

# Kit de survie en lignes de commandes ...

pour les nuls.

Ceci est la version légèrement corrigée, étendue et complétée de la présentation du jeudi 2 juin 2018.

## Pourquoi la ligne de commande ?

La ligne de commande est une chose qui a, pour des raisons marketing et en lien avec la massification de l'usage de l'outil informatique, peu à peu disparue alors qu'elle faisait partie intégrante des ordinateurs dans les années 80 et jusqu'en 1995 pour le grand public.

Actuellement, la ligne de commande est perçue par les néophytes comme un objet de toute puissance ou de diabolisation. Les stéréotypes véhiculés par les media traditionnels associent souvent la ligne de commande, et in extenso le terminal, à un outil de hacker.



Alors que cela a été pendant longtemps le quotidien de beaucoup d'utilisateurs.

Posons un peu de vocabulaire : Linux est **un noyau** c'est à dire un binaire qui communique avec le matériel. Autour de lui est le système **GNU** (pour GNU is Not Unix). Le système GNU a été pensé dans les années 80 comme étant une alternative Libre au système (gratuit) mais propriétaire Unix.

Dans les années 80, à la préhistoire informatique, les systèmes d'ordinateurs n'avaient pour ainsi dire pas d'interface graphique (et encore moins de souris) : un noyau, un interpréteur de commande (shell) et quelques utilitaires suffisaient à faire un ordinateur fonctionnel.

```
64K High Memory Area is available.
```

```
This driver is provided by Oak Technology, Inc..  
OTI-91X ATAPI CD-ROM device driver, Rev D91XV352  
(C)Copyright Oak Technology Inc. 1987-1997
```

```
Device Name       : 12345678  
Transfer Mode     : Programmed I/O  
Number of drives  : 1
```

```
C:\>C:\DOS\SMARTDRV.EXE /X
```

```
MODE prepare code page function completed
```

```
MODE select code page function completed
```

```
MSCDEX Version 2.23
```

```
Copyright (C) Microsoft Corp. 1986-1993. All rights reserved.
```

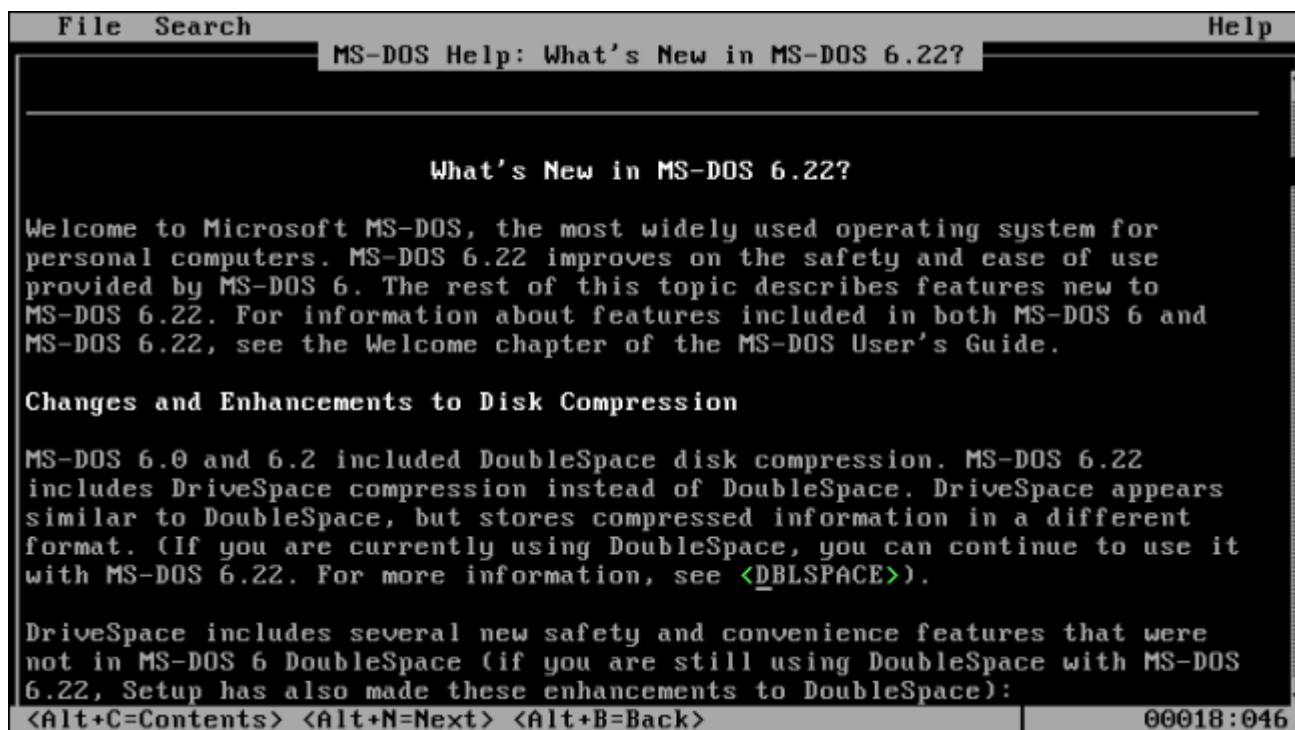
```
Drive D: = Driver 12345678 unit 0
```

```
CuteMouse v1.9 [FreeDOS]
```

```
Installed at PS/2 port
```

```
C:\>
```

Ici le système Microsoft Disk Operating System fonctionnant avec les fichiers : msdos.sys et io.sys (l'équivalent du noyau), config.sys et autoexec.bat, à partir de là l'interpréteur de commande (ou shell) : COMMAND.COM s'exécute et affiche l'image précédente.



Dans MS-DOS on trouve des outils basiques (dont BASIC.COM) qui permettent des tâches de gestion et d'édition simple, l'image affichée ci-dessus est EDIT.COM l'éditeur de textes simple mais toujours utile.

```
CP/M Bios for C-280 Computer Copyright 2016 (c) by Michal Cierniak
CP/M 2.2 Copyright 1979 (c) by Digital Research

a>dir
A: LOAD      COM : STAT      COM : PIP      COM : ED      COM
A: DUMP      COM : DDT      COM : ASM      COM : CAT      COM
A: SUBMIT    COM
a>dir b:
B: OBASIC    COM : TABLICA  BAS : SIEMA    BAS : STARS    BAS
B: MBASIC    COM : SALVO      BAS : LEM      BAS : SINEMAVE  BAS
B: STARTREX  BAS : TREKINST  BAS : TURBO    COM : TURBO    MSG
B: TURBO     OUR : TINST     COM : TINST    DTA : TINST    MSG
B: TEST      BAK : TEST      PAS : PROG     BAS
a>stat
A: R/W, Space: 199k
B: R/W, Space: 79k

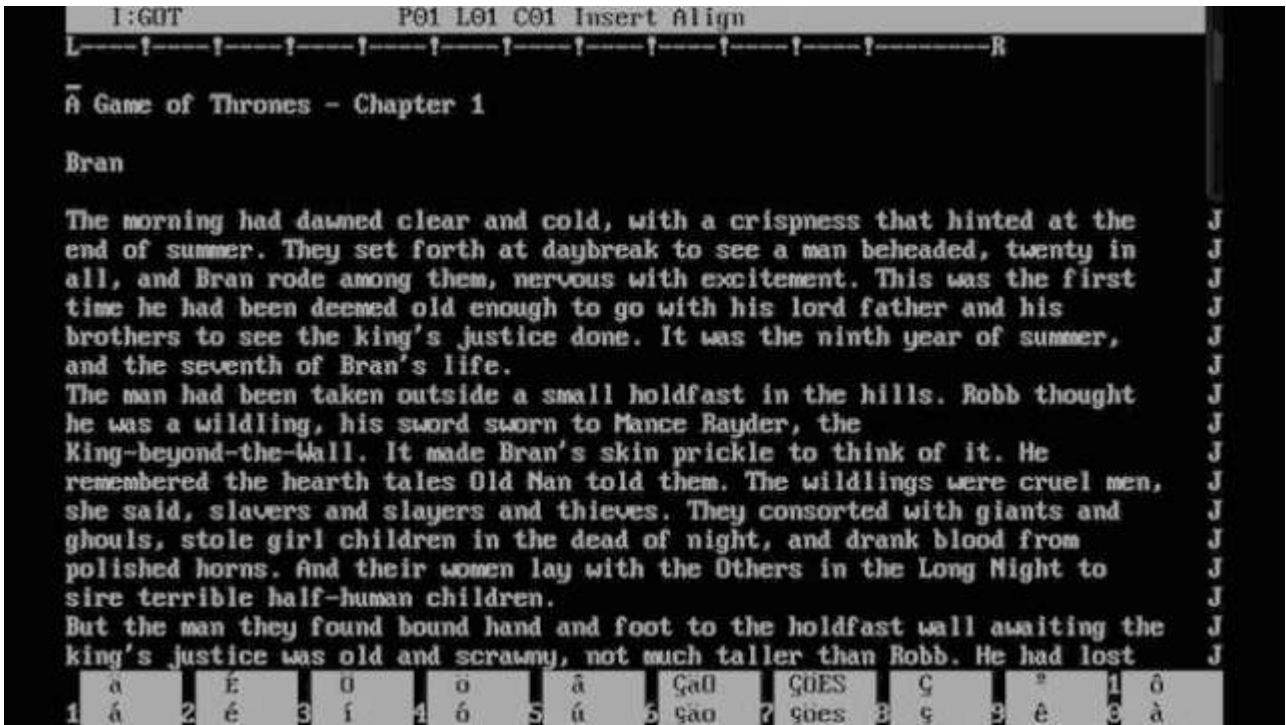
a>_
```

Et dans cet exemple l'ancêtre de MS-DOS à savoir CP/M. Ce qui est intéressant c'est de constater que MS-DOS ou Windows utilisent encore la notation A: B: C: etc... pour accrocher les périphériques locaux ou mapper les ressources distantes.

D'ailleurs de nos jours, certain-e-s utilisent encore des systèmes en mode texte sans interface graphique :

- Les administrateurs / administratrices système travaillant sous Linux, Unix, AIX ...
- Certain-e-s inconditionnels d'anciens systèmes, comme Georges R.R. Martin pour écrire son Game of Thrones (c.f. video youtube)





Un extrait de *The Secret History of Westeros or a game of thrones*. Lors d'une émission (lien youtube suivant) télévisée, l'auteur, G.R.R. Martin avoua que son arme secrète pour éviter de se faire pirater à distance les extraits de romans en cours d'écriture, utilisait "an old DOS machine" (une vieille machine sous MS-DOS) et comme traitement de textes il utilisait WordStar 4.0.

