

«Intelligence is the ability to avoid doing work, yet getting the work done.»

#Linus Torvalds



GNU/Linux

Terminal - Network

-
- ✓ Les bases du système GNU/Linux
 - ✓ Notion sur Internet et réseaux
-

SOMMAIRE

#HISTORIQUE & #BIOGRAPHIE & #PROJET_GNU & #LINUX



[HISTORIQUE] ----- Page 1

Bell Labs (AT&T), Famille Système V, Sun Solaris, Famille BSD, MacOS X, GNU/Linux

[BIOGRAPHIE] ----- Page 2

Ken Thompson ----- Page 2

Dennis Ritchie ----- Page 2 - 3

Bill Joy ----- Page 4

Richard Stallman ----- Page 5 - 6

[PROJET GNU] ----- Page 7

Les logiciels libres ----- Page 7

Les logiciels libres sous licence BSD

La licence GNU General Public License (GPL) ----- Page 8

GNU Lesser General Public License

Logiciel libre et open source

Abstraction du système ... ----- Page 9

Linus Torvalds ----- Page 10 - 11

[LINUX] ----- Page 12

Méthode de développement (Linux) ----- Page 12

Fonctionnalités de Linux (Architecture interne) ----- Page 13

Noyau

Applications ----- Page 14

Compatibilité avec d'autres systèmes

Disponibilité de Linux ----- Page 15

Sources d'informations sur Linux ----- Page 16

Evolution future de Linux

Distributions ----- Page 17

- Distributions commerciales (RedHat, Suse, Mandriva ...) ----- Page 17 - 18
- Distributions communautaires (Fedora, Ubuntu, Debian, Kali, Arch ...) --- Page 17 - 18 -19
- Distributions live ----- Page 17 - 19
- Autres systèmes Unix libres (Hurd, BSD, Systeme V) ----- Page 19 -20

Système de fichier (Linux) ----- Page 21

- ARBORESCENCE (Données hiérarchique de type arbre.)
- LES SYMBOLES ASSOCIES A L'ARBORESCENCE (Repertoire parent ou repertoire courant) -Page 22
- LE « PROMPT » BASIC DE GNU/LINUX
- SPECIFICATION D'UN CHEMIN DANS LE SYSTEME DE FICHIERS (Chemin relatif et absolu) – Page 23
- HDD (Hard Disk Drive) ou LES DISQUES DURS (MBR, formatage, capacite, organisation, type de partition, type de bus) ----- Page 23 - 24
- LES FORMATS DES SYSTEMES DE FICHIERS (Sous Linux et sous Windows) --- Page 25
- POINTS DE MONTAGE (creation manuel)----- Page 25 - 26
- MISE EN PLACE D'UN NOUVEAU DISQUE (En 3 etapes) ----- Page 26 - 28
- LE REPERTOIRE SPECIAL « /proc » (cpuinfo, meminfo, loprts) ----- Page 28
- AUTRES PERIPHERIQUES SPECIAUX (/dev/null et dev/zero) ----- Page 28 - 29

#EDITEUR_DE_TEXTE & TERMINAUX et INTERPRÉTEURS & COMMANDES & COLOR-INDEX & EXTENSIONS & LIENS



[EDITEUR_DE_TEXTE]	-----	Page 30
Editeur à partir du terminal	-----	Page 30
Vi		
Nano	-----	Page 31
Editeur de texte (graphique)	-----	Page 31
Sublime text		
Atom	-----	Page 32
Notepad	-----	Page 32 -33
Visualisation (non destructeur) avec Less, More et Cat	-----	Page 33
Fonctionnement more	-----	Page 34
Fonctionnement less		
[TERMINAUX]	-----	Page 35
Le terminal, c'est quoi?	-----	Page 35
Pourquoi le terminal ?		
Les shell	-----	Page 36
Origines: sh et POSIX	-----	Page 36
CSH Le C shell		
TCSH Le TC shell		
Bash : l'interpréteur GNU		
Dash : l'interpréteur de Debian		
Zsh : encore plus de fonctionnalités	-----	Page 37
ipython		
Le Friendly Interactive SHell		

[COMMANDES] ----- Page 38 - 39

Creer un nouveau repertoire avec MKDIR

Copier un ou plusieur fichier(s)/repertoire(s) avec CP

Deplacer un ou plusieur fichier(s)/repertoire(s) avec MV

Creer un fichier avec TOUCH

Sipprimer un ou plusieur fichier(s)/repertoire(s) avec RM

Identifier votre emplacement dans l'arborescence du systeme de fichier GNU/Linux avec PWD

Changer de repertoire avec CD

Affichage de bash avec ECHO

Lister des éléments avec LS

Gestionnaire de disque avec DU et DF

[COLOR_INDEX] ----- Page 40

Bleu, Cyan, Blanc, Rouge, Vert, Rouge(fond), Violet

[EXTENSION] ----- Page 41

.conf et .sh

[LIENS] ----- Page 42 - 43

Les liens symboliques

Les liens physiques

La commande LN

#PAQUETAGES DES DISTRIBUTIONS #INTERNET & RESEAUX #IMPRESSION #GREP & FIND #REDIRECTION #PIPES #RSYNC #ARCHIVAGE & COMPRESSION



[PAQUETAGES DES DISTRIBUTIONS] ----- Page 44

CONCEPT DE GESTION DES PAQUETAGES ----- Page 44

Apt-get, Dpkg, rpm, zypper, yum

TROUVER DES PAQUETAGES ----- Page 45

Paquetages debian, Rpm find

IDENTIFIER LES PAQUETAGES

Distributions avec des paquetages RPM

Pour Debian

INFORMATION SUR LES PACKAGES ----- Page 46

Distributions basées sur RPM

Poue Debian

/etc/apt/sources.list (Explication et exemple) ----- Page 46 - 47

CD-ROM ----- Page 48

Gestion des paquets avec APT ----- Page 48 - 56

Installation d'un paquet ----- Page 48

Recherche des paquets ----- Page 49

Description d'un paquet ----- Page 50

Affichage des dépendances ----- Page 51

Installer les dépendances ----- Page 52

Identification du dépôt d'un paquet

Provenance d'un paquet ----- Page 53

Lister le contenu d'un paquet

Versions des paquets disponibles ----- Page 54

Mise à jour des listes de paquets

Mise à jour des dépôts ----- Page 55

Mise à jour de tous les paquets installés

Désinstallation d'un paquet ----- Page 56

Suppression des paquets d'installation

Gérer des paquets deb avec dpkg

GENERALITE ----- **Page 57****Internet****Notion de réseau** ----- **Page 58**

- Une histoire de mailles ----- Page 59
- Internet aujourd’hui ----- Page 60
- Comment communiquer ? ----- Page 61 - 62
- Le modèle OSI ----- Page 62 – 63 – 64 – 65 – 66 – 67
- Règle d’or du modèle OSI
- Le succès de TCP/IP face à OSI ----- Page 67
- Le modèle TCP/IP ----- Page 68 – 69
- Classification des réseaux ----- Page 70
- Installation et configuration du réseau ----- Page 70 - 71

CONNEXION A INTERNET (configuration du réseau sous Linux) ----- **Page 72****Network Manager** ----- **Page 72 - 82**

- Installation / réinstallation ----- Page 72
- Utilisation (sous Ubuntu) ----- Page 73 - 79
- Utilisation avancée (Network-Manager) ----- Page 79 - 82

Wi-Fi ----- **Page 83**

- Identifier sa carte réseau WIFI ----- Page 83 - 84
- Utilisation graphique du WIFI ----- Page 84
- Utilisation en ligne de commande du WIFI ----- Page 85
- Modification du fichier interfaces ----- Page 85 – 86 – 87 – 88 – 89 – 90
- WPA supplicant ----- Page 90 – 91 – 92 – 93 – 94 – 95 – 96 – 97 – 98
- Le script Wifichck ----- Page 98 - 99
- Utilisation avancée (WIFI) ----- Page 99
- Optimisation du WIFI
- Problèmes et cas particuliers ----- Page 99 – 100 – 101 – 102 – 103 – 104

Modems ----- **Page 105**

- Modems bas débit ----- Page 105
- Modems haut débit ----- Page 108
- Modems câble ----- Page 109

NAVIGATION SUR LE WEB (sous Linux) -----	Page 110
HTTP et navigation -----	Page 110
URL	
HTTP -----	Page 111
Clients -----	Page 111
Serveurs -----	Page 111 - 112
Les navigateurs web -----	Page 112
Les principaux navigateurs -----	Page 112 – 113 – 114 – 115
Navigateur en ligne de commande -----	Page 115
COURRIER ELECTRONIQUE (sous Linux) -----	Page 116
MESSAGERIE INSTANTANEE (sous Linux) -----	Page 116 – 117 – 118
ECHANGE DE FICHIER P2P (sous Linux) -----	Page 118
LES OUTILS -----	Page 119 - 120
LEXIQUE -----	Page 121
[IMPRESSION] -----	Page 122
COMMANDÉ D'IMPRESSION -----	Page 122
CONTRÔLE DE TRAVAUX D'IMPRESSION -----	Page 122 – 123
[GREP & FIND] -----	Page 123
GREP _ Recherche de contenu -----	Page 123
LES DIFFERENTS EXPRESSIONS RATIONNELS de GREP utiles	
FIND _ Recherche d'un fichier -----	Page 124
SYNTAX	
LES DIFFERENTS EXPRESSIONS RATIONNELS de FIND utiles	
Alternative de la commande find avec locate -----	Page 125
[REDIRECTION] -----	Page 126
Sortie standard -----	Page 126
L'erreur standard -----	Page 126 – 127
[PIPES] -----	Page 128
STD-IN	
STD-OUT	

[RSYNC] ----- Page 129

EXEMPLE D'UTILISATION ----- Page 129

Copie vers une machine distante

Copie depuis une machine distante à travers ssh ----- Page 130

[ARCHIVAGE & COMPRESSION] ----- Page 130

Archivage ----- Page 130

LES DIFFERENTES OPTIONS pour TAR

SYNTAX ----- Page 131

Compression ----- Page 131

#LE_SUPER-UTILISATEUR #PERMISSION & PROPRIETAIRE

#GESTIONS DES COMPTES UTILISATEURS

#GESTIONS DES GROUPES UTILISATEURS

=====

[LE SUPER-UTILISATEUR] ----- Page 132

Définition et utilisation ----- Page 132

[PERMISSION & PROPRIETAIRE] ----- Page 134

LES 4 TYPES D'ACCES (r, z, x et -) ----- Page 134

LES 3 GROUPES D'UTILISATEURS (User, group et others) ----- Page 135

LE SCHEMA GENERAL

MANIPULATION avec CHMOD, CHOWN et UMASK----- Page 136 -

 Les 2 modes de chmod ----- Page 136

 Mode symbolique

 Mode octal ----- Page 137

 LES DROITS ÉTENDUS SUID, SGID et Sticky Bit ----- Page 138

 SUID _ Set User ID

 SGID _ Set Group ID ----- Page 139

 Sticky Bit _ bit collant ----- Page 140

[UMASK] _ Le masque de protection ----- Page 141

[CHOWN] _ Propriétaire

[GESTIONS DES COMPTES UTILISATEURS] ----- Page 142

Création de l'utilisateur ----- Page 142

 Methode 1

 Méthode 2

Création de mots de passe pour un utilisateur

Suppression d'un compte de l'utilisateur (non connecté) ----- Page 143

Modification d'un compte de l'utilisateur

Autres commandes diverses sur les informations des utilisateurs

[GESTIONS DES GROUPES UTILISATEURS] ----- Page 144

Ajout d'un utilisateur à un groupe ----- Page 144 – 145 – 146

Lister tous les groupes (primaire et secondaires d'un utilisateur)

Création d'un nouveau groupe

Suppression d'un groupe

Autres commandes diverses sur les informations des groupes ----- Page 146

#PROCESSUS #VARIABLES_D'ENVIRONNEMENT #SYSLOG & #GLOBBING_BASH #TRUX #EXTRA



[PROCESSUS] ----- Page 147

Définition -----	Page 147
Données associée aux processus	
Programmes en tâche de fond	
Parfaite maîtrise des tâches grâce à #PS, #TOP et #KILL -----	Page 148 – 151
PS _ Liste de tous les processus -----	Page 148
TOP _ Activité en temps réel des processus-----	Page 149
KILL _ Arrêt de processus-----	Page 150 – 151

[VARIABLES D'ENVIRONNEMENT] ----- Page 152

Env -----	Page 152
Exemples de variables de shell	
Variables de shell (bash)	
Variables d'environnement (bash)	
Variables d'environnement standards -----	Page 152 – 153
Variables d'environnement \$PATH -----	Page 153
ALIAS -----	Page 154
Fichier .bashrc	

[SYSLOG] ----- Page 155

/var/log/syslog -----	Page 155
-----------------------	----------

[GLOBBING_BASH] ----- Page 155

L'étoile (*) -----	Page 155
? (variable_char)	
Le backslash (\)	

[TRUX] ----- Page 156 - 160

« LES TRICKS de LINUX POUR TRAVAILLER PLUS EFFICACEMENT. » -----	Page 156 – 157 – 158 – 159 – 160
--	----------------------------------

