.5 Commandes d'arrêt du serveur

Il faut noter dans un premier temps que seul l'administrateur système peut exécutés l'ensemble de ces commandes qui permet d'arrêter ou redémarrer le système d'exploitation.

Il existe deux commandes principales pour arrêter le système d'exploitation (OS) Linux, l'un c'est la commande `halt`, et l'autre c'est la commande `shutdown`.

Plusieurs options peuvent l'utiliser avec ces deux commandes, on trouve par exemple :

`halt` ou `shutdown -n now`

Lors de l'application de ces deux commandes, on passe à un arrêt immédiate du système et donc de la machine.

`shutdown -h` avec cette commande, la machine est programmer d'arrêter après une minute, dans cette petite période, on peut annuler l'arrêt du système utilisant la command :

shutdown -c, c'est-à-dire cancel ou annuler.

```
[root@system ~]# shutdown -r now
Broadcast message from root (tty1) (Sat Nov 28 09:31:29 2012):
The system is going down for system halt NOW!
```

Fig. 2.2 L'exécution de la commande Shutdown -r now

Le redémarrage du système d'exploitation s'effectue avec les deux commandes :

reboot ou shutdown -r now (arrêter et relancer le système)

Le redémarrage logiciel est le plus préférées par rapport au redémarrage matériel en cliquant sur le bouton power.

De même comme les commandes d'arrêt du système, seul l'administrateur système peut exécutés les commandes qui permet le redémarrage de système d'exploitation.

Donc, ne pas utiliser cette procédure peut entraîner des pertes de données, il faut bien vérifier l'état de toutes vos taches exécutées à la console (l'invite du Shell) avant l'exécution de ces commandes. On note que ces commandes sont limitées aux administrateurs.

Voici dans ce tableau quelques options de la commande shutdown

Tab. 2.2 Options de la commande shutdown

Commande	Description
shutdown -h +6m	éteindre le système après 6 min
shutdown -r +20m	redémarrer le système après 20 min
shutdown -h now	éteindre le système immédiatement
shutdown -r now	redémarrer le système immédiatement
shutdown -c	annuler shutdown
halt	éteindre le système immédiatement
reboot	redémarrer le système immédiatement

Pour avoir le help sur une commande quelconque, il suffit d'utiliser la commande man

Exemples :

man ls plus d'options sur la commande man
man cd plus d'options sur la commande cd
man cp plus d'options sur la commande cp
ou même on utilise la commande lui-même suivant de l'option --help

Exemples :

ls --help plus d'options sur la commande man
cd --help plus d'options sur la commande cd
cp --help plus d'options sur la commande cp

Remarque : dans certains cas des distributions Linux, on note le manque du help pour quelques commandes, et donc il suffit d'utiliser les deux commandes : si l'un des deux commandes (man ou --help) ne marche pas, il est impérativement d'utiliser l'autre commande. Soit on utilise :

man + la commande
la commande + --help

on principe l'un des deux commandes marche très bien.

2.4 Exemples d'application

Voici quelques exemples simples de manipulation de commandes Linux [10], dans ces exemples, on essaie d'exécuter les commandes les plus simples et les commandes pratiques pour nos étudiants, aussi, pour rendre dans un premier temps le système d'exploitation linux plus simple et plus pratique. Dans les trois premiers exemples, on aborde la notion d'une séquence de commandes Shell de Linux et leurs résultats.

Exemple : 1

Démarrez votre machine sous Linux. Si vous êtes sous Windows, fermez votre compte et redémarrez votre machine sous Linux. Au démarrage et lors du choix des systèmes laissez la valeur par défaut. (le choix de la machine pas de l'utilisateur connecté) Connectez-vous.

Exemple : 2

Lancer une invite du shell (terminal). Vous vous trouvez face à une invite de commandes. Chaque commande s'écrit après le signe \$ et se valide par la touche Entrée.

- Essayez avec des commandes simples comme le cas de ls, cd, rm, mkdir

Exemple : 3

La commande ls liste le contenu d'un répertoire, pwd affiche le répertoire courant et cd change le répertoire courant. Exécutez la séquence suivante de commandes

1. pwd

2. ls

3. cd Image

4. pwd

5. ls

- Quel est le répertoire par défaut à l'ouverture d'un terminal ? Comparer avec celui de votre voisin.

- Quelle séquence permet de lister le contenu du répertoire nommé med ?