Georges R.R. Martin at CONAN / Youtube

- Des aficionado-a:s qui échantent leurs personnalizations sur le subreddit appelé "unixporn" -> [unixporn](https://www.reddit.com/r/unixporn/) on reddit



```

file.c
libtool: compile: x86_64-pc-linux-gnu-gcc -DHAVE_CONFIG_H -I. -I. -I./../in
clude -I./../include -I./ref -I/usr/include/et -pipe -O2 -march=native -D_LA
RGE_FILES= -Wall -Wmissing-prototypes -Wpointer-arith -Wbad-function-cast -Wm
issing-declarations -Wnested-externs -pipe -O2 -march=native -c file.c -fPIC
-DPIC -o .libs/libhx589_la-file.o
/bin/sh ./../libtool --tag=CC --mode=compile x86_64-pc-linux-gnu-gcc -DH
AVE_CONFIG_H -I. -I. -I./../include -I./../include -I./ref -I/usr/include
/et -pipe -O2 -march=native -D_LARGE_FILES= -Wall -Wmissing-prototypes -Wpoi
nter-arith -Wbad-function-cast -Wmissing-declarations -Wnested-externs -pipe
-O2 -march=native -c -o libhx589_la-sel.lo `test -f 'sel.c' || echo './'`sel
.c
libtool: compile: x86_64-pc-linux-gnu-gcc -DHAVE_CONFIG_H -I. -I. -I./../in
clude -I./../include -I./ref -I/usr/include/et -pipe -O2 -march=native -D_LA
RGE_FILES= -Wall -Wmissing-prototypes -Wpointer-arith -Wbad-function-cast -Wm
issing-declarations -Wnested-externs -pipe -O2 -march=native -c sel.c -fPIC
-DPIC -o .libs/libhx589_la-sel.o
/bin/sh ./../libtool --tag=CC --mode=compile x86_64-pc-linux-gnu-gcc -DH
AVE_CONFIG_H -I. -I. -I./../include -I./../include -I./ref -I/usr/include
/et -pipe -O2 -march=native -D_LARGE_FILES= -Wall -Wmissing-prototypes -Wpoi
nter-arith -Wbad-function-cast -Wmissing-declarations -Wnested-externs -pipe
-O2 -march=native -c -o libhx589_la-sel-gran.lo `test -f 'sel-gran.c' || ech
o './'`sel-gran.c
libtool: compile: x86_64-pc-linux-gnu-gcc -DHAVE_CONFIG_H -I. -I. -I./../in
clude -I./../include -I./ref -I/usr/include/et -pipe -O2 -march=native -D_LA
RGE_FILES= -Wall -Wmissing-prototypes -Wpointer-arith -Wbad-function-cast -Wm
issing-declarations -Wnested-externs -pipe -O2 -march=native -c sel-gran.c -
fPIC -DPIC -o .libs/libhx589_la-sel-gran.o
HOST="x86_64-pc-linux-gnu"
x86_64-pc-linux-gnu_CFLAGS="-pipe -O2 -march=native"
x86_64-pc-linux-gnu_CFLAGS="-pipe -O2 -march=native"

case "${CATEGORY}/${PWD}" in
sys-apps/paludis)
NORMAL >> /etc/paludis/bashrc << sh << %* : 1: 1
[exbull:~] [1:vim] 2:24h

```

PID	USER	PRI	NI	VRT	RES	SHR-S	CPU%	MEM%	IORM	TIME+	Command	
7583	paludisbu	20	0	8652	2312	1788	5	59.3	0.8	0	1:26.62	sydbox
72	root	20	0	19132	2452	2304	5	0.6	1.0	0	0:01.11	/usr/lib
271	tureba	20	0	23928	6660	2388	5	0.0	2.6	0	0:04.87	tmux -u2
14177	paludisbu	20	0	6952	2498	1736	5	0.0	0.9	0	0:00.03	make all
12147	root	20	0	480M	15828	15832	5	0.0	7.8	0	0:04.67	cave exe
16059	tureba	20	0	14272	2928	2404	8	0.0	1.2	0	0:00.76	htop
14930	paludisbu	20	0	6988	2316	1656	5	0.0	0.0	0	0:00.04	make all
235	tureba	20	0	76444	3464	2740	5	0.0	1.4	0	0:01.78	sshd: tu
7584	root	20	0	118M	17788	15108	5	0.0	7.0	0	0:00.11	cave per
984	tureba	20	0	29212	8012	4508	5	0.0	3.2	0	0:00.22	vim /etc
26896	root	20	0	118M	17788	15168	5	0.0	7.8	0	0:00.50	cave per

```

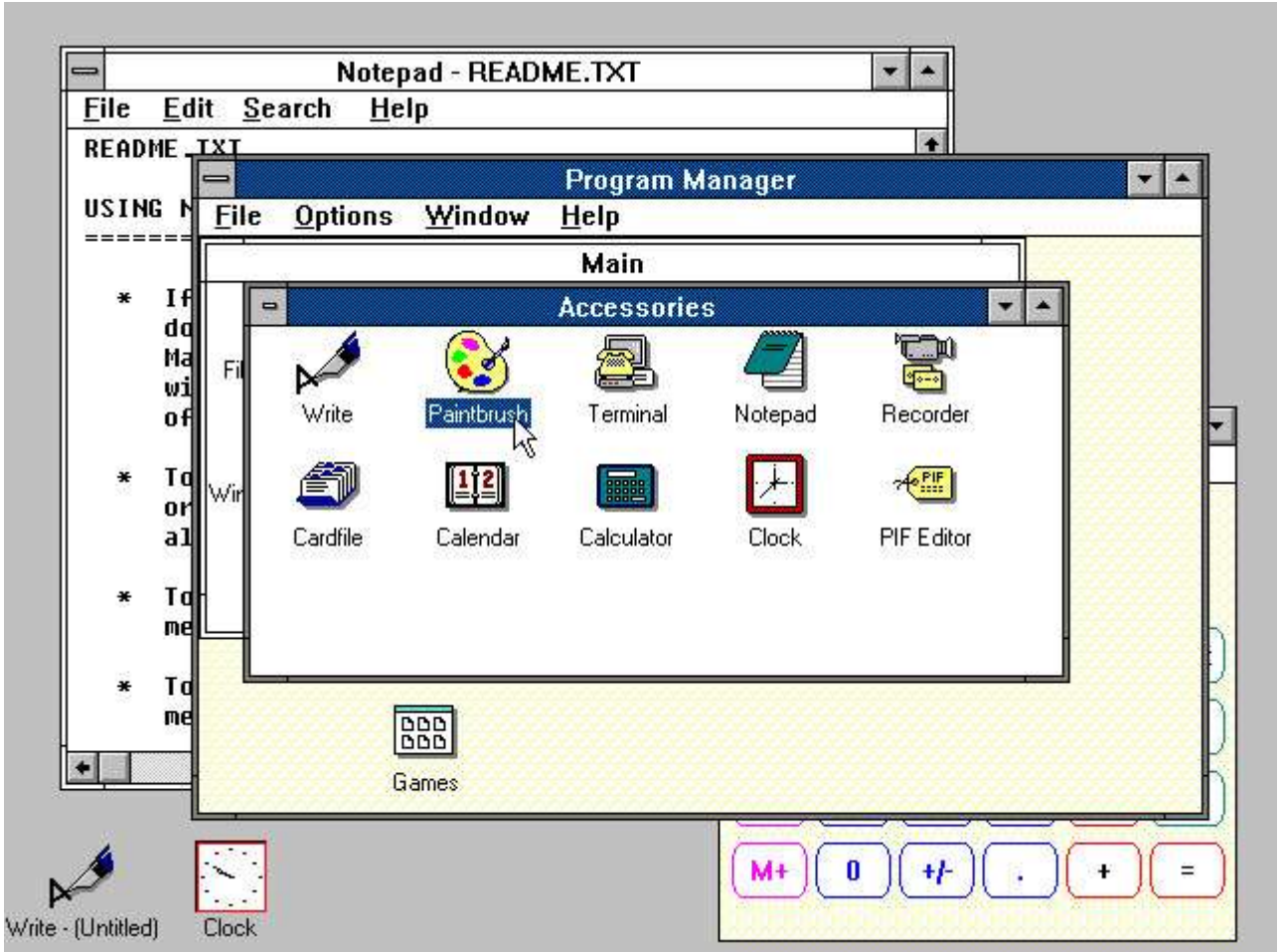
File: help, F: Setup, F: Search, F: Filter, F: Tree, F: Sort, F: View, F: Filter, F: Ctrl, F: Ctrl
README autp4ite.cache configure lnst armp
Rules build configure.ac luatru stamp-h1
[11:04:40]1023] (tureba@exbull) cd ../comp
[11:04:46]1024] (tureba@exbull) ls (master 952ba15 ~/o
ool)
AUTHORS Makefile.am VERSION config.lt libtool
Doxyfile Makefile.in aclocal.m4 config.status opal
HACKING Makefile.ospi.rules autogen.pl configure opal
INSTALL NEWS autop4ite.cache configure.ac orfe
LICENSE README config contrib ocbom
Makefile README_JAVA.txt config.log examples test
[11:04:46]1025] (tureba@exbull)
ool)
[11:07:11]1025] (tureba@exbull) (master 9[11:07:27]
1025][11:07:32]1025] (tureba@exbull) (master 952ba1
5 -/om[11:11:00:44]1025] (tureba@exbull) (master 9
[11:09:32]1025] (tureba@exbull) (master 952ba15 -/omp)
[11:11:09:32]1025] (tureba@exbull)

```

- Des gens qui ont des machines à très faibles capacités et ont des besoins spartiates en terme d'interfaces graphiques voire aucune.

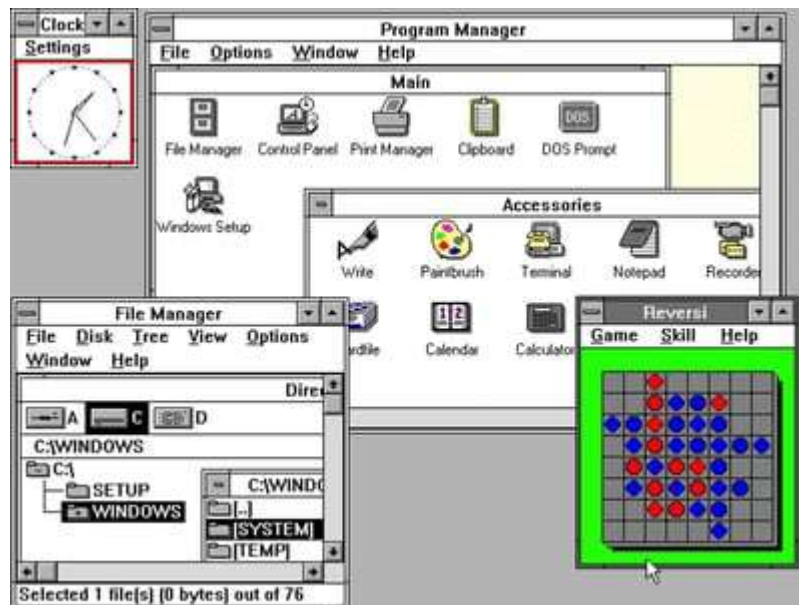
## De l'interface textuelle vers l'interface graphique

Rares étaient les interfaces graphiques, avant 1984 on pouvait trouver des stations Xerox, et peut être des amiga (à vérifier), passé 1984 on a eu le "Macintosh classique" d'apple, windows est devenu réellement graphique en 1990/1 (windows 3.0).