[EXTRA] ----- Page 161 - 164

Comment prononcer "Linux" ? ----- Page 161

Pages info

Recherche de ressources sur Internet ----- Page 161 - 162

Quelques applications de bureau ----- Page 162

Alternatives aux outils sous Windows

GNU / Linux à la maison ----- Page 162 - 163

Utiliser les distributions GNU / Linux ----- Page 163

Ne ratez pas le prochain train ! ----- Page 164

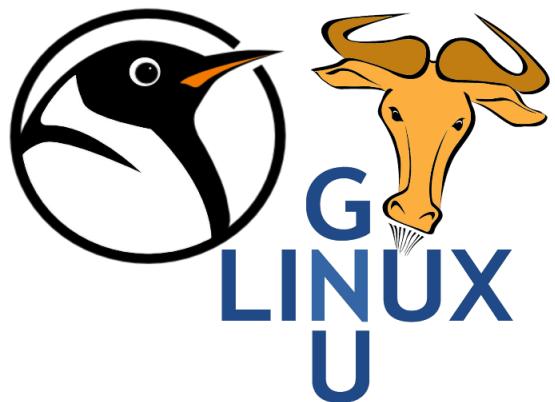
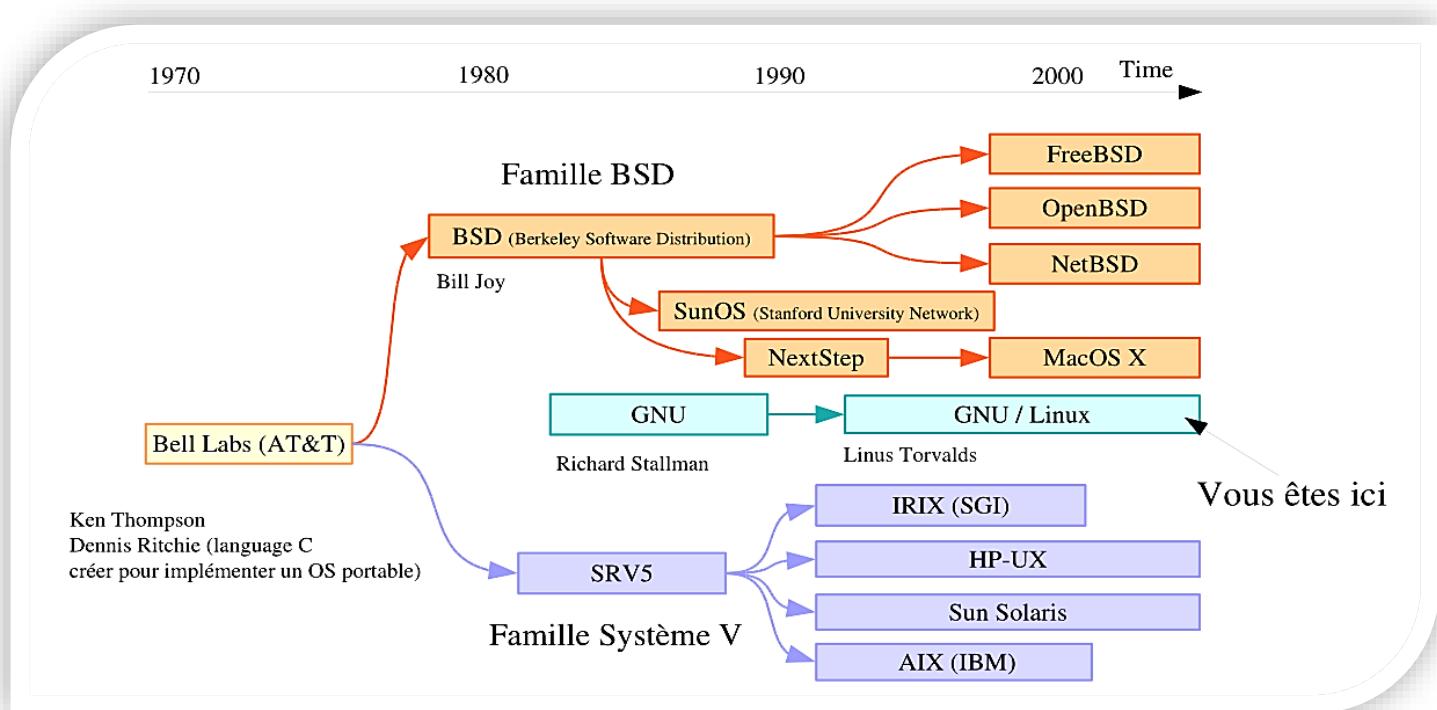
#HISTORIQUE #BIOGRAPHIE #PROJET_GNU &
#LINUX

« Savoir c'est pouvoir »

Historique



MacTM OS



Biographie

Ken Thompson



Kenneth Lane Thompson dit **Ken Thompson** (né le 4 février 1943 à La Nouvelle-Orléans) est un informaticien américain, concepteur des systèmes UNIX et Plan 9 ainsi que des langages B et Go.

Après avoir obtenu ses diplômes à l'université de Californie à Berkeley, il rejoint les Laboratoires Bell en 1966. Il y travaille sur les systèmes d'exploitation à temps partagé, notamment Multics, puis Unix à partir de 1969 et plus tard Plan 9. En 1970, il met au point le langage B, précurseur du C.

Il participe à de nombreuses évolutions du système Unix : portage sur PDP-11, réécriture en langage C, et introduction des *pipes*.

Dans les années 1970, il crée Belle avec Joseph Condon, un ordinateur spécialisé dans le jeu d'échecs. *Belle* remporte à plusieurs reprises le championnat nord-américain des ordinateurs d'échecs entre 1978 et 1986. Il est également à l'origine des recherches sur les tables de finales.

En 1979, AT&T publie dbm, un moteur de base de données qu'il a initié.



Il est également l'inventeur de l'éditeur *ed* et avec Rob Pike du codage des caractères UTF-8.

Il reçoit conjointement avec Dennis Ritchie le prix Turing de l'ACM en 1983, et la médaille nationale de la technologie des États-Unis en 1998 pour leur travail sur le système Unix. À la réception du prix il donne une conférence qui met en garde contre les chevaux de Troie.

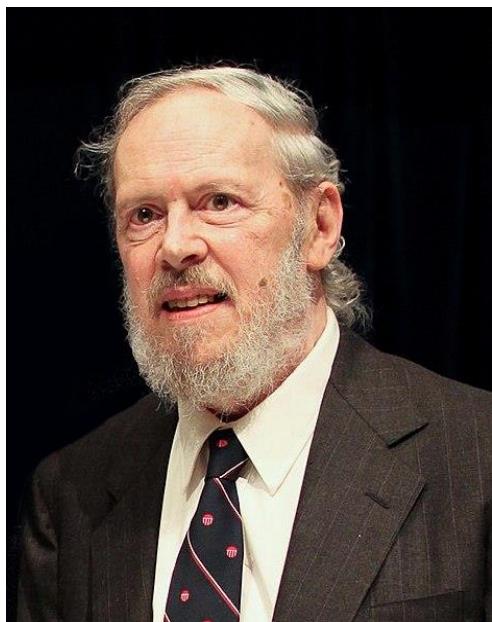
Il a pris sa retraite des Bell Labs en 2000. Il a travaillé à Entrisphere comme conseiller scientifique jusqu'en 2006 et travaille désormais chez Google.

C'est chez Google qu'il participe à la création d'un nouveau langage de programmation, issu du C appelé Go : « Chez Google, nous pensons que la programmation devrait être rapide, productive et surtout, « fun ». C'est pourquoi nous sommes ravis de proposer ce nouveau langage de programmation expérimental. Les opérations de compilation sont presque instantanées, et le code compilé propose une vitesse de fonctionnement proche de celle du C ».

Il est corécipiendaire avec Dennis Ritchie du prix japonais de 2011.

De par ses contributions, Ken Thompson est considéré par certains comme un *übergeek*.

Dennis Ritchie



Dennis MacAlistair Ritchie, né le 9 septembre 1941 à Bronxville dans l'État de New York et mort le 12 octobre 2011 à Berkeley Heights dans le New Jersey, est un des pionniers de l'informatique moderne, inventeur du langage C et co-développeur de Unix. Il est parfois désigné par *dmr*, son adresse électronique aux Laboratoires Bell.

Au début des années 1970, programmeur aux Laboratoires Bell, il travaille avec Ken Thompson au développement d'Unix. Le langage B de Thompson étant trop limité pour les besoins du nouveau système, Ritchie est amené à créer sur les mêmes bases le langage C. Par la suite, avec l'aide de Brian Kernighan, il promeut le langage et rédige notamment le livre de référence *The C Programming Language*.

Il reçoit conjointement avec Ken Thompson le prix Turing de l'ACM en 1983 pour leur travail sur le système Unix.

Son père, Alistair E. Ritchie, est un scientifique des Laboratoires Bell et connu pour le livre *The Design of Switching Circuits* sur la théorie des commutations de circuits, dont il est co-auteur. Dennis Ritchie étudie la physique ainsi que les mathématiques appliquées à Harvard, y obtenant un doctorat.

En 1967, il commence à travailler également aux Laboratoires Bell, où son père fait lui-même carrière. Un an plus tard, en 1968, il soutient sa thèse, nommée *Program Structure and Computational Complexity*, à Harvard sous la supervision de Patrick C. Fischer (en). Ritchie n'obtiendra cependant jamais son doctorat.

En 1978, Ritchie co-écrit avec Brian Kernighan *The C programming language*, un livre sur le langage C. C'est l'une de ses contributions les plus célèbres, puisque le livre est encore aujourd'hui connu sous le nom de *K&R*, les initiales des deux auteurs.

Ritchie travaille également dans les années suivantes avec Ken Thompson sur le système d'exploitation UNIX. Ritchie est notamment le contributeur principal du portage d'UNIX sur différentes machines et plateformes.

En 1983, Ritchie et Thompson reçoivent le prix Turing pour leurs travaux sur la théorie d'un système d'exploitation générique et l'implémentation du modèle sur le système UNIX. Le prix de Ritchie est par ailleurs intitulé « Reflections on Software Research ». En 1990, Ritchie et Thompson seront de nouveau récompensés, par la médaille *Richard W. Hamming* de l'IEEE pour le système UNIX et le langage C.

Le 21 avril 1999, Thompson et Ritchie reçoivent la *National Medal of Technology and Innovation* de 1998, par le président américain Bill Clinton, toujours pour l'invention d'UNIX et du langage C. Ces inventions ont, d'après le jury, « permis d'énormes avancées dans l'informatique matérielle et logicielle et les réseaux et a stimulé la croissance de toute l'industrie, confortant le leadership des États-Unis dans le domaine des systèmes d'information ».

De santé précaire, après le traitement d'un cancer de la prostate et de problèmes cardiaques, il meurt à l'âge de 70 ans à son domicile de Berkeley Heights dans l'État du New Jersey où, vivant seul, il est retrouvé le



mercredi 12 octobre 2011, inerte. La date de sa mort n'est pas connue avec certitude, mais pourrait se situer autour du 8 ou 9 octobre, selon les sources inspirées par les proches. Son décès est occulté par celui de Steve Jobs, survenu quelques jours auparavant, bien que Mac OS et iOS se basent sur les travaux de Dennis Ritchie effectués sur UNIX.

Bill Joy



William N. Joy (né le 8 novembre 1954 à Farmington Hills dans le Michigan), plus connu sous le nom de **Bill Joy**, est un informaticien américain. Il est notamment connu pour son travail de développement du système d'exploitation Unix BSD et pour avoir été l'un des cofondateurs de Sun Microsystems en 1982 avec Vinod Khosla, Scott McNealy et Andy Bechtolsheim. Il a travaillé en tant que scientifique en chef dans cette entreprise jusqu'en 2003.

Joy commence ses études à l'université du Michigan, où il obtient une licence (*Bachelor of Science*) en génie électrique, et les poursuit à partir de 1975 à l'université de Californie à Berkeley qui lui délivre un master d'ingénieur en génie électrique et en informatique en 1979.

À Berkeley, Joy joue un rôle important dans le développement de l'Unix de Berkeley, c'est-à-dire BSD, sur lequel sont basés un grand nombre de systèmes Unix modernes tels que FreeBSD, NetBSD et OpenBSD.

Certaines de ses plus importantes contributions sont la pile TCP/IP, l'éditeur vi et l'interpréteur de commandes csh.

Au début des années 1980, DARPA fait appel à la société BBN pour ajouter le support de TCP/IP, sur les conseils de Vinton Cerf et de Bob Kahn, à l'UNIX de Berkeley. On demande à Joy d'ajouter la pile développée par BBN dans BSD. Mais Joy s'y refuse, car il estime le support TCP/IP développé par BBN insatisfaisant, et décide de plutôt programmer sa propre pile TCP/IP. John Gage (en) décrit ainsi la suite des événements : « BBN avait un énorme contrat pour implémenter TCP/IP, mais leur travail ne marchait pas, et celui de Joy fonctionnait. Alors ils ont eu une réunion, et quand cet étudiant en T-shirt apparut, ils lui demandèrent *comment avez-vous fait ça ?* et Bill répondit *C'est très simple : vous lisez le protocole et vous écrivez le code* ». Cette version des événements est toutefois contestée, notamment par Rob Gurwitz, l'ingénieur de BBN responsable de l'implémentation de TCP/IP développée par cette entreprise.

En 1986, il reçoit le prix Grace Murray Hopper de l'ACM.

Joy est aussi une figure importante dans le développement des microprocesseurs SPARC, du langage Java et de Jini.

Le 9 septembre 2003, Sun publie un communiqué annonçant le départ de Bill Joy de la firme. D'après ce texte, celui-ci va « prendre du temps pour réfléchir à ce qu'il va faire » et n'a pas de « plan définitif ».

Depuis son article *pourquoi le futur n'a pas besoin de nous* consacré au futur de l'humanité dans *Wired*, il est régulièrement invité pour participer à des débats et des colloques sur les suites de cet article.