Exemple : 4

Exécutez la séquence suivante :

1. cd

2. pwd

3. cd .

4. pwd

5. cd ..

6. pwd

7. cd ..

8. pwd

9. cd .

10. pwd

11. cd

12. pwd

Quelle est la dernière réponse dans le terminal ? Pourquoi ?

Exemple : 5

La commande `mkdir med` permet de créer un répertoire nommé `med`. Dans le bureau créez un répertoire `med1`.

- Est-ce que c'est facile de supprimer ce répertoire ? oui
- Quelle est la commande pour supprimer ce répertoire ? `rmdir med`
- Créer un fichier nommé `exemple1` dans ce répertoire
- Comment supprimer le répertoire qui porte le fichier nommé `exemple1` ? `rm -r med`

Ici la commande `rm -r` permet la suppression récursivement.

Exemple : 6

Pour n'importe quelle commande, le manuel est accessible via la commande `man`. Donc on utilise la commande `man` comme suit :

`man nom de la commande`

Lisez le manuel de la commande `ls` et répondez aux questions suivantes :

1. Quelle option permet d'afficher tous les fichiers ?
2. Quelle option permet d'afficher les fichiers avec un format long en liste (long listing format) ?
3. Quelle option permet d'afficher la taille des fichiers avec un format plus lisible ?

Exemple : 7

Tester la commande `man` pour obtenir l'aide sur `man`. `man man`

Quelles sont les commandes susceptibles d'être documentées dans `man`? `man man`

Exemple : 8

Le répertoire `/bin` est répertoire parmi les répertoires les plus important dans Linux qui porte l'ensemble des commandes de Linux. Sous votre terminal :

Lister le contenu du répertoire `/bin`.

`ls /bin`

Lister de manière détaillée le contenu du répertoire `/bin`.

`ls -l /bin`

`ls -la /bin`

Exemple : 9

Le répertoire /dev est répertoire parmi les répertoires les plus important dans Linux qui porte des fichiers systèmes de Linux très importantes. Et donc si vous n'êtes pas capable de connaitre les résultats lors des manipulations des fichiers sous Linux utilisant le mode Shell, je te conseil de n'est exécutés pas ces fichier et les commandes pour les modifier sous votre terminal, car, la modification de ces fichier endommage votre système d'exploitation Linux :

Exemples :

Utilisant la commande ls pour voir le contenu du répertoire /dev

Lister le contenu du répertoire /dev.

```
ls /dev
```

Lister le contenu du fichier /etc/passwd.

```
cat /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash
```

```
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
```

```
bin:x:2:2:bin:/bin:/bin/sh
```

```
sys:x:3:3:sys:/dev:/bin/sh
```

```
sync:x:4:65534:sync:/bin:/bin/sync
```

```
games:x:5:60:games:/usr/games:/bin/sh
```

```
man:x:6:12:man:/var/cache/man:/bin/sh
```

```
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
```

```
mail:x:8:8:mail:/var/mail:/bin/sh
```

```
news:x:9:9:news:/var/spool/news:/bin/sh
```

...

```
user,,,:/var/run/hplip:/bin/false
```

```
nico:x:1000:1000:Nico,,,:/home/nico:/bin/bash
```

Exemple : 10

Utilisant la commande wc pour afficher le nombre de lignes + mots + caractères

Combien de lignes, de mots et de caractères comportent les fichiers monfichier.unix

```
wc monfichier.unix
```

```
1 4 15 monfichier.unix
```

Exemple : 11

Quel est le système Linux installé sur votre machine? `uname -a`

Quelle est la distribution Linux installée ? `cat /etc/issue`

Quels sont les utilisateurs connectés au système ? `Who`

Fermer ce terminal `exit`

Exemple : 12

Le manuel d'une commande quelconque est accessible via la commande `man`. Ou on utilise la commande suivie par `-help`.