Sur le site de Toastytech on trouve d'ailleurs des captures d'écran des systèmes d'exploitation de cette époque : <http://toastytech.com/guis>

A cette époque, et hormis les macintosh et les amiga (avec GEM) les interfaces graphiques ne sont que des surcouches du système, ainsi les "PC" grand public sont livrés avec MS-DOS (ou d'autres DOS jusqu'à ce que Micro\$oft bloque le démarrage de windows s'il détecte un DOS autre que le sien en cours d'exécution, c.f. l'affaire "4DOS")... pour un démarrage automatique de windows lors de l'allumage de l'ordinateur il fallait ajouter la ligne "win" à la fin du fichier `C:\AUTOEXEC.BAT` (oui je sais, ça fait vieux barbu).

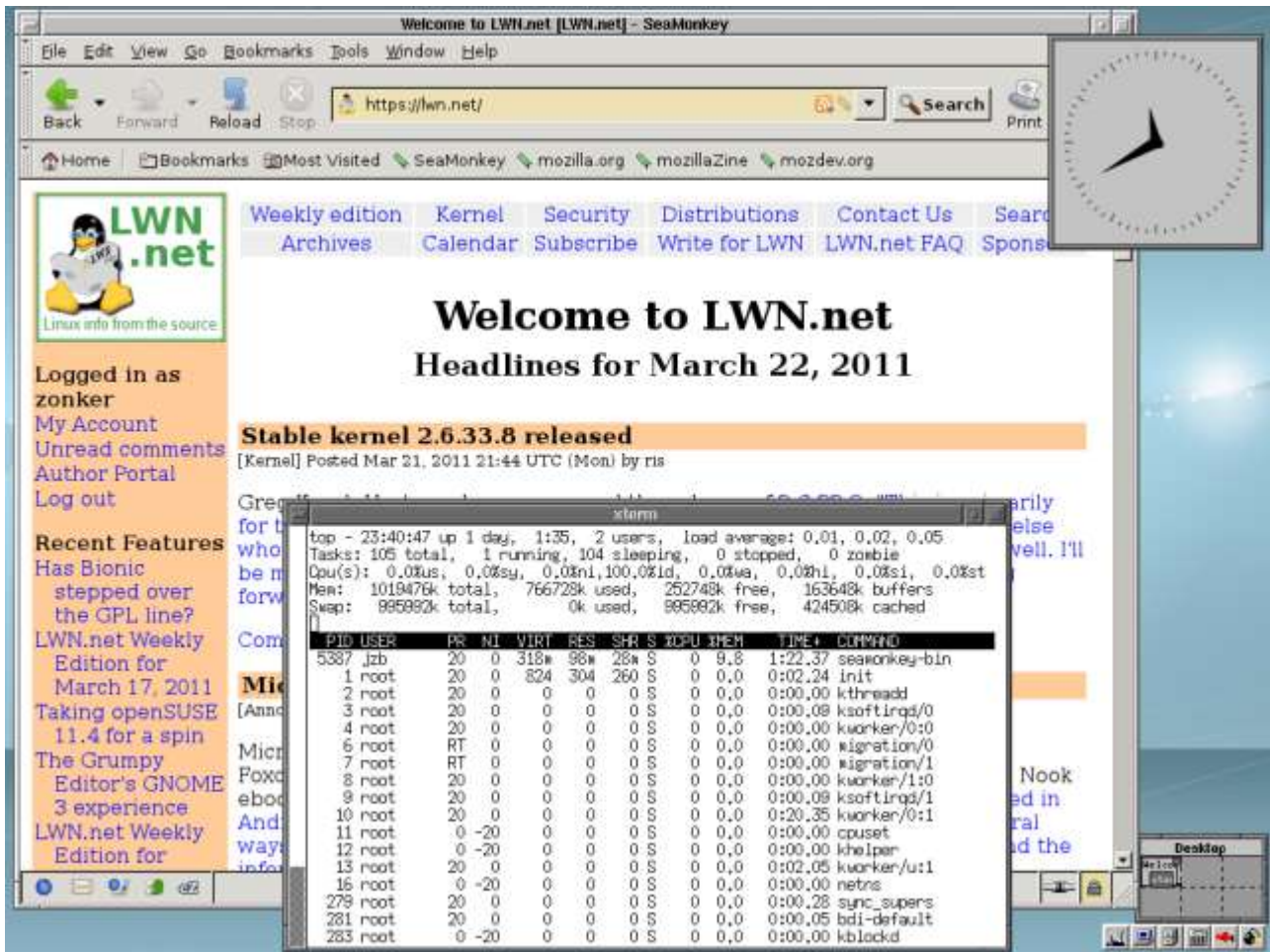


*Quelques captures d'écran de Windows 3.0 qui fût en son temps une révolution sur PC.*

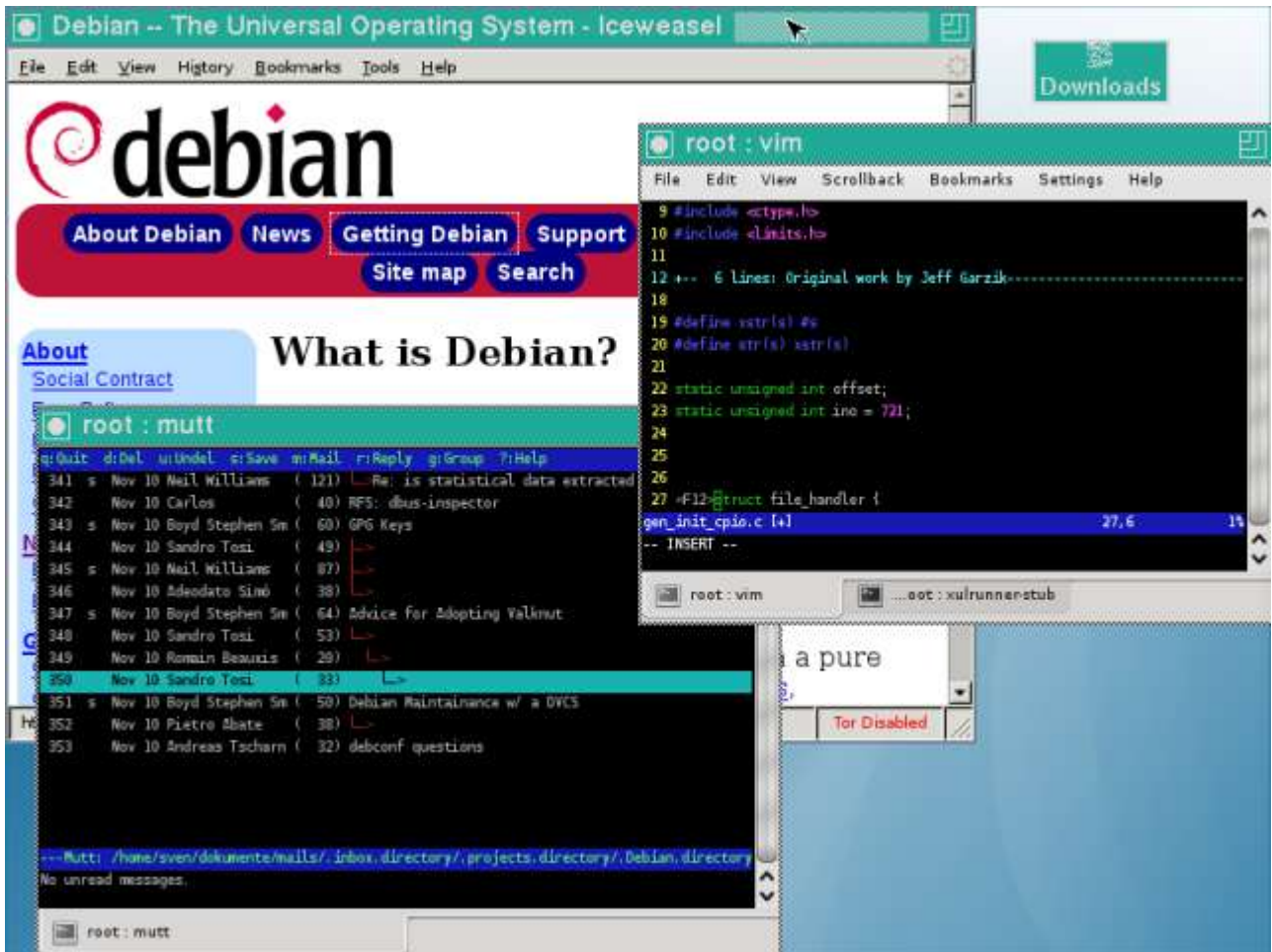
GNU/Linux ont donc été créés à cette époque là. Et, malgré l'arrivée du serveur X Window / XFree86, et comme Unix/Linux étaient plutôt utilisés par des ingénieurs et des passionnés, l'interface graphique n'a pas été leur intérêt. Si bien qu'au milieu des années 90 peu d'interfaces graphiques existent : TWM, KDE, GNOME, sawfish, GNUStep, ICEWm, blackbox ... Le soucis a toujours été à l'époque comme celui que nous expérimentons en tant que libristes actuellement : que ce soient les SoC ou les cartes propriétaires wifi, il faut toujours trouver un driver quelque part, alors imaginez vous dans les années 90, avec une connexion à 56 kbps (7 ko/s) télécharger un fichier de quelques mégas pour un driver sur une ligne analogique qui est inutilisable tant qu'on reste connecté à internet...

L'intérêt progressif du public et de développeurs et développeuses pour les Linux s'est traduite progressivement par plus de développements et désormais une jungle d'interfaces. Au milieu des années 90 commencent à arriver les premières applications graphiques et interfaces graphiques dignes de ce nom, TWM, XFCE, FVWM, GNUStep, KDE, GNOME, metacity, ICEWm, blackbox ... mais hormis ces interfaces, peu de logiciels graphiques "grand public". Le seul outil réellement développé est le terminal graphique où peuvent ainsi s'exécuter les logiciels dont les gens ont l'habitude.