Richard Stallman



Richard Matthew Stallman, né le 16 mars 1953 à Manhattan, connu aussi sous les initiales **rms** (en minuscules), est un programmeur et militant du logiciel libre. Initiateur du mouvement du logiciel libre, il lance, en 1983, le projet GNU et la licence publique générale GNU connue aussi sous le sigle GPL. Il a popularisé le terme anglais « *copyleft* ». Programmeur renommé de la communauté informatique américaine et internationale, il a développé de nombreux logiciels dont les plus connus des développeurs sont l'éditeur de texte GNU Emacs, le compilateur C de GNU (GCC), le débogueur GNU (*gdb*), mais aussi, en collaboration avec Roland McGrath, le moteur de production GNU Make.

Depuis le milieu des années 1990, il consacre la majeure partie de son temps à la promotion des logiciels libres auprès de divers publics un peu partout dans le monde. Depuis quelques années, il fait campagne contre les brevets logiciels et la gestion des droits numériques (DRM). Le temps qu'il alloue encore à la programmation est consacré à GNU Emacs, bien qu'il ne soit plus le mainteneur principal depuis février 2008. Il gagne sa vie en partie avec les cachets de conférencier qu'on lui donne à l'occasion ou des prix qu'on lui remet.

En 2010, un récit biographique qui relate la vie de Richard Stallman paraît en français sous le titre *Richard Stallman et la révolution du logiciel libre*.

Doté de fortes capacités en science, il participe régulièrement à des rencontres de jeunes passionnés par les sciences. Il utilisa un ordinateur pour la première fois durant ses années de lycée en 1969. L'été suivant, à la fin de ses études secondaires, il fut engagé par le centre scientifique d'IBM de New York et il s'attaqua à l'écriture de son premier programme : le préprocesseur du langage de programmation PL/I destiné aux ordinateurs IBM 360.



En 1971, alors étudiant en physique et en mathématiques à l'université Harvard où il obtint d'excellents résultats, il devient *hacker* au département de recherche en intelligence artificielle du *Massachusetts Institute of Technology* (MIT) pendant des années. Ce milieu plaît davantage à Stallman, qui y apprécie notamment la souplesse de la politique d'accès aux ordinateurs.

En 1975, il met fin à ses études à Harvard pour se consacrer uniquement à la programmation. Il découvrit au MIT l'éthique des *hackers* : le partage des connaissances, le refus de l'autorité et le perfectionnisme. Cependant, peu à peu, l'ambiance des débuts change et il est de plus en plus rejeté par ses pairs qui acceptent des postes dans des entreprises créant du logiciel propriétaire. Il décide de maintenir à jour les fonctionnalités de la machine Lisp qui était passée sous l'égide de deux entreprises : Symbolics et LMI. Durant des mois, seul contre des dizaines de développeurs, à partir de la documentation il parvint à s'acquitter de ce projet, exploit qui sera reconnu par ses collègues.

Tout bascula au début des années 1980. Lorsque l'imprimante Xerox de son laboratoire se mit à avoir des soucis de bourrage, il décida d'améliorer le pilote existant afin de régler le problème. Il fut alors surpris de voir que celui-ci est uniquement disponible sous la forme d'un binaire; le code source en est inaccessible et personne ne veut le lui fournir.

Il comprend que l'éthique du *hacker* est en train de disparaître et qu'il faut agir. C'est ce qui explique en partie la conception du projet GNU quelques mois plus tard après un message publié sur Usenet.

Le journaliste américain Sam Williams (en) lui a consacré une biographie, *Free as in freedom*, que Stallman n'a jamais agréé. La communauté Framasoft lui ayant demandé son concours pour une édition en français de cette biographie, il a accepté d'en écrire la préface à condition de pouvoir annoter largement l'œuvre originale.

Il parle couramment anglais et français , assez couramment espagnol et un peu indonésien.



Projet GNU (philosophie)

« *GNU is Not Unix* »



Projet de réaliser un système à la Unix entièrement libre.

Lancé en 1984 par **Richard Stallman**, un chercheur du MIT, à une époque où les sources d'Unix n'étaient plus libres d'accès.

Composants initiaux : compilateur C (gcc), make (GNU make), Emacs, bibliothèque C (glibc), outils de base (ls, cp ...).

Cependant, en 1991, le projet GNU n'avait toujours pas de noyau et tournait sur des Unix propriétaires.

> Les logiciels libres

Les **logiciels libres** garantissent les **4 libertés** suivantes aux utilisateurs :

1. La liberté d'exécuter le programme, qu'elle que soit le but
2. la liberté d'étudier son fonctionnement, et de l'adapter à ses besoins.
3. La liberté de redistribuer des copies pour aider autrui.
4. La liberté d'améliorer le programme, et de partager ses améliorations avec autrui.

Voir <http://www.gnu.org/philosophy/free-sw.html>



> Les logiciels libres sous licence BSD

Bien sûr, elle garantit les 4 libertés aux utilisateurs.

Cependant, elle permet de rendre propriétaire le programme pour soi.

Exemple de licences : BSD, Apache

> La licence GNU General Public License (GPL)



La contribution majeure du projet GNU !

Les licences **Copyleft** utilisent la loi sur le **copyright** pour permettre aux auteurs d'exiger que toute modification d'un logiciel libre reste un logiciel libre.

La licence **GNU GPL** exige que toutes modifications et travaux dérivés soient aussi publiés sous licence **GPL** :

Voir <http://www.gnu.org/copyleft/copyleft.html>

- Ne s'appliquent qu'aux logiciels publiés
- Tous les programmes incluant le code **GPL** (que ce soit par lien statique ou dynamique) sont considérés comme une extension de ce code.

FAQ **GPL** : <http://www.gnu.org/licenses/gpl-faq.html>

> GNU Lesser General Public License

Licence Copyleft similaire à GNU GPL :

Les modifications doivent être échangées selon les mêmes conditions.

Cependant, permet l'utilisation au sein de programmes propriétaires.

Utiliser par plusieurs librairies de logiciels libres.

Exemples : glibc, GTK, Wine, SDL

Voir <http://www.gnu.org/copyleft/lesser.html>

> Logiciel libre et open source

Le mouvement des logiciels libres :

- Approche fondée sur des principes.
- Basé sur la liberté individuelle et l'utilité sociale de la coopération.

Voir <http://www.gnu.org/philosophy/free-software-for-freedom.html>

Le mouvement open source :

- Approche pragmatique.
- Invoque principalement les avantages de partager les sources et faire ses choix selon la supériorité technique.

Bien que les motivations de départ sont différentes, les deux mouvements travaillent très bien ensemble !

> De nos jours, les systèmes les plus puissants sont basés sur un design vieux de plus de 35 ans !

- o Ce qui est simple est beau.
- o Faire que chaque programme fasse une seule chose et bien.
- o Préférer la portabilité à l'efficacité.
- o Éviter les interfaces captives.

> Abstraction du système

Noyau : niveau matériel.

Shell (coquille) : niveau texte.

X Window : niveau graphique.

> Au départ, Unix a été créé pour les ordinateurs multi-utilisateurs.

Multi-utilisateur et sécurisé :

Par défaut, les utilisateurs ordinaires ne peuvent pas toucher aux fichiers d'autres utilisateurs.

En particulier, ils ne peuvent ni modifier les paramètres du système, ni supprimer des programmes, etc.

root :

utilisateur administrateur avec tous les priviléges.

- o Multi-tâches.
- o Supporte plusieurs processeurs.
- o Extrêmement flexible.
- o Prise en charge du réseau.
- o Portable.
- o Scalable.

Linus Torvalds



Linus Benedict Torvalds, né le 28 décembre 1969 à Helsinki en Finlande, est un informaticien américano-finlandais.

Il est connu pour avoir créé en 1991 (à 21 ans) le noyau Linux dont il continue de diriger le développement. Il en est considéré comme le « dictateur bienveillant à vie » (*Benevolent Dictator for Life*).

Il a également créé le logiciel de gestion de versions décentralisée Git et le logiciel d'enregistrement et de planification des plongées Subsurface (en). Il a été honoré, avec Shinya Yamanaka, du prix Millennium Technology, en 2012, décerné par la Technology Academy Finland (en) « en reconnaissance pour sa création d'un nouveau système d'exploitation open source pour les ordinateurs ayant conduit au noyau Linux, largement utilisé ». Il est également lauréat 2014 du Computer Pioneer Award, décerné par l'IEEE Computer Society.

Linus Torvalds fait partie de la communauté des Finlandais suédophones (*Finlandssvensk*), une population représentant 6 % des habitants de la Finlande. Son père, Nils Torvalds, est journaliste de télévision et de radio et aussi membre actif du Parti populaire suédois (le parti des suédophones finlandais), élu député européen en 2012. Sa mère, Anna, est également journaliste, et il compte parmi ses grands-parents le statisticien Leo Törnqvist et le poète Ole Torvalds.

Le prénom de Linus lui vient de Linus Pauling, chimiste américain récipiendaire du prix Nobel, mais Torvalds lui-même explique qu'il « pense avoir aussi été nommé d'après Linus, le personnage des Peanuts », ce qui le rend « à moitié chimiste nobélisé et à moitié personnage de BD portant une couverture ».

Il découvre l'informatique vers l'âge de 11 ans grâce à l'ordinateur de son grand-père, un Commodore VIC-20.



Torvalds étudie à l'université d'Helsinki entre 1988 et 1996, où il obtient un diplôme de master en informatique, auprès du groupe de recherche NODES. Sa carrière académique connaît une interruption lorsqu'il part effectuer son service militaire (obligatoire en Finlande). Il rejoint la brigade d'Uusimaa de l'armée finlandaise à l'été 1989, et sert onze mois, au rang de sous-lieutenant, chargé des calculs balistiques.

En reprenant ses études en 1990, il est amené à travailler avec un ordinateur DEC MicroVax sous Ultrix, ce qui constitue ses premiers contacts avec Unix. Après le Commodore VIC-20, il utilise un ordinateur Sinclair QL, dont il modifie le système d'exploitation ; il écrit un éditeur, un assembleur, et quelques jeux (ainsi que des bibliothèques graphiques pour Pac-Man).

Inspiré par le système Minix développé par Andrew S. Tanenbaum, il développe un noyau de système d'exploitation pour tirer pleinement parti des fonctionnalités de son nouvel ordinateur personnel, un compatible PC doté d'un microprocesseur Intel 80386. Ce noyau est celui du système d'exploitation GNU/Linux, sous licence GPL, et constitue sa thèse de master, intitulée *Linux: A Portable Operating System*.

Après avoir quitté l'Université d'Helsinki, il travaille de février 1997 à juin 2003 chez Transmeta, une société de la Silicon Valley qui fabrique des microprocesseurs à faible consommation électrique. Chez Transmeta, il était autorisé à consacrer une partie de son temps au développement du noyau Linux.

Il entre en 2003 aux Open Sources Development Labs (intégrés en 2007 à la Fondation Linux) et s'installe à Beaverton dans l'Oregon (États-Unis) avec sa famille.

Depuis, il se consacre principalement à diriger l'équipe de développement du noyau de Linux, en tant que « Dictateur bienveillant ». Son comportement notoirement intransigeant, franc et direct, s'il est évoqué comme une raison de la qualité du projet le conduit toutefois à faire des excuses publiques et faire une pause en 2018.

Contrairement à la majorité des adeptes du logiciel libre, Linus Torvalds reste relativement discret et refuse généralement de commenter les avantages et inconvénients des autres systèmes d'exploitation, comme le système Windows de Microsoft. Il réagit en revanche avec vigueur lorsque l'on remet en cause les qualités techniques du noyau Linux. Sa discussion houleuse avec Andrew S. Tanenbaum, opposant les noyaux Minix et Linux sur un choix de conception central (micro-noyau contre noyau monolithique), est d'ailleurs restée un exemple célèbre.

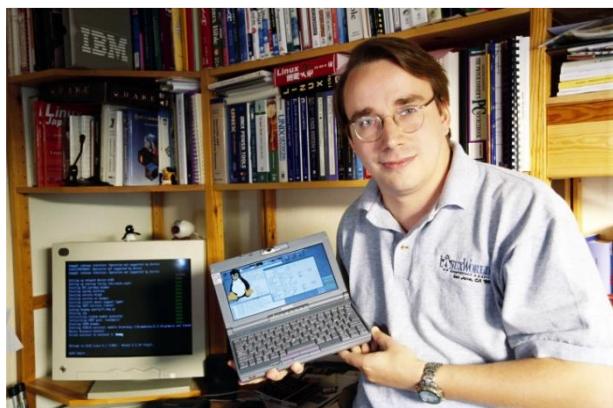
Il a également créé en avril 2005 le logiciel de gestion de versions Git, initialement prévu pour le développement du noyau Linux.

En septembre 2009, évoquant lors d'une intervention à la LinuxCon (en) le développement du noyau Linux, il déclare : « Nous n'avons certainement pas le noyau léger et hyper efficace envisagé lorsque j'ai commencé à développer le noyau Linux ».

En 2017, lors de l'Open Source Leadership Summit organisé par Jim Zemlin, le dirigeant de la Fondation Linux, il affirme que « Les discours sur l'innovation de l'industrie sont des conneries [bullshit] [...] n'importe qui peut innover, "penser différemment", c'est vide de sens. 99 % de l'innovation provient du travail réalisé [...] Tout ce tapage est loin du travail réel. Ce dernier se mesure dans les détails [...] les projets qui réussissent, c'est 99 % de transpiration et 1 % d'innovation ».