Lisez le manuel de la commande `ls` et répondez aux questions suivantes :

1. Quelle option permet d'afficher tous les fichiers dans un répertoire ? `ls -a`
2. Quelle option permet d'afficher les fichiers avec un format long en liste (long listing format) ? `ls -l`
3. Quelle option permet d'afficher la taille des fichiers avec un format plus lisible ? `ls -la`

Chapitre 3. L'éditeur Vi

3.1 Intérêt de Vi

3.1.1 Introduction

L'éditeur de texte Vi est l'éditeur de base utilisant le système d'exploitation Linux, on le retrouve donc dans toutes les versions de ce système d'exploitation et donc dans tous les systèmes dérivés, qu'il s'agisse de LINUX ou d'autres produits apparentés. On trouve aussi d'autres éditeurs de texte en Linux comme le cas de Emacs en particulier qu'il est généralement en compagnie, , dans des distributions musclées, on le retrouve souvent seul dans les micros distributions de Linux, celles qui tiennent que sur une disquette [11].

Vi est un éditeur en mode text (toutes les commandes se font grâce au clavier) qui existe sur les systèmes d'exploitation Unix (Linux entres autres). Vi peut être très utile si l'interface graphique ne fonctionne plus. Pour éditer les fichiers type text.

Il s'agit certainement l'aspect le plus déroutant pour le débutant. Au lancement de Vi, il se trouve automatiquement en commande. Tout ce que vous taperez sera interprété comme une commande et non comme du texte à enregistrer. L'autre mode, appelé mode insertion et sert, lui, à entrer et écrit du texte. Selon la version de Vi utilisée l'indication du mode courant n'apparaît pas clairement dans la barre de menu. Ceci est particulièrement vrai lorsqu'on utilise Vi via TELNET.

Une bonne pratique consiste donc à ne jamais rester en mode insertion mais à toujours revenir au mode commande dès la fin de la frappe. Ce retour s'obtient en tapant Ctrl-C



Pour les utilisateurs débutants avec Linux, Vi est pénible à apprendre pour les débutants, surtout aujourd'hui où les nouveaux éditeurs sont tous munis de belles interfaces graphiques comme les éditeurs de texte de Windows par exemple EditPlus. Mais la puissance de Vi dans Linux surpasse de beaucoup d'autres éditeurs plus conviviaux. On peut quasiment tout faire sous Vi, la difficulté est de savoir comment Vi peut le faire.

Attention, au premier contact de Vi, il est probable que vous soyez d'étéconcerté, totalement perdu.

Il y a certaines choses à connaître :

- L'interface avec l'utilisateur se fait uniquement par le biais du clavier. Donc, ne vous énervez pas sur celui-ci, la situation ne fera qu'empirer.
- La touche DEL n'efface pas les caractères, ça trouble beaucoup au début.
- Vi a deux modes : le mode insertion (on saisit du text) et le mode commande (on applique des actions sur le text) que nous détaillerons ci-dessous.
- Vi a énormément de commandes. Pour l'utiliser efficacement, pas nécessaire de toutes les connaître (y a-t-il seulement quelqu'un qui les connaît toutes ? j'en doute...) ; m'mémorisez les principales, et au fur et à mesure de son utilisation, vous en apprendrez d'autres.

... alors bon courage et ne vous inquiétez pas si c'est la galère totale au début !

Le choix de Vi dans le programme du module de l'informatique pour la chimie est justifié par le fait que les étudiants de chimie physique en générale utilisent ce mode comme éditeur de text pour lire leurs fichiers .txt. Donc, la connaissance de ce type des éditeurs est important pour facilite la tâche de l'utilisation direct de leurs fichiers .txt sous Linus, sans aller et retour au Windows et au Linux plusieurs fois.

3.2 Les deux modes Vi

Vi se lance directement en mode console ou dans une fenêtre terminal.

Le déplacement entre les deux modes de Vi, se fait à l'aide de la touche 'i' (pour "insertion") et la touche échap <ESC>.

Une fois lancé Vi occupe toute la fenêtre [12].

Une syntaxe très classique comme Vi file.txt lance Vi en affichant le contenu du fichier file.txt. Si ce fichier n'existe pas Vi le crée puis affiche des lignes vides commençant par un symbole ~ . En fait ces lignes sont fictives, le remplissage est purement visuel. Selon la version de Vi que vous utilisez et son paramétrage vous aurez peut-être une barre de menu dans le bas de l'écran qui vous donnera quelques (maigres) informations sur la session en cours, nom du fichier édité, position du curseur dans le texte [13].

3.2.1 Mode commande

C'est le mode initial au chargement. Dans ce cas Vi est non opérationnel pour l'édition. On peut manipuler deux types de commandes, les commandes directes et les commandes utilisant les deux points ":". [14]

Les commandes directes sont celles qui s'exécutent directement, sur la frappe d'une seule lettre, par exemple y pour copier, p pour coller, d pour effacer.