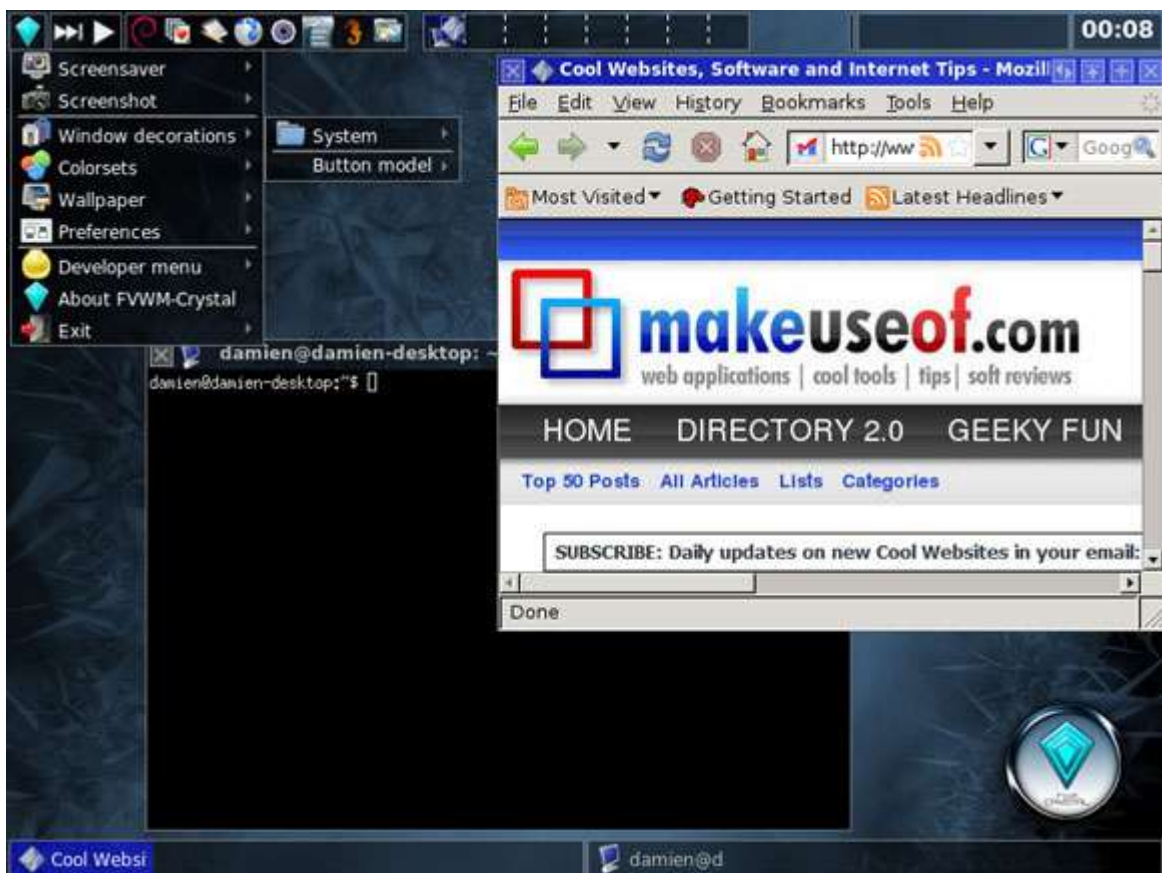




Deux des plus anciennes interfaces graphiques ci-avant FVWm (ou sa variante FVWm95) exécuté sur une antique distribution " slackware " et ci après TWM sur un système " debian ".

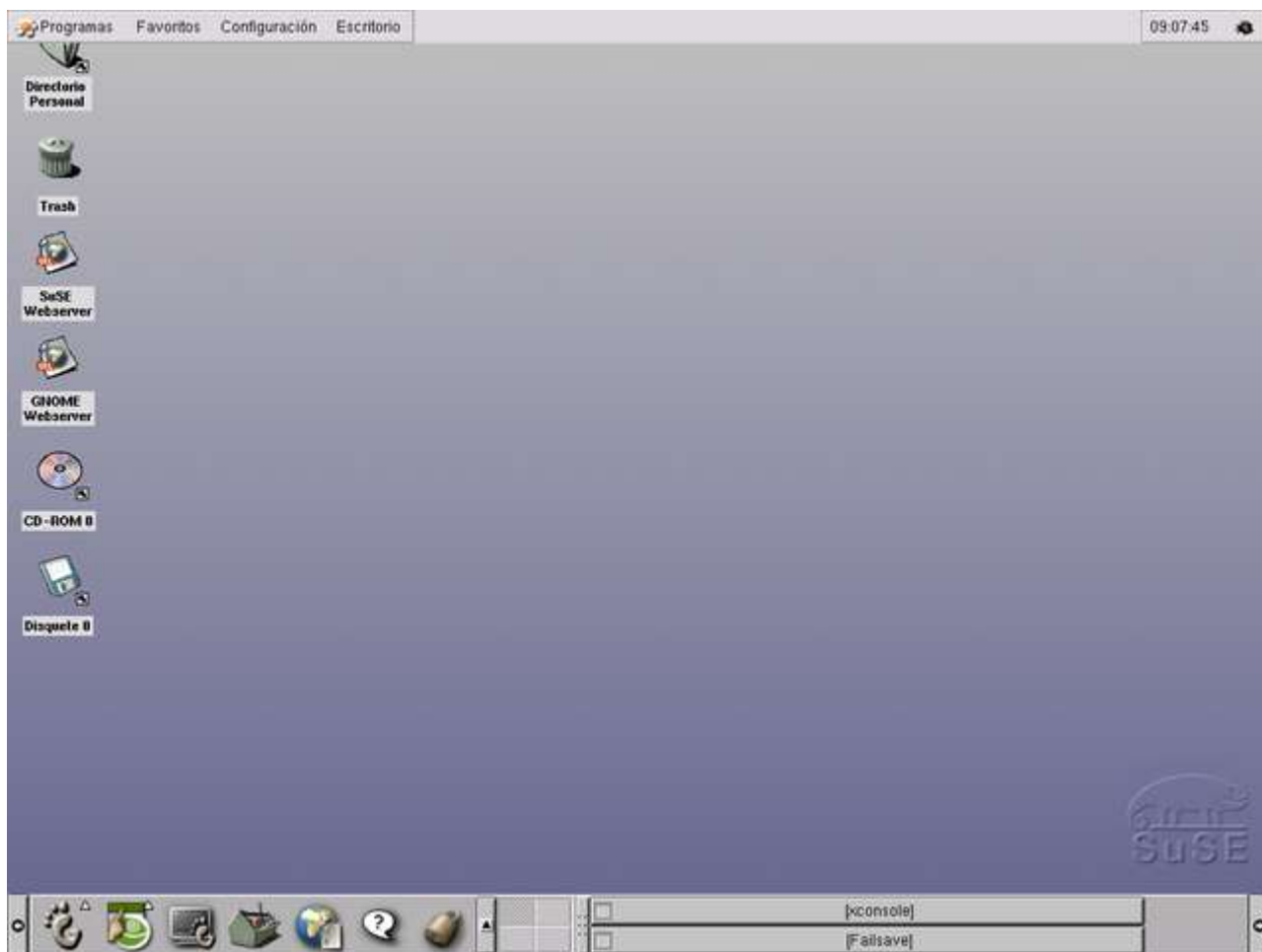


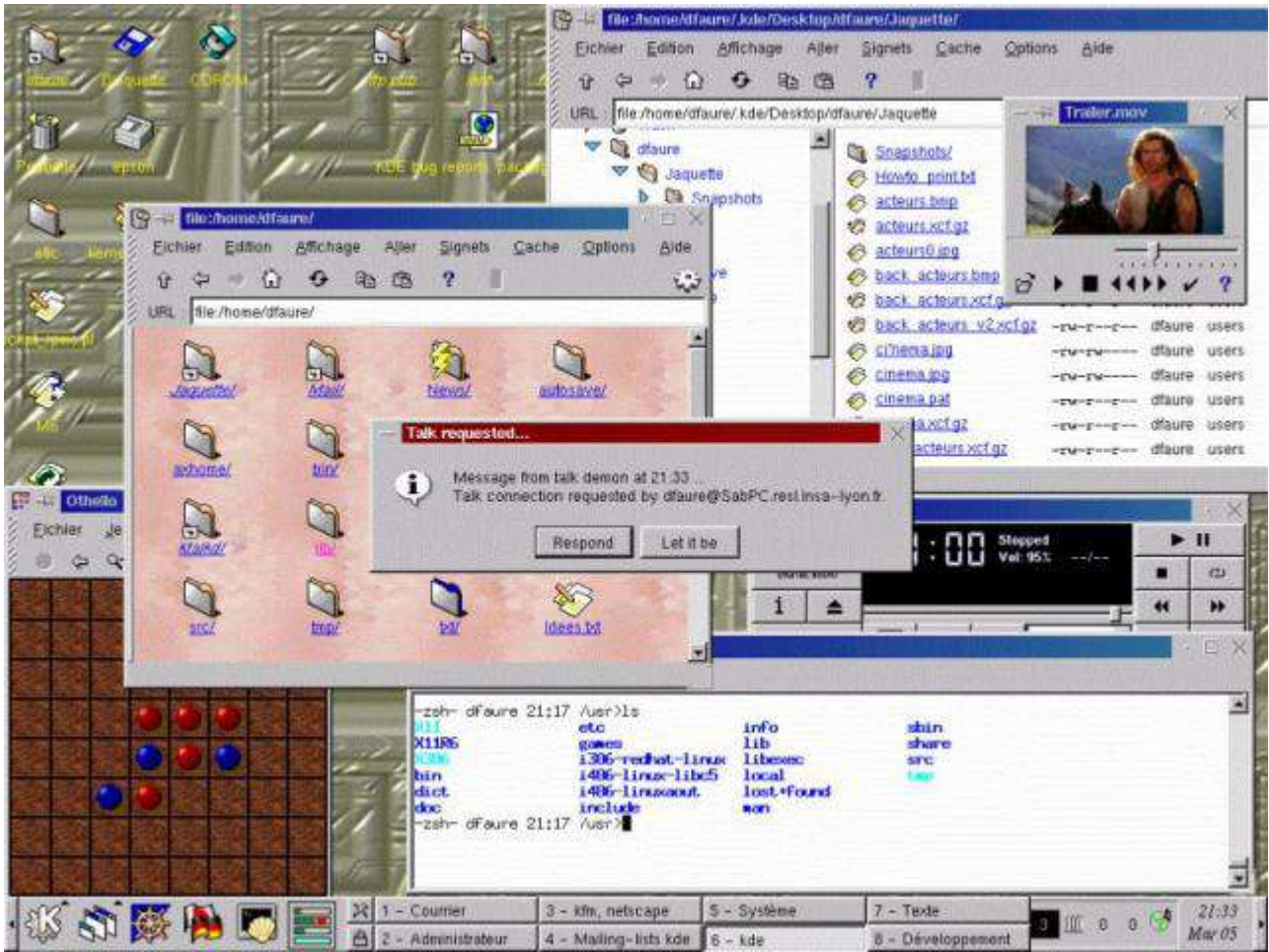
Une autre version de FVWM dite "Crystal" ...