Linux

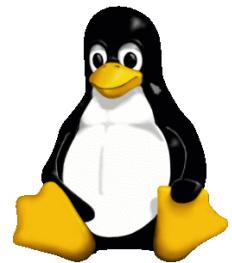
« Linus + Unix »



Au cours de l'année 1991, un étudiant finlandais, nommé Linus Torvalds, a acheté un micro-ordinateur de type PC, afin d'étudier la programmation du microprocesseur i386. Ne voulant pas être limité par MS/DOS, il a tout d'abord utilisé un clone d'Unix, peu cher, appelé Minix. Minix possède lui-même certaines limitations qui, bien que moins importantes que celles de MS/DOS, sont assez gênantes (limitation de la taille des exécutables à 64 kilo-octets, limitation des systèmes de fichiers à 64 mégaoctets, temps de réponse déplorable ...). Aussi, Linus Torvalds a commencé à ré-écrire certaines parties du système afin de lui ajouter des fonctionnalités et de le rendre plus efficace et a diffusé une distribution source de son travail via Internet. La première version de Linux (version 0.1 en août 1991) était née.

Cette première version était très limitée (elle ne comportait même pas de gestionnaire des disquettes) et n'était utilisable que sous Minix. Aussi, il est fort probable qu'elle ait été utilisée par très peu de personnes. Néanmoins, un petit nombre de "hackers" ont découvert, à cette époque, cet embryon de système et ont vu en lui la base d'un système exploitable. Aussi, un certain nombre de personnes ont commencé à travailler avec Linus Torvalds afin d'enrichir le noyau.

Au cours des quatre dernières années, le développement du noyau Linux, ainsi que des utilitaires nécessaires pour en faire un système compatible avec Unix, a été mené de manière intensive par un ensemble de programmeurs, situés aux quatre coins du monde, communiquant via le réseau Internet. Ces développeurs ont implémenté de nombreuses fonctionnalités qui font de Linux aujourd'hui un clone efficace d'Unix pour micro-ordinateurs PC-386, Amiga et Atari.



Méthode de développement (Linux)

La façon dont Linux a été développé (et continue à être développé) est assez originale. En effet, le développement de Linux n'est pas contrôlé par une organisation responsable du système : un ensemble de développeurs, répartis dans le monde entier, collabore pour étendre les fonctionnalités du système, le plus souvent en dialoguant via Internet. Tout programmeur disposant des compétences nécessaires, de temps libre, et d'une certaine dose de courage, peut participer au développement du système.

Bien que le nom "Linux" se réfère au noyau du système, le développement ne se confine pas à ce seul noyau : certaines équipes travaillent sur d'autres projets, comme la conception et le développement de nouveaux utilitaires ou encore le développement des bibliothèques partagées utilisées pour programmer.

Chaque équipe travaille selon une structure hiérarchique informelle : une personne est responsable d'un projet et plusieurs autres programmeurs participent au développement en contribuant du code. L'exemple

typique de cette méthode de développement est le noyau lui-même : Linus Torvalds maintient le noyau et c'est lui qui effectue les distributions sources quand il estime que le code est utilisable ; chaque partie importante du noyau (comme la gestion du réseau, les gestionnaires de périphériques, le système de fichiers ...) est sous la responsabilité d'un développeur qui centralise le travail d'autres programmeurs et les transmet à Linus Torvalds pour inclusion dans le noyau officiel.

Bien que cette méthode de développement puisse surprendre au premier abord, elle est très efficace. La totalité du noyau de Linux a été écrite par des volontaires, qui ont souvent passé des nuits entières à programmer et à corriger des bogues.

Le code développé dans le cadre de Linux est le plus souvent diffusé sous forme de programme source, sous la licence GNU ("General Public License", ou GPL). La GPL stipule que les programmes sources sont la propriété de leurs auteurs et qu'ils doivent être distribués sous forme de source. Cette licence autorise quiconque à revendre ces programmes, mais elle impose que tout utilisateur puisse avoir accès aux programmes sources. De plus, la GPL impose que toute modification de ces programmes ait couvert par la même licence, et donc que les programmes seront toujours librement disponibles.

Fonctionnalités de Linux (Architecture)

Le noyau



Linux offre toutes les fonctionnalités d'un clone Unix sur micro-ordinateurs PC-386. Il fournit un environnement de travail multi-utilisateurs, plusieurs personnes peuvent utiliser la machine au même moment, et multi-tâches, chaque utilisateur peut exécuter plusieurs programmes en parallèle.

Le système fonctionne en mode protégé, exécute du code 32 bits, contrairement à d'autres systèmes qui s'exécutent en mode 16 bits et sont donc moins performants..., et utilise les mécanismes de protection du processeur pour garantir qu'aucun processus ne peut perturber l'exécution des autres ou du système lui-même.

Le noyau implémente les sémantiques Unix : processus concurrents, chargement à la demande des programmes exécutables avec partage de pages et copie en écriture, pagination, systèmes de fichiers, support des protocoles réseau TCP/IP.

Il supporte, de plus, la majorité des périphériques existant dans le monde PC (y compris les cartes sonores) et permet de relire les partitions MS/DOS, OS/2 et tous les formats standards de CD/ROM.

Applications

Les librairies de développement dans Linux sont basées sur les librairies GNU, de la "*Free Software Fundation*". Ces librairies offrent un haut degré de comptabilité avec les différents "standards" Unix (Posix, BSD, System V), ce qui permet de compiler facilement tout type d'application disponible au niveau source pour Unix. Ces librairies existent sous forme de bibliothèques partagées, ce qui signifie que le code des fonctions de librairie n'est chargé qu'une seule fois en mémoire et que les programmes exécutables sont plus petits en taille sur les disques.

La plupart des utilitaires standards Unix sont disponibles sous Linux, aussi bien les commandes de base que des applications plus évoluées, comme les compilateurs et éditeurs de texte. La plupart de ces utilitaires sont des programmes GNU, qui supportent des extensions qu'on ne retrouve pas dans les versions BSD ou System V de ces programmes, mais qui restent compatibles avec ces dernières. Certains programmes, notamment les utilitaires réseau, sont des programmes BSD. En résumé, pratiquement tout programme Unix diffusé sous forme de source peut être compilé sous Linux et s'exécute parfaitement, grâce à la compatibilité implémentée dans le noyau et dans les librairies.

En plus des programmes standards, Linux supporte de "grosses" applications. On retrouve l'interface graphique X Window (XFree86 3 basé sur X11R6), un environnement de développement très complet comprenant toutes les bibliothèques standard, compilateurs et débogueurs disponibles sous Unix (C, C++, Objective-C, Smalltalk, Fortran, Pascal, Lisp, Scheme, Ada, gdb, ...). L'utilisateur dispose également d'outils très puissants de formatage de texte, comme nroff, TeX, et LaTeX.

Compatibilité avec d'autres systèmes

Linux n'est pas compatible directement avec les applications développées pour d'autres systèmes d'exploitation. Afin de permettre aux utilisateurs de Linux de bénéficier des applications qu'ils possèdent déjà, que ce soit sous MS/DOS, Windows ou des systèmes Unix commerciaux, des émulateurs sont en cours de développement et permettent déjà d'exécuter des applications "étrangères".

L'émulateur MS/DOS utilise le mode virtuel 8086 du processeur i386 pour exécuter des applications DOS. Il implémente les fonctionnalités de MS/DOS dans un processus et assure ainsi l'interfaçage entre l'application et le système en émulant les appels système effectués par le programme. À ce jour, de nombreuses applications fonctionnent correctement sous l'émulateur MS/DOS et la liste s'allonge tous les jours.

L'émulateur WINE est assez similaire à WABI, développé par Sun Microsystems : il permet d'exécuter des applications Windows en convertissant leurs appels graphiques en requêtes adressées à l'environnement X Window. À ce jour, seul un petit nombre d'applications Windows fonctionne correctement, mais le développement de WINE n'en est qu'à ses débuts et les progrès semblent prometteurs.

Le module de compatibilité iBCS2 permet d'exécuter des applications développées pour des systèmes Unix commerciaux sur micro-ordinateurs PC-386. Cet émulateur convertit les appels système se conformant au standard iBCS2 (qui définit le format des primitives systèmes ainsi que celui de leurs arguments) en appels natifs traités par le noyau Linux. Il est ainsi possible d'exécuter de manière transparente des programmes développés pour d'autres systèmes, comme SCO par exemple.

Le but de ces différents émulateurs est de permettre d'utiliser des applications commerciales sous Linux. Il faut désormais signaler que certains éditeurs de logiciels considèrent maintenant Linux comme un marché potentiel pour leurs produits et envisagent de porter leurs applications sous Linux. De la sorte, il est probable qu'un certain nombre d'applications commerciales tourneront bientôt en mode natif sous Linux, sans nécessiter d'émulateur. L'exemple le plus frappant de cette tendance consiste en le portage du jeu DOOM qui fonctionne sous Linux dans l'environnement X Window.

Disponibilité de Linux

Linux, et ses programmes associés, étant diffusés librement sous la licence GNU, ils sont disponibles au téléchargement sur de nombreux serveurs FTP de part le monde. Les trois sites primaires de distribution sont les suivants :

- tsx-11.mit.edu, dans le répertoire /pub/linux ;
- ftp.funet.fi, dans le répertoire /pub/OS/Linux ;
- sunsite.unc.edu, dans le répertoire /pub/Linux.

De nombreux serveurs FTP effectuent des miroirs réguliers de ces sites, permettant ainsi d'accéder rapidement aux mêmes données en raison de la proximité géographique. En France, on peut citer les serveurs suivants :

- ftp.ibp.fr, dans le répertoire /pub/linux ;
- ftp.fdn.org, dans le répertoire /pub/Linux ;
- ftp.loria.fr, dans le répertoire /pub/linux ;
- ftp.univ-angers.fr, dans le répertoire /pub/Linux.

De nombreuses distributions binaires du système sont également disponibles pour un prix raisonnable sous forme de CD-ROM. Ceci permet aux personnes intéressées d'installer un système Linux, même si elles n'ont pas accès au réseau Internet. Il est à noter que ces CD-ROM ne sont pas toujours à jour, vu l'évolution rapide de Linux, mais ils sont généralement édités périodiquement afin d'incorporer les dernières versions disponibles.

Sources d'informations sur Linux

Linux fait l'objet de plusieurs forums de discussion Usenet, aussi bien en langue anglaise (*comp.os.Linux.advocacy*, *comp.os.Linux.announce*, *comp.os.Linux.answers*, *comp.os.linux.development.apps*, *comp.os.Linux.development.system*, *comp.os.Linux.hardware*, *comp.os.Linux.networking*, *comp.os.linux.misc*, *comp.os.Linux.setup*, *comp.os.Linux.x*) qu'en français (*fr.comp.os.Linux*).

De nombreuses documentations sont désormais également accessibles. Les "HOWTO" décrivent, brièvement, mais clairement, l'installation et l'utilisation d'une partie du système. De plus, un groupe de documentation, le "*Linux Documentation Project*", publie des guides complets qui sont diffusés librement. Le serveur FTP [ftp.ibp.fr](ftp://ftp.ibp.fr) offre ces documentations dans le répertoire /pub/Linux/docs ainsi que leurs traductions françaises dans le répertoire /pub/Linux/french.

Enfin, il faut signaler que certains éditeurs commencent à publier des livres complets sur Linux. Les premiers livres publiés sur Linux étaient rédigés en allemand, mais il existe maintenant des livres en anglais sur Linux, certains de ces livres étant traduits en français.

Évolution future de Linux

La version actuelle du noyau de Linux est 1.2. Bien que Linux soit déjà très complet et compétitif avec des systèmes d'exploitation commerciaux, le développement du système continue.

Des portages du noyau sont en cours afin de réaliser des versions s'exécutant sur stations de travail Sparc et PC Alpha, en plus des versions actuelles fonctionnant sur micro-ordinateurs PC-386, Amiga et Atari. De nouvelles fonctionnalités sont également à l'étude et devraient être disponibles prochainement.

Le noyau n'est pas la seule partie du système à évoluer. Des équipes de développement travaillent pour étendre les fonctionnalités des utilitaires et des librairies. On peut citer, notamment, l'utilisation imminente du format de fichier exécutable ELF qui permettra la création de librairies dynamiques et une compatibilité plus grande avec les versions d'Unix commercial s'exécutant sur PC-386.

Distributions

Se charge de publier un ensemble cohérent de versions compatibles du noyau, de la bibliothèque C, des compilateurs, des outils... Cela représente un travail très conséquent !

Les outils sont disponibles sous forme de paquetages qui peuvent facilement être installés, supprimés ou mis à jour. Les dépendances entre outils sont gérées automatiquement.

■ Distributions commerciales :

Incluent de l'assistance technique. Le code source est libre, mais les binaires ne sont pas libres d'accès.

■ Distributions communautaires :

Sources et binaires sont librement disponibles. Fourni sans assistance technique obligatoire.

■ Distributions live :

Linux s'amorce à partir d'un périphérique de stockage (cd-rom, dvd-rom ou usb) et démarre tout à partir de ce périphérique.

Idéal pour essayer GNU / Linux et les applications des logiciels libres sans avoir à installer quoi que ce soit sur le disque dur !

Le système est prêt et démarre en 2-3 minutes. Plus rapide qu'installer et configurer GNU / Linux !

Aussi efficace pour récupérer des données lorsque le système d'origine ne démarre plus.

Utilise un système de compression pour mettre en mémoire 3 à 4 fois la capacité de stockage !

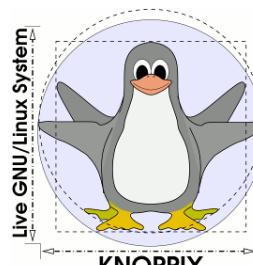
Liste des distributions live : <http://frozentech.com/content/livecd.php>

➤ Knoppix :

La plus populaire. Disponible en CD et DVD.

Idéale pour l'auto configuration de votre matériel !

voir <http://knoppix.net/>



➤ Ubuntu :

Distribue un CD live avec chaque version (tous les 6 mois).

voir <http://ubuntu-linux.org/>



DISTRUBUTIONS COMMERCIALE



RED HAT

<http://www.redhat.com/>

La plus populaire. Fiable, sûre, conviviale et facile à installer, prise en charge par tous les fournisseurs de logiciel et de matériel.