Les commandes ":" attendent des arguments éventuels et surtout la frappe de la touche entrée pour s'exécuter, exemple : w foo.txt qui demande la sauvegarde sous le nom foo.txt.

Aspect particulièrement précieux, le mode commande de Vi présente cette analogie avec certains Shell d'offrir un historique de commande consultable avec la flèche "haut". Cet aménagement permet souvent d'éviter de retaper une commande déjà saisie précédemment.

3.2.2 Mode insertion

Pour écrire dans le mode text Vi il suffit de passer du mode commande vers le mode insertion. La frappe de i ou de A permet de passer du mode commande au mode insertion. La différence essentielle est que i laisse le curseur à sa place alors que A déplacé ce curseur en fin de ligne ce qui est très pratique pour réaliser un ajout. ESC ou CTRL-C permet de revenir en mode commande [14].

3.3 Exemples d'application

3.3.1 Sauvegarder un fichier et quitter

Pour sauvegarder un fichier sous le mode Vi on a les trois principaux commandes :w, :wq, et :q !

- Quitter sans sauvegarder
:q!
- Sauvegarder un fichier sans quitter
:w pour sauvegarder (write) ;
- Sauvegarder un fichier et quitter
:wq pour sauvegarder (write) ET quitter (quit) ;

3.3.2 Autres commande de Vi

Voici quelques commandes les plus importantes au mode Vi:

Dans ce tableau, on écrit juste les commandes les plus utilisées, et le commandes les plus pratiques en mode Vi, comme le cas sauvegarder un fichier, passer en mode commande ou en mode insertion, copier des mots, phrases et même des paragraphes, coller copier des mots, phrases et même des paragraphes et effacer un test.

Tab. 3.1 Commandes de base du mode Vi

Commande	Description
:q	Quitte Vi sans sauvegarder.
:q!	Quitte Vi sans sauvegarder même si le fichier a été modifié.
:wq	Sauvegarde le fichier et quitte.
:x	Sauvegarde le fichier et quitte.
i	Mode insertion de text. (Echap pour quitter).
r	Mode remplacement de text. (Echap pour quitter).
x	Efface le caractère sélectionné.
yy	Copie (4yy copie 4 lignes).
dd	Coupe (4dd coupe 4 lignes).
p	Colle.

Voici un exemple de fenêtre terminal ou la présence du mode Vi dans le cas d'enregistrement d'un fichier text :

C'est le cas d'exécution de la commande :wq

Références bibliographiques

1. A. Tanenbaum, Modern Operating Systems, Prentice–Hall, 1992.
2. M. J. Bach, La conception du système Unix, Elsevier-Masson, 1993.
3. S. Bourne, Le système Unix, InterEditions, 1991.
4. N. Pons, Linux, Entraînez-vous sur les commandes de base : Exercices et corrigés (3ième édition), ISBN : 9782746095687, 2015.
5. J. F. Bouchaudy, Linux Administration : Tome 1, Les bases de l'administration système, 2ième édition, 2009.
6. J.L. Nebut, Unix pour l'utilisateur : Commandes et langages de commandes, Technip, 1990.
7. B. Guérin, Linux, Administrer le système, 2015.
8. T. Bartolone, Red Hat Enterprise Linux - CentOS, Mise en production et administration de serveurs (2ième édition), ISBN : 9782746091610, 2015.
9. M. Dutreix, P. Fauquembergue, Debian GNU/Linux - Services réseau, ISBN : 9782409012068, 2018.
10. S. Blondeel, D. Carton, H. Singodiwirjo, Débuter sous Linux, 2em édition, Groupe Eyrolles, 2004, 2005, ISBN : 2-212-11559-8.
11. P. Pinchon, Linux, Administration avancée - Maintenance et exploitation de vos serveurs (2e édition), ISBN : 9782746099845, 2016.
12. E. Siever, S. Spainhour, S. Figgins and J. P. Hekman , Linux in a Nutshell, 3rd Edition, Third Edition, ISBN 0-596-00025-1, 2000.
13. A. Robbins, L. Lamb, E. Hannah, Learning the vi and Vim Editors, 7th Editio, O'Reilly Media, 2009.
14. A. Robbins, vi Editor Pocket Reference, 1st Edition, O'Reilly, ISBN 1-56592-497-5vi, 1999.