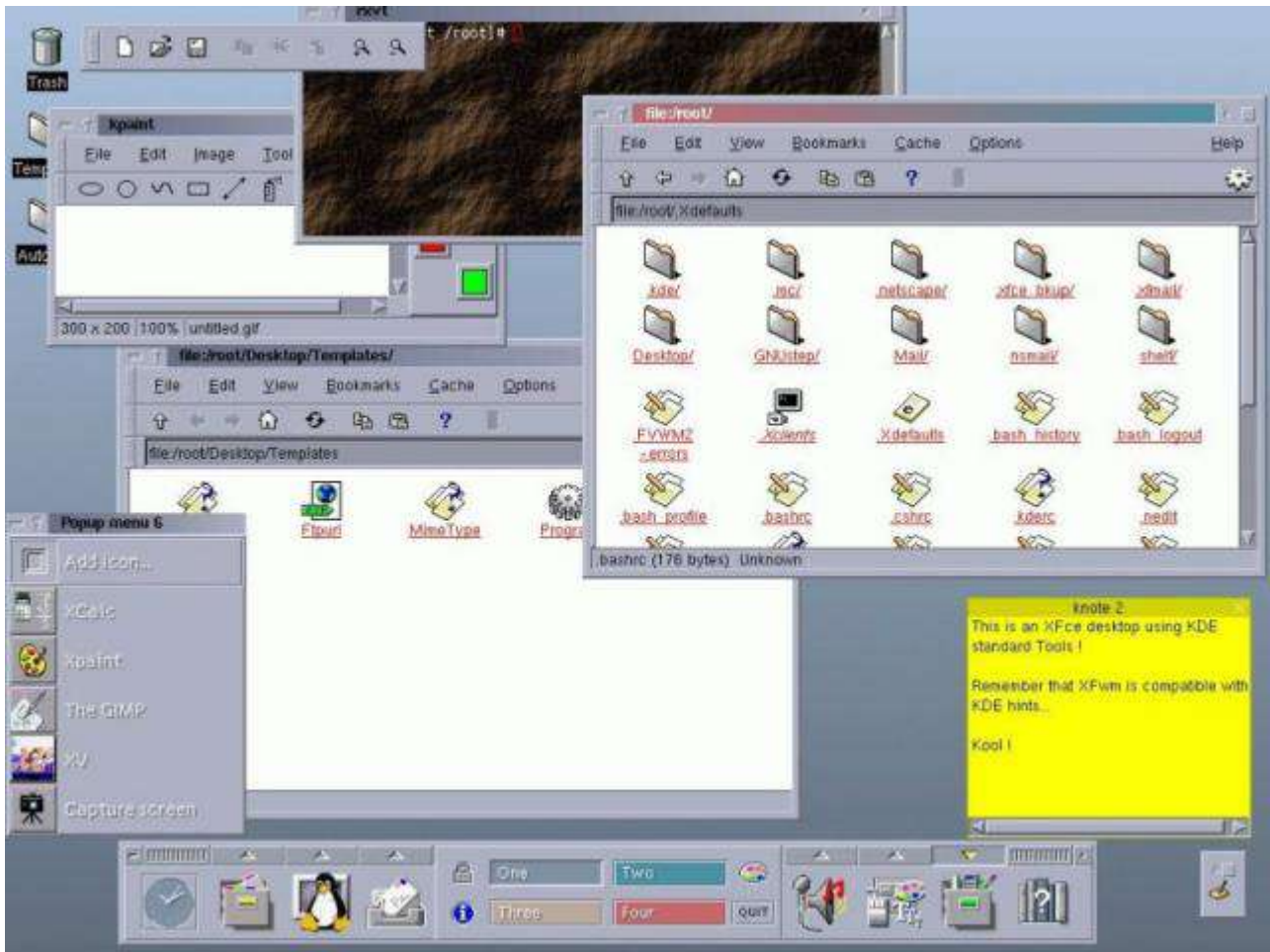


Et bien sûr n'oublions pas les interfaces graphiques qui sont encore parmi les plus connues du monde linux actuel dans les premières versions stables : GNOME 1 et KDE 1 !





Et aussi Xfce version 2



Le monde des Linux permet ce que le monde Windows ou Apple ne permet pas : outre le changement de style d'une interface par la personnalisation de l'interface (couleurs, transparences, formes des barres de fenêtres, décorations ou non, position dessus/dessous) il est possible de choisir l'interface qui est notre petite chou-chou et ce même si des gens restent sur l'interface "par défaut" ou choisissent leur distribution en fonction du *look* qu'elle montre par défaut (d'où à mon avis l'un des critères de la réussite d'ubuntu).

Cependant, il y a toujours eu des "pro" qui ne jurent que par la ligne de commande pour des raisons diverses et variées.

L'objet de la présentation actuelle est de vous donner quelques commandes de base et quelques exemples pratiques pour vous familiariser à la manipulation des dossiers et fichiers, quelques outils d'édition et de mises à jour. Pourquoi ?

Tout simplement car lorsqu'on est un-e habitué-e des interfaces graphiques les raisons d'utiliser une interface textuelle sont (entre autres) :

- une panne de l'interface graphique (et oui ça arrive... nvidia powaaa !)
- un goût pour certaines applications plus rapides, moins gourmandes en mode texte qu'en mode graphique (personnalisation à la mode h4ck3rzzz) et une personnalisation à l'extrême du système avec uniquement les "paquets nécessaires".



```
~/home/anonymous/Musique/Clint Mansell
../
Clint Mansell - Moon OST #10 - Sacrifice-vbscRHUM6ho.flac [ |FLC]
Clint Mansell - Moon OST #11 - We're Going Home-fZsW0_cX0rI.flac [ |FLC]
Clint Mansell - Moon OST #12 - Welcome to Lunar Industries (Three Y[ |FLC]
Clint Mansell - Moon OST #1 - Welcome to Lunar Industries-_lAfMT5FI[07:14|FLC]
Clint Mansell - Moon OST #2 - Two Weeks Counting-o0wLmpTPH9A.flac [ |FLC]
Clint Mansell - Moon OST #3 - I'm Sam Bell-hABLaUbNME4.flac [ |FLC]
Clint Mansell - Moon OST #4 - I'm Sam Bell, Too-aNKusMCaxvs.flac [ |FLC]
Clint Mansell - Moon OST #5 - Memories (Someone We'll Never Know)-b[ |FLC]
Clint Mansell - Moon OST #6 - Are You Receiving-RyNVZ3blKtg.flac [ |FLC]
Clint Mansell - Moon OST #7 - Can't Get There From Here-Jl8odSveiuM[ |FLC]
Clint Mansell - Moon OST #8 - We're Not Programs, Gerty, We're Peop[ |FLC]
Clint Mansell - Moon OST #9 - The Nursery-0lqRdN01FCU.flac [ |FLC]

Playing... | Master 100% | >000:07:14
> Clint Mansell - Moon OST #1 - Welcome to Lunar Industries-_lAfMT5FIZE.flac
00:57 06:17 [07:14] 48kHz 901kbps [STEREO] [NET] [SHUFFLE] [REPEAT] [NEXT]
```

- un pc avec des capacités faibles qui permet d'exécuter peu de logiciels graphiquement mais qui supporte aussi quelques applications textuelles complémentaires.

Historiquement, les années 80 et début 90 offrent aux utilisateurs de PC à but professionnel un logiciel de démarrage, un noyau, un shell et quelques applications. Cela suffit amplement pour beaucoup de gens.

Avant de commencer à faire mumuse dans le terminal, il est nécessaire de reposer quelques bases de l'organisation hiérarchique du système de fichier, car autant dans les interfaces graphiques en comparant celle des 3 systèmes qui en utilisent (je passerai outre les systèmes de téléphones portables ou des tablettes iOS où justement il n'y a plus que des icônes accédant directement aux données mais où le système de fichier sous-jacent est masqué à l'utilisateur basique.

Critère	Windows	Linux / Unix	Mac Operating System / MacPS
Le système (ordinateur)	"Ordinateur" ou "Bureau (XP)"	/	/
Désignation des systèmes attachés non-amovibles	C:\ D:\ ... Z:\ (sauf A:\ et B:\ réservés par défaut aux disquettes)	/dev/sda.. /dev/sdb... /dev/hd... (vieux linux) /dev/mmcbk0p... le premier de ces systèmes est accroché à "/"	/ (je n'ai jamais eu à tester des systèmes mac sur plusieurs DD)
Désignation des systèmes amovibles	A:\ B:\ D:\ E:\ ... Z:\ (toujours A et B réservés aux disquettes)	/mnt/device /media/device /run/media/device (vieux) /mnt/login/device /media/login/device /run/media/login/device ...	/Volumes/device
Localisation des dossiers des utilisateurs.	C:\Documents and Settings (XP)\login C:\Utilisateurs\login C:\Users\login	/home/login	/users/login
Les programmes	C:\Program Files C:\Program Files_(x86)	/bin /sbin /usr/bin /usr/sbin /usr/local/bin /usr/local/sbin	?????
Les réglages "par défaut" des programmes	C:\Users C:\Utilisateurs C:\Documents and Settings Soit dans fichiers soit dans BDR.	/etc/ ou /usr/lib ou /home/login/.config ou /home/login/.local	/Users/login/.Library ?

## Où configurer le shell des utilisateurs ?

Ce paramètre se configure dans `/etc/passwd`

Note : Si on regarde les lignes `/usr/sbin/nologin`

Donc on commence à ouvrir un terminal pour nous entraîner.

Où démarre le shell du terminal ? Il démarre dans le dossier personnel de l'utilisateur donc `/home/login` . et si c'est l'utilisateur `root` à savoir l'administrateur c'est dans le dossier `/root` .

## Quelques commandes pour commencer :

## Où suis-je ? :

`pwd`

Cette commande veut dire : Print Working Directory, elle affiche (imprime) le dossier actuel (working directory).

```
1 anonymous@Ars3n1c:~$ pwd
2 /home/anonymous
3 anonymous@Ars3n1c:~$
```

## Qui suis-je ? :

`who` et / ou `whoami`

`who` : qui est-ce ? `whoami` : Who Am I qui suis-je ?

```
1 anonymous@Ars3n1c:~$ who
2 anonymous tty7          2018-06-02 09:19 (:0)
3 anonymous@Ars3n1c:~$ whoami
4 anonymous
5 anonymous@Ars3n1c:~$
```

## Qui est connecté au système ? :

`w`

```
1 anonymous@Ars3n1c:~$ w
2 11:56:29 up 2:39, 1 user, load average: 0,36, 3,21, 2,72
3 USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
4 anonymou tty7      :0            09:19   2:38m  4:03   0.78s /usr/bin/lxsession -s
5 LXDE -e LXDE
6 anonymous@Ars3n1c:~$
```

montre qui est connecté au système et le programme qu'il exécute.

## Je veux aller au dossier `/usr/share` (par exemple) :

`cd /usr/share`

`cd` pour "change directory".

## Je veux afficher l'organisation des disques :

`lsblk` est la commande à utiliser pour liste des blocs.





19	tmpfs	5,0M	4,0K	5,0M	1%	/run/lock
20	tmpfs	993M	0	993M	0%	/sys/fs/cgroup
21	/dev/sda1	453M	74M	352M	18%	/boot
22	/dev/mapper/TaGueule-maison	257G	60G	184G	25%	/home
23	tmpfs	199M	4,0K	199M	1%	/run/user/110
24	tmpfs	199M	12K	199M	1%	/run/user/1000
25	anonymous@Ars3n1c:~\$					

Vous aurez noté bien sûr que l'option `-h` modifie toujours les affichages pour les rendre humainement lisibles et intelligibles.