SUSE (Novell)

<http://fr.suse.org/>

L'alternative principale. Facile à installer, conviviale et stable. Obtiens le support des fournisseurs de logiciel et de matériel.



MANDRIVA (Mandrake)

<http://mandriva.fr.maladiva.com/>

Conviviale, facile à installer, plus innovante, mais moins stable.

Cible principalement les utilisateurs individuels. Peu pris en charge par les fournisseurs de logiciel et de matériel.



FEDORA (Core)

<http://fedora-fr.org/>

Stable, sûre, conviviale, facile à installer. Sortie fréquente de nouvelles versions complètes.



UBUNTU

<http://ubuntu-fr.org/>

La distribution communautaire qui progresse le plus.

Basé sur Debian, mais avec une version stable tous les 6 mois.

Conviviale pour les utilisateurs. Bonne pour les débutants.



MINT

<http://linuxmint.com/>

Crée à partir d'Ubuntu, s'affirme comme un « système moderne, élégant et confortable ».

À la fois puissant et facile d'utilisation.

Selon le site Alexa, Linux Mint est la distribution Linux la plus Consultée sur Internet, après Ubuntu et devant Debian.



DEBIAN

<http://debian.org/>

Très stable et sûre, mais plus difficile à configurer et à installer.

Conviviale pour les développeurs, mais pas encore pour les utilisateurs.

Version stable pas assez fréquente (tous les 2 ou 3 ans). La meilleure pour les serveurs, mais pas pour les débutants.

DISTRIBUTIONS COMMUNAUTAIRES



KALI

<http://kali.org/>

Base sur debian, l'objectif de Kali Linux est de fournir une distribution regroupant **l'ensemble des outils nécessaires aux tests de sécurité d'un système d'information**, notamment le **test d'intrusion**. (Alternative : Parrot Linux)



ARCH

<http://archlinux.org/>

Arch Linux a été conçue pour être le système d'exploitation **parfait** pour les **utilisateurs avancés**. Sa philosophie simple et sans outils de configuration demande.



CentOS

<http://centos.org/>

Une distribution Linux orientée **entreprise**, basée sur les sources de Red Hat Enterprise Linux, disponibles de manière libre et **gratuite**.



ElementaryOS

<http://elementary.io/>

Elementary OS est un système d'exploitation libre dérivé d'Ubuntu proposant une **interface graphique** proche de celle de **macOS**.

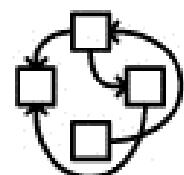
■ Autres systèmes Unix libres :

➤ GNU / Hurd :

Outils GNU avec le Hurd, le micro-noyau de GNU.

De plus en plus mûr, mais pas encore assez pour être utilisé par tous. Jusqu'à présent (2005), surtout utilisé par ses développeurs eux--mêmes.

voir <http://www.gnu.org/software/hurd/hurd.html>



➤ Famille BSD :

FreeBSD : Système BSD puissant, multi-plateforme, sûr et populaire.

voir <http://www.freebsd.org/>

OpenBSD : Système BSD puissant, multi-plateforme, sûr et populaire.

Construit pour une fiabilité et une sécurité extrême. Populaire pour serveurs sur Internet.

voir <http://openbsd.org/>

NetBSD : Distribution BSD dont le but est d'être extrêmement portable.

Disponible sur ARM et autres.

voir <http://netbsd.org/>



➤ Famille SYSTEME V :

OpenSolaris : Le noyau open source de Sun Solaris.
À débuter en juin (2005). Pas encore de version stable.

voir <http://opensolaris.org/>



eCos : Système embarqué à temps réel très léger fourni par Red Hat / Cygnus.
API compatible avec POSIX.

voir <http://ecos.sourceforge.net/>



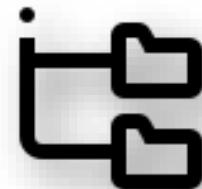
N. B. Ne confondez pas la version de distribution avec celle du noyau !

Système de fichier (Linux)

« Sous Linux, tout est fichier »

■ SYSTÈME DE FICHIER

Les **systèmes de fichiers** (il en existe de multiples) gèrent l'**organisation physique** des informations mémorisées sur les périphériques de stockage de l'ordinateur (HDD, clé USB, DVD, ...). Chacun offre des avantages pour certains types d'utilisation... mais des désavantages pour d'autres utilisations. Connaître leurs principales caractéristiques est nécessaire pour l'utilisation optimale d'un équipement. C'est comme un classeur.



■ ARBORESCENCE

L'**arborescence** c'est l'**organisation logique** des fichiers sur un ou plusieurs systèmes de fichiers, il s'agit d'une structure de données hiérarchique de type arbre.

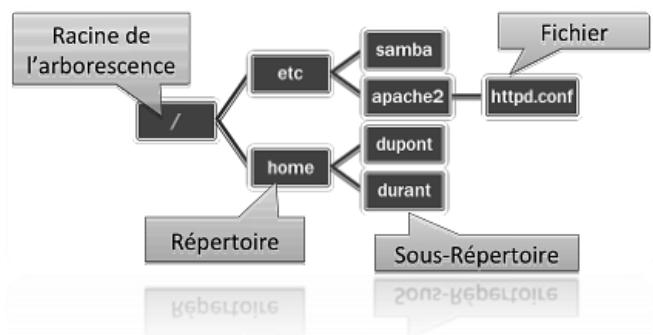
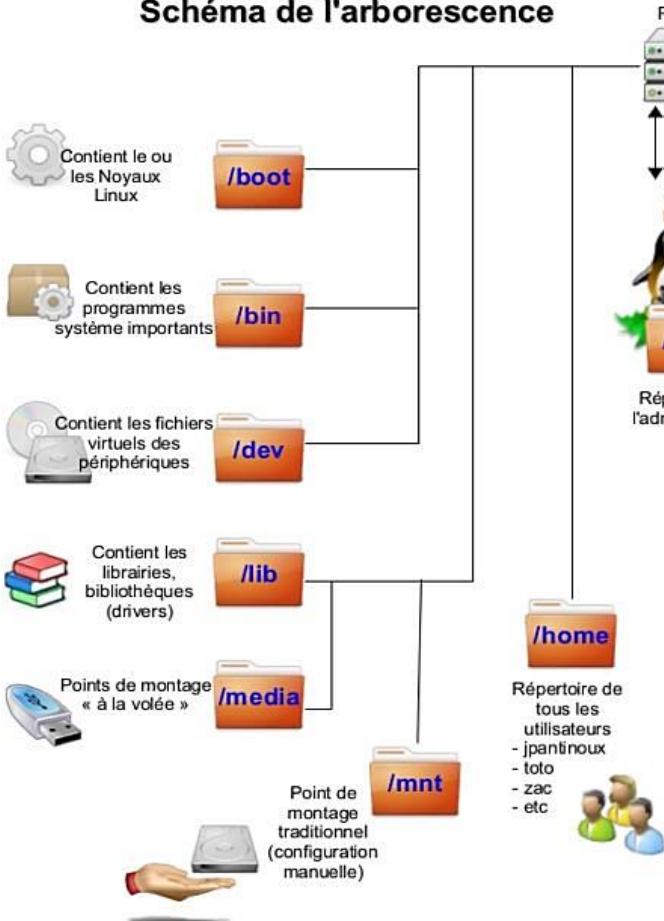
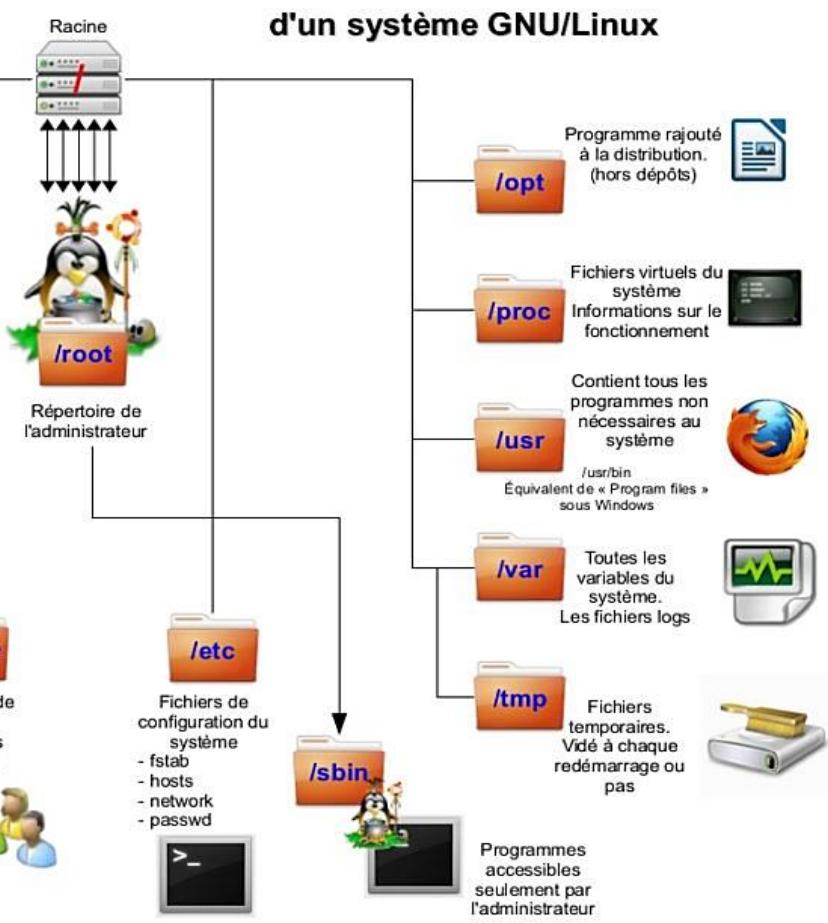


Schéma de l'arborescence



d'un système GNU/Linux



■ LES SYMBOLES ASSOCIENT À L'ARBORESCENCE

.. / ou .. Répertoire parent.

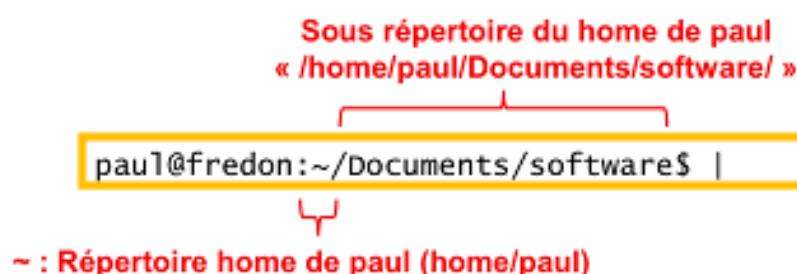
. / ou . Répertoire courant.

~ (tilde) Répertoire « home » (personnel) de l'utilisateur courant.

- ✓ Le ". ." est une commande qui signifie "ICI" (ex : #cp /etc/fstab .). Cette commande signifie littéralement : copier le fichier fstab dans le répertoire "etc" ici (répertoire actuel).

■ LE « PROMPT » BASIC DE GNU/LINUX

Il désigne « l'invite de commande » des caractères indiquant que l'ordinateur est prêt à recevoir une saisie au clavier. Ces caractères donnent des informations à l'utilisateur.



■ SPÉCIFICATION D'UN CHEMIN DANS LE SYSTÈME DE FICHIERS

Il existe 2 méthodes pour spécifier un chemin dans le système de fichiers :

Chemin relatif : dépend du répertoire courant.

Chemin absolu : débute à la racine du système (« / »).

■ Exemples

relatif

root@fredon:/home/paul# cd Documents

absolu

root@fredon:/home/paul# cd /home/paul/Documents

root@fredon:/home/jean# cd Documents
root@fredon:/home/jean/Documents/# |

✗ root@fredon:/home/jean# cd /home/paul/Documents
root@fredon:/home/paul/Documents/# |

root@fredon:/etc/apache# cd ../
root@fredon:/etc/# |

= root@fredon:/etc/apache# cd /etc
root@fredon:/etc/# |

paul@fredon:~ # cd Docs
paul@fredon:~/Docs# |

✗ jean@fredon:/etc# cd /home/paul/Docs
root@fredon :/home/paul/Docs # |

✓ Attention aux chemins relatifs à l'intérieur d'un script

- Le script peut être exécuté depuis n'importe où.
- Le répertoire courant est donc différent à chaque fois.

■ HDD (Hard Disk Drive) ou LES DISQUES DURS

Il est constitué d'un ou plusieurs plateaux dont :

Chaque plateau est divisé en pistes (tracks).

Chaque piste est divisée en secteurs (sectors) et le cylindre est formé par les pistes de même rayon sur chaque plateau.

Le formatage est effectué à 2 niveaux :

En usine : le formatage bas niveau des pistes et secteurs.

Par l'utilisateur : effacement ou réécriture des données.

(Ex : lors de l'installation d'un système d'exploitation).

La capacité d'un disque dépend :

De la taille des secteurs.

Du nombre de cylindres et donc du nombre de plateaux.

L'organisation du disque :



Le **MBR** ou **Master Boot Record** est situé dans le 1^{er} secteur du disque, il est constitué de 2 parties :

La table des partitions.

Le programme d'amorçage qui charge le noyau du système.

Les partitions (types) :

Les partitions principales qui sont au maximum de 4 et acceptent tout type de système de fichiers.

Les partitions étendues qui sont destinées à contenir des partitions logiques et non un système de fichiers. Mais nécessitent au moins une partition principale.

Les partitions logiques qui sont contenues dans une partition étendue et acceptent tout type de système de fichiers.