## Quelles sont les tâches en cours d'exécution ? :

`top` ou `htop`

```
top - 11:58:46 up 2:41, 1 user, load average: 0,32, 2,15, 2,38
Tasks: 184 total, 2 running, 129 sleeping, 0 stopped, 0 zombie
%Cpu(s): 5,6 us, 2,7 sy, 0,0 ni, 91,5 id, 0,2 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 2032872 total, 1159284 free, 438748 used, 434840 buff/cache
KiB Swap: 7999484 total, 7817468 free, 182016 used. 1406084 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
 32571 anonymo+ 20   0 550732 35748 27112 R  18,4  1,8   0:01.90 gnome-scre+
   661 root      20   0 411740 30068 19368 S  10,2  1,5   4:08.05 Xorg
 32569 anonymo+ 20   0 42236 3680 3036 R  1,6  0,2   0:01.81 top
  1183 anonymo+ 20   0 675416 29952 11280 S  1,0  1,5   0:32.27 pcmanfm
  1235 anonymo+ 20   0 522348 10468 6272 S  0,7  0,5   1:10.21 clipit
   221 root      20   0 0 0 0 S  0,3  0,0   0:05.98 dmccrypt_wr+
  1175 anonymo+ 20   0 201624 8388 3632 S  0,3  0,4   0:08.32 openbox
  1180 anonymo+ 20   0 788716 23704 14028 S  0,3  1,2   0:41.94 lxpanel
  2967 anonymo+ 20   0 328720 16880 12512 S  0,3  0,8   0:09.65 lxterminal
27634 root      20   0 0 0 0 I  0,3  0,0   0:00.91 kworker/0:2
27831 root      20   0 0 0 0 I  0,3  0,0   0:00.32 kworker/1:1
28638 root      0 -20 0 0 0 I  0,3  0,0   0:03.23 kworker/u9+
32570 root      20   0 0 0 0 I  0,3  0,0   0:00.03 kworker/u8+
    1 root      20   0 219608 5284 3648 S  0,0  0,3   0:24.38 systemd
    2 root      20   0 0 0 0 S  0,0  0,0   0:00.02 kthreadd
    4 root      0 -20 0 0 0 I  0,0  0,0   0:00.00 kworker/0:+
    6 root      0 -20 0 0 0 I  0,0  0,0   0:00.00 mm_percpu +
```

Affiche l'état du système et la table des processus en cours d'exécution ou en attente (table of processes). On peut aussi utiliser la commande `ps`

```
1 | ps
```

et pour afficher tout ce qui tourne (en arrière plan etc..)

```
1 | ps -a
2 | ps -e
3 | ps -x
4 | ps -auex
```

## Quels sont les fichiers et "nodes" ouverts ? :

`lsdf` pour list open files

Note : Il y a l'autocomplétion par la touche `[tab]` qui parcourt alphabétiquement les différents choix possible qui complètent le début saisi.

Deux liens physiques importants à connaître : `./` = dossier actuel, `../` = dossier parent.

## Maintenant entraînement à manipuler des dossiers :

Je veux créer un dossier `Mars` comme sous-dossier du dossier actuel : `mkdir Mars` (attention à la casse).

Tel quelle la commande `mkdir` oblige à créer les sous-arborescence les unes après les autres mais en passant quelques options on peut créer tout en une seule fois.

```
mkdir -p Mars/Satellites/Phobos
```

Depuis le dossier actuel je veux aller dans le dossier `Phobos`.

```
cd Mars/Satellite/Phobos
```

 ou bien :

```
cd ../Mars/Satellite/Phobos
```

 et bien sûr `pwd` pour voir où vous êtes.

Je veux remonter de plusieurs dossiers en une fois :

1. retour dans Phobos : `cd Phobos`
2. remontée des deux dossiers : `cd ../../` suivi de `pwd`.

Je veux remonter d'un dossier : `cd ..` la preuve par `pwd`

ET retourner maison

ex : `cd /usr/share/icons` puis vérification par `pwd` suivi de `cd` suivi de `pwd` pour vérifier à la fin.

Petit entraînement et préparation de la suite, saisie des caractères non reconnus nativement et des espaces :

```
1 cd
2 mkdir Mars/Satellites/Deimos
3 mkdir Mars/Satellites/Belle Mere
4 mkdir Mars/Satellites/Belle\ Soeur
5 mkdir 'Mars/Satellites/Beau Pere'
6 mkdir "Mars/Satellites/Beau Frere"
7 echo "Salut" > machin
```

Note : `cd` tout seul permet de revenir au dossier de base de l'utilisateur.

Note 2 : la dernière ligne commençant par `echo` sera expliquée plus tard.

## Créer plusieurs sous-dossiers en une seule fois :

```
1 mkdir {Jupiter,"Mini moi"}/Satellites/{Io,Europe,Ganymede,Callisto,Bidule}
```

Attention quand on supprime c'est d é f i n i t i f !

## Supprimer un dossier non vide et / ou une sour-arborescence



```
1 rmdir Mini\ moi
```

cela crée une erreur, du coup :

```
1 rm -fR ./Mini\ moi
```

suivi de `ls -lh` pour vérifier.

Attention au gag : en tant que root, taper un `rm -fR /` efface la totalité du disque dur :p

## La commande ls, . et ..

Testez ces commandes les unes après les autres en observant les résultats.

```
1 cd Mars
2 mkdir ../{Gaia,Venus,Mercure}
3 cd ..
4 ls
5 ls -l
6 ls -la
7 ls -lh
8 ls -lah
9 ls -rl
```

comment analyser la sortie d'un ls -l ?

```
1 ls -lh
2 ...
3 drwxr-xr-x 3 anonymous anonymous 4,0K mai 4 23:47 Jupiter
4 -rw-rw-r-- 1 anonymous anonymous 6 mai 4 23:56 machin
5 ...
```

type d'entrée	droits propriétaire	droits groupe propriétaire	droits autres	nombre de liens (dossiers + . + ..)	propriétaire	groupe propriétaire	taille sur disque	date de dernière modification	heure de dernière modification	nom du fichier
d	rwx	r-x	r-x	3	anonymous	anonymous	4,0K	mai 4	23:47	Jupiter
-	rw-	rw-	r--	1	anonymous	anonymous	6	mai 4	23:56	machin

le `-` veut dire que le fichier n'a rien de spécial (c'est un simple fichier d'archives) d -> dossier, S -> Suid bit activé r -> lecture x -> exécution (ou parcours d'un dossier) w -> écriture

## Les informations de droits sont modifiables par la commande

`chmod` (change mode) La commande permet de changer les modes d'un fichier, d'un dossier ou d'une arborescence, la marque "r" vaut 4, le "w" vaut 2 et le "x" vaut 1.

Par exemple : `chmod 0640 fichier.txt` permet de rendre le fichier `fichier.txt` accessible en lecture et écriture (6) pour le propriétaire, accessible en lecture (4) pour les membres du groupe associé au fichier et aucun accès (0) au reste des utilisateurs (sauf root évidemment).

## Les informations de propriété sont modifiables par la commande

`chown` (change owner) permet de changer l'utilisateur.

```
1 chown anonymous.anonymous bidule.txt
2           // attribue le fichier bidule.txt à anonymous et son groupe
3 chown anonymous.root bidule.txt
4           // attribue le fichier bidule.txt à anonymous et au groupe
   root
5 chown root.root bidule.txt
6           // attribue le fichier bidule.txt à root et son groupe
7 chown 1000:1000 bidule.txt
8           // attribue le fichier bidule à l'utilisateur d'UID 1000 et au groupe de GUID
   1000
9 chown -R anonymous.anonymous /home/anonymous
10          // attribue récursivement le dossier /home/anonymous à l'utilisateur
   anonymous et à son groupe
```

## Rendre un fichier invisible :

```
1 ls -l
2 mv machin .machin
3 ls -l
```

à retenir : `mv` = move (déplacement du fichier).

à retenir 2 : un fichier "invisible" sous linux est un fichier (ou un dossier) dont le nom commence par ".".

## Créer un fichier ligne par ligne directement depuis le shell sans éditeur :

```
1 cd
2 cd Jupiter
3 echo Ceci est le dossier de Jupiter > Readme
4 less Readme
```

sortie de less par la touche "Q" (note : c'est la même chose quand on a une mise à jour critique ou grave pour raison de sécurité, un message de post-installation s'affiche, et pour quitter on appuie sur "Q").

```
1 cat Readme
```

vous voyez la différence entre less et cat.

```
1 echo ----- > Readme
```

regardons le résultat :

```
1 | cat Readme
```

et oui ! Le début à été effacé. Utilisons la touche [flèche haut] pour revenir à la ligne `echo Ceci est le dossier de Jupiter > Readme` puis ensuite tapez `echo ..... >> Readme` et regardons par un `cat Readme`.

```
1 | echo "Bonjour"
```

*A Retenir : less et cat permettent d'afficher des contenus de fichiers, echo permet soit d'afficher du texte, soit d'écrire un texte vers un périphérique ou un fichier en mode écrasement > ou ajout >> .*

## Edition par bloc

il est possible directement d'utiliser la commande echo pour éditer un fichier ligne par ligne et l'enregistrer en une seule fois.

```
1 | echo "  
2 | `> écrivez ici le texte que vous voulez mais sans pouvoir`  
3 | `> revenir à la ligne précédente. Chaque nouvelle ligne est`  
4 | `> précédée du signe > mais cela permet quand même d'éditer`  
5 | `> un fichier au fur et à mesure et quand c'est fini on a 3`  
6 | `> choix disponibles : annuler par [CTRL]+[Z]`  
7 | `> finir et afficher le texte à l'écran par [CTRL]+[D]`  
8 | `> finir et enregistrer dans un fichier en une seule fois`  
9 | `> par la redirection finale " >> Readme`
```

regardez ensuite le résultat obtenu par : `cat Readme`.

Note : vous pouvez imaginer les possibilités de cette option en cas de manque d'éditeur sur un système ultraminimaliste.

## Copier un fichier (Readme) dans un autre dossier

```
1 | cp Readme ../Mars/
```

L'option "-v" rend la commande verbeuse.

## Copier un dossier et son contenu en une fois

```
1 | cd ..  
2 | pwd (vérification si on est au dessus de Jupiter)  
3 | cp -r Jupiter Saturne  
4 | ls -l Saturne (pour vérifier que tout à été copié).
```

## Effacer un dossier et son contenu

```
1 rm -fR Saturne
```

## Filtrer une sortie d'après un mot clé :

```
1 grep
```

ex : Afficher tous les paquets de debian contenant le mot clé "theme" puis filtrer les sorties pour n'afficher que celles contenant "gtk"

```
1 apt-cache search theme | grep gtk
```

Maintenant qu'on a fini de faire mumuse avec les sorties, les fichiers et les dossiers on va passer aux commandes dangereuses.

Notez aussi l'utilisation du tuyau `pipe` permet de chaîner les commandes (le résultat de la première est envoyé vers la 2nde etc...)