Voici un exemple d'organisation de partition permettant d'installer plusieurs systèmes d'exploitation



Prise en charge des disques sous GNU/Linux :

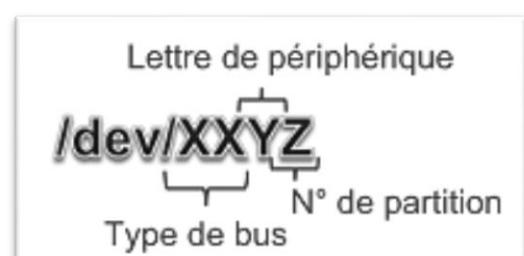
Le pointeur spécial **/dev** (device) permet l'accès aux disques

➤ Type de bus :

hd périphériques IDE.

sc Périphériques SCSI.

sd Périphériques SATA.



➤ Exemples :

/dev/hda1 Partition 1 sur le 1^{er} disque IDE.

/dev/sdb2 Partition 2 sur le 2^e disque SATA.

■ LES FORMATS DES SYSTÈMES DE FICHIERS

À chaque système est associé un format qui définit la structure des données sur le support

Sous GNU/Linux :

Ext2, Ext3, Ext4 (qui est la plus courante), jfs, xfs ...

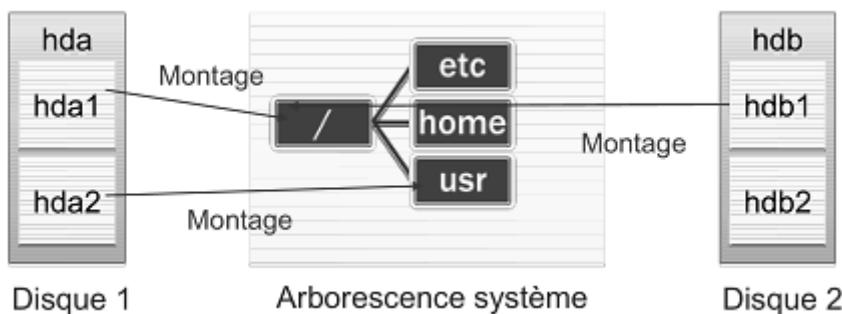
Sous Windows :

FAT, FAT32 et NTFS.

- ✓ Toujours préférer un système de fichier « journalisé ». Chaque séquence de lecture/écriture est d'abord inscrite dans un journal avant d'être effectuée.
- ✓ Si le système se bloque pendant une séquence, elle sera achevée après le redémarrage.
- ✓ On évite les erreurs dans le système de fichiers.
- ✓ Il existe un format de type « swap » qui est utilisé comme « mémoire virtuelle » :
 - *Dans le cas où la mémoire vive est saturée par le système pour améliorer les performances.*
 - *La taille du « swap » est fixée au double de la mémoire vive (Si 512Mo de mémoire vive -> 1024Mo de swap).*
- ✓ GNU/Linux peut lire la plupart des formats. Notamment ceux de Windows.
- ✓ Avant d'être utilisé, un disque doit être partitionné, à l'aide de la commande "#fdisk" si Linux est déjà installé. Sinon par le programme d'installation qui dépend de la distribution. Il faut ensuite créer un système de fichier avec l'utilitaire générique "#mkfs" (ex : mkfs.ext3, mkfs.ext4, mkfs.xfs, ...).

■ POINTS DE MONTAGE

L'arborescence est construite à partir de « point de montage », ce qui signifie qu'un « point de montage » est une association entre une partition physique et l'arborescence du système.



Avantages :

Mettre à l'abri certaines données stratégiques comme **/home**.

La défaillance du disque hdd n'entraîne pas une réinstallation totale.

Tant qu'ils ne sont pas effectués, le système de fichiers est inaccessible.

Seulement ceux qui figurent dans le fichier « **/etc/fstab** » sont réalisés automatiquement au démarrage du système.

Création d'un point de montage manuellement :

#**mount** pour **monter** un périphérique et #**umount** pour **supprimer** le point de montage.

```
root@fredon:~# mount /dev/hdd /mnt/cdrom  
root@fredon:~# umount /dev/hdd
```

-
- ✓ Une partition est toujours associée à un système de fichiers, il faut parfois préciser le type de ce système (Ext2, Ext3, Ext4, xfs, swap, jfs, iso9660, vfat, ...).
-

■ MISE EN PLACE D'UN NOUVEAU DISQUE

3 étapes sont nécessaires pour la mise en place d'un nouveau disque :

ÉTAPE 1 : Crédit d'une partition par « fdisk »

Partition principale de **100Mo** avec « **fdisk** » sur **/dev/sda**.

```
root@fredon:~# fdisk /dev/sda  
Commande (m pour l'aide): n  
Action de commande  
 e étendue  
 p partition primaire (1-4)  
p  
Numéro de partition (1-4): 1  
Premier cylindre (1-26, par défaut 1):  
Utilisation de la valeur par défaut 1  
Dernier cylindre ou +taille or +tailleM ou +tailleK (1-26, par défaut 26): +100M  
Commande (m pour l'aide): w  
La table de partitions a été altérée!  
Appel de ioctl() pour relire la table de partitions.  
Synchronisation des disques.  
Commande (m pour l'aide): q
```

Vérification de la partition sous « fdisk ».

```
Commande (m pour l'aide): p
Périphérique Amorce     Début      Fin      Blocs   Id Système
/dev/sdal               1          13      104391  83 Linux
```

ÉTAPE 2 : Initialisation du système de fichier par « mkfs »

```
root@fredon:~# mkfs.ext3 /dev/sda1
mke2fs 1.40.8 (13-Mar-2008)
Étiquette de système de fichiers=
Type de système d'exploitation : Linux
Taille de bloc=1024 (log=0)
Taille de fragment=1024 (log=0)
26104 i-noeuds, 104388 blocs
5219 blocs (5.00%) réservés pour le super utilisateur
Premier bloc de données=1
Nombre maximum de blocs du système de fichiers=67371008
13 groupes de blocs
8192 blocs par groupe, 8192 fragments par groupe
2008 i-noeuds par groupe
Superblocs de secours stockés sur les blocs :
    8193, 24577, 40961, 57345, 73729
Écriture des tables d'i-noeuds : complété
Création du journal (4096 blocs) : complété
Écriture des superblocs et de l'information de comptabilité du système de
fichiers : complété
```

ÉTAPE 3 : Création du point de montage par « mount » de type Ext4

Le répertoire `/mnt/toto` doit déjà exister.

```
root@fredon:~# mount -t ext3 /dev/sda1 /mnt/toto/
```

Vérification du point de montage avec la commande « df ».

```
root@Fredon:~# df
Sys. de fich.      1K-blocs   Occupé   Disponible Capacité Monté sur
/dev/sdc1           7913216   2716248   4798156   37% /
varrun              127856     108     127748    1% /var/run
varlock              127856      0     127856    0% /var/lock
udev                127856      64     127792    1% /dev
devshm              127856      12     127844    1% /dev/shm
lrm                 127856    38176     89680   30% /lib/modules/2.6.24-16-generic/vol...
/dev/sdd1            8045180   6987320   1057860   87% /media/GEN-USB
/dev/sdal             101086     5664     90203    6% /mnt/toto
```

-
- ✓ Si le doute s'installe quant à l'intégrité du système de fichiers. Une vérification s'impose et il faut parfois démonter le système en préalable la vérification.
 - ✓ Supprimer d'abord le montage avec la commande « **umount** ». Puis utiliser la commande générique « **fsck** » (ex : `#fsck /dev/sda1`). Effectuer le montage après la vérification pour accéder au système de fichiers.
-

■ LE RÉPERTOIRE SPÉCIAL « /proc »

Voici quelques fichiers intéressants à consulter :

Cpuinfo : informations sur le(s) processeur(s).

Meminfo : utilisation de la mémoire.

Ioports : adresses physiques des différents périphériques matériels.

On peut le visualiser avec la commande « **cat** » :

```
root@fredon:~# cat /proc/cpuinfo
processor : 0
vendor_id : GenuineIntel
cpu family    : 6
model        : 23
model name   : Intel(R) Core(TM)2 Duo CPU      T9300 @ 2.50GHz
stepping     : 8
cpu MHz      : 2500.585
cache size   : 6144 KB
fdt_bug      : no
hlt_bug      : no
f00f_bug     : no
coma_bug     : no
...
...
```

■ AUTRES PÉRIPHÉRIQUES SPÉCIAUX

Ils ressemblent à des fichiers, mais

/dev/null

Le destructeur de données ! Fais disparaître toutes données écrites dans ce fichier. Utile pour se débarrasser d'une sortie indésirable (telles que des « **logs** ») :

```
mplayer black_adder_4th.avi &> /dev/null
```

/dev/zero

Les lectures à partir de ce fichier renverront toujours des caractères \0 utile pour créer un fichier rempli de zéros :

```
dd if=/dev/zero of=disk.img bs=1k count=2048
```

#EDITEUR_DE_TEXTE #TERMINAUX &
INTERPRÉTEURS #COMMANDES #COLOR-INDEX
#EXTENSIONS #LIENS

« Premier pas avec GNU/Linux »

Éditeur de texte

« L'outil fondamental »

➤ Éditeur à partir du terminal :



Voir <http://ex-vi.sourceforge.net/>

#vim ou #vi // c'est un éditeur de texte en mode texte disponible sur tous les systèmes Unix, créé avant même l'apparition de la souris, open source très puissant développé par Bram Moolenaar et Al (idéal pour un administrateur système, programmeur et parfaits pour les utilisateurs expérimentés), la commande "#vimbutor" est très efficace pour apprendre l'éditeur vim. A savoir que Vim (Vi IMproved) est la version améliorée de Vi. Il contient **30 commandes** facile à apprendre et suffisante pour **99%** des besoins quotidiens !

- 💻 Difficile à apprendre pour les débutants habitués aux éditeurs graphiques.
- 💻 Très productif pour les utilisateurs avancés.
- 💻 Souvent incontournable pour modifier des fichiers en administration de système ou dans les systèmes embarqués, quand vous ne disposez que d'une console texte.

```

    111
ILE88D]: :JD8888BD];
LGIT888D:f8GjjjlB888E;
IE :8888Et. ,G8888.
;i EBBB, ,8888,
DBBB, :8888:
DBBB, :8888:
DBBB, :8888:
DBBW, :8888:
MSBW, :8888:
MSBW: :8888:
DGGO: :8888:
:8888:
:W888:
:8888:
EB8881
tw8880

```

Voir <https://www.nano-editor.org/>

#nano // un autre éditeur de texte léger en mode texte, clone amélioré de Pico (éditeur non libre dans Pine) permet de créer ou modifier un fichier/text, il suffit de taper directement "#nano text" sur la console pour créer un fichier(texte) nommé "text".

- 💻 Convivial et plus facile à prendre en main grâce à un résumé des commandes affichées à l'écran.
 - 💻 Disponible sous forme de paquetages binaires pour plusieurs plateformes.
 - 💻 Une alternative à vi dans les systèmes embarqués. Cependant, pas encore disponible à travers **busybox**.

➤ Éditeur de texte (graphique) :



Voir <https://www.sublimetext.com/>

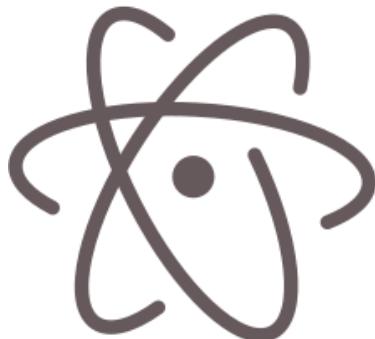
Sublime Text est un éditeur de texte générique codé en C++ et Python, créé par [Jon Skinner](#), disponible sur Windows, Mac et Linux. Le logiciel a été conçu tout d'abord comme une extension pour Vim, riche en fonctionnalités. Son créateur quitta son travail chez Google pour poursuivre un de ses rêves : créer un meilleur éditeur de texte. Il se donna trois principes pour ce logiciel :

- Discret, interface minimum : on doit pouvoir se focaliser sur le texte et non une myriade de barres d'outils ;
 - Ne pas cacher le texte par des fenêtres ;
 - Utiliser toute la place possible : plein écran, multiécran, édition de fichiers côté à côté devraient être possibles.

Sublime Text est le plus approprié pour les codeurs lourds. Ces développeurs apprécieront la fonctionnalité de raccourci et le haut niveau de personnalisation. Les capacités multiplateformes permettent également de lancer le même éditeur de texte sur toutes les machines.

- Sublime Text vous permet de tester l'éditeur de texte avant de vous engager sur l'investissement. Bien qu'il soit dommage que vous deviez payer pour cet éditeur de texte, vous pouvez au moins l'essayer – et le prix n'est pas si cher que ça.
 - L'éditeur de texte fonctionne sur plusieurs plateformes telles que Mac, Windows et Linux. Il est également multiplateforme, donc une seule licence fonctionnera sur tous vos appareils – pas besoin d'en acheter d'autres.
 - Sublime Text permet l'édition fractionnée pour gérer et éditer les fichiers les uns à côté des autres. Vous pouvez également ouvrir plusieurs fenêtres et les placer sur différents moniteurs.
 - L'API Python offre la possibilité de mettre à niveau Sublime Text avec les plug-ins que vous ou d'autres personnes créez.
 - Sublime Text a des raccourcis extrêmement simples et puissants. De la recherche et la modification de plusieurs lignes à la localisation de certaines fonctions dans le menu, Sublime Text devrait faire le bonheur des amateurs de raccourcis.