## Les téléchargements :

### Télécharger en mode texte :

Suivant les distributions il y a par défaut l'un, l'autre ou les deux, les `....` doivent être remplacés par l'url du fichier. À noter que `wget` permet d'aspirer des sites webs alors que `curl` n'est prévu que pour un fichier à chaque fois, et qu'il est possible de forcer `curl` à passer par des réseaux anonymisants tel que tor.

```
1 wget -c ....
2 curl -L -C- -O ....
```

## Passer en super utilisateur / administrateur

Les mesures de sécurité élémentaires font qu'on sépare

si sudo disponible :

```
1 sudo su
```

si sudo pas disponibles :

```
1 su -
```

## exécuter 1 commande unique en superutilisateur :

```
1 sudo commande options
2 su -c "commande options"
```



ATTENTION : En tant que root vous avez le pouvoir de vie et de mort sur le système et il faut savoir que les applications unix ne disent rien si tout s'est passé comme prévu (si elles donnent des informations c'est qu'il y a une erreur). Si vous voulez forcer le bavardage des commandes alors il faut passer l'option "-v" la plupart du temps (v pour verbose c'est à dire verbeux).

## Enchaînement des commandes : & et &&

& : les commandes s'exécutent en parallèle dans des sous-shells

&& : la seconde commande s'exécute si la première renvoie un code d'erreur "0" (donc pas d'erreur).

Par exemple :

```
1 anonymous@Ars3n1c: ~$ echo "hello" & echo "salut"
2 [1] 1725
3 salut
4 hello
5 anonymous@Ars3n1c: ~$ echo "hello" && echo "salut"
6 hello
7 salut
8 [1]+  Fini                  echo "hello"
9 anonymous@Ars3n1c: ~$
```

Dans le premier cas les deux commandes s'exécutent en simultanée (le [1] 1725 indique que la 2e commande est envoyée dans un sous-terminal dont le PID (process ID) est 1725). La 2nde commande a été plus rapide que la première. Dans le 2nd cas, les commandes s'exécutent dans l'ordre car la 2nde commande s'exécute seulement si la première n'a pas posé problème.

Ce type de chaînage est pratique pour effectuer des opérations de mises à jour automatisées plus ou moins contrôlées :

```
1 root@Ars3n1c: ~# apt update && apt clean && apt upgrade -y && apt clean && init 0
```

Cette série de commandes fait :

- une mise à jour de la liste des programmes `apt update` puis si l'exécution est correcte ;
- un nettoyage du cache local des paquets `apt clean` puis si tout s'est bien passé ;
- une mise à jour du système en répondant "Oui" automatiquement si une question est posée grâce à la commande `apt upgrade -y` et si tout s'est bien passé suivra ...
- un nettoyage du cache des paquets téléchargés grâce à `apt clean` puis si tout est ok, alors ...
- l'ordinateur sera éteint aussitôt après `init 0`.

## Avoir de l'aide sur une commande

```
1 man commande
```

La commande `man` affiche une aide formatée, pendant longtemps, quand les connexions internet étaient coûteuses et lentes, l'administrateur ou l'administratrice d'un système Unix/Linux utilisait principalement les pages `man` (c'est le nom donné à toutes ces aides des commandes).

## Mettre à jour le système

debian et clones (\*buntu ; lxle; !#; kali; tails; ...):

```
1 apt update
2 apt upgrade
```

## redhat et fedora et clones

```
1 yum ou dnf :
2 dnf system-upgrade clean
3 dnf clean all
4 dnf upgrade
5 dnf system-upgrade download --releasever=28 --allowdowngrade --best
6 dnf system-upgrade reboot
```

## les distributions "snap"-isées

Snap est un nouveau système d'emballage. Mon avis sur le sujet sera exprimé une prochaine fois lors d'une présentation dédiée à ubuntu 18.04 LTS

```
1 snap refresh
```

## archlinux, parabola gnu/linux-libre, manjaro (pour les paquets qui ne sont pas dans AUR)

```
1 pacman -Syu
```

et pour les paquets dans AUR en plus des autres paquets des dépôts par défaut.

```
1 yaourt -Syua
```

## voidlinux

```
1 xbps-install -Su
```

## nettoyer le cache des paquets

```
1 debian et clones : apt clean
2 archlinux et clones : pacman -Sc / (yaourt -Sc ?)
3 voidlinux : xbps-remove -O (attention à la casse)
4 fedora, redhat : dnf clean all
5 snap-isées : mes recherches n'ont rien montré'
```

# Vider les versions obsolètes et / ou le cache des paquets et orphelins

Tout comme les opérations de mise à jour ces opérations de nettoyage des paquets obsolètes ou du cache inutile.

## debian et clones

```
1 apt autoremove --purge (orphelins / dépendances inutiles)
2 apt-get autoremove --purge (raspberry pi, vieilles versions de debian)
```

A noter: `aptitude` ou `dselect` → applications en curses comparables graphiquement à synaptic / kynaptic

## archlinux et clones

```
1 pacman -Sc
2 pacman -Scc
3 pacman -Rsn $(pacman -Qdtq)
```

La première commande nettoie le cache des fichiers antérieurs dont on a une version plus récente, la deuxième commande nettoie quant à la troisième elle recherche tous les paquets orphelins ou inutiles pour faire fonctionner d'autres paquets installés.

## voidlinux

```
1 xbps-remove -O
2 xbps-remove -Oo
```

La première commande supprime les paquets obsolètes et la 2nde les paquets orphelins aussi.

Pour les autres distributions telles que les fedora et autres snaps à vous de chercher :p

## supprimer un paquet unique (avec ou sans dépendances)

```
1 debian et clones :
2 apt remove --purge nom-du-paquet
3 apt autoremove --purge (pour les dépendances)
4
5 Archlinux et clones :
6 pacman -Rs
7 pacman -Rdd
8 pacman -Rsn $(pacman -Qdtq)
9
10 Voidlinux :
11 xbps-remove paquet
12
13 Versions snaps :
14 snap remove paquet
15
```

## Quelques utilitaires

- mc

midnight commander est le logiciel qui s'exécute par la commande `mc` c'est un véritable couteau-suisse.

Rapide. Simple. Capable d'édition de visualisation, copie, effacement et déplacement, il fonctionne en double-fenêtre à la sauce du mythique Norton Commander.

Gauche				Droite			
Fichier				Options			
Commande				~/Bureau/En Cours			
ln	Nom	Taille	Date de Modifi	ln	Nom	Taille	Date de Modifi
.	.pki	4096	27 avril 15:30	.	experie~es.png	50535	30 avril 07:36
.	.purple	4096	9 mai 22:55	.	image000.png	27589	30 avril 07:36
.	.ssh	4096	28 avril 23:04	.	image000s.png	21949	30 avril 07:36
.	.texlive2017	4096	30 avril 20:22	.	les-3-t~es.png	355534	30 avril 00:59
.	/Aurora	4096	9 mai 21:07	.	methode~ue.odt	16669	6 mai 22:26
.	/Bureau	4096	9 mai 22:38	.	methode~ue.pdf	29041	6 mai 22:26
.	/Desktop	4096	27 avril 19:50	.	paranoi~de.txt	141473	30 avril 07:36
.	/Documents	12288	8 mai 23:15	.	paranoi~de.mkd	140772	30 avril 07:36
.	/Downloads	4096	27 avril 19:50	.	present~ed.odp	21218K	30 avril 07:36
.	/Gentoo_amd64	4096	8 mai 14:40	.	present~cy.odp	21721K	30 avril 07:36
.	/Images	4096	8 mai 21:53	.	present~cy.pdf	1272206	30 avril 07:36
.	/Localmount	4096	9 mai 23:05	.	revisio~ts.pdf	256061	2 mai 07:53
.	/Localmount2	4096	9 mai 23:05	.	sunage1.en.png	54469	29 avril 23:07
.	/Musique	4096	7 mai 12:21	.	traject~es.png	8867	30 avril 00:24
.	/Nightly	4096	9 mai 13:02	.	vademec~e4.mkd	2483	30 avril 07:36

/Localmount 77G/144G (53%)      image000s.png 77G/144G (53%)  
 Astuce: Vous pouvez désactiver toutes les demandes de confirmation dans le menu  
 anonymous@lvclf3r:~/Bureau/En Cours\$ [^]  
 1Aide 2Menu 3Voir 4Modif 5Copier 6RenDep 7CréRep 8Suppr 9Men~ér 10Qui~er

- nano

C'est devenu peu à peu l'éditeur de base des distributions courantes. Ici l'exécution de la commande `nano - cil`



```

GNU nano 2.9.3                                Nouvel espace                                Modifié
1 Nano est un logiciel capable de vous permettre l'édition de documents
2 textuels simples mais tout à fait utiles pour de l'échange qui passe
3 par ce medium.
4
5 Il sait reconnaître aussi les colorations syntaxiques grâce aux extensions
6 de fichiers que vous enregistrez.
7
8 Certaines options (ici -cil) permettent l'affichage des lignes à gauche,
9 la position du curseur et les informations sur le contenu du fichier en bas
10 une barre en dessous avec les commandes et des informations sur le fichier
11 en haut.
12

[ ligne 11/12 (91%), col. 9/9 (100%), car. 497/498 (99%) ]
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^J Justifier ^C Pos. cur.
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^T Orthograp.^_ Aller lig.

```

- moc / ncpdcpb

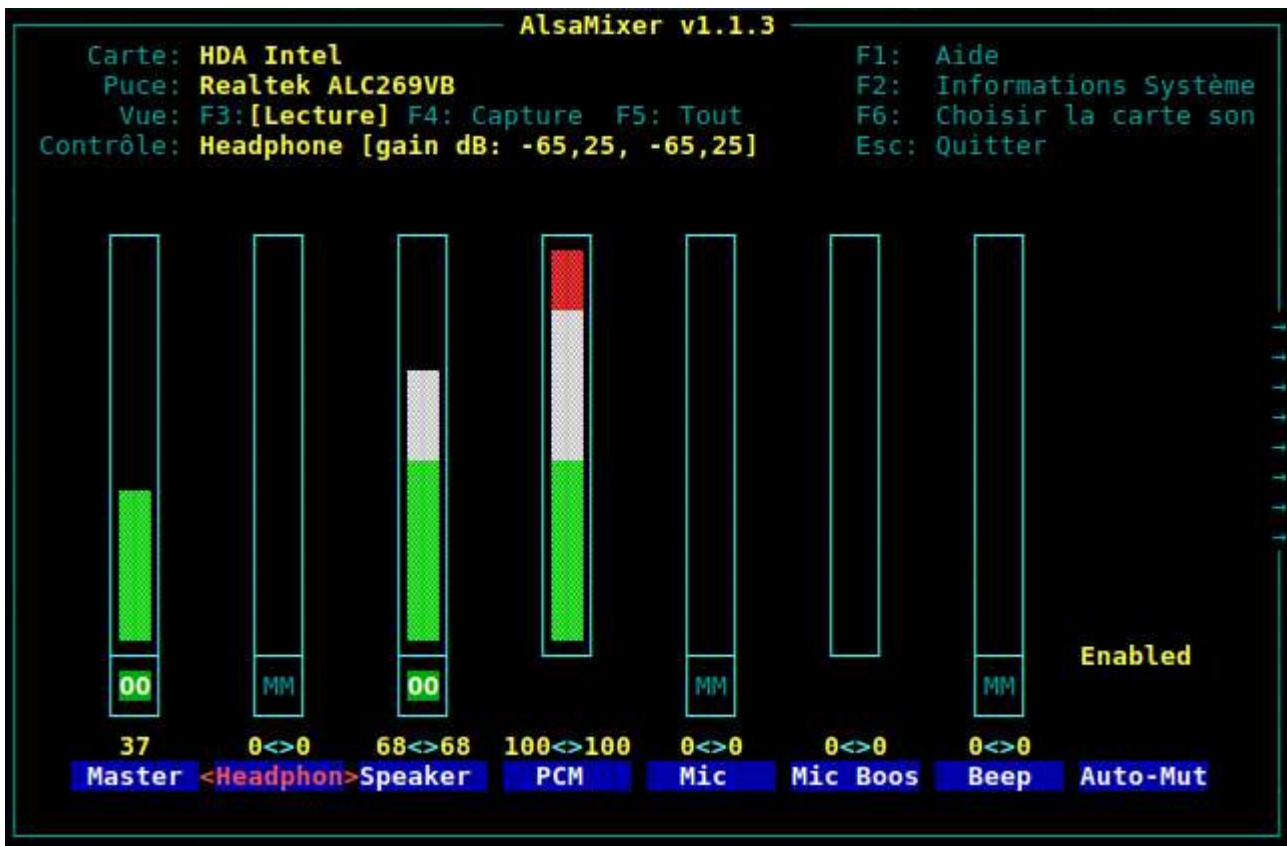
```

|-----|/home/anonymous/Musique/Clint Mansell|-----
../
Clint Mansell - Moon OST #10 - Sacrifice-vbsCRHUM6ho.flac      [      |FLC]
Clint Mansell - Moon OST #11 - We're Going Home-fZsW0_cX0rI.flac [      |FLC]
Clint Mansell - Moon OST #12 - Welcome to Lunar Industries (Three Y[      |FLC]
Clint Mansell - Moon OST #1 - Welcome to Lunar Industries- lAfMT5FI[07:14|FLC]
Clint Mansell - Moon OST #2 - Two Weeks Counting-oOwlmpTPH9A.flac [      |FLC]
Clint Mansell - Moon OST #3 - I'm Sam Bell-hABLaUbNME4.flac   [      |FLC]
Clint Mansell - Moon OST #4 - I'm Sam Bell, Too-aNKusMCaxvs.flac [      |FLC]
Clint Mansell - Moon OST #5 - Memories (Someone We'll Never Know)-b[      |FLC]
Clint Mansell - Moon OST #6 - Are You Receiving-RyNVZ3blKtg.flac [      |FLC]
Clint Mansell - Moon OST #7 - Can't Get There From Here-Jl8odSveiuM[      |FLC]
Clint Mansell - Moon OST #8 - We're Not Programs, Gerty, We're Peop[      |FLC]
Clint Mansell - Moon OST #9 - The Nursery-0lqRdN01FCU.flac    [      |FLC]

|-----|Playing...|-----|Master 100%|-----|>00:07:14|
> Clint Mansell - Moon OST #1 - Welcome to Lunar Industries- lAfMT5FIZE.flac
00:57 06:17 [07:14] 48kHz 901kbps [STEREO] [NET] [SHUFFLE] [REPEAT] [NEXT]

```

- alsamixer



- links / lynx / w3m

Ahhh les joies de la navigation en mode texte.... si touchante et nostalgique...

Debian -- Le système d'exploitation universel (p1 of 6)

Link: [author](#)  
 Debian

[ Recherche ]

Sauter le menu

- \* À propos de Debian
- \* Obtenir Debian
- \* Assistance
- \* Le coin du développeur

Télécharger Debian 9.4 (installation par le réseau pour PC 64 bits)

Debian

Debian est un système d'exploitation libre pour votre ordinateur. Un système d'exploitation est une suite de programmes de base et d'utilitaires permettant à un ordinateur de fonctionner.

Debian est bien plus qu'un simple système d'exploitation : il est fourni avec plus de 51000 paquets . Ce sont des composants logiciels précompilés, assemblés dans un format ingénieux conçu pour une installation aisée sur une machine. Lire la suite...

<mailto:webmaster@debian.org/>

Ces trois navigateurs permettent ainsi de naviguer, sans JS, sans CSS, sans images (si dans un terminal), à l'ancienne...

- dhcpcd ou dhclient (debian)

C'est avec ces commandes qu'on peut configurer la connexion réseau filaire automatiquement

- nm-cli ou wicd ou wifi-menu (archlinux et clones)

Ces commandes sont là pour configurer le wifi en mode texte.

- ssh

```
1 anonymous@Ars3n1c:~$ ssh -l memyselfi 192.168.1.107
2 memyselfi@192.168.1.107's password:
3 Linux raspberrypi 4.14.34+ #1110 Mon Apr 16 14:51:42 BST 2018 armv6l
4
5 The programs included with the Debian GNU/Linux system are free software;
6 the exact distribution terms for each program are described in the
7 individual files in /usr/share/doc/*/copyright.
8
9 Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
10 permitted by applicable law.
11 Last login: Thu May 31 00:41:10 2018 from 192.168.1.19
12 memyselfi@raspberrypi ~ $
```

- sshfs

Merci à Tristan de m'avoir fait découvrir cette commande. Un régal ! Elle permet l'équivalent d'un ssh:// ou d'un sftp:// que j'utilise très souvent dans mon explorateur de fichiers graphique (pcmanfm). Grâce à cela on peut attacher une ressource du réseau à un point d'ancrage sur sa propre machine et l'explorer comme un simple sous-dossier (attention cependant aux latences de la connexion qui influent sur la réactivité).

```
1 sshfs login@ressource:/chemin point-ancrage-local
```

- fusermount

```
1 fusermount -u point-ancrage-local
```

permet de démonter le point d'ancrage

- gpg

gpg est un véritable petit chenapan qui permet de chiffrer des données, des fichiers, etc.....

- systemctl / sv

la commande `systemctl` (anciennement `service` pour les systèmes) permet de démarrer, d'arrêter ou de rendre automatique au démarrage certains programmes du système appelés services.

La commande `sv` fait de même pour les systèmes utilisant `runit` comme système de démarrage (typique de voidlinux), d'autres utilisent encore la commande `rc`.

- ln

Avec la commande `ln source raccourci` avec quelques options type `-s` ou `-sf`

## Gérer les périphériques et les liens

---

Formatage, partitionnement, vérification, gravure d'iso, création d'iso...

### Partitionnement de périphériques (disques, clés, etc...)

#### Partitionner un disque (inférieur à 2,3 To)

```
1 fdisk /dev/...
2 cfdisk /dev/...
```

#### Partitionner un disque (supérieur à 2,3 To)

```
1 gpt ...
```

### Formater une partition

#### type ext

```
1 mkfs.ext2 /dev/....
2 mkfs.ext3 /dev/....
3 mkfs.ext4 /dev/....
```

#### type swap

```
1 mkswap /dev/....
```

#### Formater une clé (par exemple) en FAT32 ou EFI

```
1 mkfs.vfat -n NOM /dev/....
```

### Créer un lien symbolique

```
1 ln -s cible alias
```

### Graver une image

```
1 dd status=progress ... if=... of=....
```

### Mounter un périphérique manuellement



C'est à l'aide de la commande `mount` qu'on pourra effectuer cela. Cette commande permet d'accrocher à un endroit de l'arborescence un support physique et la commande `umount` permet de le décrocher.

## un périphérique réel

Imaginons une clé USB branchée sur le système, son noeud système (la manière dont le noyau linux communique avec elle) est `/dev/sdb`. Elle possède une partition de type `FAT32` dont la désignation est `/dev/sdb1` (la désignation se fait automatiquement par le système). Pour avoir accès à ces infos on utilise bien sûr la commande `lsblk` et ensuite je décide de *mounter* cette clé dans le dossier `/mnt` :

```
1 | mount /dev/sdb1 /mnt
```

Les noyaux de Linux sont capables de reconnaître automatiquement bon nombre de systèmes de fichier aussi l'option `-t . . . .` n'est pas nécessaire.

## un ramdisk

Le ramdisk ou disque de mémoire volatile est vraiment très pratique. Il permet de créer dans la mémoire volatile de l'ordinateur un espace assimilé par le système à un stockage permanent, en gros, un peu de ram devient un peu de disque. Quel(s) intérêt(s) ? Ils sont multiples tout autant légaux que non.

Dans les choses légales citons :

- économiser des cycles d'écriture sur un SSD
- aider un vieux (ou un lent) disque dur à trouver un souffle de jeunesse
- virtualiser
- etc.

Dans le cas d'un usage à but illégal, sachez tout de même que la RAM est vulnérable aux attaques de type "Cold Boot" et que contrairement à la légende une RAM ne s'efface pas à l'extinction mais quelques dizaines de secondes plus tard, plus si elle est vite refroidie.

Voici quand même la commande :

```
1 | mount -t tmpfs tmpfs point-de-montage -o size=xxxx
```

Attention : si l'option `-o size=xxxx` n'est pas précisée, le système alloue la moitié de la mémoire vive disponible à l'espace de stockage. La taille `xxxx` doit contenir l'unité : k, M, G ...

## une ISO en loop

```
1 | ls /mnt
2 | mount -o loop -t iso 9660 monfichierimage.iso /mnt
3 | ls /mnt
```

Par cette commande `monfichierimage.iso` qui est un fichier contenant une copie bit-à-bit d'un support de type CDRROM a été monté dans le dossier `/mnt` qui est vide au départ et non vide à l'arrivée. Bien sûr on finit proprement par le démonter :

```
1 | umount /mnt
```

## Éteindre et redémarrer

---

suivant les distributions le droit d'éteindre ou de redémarrer le système n'est pas permis directement à l'utilisateur lambda. D'autres distributions permettent l'accès à cette commande. La commande est "shutdown" à la quelle peuvent être passés des arguments divers. Il y a bien sûr la commande "init" ou la commande "telinit" aussi.

Notez que la commande "init" n'est pas forcément existante dans toutes les distributions par contre shutdown et telinit elles oui. Voici quelques options de son utilisation.

### Extinction directe du système :

```
1 | shutdown -h now ou /sbin/telinit 0 ou init 0
```

### Extinction différée du système dans un délais (en minutes) :

```
1 | shutdown -h +10
```

### Extinction différée du système à une heure spécifique :

```
1 | shutdown -h -t 23:35:00 (à vérifier)
```

## Redémarrage du système

```
shutdown -r now
```

ou

```
/sbin/telinit 6
```

ou

```
init 6
```

---

Sources :

manpages

<http://toastytech.com/guis/>

[https://fr.wikipedia.org/wiki/Xerox\\_Star](https://fr.wikipedia.org/wiki/Xerox_Star)

[https://fr.wikipedia.org/wiki/Interface\\_graphique](https://fr.wikipedia.org/wiki/Interface_graphique)

<https://fr.wikipedia.org/wiki/Amiga>

<http://www.catb.org/~esr/writings/taouu/html/ch02s05.html>