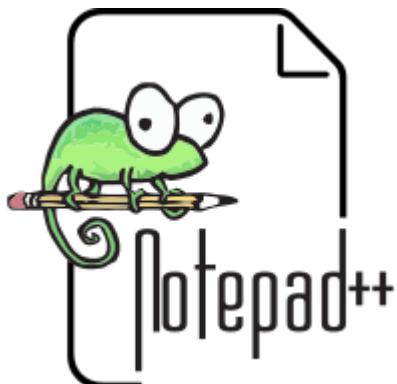
- Vous pouvez également personnaliser à peu près n'importe quoi dans Sublime Text. C'est particulièrement vrai lorsqu'il s'agit de raccourcis et de menus. Nous vous recommandons de modifier les paramètres pour ouvrir des fichiers dans la même fenêtre (nouvel onglet).
- A quelques grands thèmes communautaires disponibles. Découvrez le thème Dracula Sublime.



Voir <https://atom.io/>

Atom est un éditeur de texte libre pour macOS, GNU/Linux et Windows développé par **GitHub**. Il supporte des *plug-ins* écrits en Node.js et implémente Git Control. La plupart des extensions sont sous licence libre et sont maintenues par la communauté. Atom est basé sur Chromium et Electron et est écrit en CoffeeScript. Il est aussi utilisé en tant qu'environnement de développement (EDI).

- Atom est un éditeur de texte open source avec une large communauté de développeurs. Cela signifie que vous obtenez des mises à jour cohérentes et de nouveaux thèmes et paquets. Découvrez le thème Dracula pour Atom (<https://draculatheme.com/atom/>).
- C'est une solution multiplateforme qui fonctionne sur les principaux systèmes d'exploitation.
- L'éditeur est utile pour coder seul, mais ses vraies forces entrent en jeu lorsque vous avez besoin de collaborer avec d'autres personnes. Toute l'édition et la création peuvent se faire en temps réel.
- Atom fournit un paquet GitHub pour l'intégration et l'exécution de choses telles que les requêtes pull et la résolution des conflits de fusion.
- Vous pouvez rechercher de nouveaux paquets et thèmes directement depuis l'éditeur de texte.
- Il est assez facile d'écrire votre code avec l'autocomplétion intelligente, un navigateur de système de fichiers, et une fonction de recherche et de remplacement.
- Atom offre plusieurs volets pour comparer et éditer le code côte à côte.



Voir <https://notepad-plus-plus.org/>

Probablement l'éditeur de texte avancé le plus populaire sur le marché, Notepad++ se présente sous la forme d'un ensemble compact. Il a été développé par **Don Ho**, un informaticien situé à Paris diplômé de l'université Paris VII, fondé sur la composante Scintilla, a pour but de fournir un éditeur léger (aussi bien au niveau de la taille du code compilé que des ressources occupées durant l'exécution) et efficace. Il est également une alternative au bloc-notes de Windows (d'où le nom), **sans frais** et avec des composants d'édition puissants. Il est distribué gratuitement sous Licence Publique Générale, ce qui signifie que tous

les développeurs et créateurs de contenu peuvent profiter de l'éditeur de texte immédiatement après un téléchargement rapide. Notepad++ fonctionne sous Microsoft Windows, et il s'efforce d'utiliser moins de puissance de calcul qu'un éditeur de texte moyen.

Notepad++ se distingue notamment par le fait qu'il a déjà été traduit dans plus de 80 langues, ce qui lui permet d'être accessible partout dans le monde. De plus, vous pouvez traduire Notepad++ dans votre langue maternelle si vous ne trouvez pas votre langue dans la liste des traductions.

- Notepad+++ est totalement gratuit pour tous.
- L'éditeur de texte est déjà traduit dans des dizaines de langues et fournit la documentation nécessaire pour traduire dans plusieurs langues.
- Vous obtenez un éditeur multi-vues avec coloration syntaxique et pliage.
- Les outils de personnalisation sont faciles à comprendre et suffisamment puissants pour les développeurs les plus avancés.
- Les paramètres d'autocomplétions vous permettent de terminer les fonctions, les paramètres et les mots sans avoir à les taper à plusieurs reprises.
- Il offre une interface multi-documents pour passer d'un onglet à l'autre et gérer plusieurs projets à la fois.
- Notepad+++ fournit une liste de plug-ins pour améliorer la fonctionnalité de l'éditeur de texte ou l'intégrer à d'autres programmes.
- Vous pouvez ouvrir une liste de fonctions, qui affiche un aperçu de toutes les fonctions trouvées dans le fichier en cours. Cela inclut également un moteur de recherche pour localiser rapidement les fonctions dans les documents volumineux.

VISUALISATION (non destructeur) avec *cat*, *more* et *less*



less

more

cat

`#cat` // pour visualiser les fichiers courts, `-n` permet de numérotter les lignes, `-s` permet de réduire toutes les lignes vides en une seule.

`#more` // pour visualiser les fichiers longs.

`#less` // pour visualiser l'intégralité des fichiers.

`#strings` // permet d'extraire les chaînes de caractères pour observer ceux qui a été écrit comme **texte**, elle est pratiquement utilisable pour des fichiers binaires (ex : `#strings /bin/ls`).

`#stat` // permet d'avoir quelques informations sur un fichier (taille, nombre de blocs, accès, périphérique rattaché, lien, modification ...).

`#wc` // consiste à savoir : [1.nombre de ligne] [2.nombre de mot] [3.taille en octet] d'un fichier.

-
- ✓ On peut afficher le contenu des plusieurs fichiers en même temps par "#cat" (ex : #cat file0 file1 file2).
-

>> FONCTIONNEMENT de #more

- [ENTRÉE] pour avancer d'une ligne.
- [ESPACE] pour avancer d'une page.
- [b] pour remonter page par page.
- [q] pour quitter.

>> FONCTIONNEMENT de #less

- [HAUT/BAS] touches directionnelles.
- [GAUCHE/DROITE] touches directionnelles.
- [/] pour faire une recherche (*par occurrence*).
- [n] ou [f] next.
- [MAJ+n] ou [b] preview.
- [q] pour quitter.

-
- ✓ Pour créer rapidement & efficacement un fichier avec un/des contenu(s), la commande suivante fera l'affaire :

```
#echo le texte/contenu > file_name
```

- ✓ Il est préférable de visualiser les fichiers sensibles par "#cat" afin d'éviter toute sorte de dommage qu'on risquait de faire avec nano ou vim.
-

Les terminaux et interpréteurs de commande (shell)

« Mieux communiquer avec votre ordinateur »

Le terminal, c'est quoi ?



Le **terminal** est un shell. On en retrouve un (ou plusieurs) dans tous les systèmes Linux ; le terminal est un programme, un émulateur de console qui nous permet de lancer des commandes en texte depuis l'interface graphique, c'est une interface de ligne de commande (CLI) par opposition à l'interface graphique utilisateur (GUI : graphical user interface) qui nous permet de passer des commandes plutôt qu'en cliquant sur des icônes de programmes, par exemple. La console, elle, est l'écran noir avec du texte qui défile rapidement lorsque l'on démarre l'ordinateur. Dites-vous que tout ce qui se passe dans votre système GNU/Linux est écrit en texte (script) quelque part dans le système de fichier. Dans Linux, tout est fichier, certains sont des fichiers exécutables (des scripts) et d'autres non. Les scripts prédéfinis permettent, entre autres, d'automatiser des opérations, des tâches, des processus, etc. Vous pouvez contrôler toutes les composantes de votre système Linux par le terminal.

*Il suffit d'appuyer simultanément la touche **CTRL + ALT + T** pour ouvrir un terminal.*

Pourquoi le terminal ?

Il est parfois plus simple de **taper une commande** que d'effectuer des manipulations demandant beaucoup de clics de souris dans une interface graphique. C'est aussi un moyen plus simple pour expliquer comment faire quelque chose à quelqu'un (sur un forum par exemple), puisqu'il suffit d'indiquer la commande et non la suite de clics à effectuer sur l'interface graphique.

Cependant, même si le terminal peut être beaucoup plus efficace qu'une interface graphique sous les doigts d'un utilisateur avancé, il est moins abordable que les interfaces graphiques.

Il est probable qu'aucune des deux méthodes (commandes ou interface graphique) ne remplacera complètement l'autre car elles se complètent plus qu'elles ne rivalisent.

Les shell

Le rôle de l'interpréteur de commandes (ou shell) est de lire l'entrée (clavier, fichier, etc), de l'interpréter et si besoin est de lancer une commande. Ou alors d'envoyer une erreur... car les entrées doivent respecter une certaine syntaxe qui peut varier sensiblement d'un interpréteur à l'autre.



Autrement dit la syntaxe du shell forme un langage de programmation de haut niveau (un langage de script directement interprété).

Il y en a plusieurs :

Alors, quelle différence entre ces différents interpréteurs, pourquoi utiliser l'un plutôt que l'autre, qu'est-ce qui justifie tel ou tel choix pour les distributions ?

- Origines : sh et POSIX (obsolète)

Sh ou **bourne shell** est le shell historique depuis le système unix 7 par Steve Bourne. Ses fonctions de base ont été étendues avec les interpréteurs ash puis bash.

ash fut le premier shell unix conforme aux spécifications POSIX. La norme POSIX a permis de mettre fin aux aspects les plus néfastes de la guerre des unixes, c'est ce qui a permis de définir une inter-compatibilité des systèmes unix (notamment entre les différents unix commerciaux, bsd et gnu/Linux).

■ CSH Le C shell (obsolète)

Shell avec une syntaxe à la C, qui a connu son heure de gloire.

- TCSH Le TC shell (toujours très populaire)

Une implémentation compatible avec le C shell, avec des fonctionnalités avancées
(Complète les noms de commandes, rappel de commandes antérieures et bien d'autres...)

- Bash : l'interpréteur GNU (le plus populaire, pour débutant)

Bourne Against SHell a donc été créé pour le compte de la FSF afin d'étendre les fonctionnalités de ash (qui lui-même a été créé pour étendre les fonctionnalités de sh).

■ Dash : l'interpréteur de Debian (pour utilisateur expérimenté)

Debian Almquist SHell implémente quasiment toutes les fonctionnalités de BASH, mais avec de meilleures performances.

■ Zsh : encore plus de fonctionnalités (le meilleur)

Le meilleur de tous vous diront les utilisateurs... Plus léger que bash et plus étendu, il reprend les éléments de plusieurs shells jugés les plus pratiques.

On citera une complétion bien plus étendue et la récursivité des commandes. Et un prompt bien geek ;)



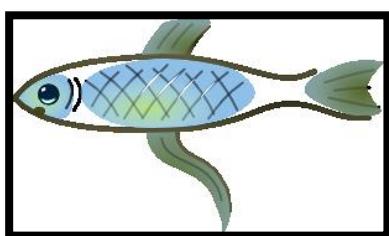
■ ipython

À l'origine, python a été créé pour fournir un shell au système d'exploitation néerlandais amoeba. Il permet assez facilement d'injecter des commandes système en bash, avec la bibliothèque os.

ipython est un interpréteur python qui inclue les commandes unix standard sans avoir à faire des dizaines d'imports.

■ Le Friendly Interactive SHell

Voir <http://roo.no-ip.org/fish/>



- Caractéristiques standards : historiques, complètent les noms de commandes et de fichiers...
- Apporte de nouvelles fonctionnalités : complète les options de commandes, description des commandes, syntaxe mise en valeur.
- Facilite l'ouverture de tous les fichiers : fournit une commande open.
- Syntaxe plus facile et consistante (pas conforme à POSIX). Rends plus facile la création de script shell.

Les débutants en ligne de commande peuvent apprendre plus facilement ! Même les utilisateurs expérimentés devraient trouver ce shell très pratique.

Commande de base (bash)

« Profiter la puissance de la commande GNU/Linux Bash »



MAKE DIRECTORY > `#mkdir $`_ permet de créer un nouveau répertoire, `-p` pour créer un sous-répertoire (ex : `#mkdir -p doc0/ doc1/ doc2`), `#mkdir -p doc/{ doc0, doc1}` est la commande pour créer deux répertoires indépendants l'une des autres dans un autre répertoire parent).

COPY > `#cp $`_ permet de copier un ou plusieurs fichier(s)/répertoire(s).

MOVE > `#mv $`_ permet de déplacer un ou plusieurs fichier(s)/répertoire(s), le renommage est possible si même répertoire (ex : `#mv ./ file0 ./ file1`).

NEW FILE "toucher la date de modification" > `#touch $`_ pour créer un fichier vide.

REMOVE > `#rm $`_ permet de supprimer un ou plusieurs fichier(s)/répertoire(s), `-r` récursive `-f` forcé (ex : `#rmdir` pour effacer un dossier vide).

-
- ✓ `#rm key_word*` pour supprimer des fichiers ayant des "mots-clés" en commun (*dans leurs noms*).
-

PRINT WORKING DIRECTORY "où suis-je ?" > `#pwd` **\$** permet d'identifier votre emplacement dans l'arborescence du système de fichier.

CHANGE_DIRECTORY "où vais-je ?" > `#cd` **\$** permet de changer de répertoire.

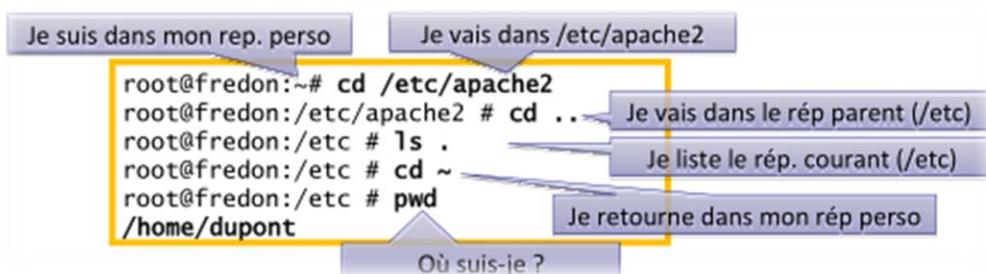
- ✓ Une façon très simple de retourner vers le répertoire parent initial sans réécrire le chemin est de taper la commande `#cd ~`

BASH_of_PRINT "affichage de bash" > `#echo` **\$** permet de retourner un caractère au terminal.

LISTER LES ELEMENTS > `#ls` **\$** permet de visualiser sous forme de liste les éléments d'un répertoire, **-l** sous forme de liste, **-h** human readable, **-a** affiche les fichiers cachés, **-t** trier par date.

UTILISATION DE DISQUE (1.0) > `#du` **[doc/file]** **\$** permet de visualiser l'utilisation de disque, **-h** pour transformer les tailles en "human readable", **-s** pour n'afficher que la taille total (ex : `#du -hs /boot`).

UTILISATION DE DISQUE (2.0) > `#df` **[doc/file]** **\$** permet de voir la taille/utilisation (en%) de tous les systèmes de fichier rattaché à l'arborescence du **[doc/file]** (chemin absolu). On peut également l'accompagner avec l'option **-h** (pour un diviseur 1024) ou **-H** (pour diviseur 1000).



Color Index (bash)

« Plus de précision »

Il peut nous arriver de poser ce genre de question : « mais que signifie toutes ces couleurs-là ? ». Ne t'inquiète pas, vous avez votre réponse ! voici les différentes **couleurs** que peuvent représentés un terminal.

BLEU Dossier/Répertoire.

CYAN Lien symbolique (équivalent de raccourci dans Windows).

BLANC Fichier admin (y compris les fichiers personnels).

ROUGE Archives (ex : splash, xpm, gz, tar...).

VERT Exécutable (ex : fichier binaire, script shell, code ...).

ROUGE exécutable qui requiert un droit administrateur (root).

VIOLET Images (ex : iso, bin, img...).



Extension (les plus courants)

« Plus de précision »

En général, un système tel que **GNU/Linux** n'a pas besoin **d'extension**, mais il est souvent nécessaire d'en connaître quelques-unes pour ne pas se tromper.



- **.conf** les fichiers texte de configuration.
- **.sh** les fichiers texte (codé par un langage shell) qui deviendront des exécutables.

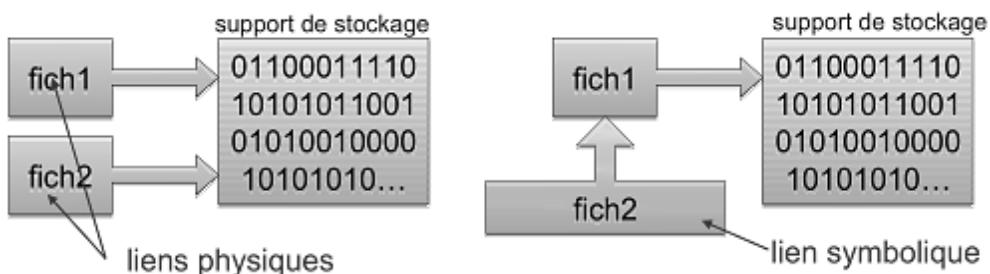
Si vous voulez consulter toutes les listes d'extensions de fichiers existants :

Voir https://fr.wikipedia.org/wiki/Liste_d%27extensions_de_fichiers

Les liens

➤ Un lien est un type spécial de fichier qui fait référence à un autre fichier, il permet également de :

- Créer des raccourcis vers des fichiers existants.
- Assurer la compatibilité des logiciels entre les distributions GNU/Linux.
- Éviter de stocker plusieurs fois le même fichier dans des répertoires différents.



■ Les liens symboliques

➤ Le lien symbolique est une référence vers un fichier cible, utile pour simplifier et réduire l'utilisation du disque quand deux fichiers ont le même contenu.

- Lorsque le fichier cible est effacé.
- Lorsque le lien est effacé, le fichier cible n'est pas effacé.

■ Exemple :

```
rout@fredon:~/Documents$ ls -l
```

total 8	Nom du lien
lrwxrwxrwx 1 rout rout 29 2008-08-25 14:23 ip -> /proc/sys/net/ipv4/ip_forward	
drwxr-xr-x 3 rout rout 4096 2008-06-02 14:20 software	
drwxr-xr-x 3 rout rout 4096 2008-07-29 15:54 vmware-tools	

Indique que c'est un lien Emplacement du vrai fichier

> La commande « ln » avec l'option **-s** est utilisé pour créer un lien symbolique :

```
rout@fredon:~/Documents$ ln -s /proc/sys/net/ipv4/ip_forward ip
```

Cible (Target)

Nom du lien (link name)

■ Les liens physiques

> Le lien physique est associé à un emplacement sur le support de stockage :

- 2 liens peuvent être associés aux mêmes « inode ».
- Similaire à la notion « pointeurs » du langage C.
- Deux liens physiques sont considérés comme 2 fichiers indépendants (même si leur contenu est au même emplacement sur le support).
- Le lien physique est vu comme un fichier régulier.

> La commande « ln » sans option pour créer un lien physique :

nom fichier

```
rout@fredon:~/Documents$ ln /home/paul/Documents/Rapport2007-2008.doc rap0708
rout@fredon:~/Documents/essais$ ls -il
total 176
470930 -rw-r--r-- 2 rout rout 84091 2008-08-25 14:48 rap0708
470930 -rw-r--r-- 2 rout rout 84091 2008-08-25 14:48 rapport-annee2007_2008.doc
```

L' « inode » est identique.
Il s'agit bien de liens physiques

Nombre de liens vers cet inode.
C'est un indice permettant de supposer qu'il s'agit d'un lien

cible

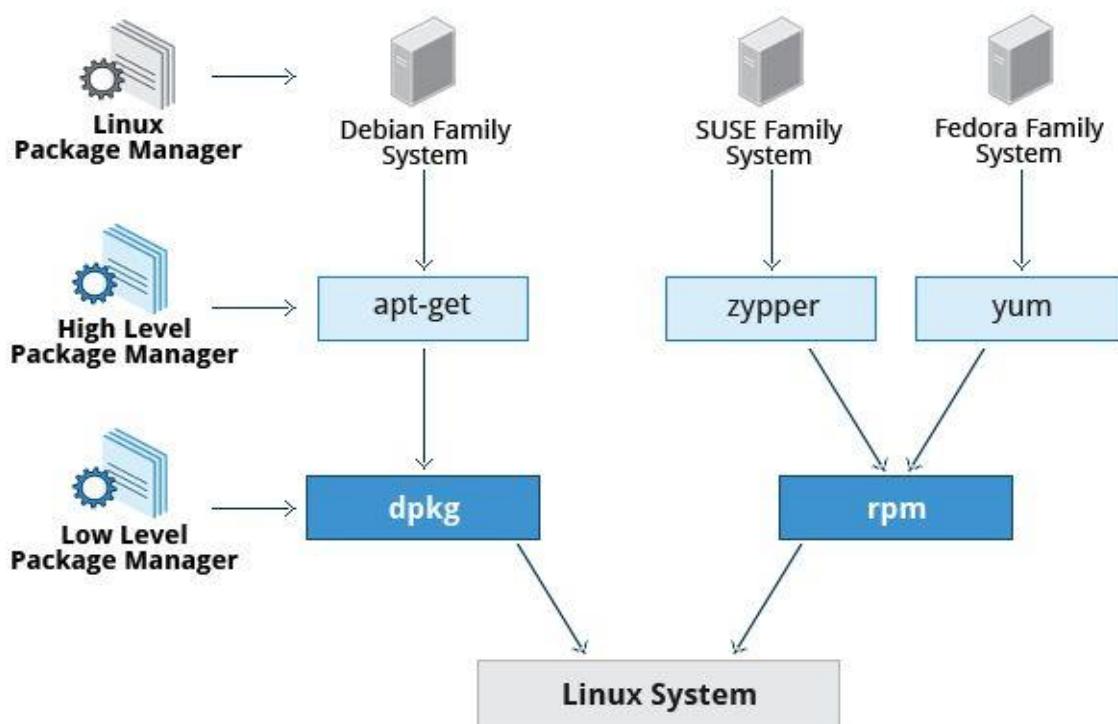
#PAQUETAGES DES DISTRIBUTIONS
#INTERNET & RÉSEAUX #IMPRESSION #GREP
& FIND #REDIRECTION #PIPES #RSYNC
#ARCHIVAGE & COMPRESSION

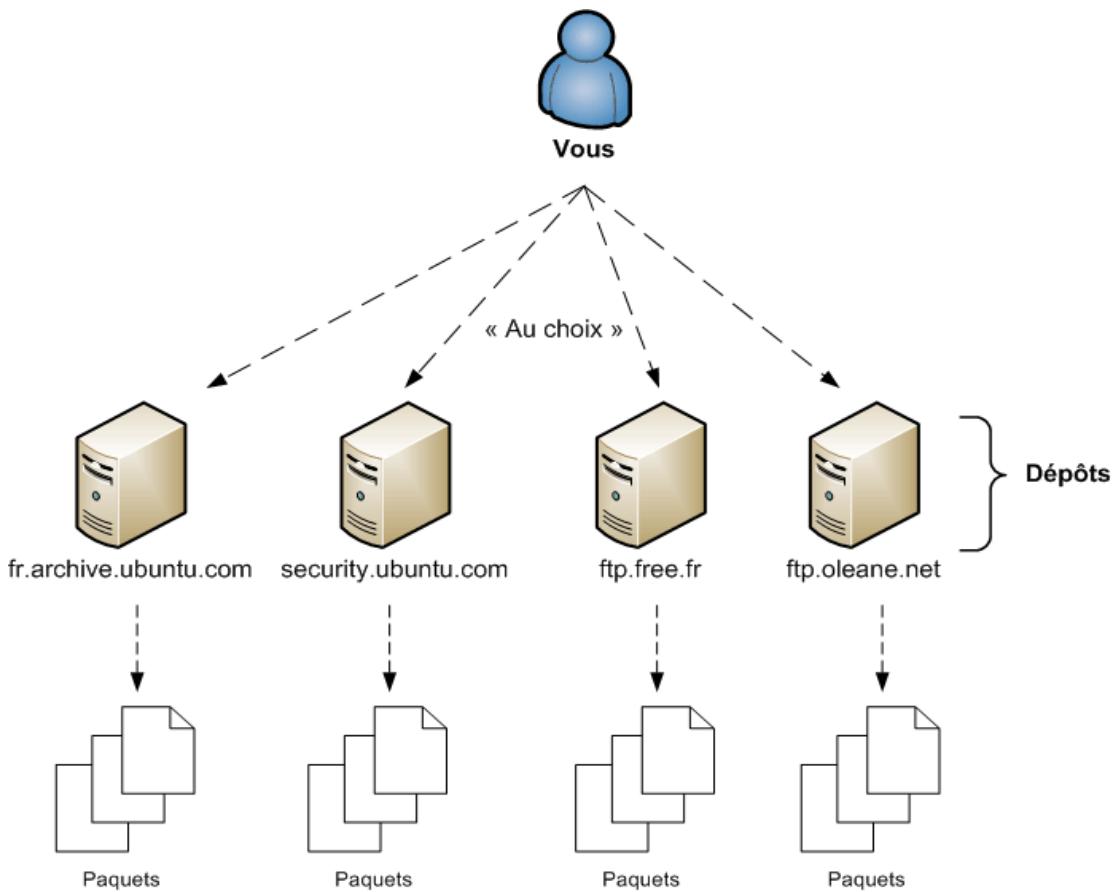
« Les outils de base »

Paquetages des distributions

« APT et DPKG »

- CONCEPT DE GESTION DES PAQUETAGES





■ TROUVER DES PAQUETAGES

- Paquetages Debian : <http://www.debian.org/distrib/packages>
Recherche par nom de paquetage ou de fichier.

- Rpmfind : <http://rpmfind.net/>
Nombreux paquetages RPM pour Red Hat, Mandriva, Suse...

■ IDENTIFIER LES PAQUETAGES

Utile pour récupérer la plupart des informations, récupérer le code, trouver des versions récentes, rapports de problèmes...

- Distributions avec des paquetages RPM :
(Red Hat, Fedora, Mandriva, Suse...)

```
#rpm -qf /bin/ls
```

coreutils-5.2.1-7

- Debian :

```
#dpkg -S /bin/ls
```

fileutils: /bin/ls

▪ INFORMATION SUR LES PACKAGES

Accéder à la description des paquetages, numéros de versions, sources, etc.

- Distributions basées sur RPM :

```
#rpm -qi [package_name]
```



- Debian :

```
#dpkg -s [package_name]
```



▪ /etc/apt/sources.list

Dans ses fonctions, Apt utilise un fichier qui liste les ‘sources’ à partir desquelles les paquets peuvent être obtenus. Ce fichier est **/etc/apt/sources.list**.

Les entrées de ce fichier suivent généralement ce format (exemple) :

```
deb http://site.example.com/debian distribution component1 component2 component3  
deb-src http://site.example.com/debian distribution component1 component2 component3
```

Explication :

✓ deb ou deb-src, indique le type d’archive dont

- **Deb** : si elle contient des paquets binaires qui sont les paquets précompilés que nous utilisons généralement.
- **Deb-src** : paquets sources qui sont les programmes Linux originaux sources plus le fichier de contrôle de Debian (.dsc) et le diff.gz contenant les changements nécessaires pour ‘l’empaquetage’ du programme.

✓ Distribution

- La ‘distribution’ peut être soit le **nom de code**, c’est-à-dire squeeze, wheezy, jessie, sid
- Soit le nom d’une catégorie de version (oldstable, stable, testing, unstable)

Exemple :

1. Si vous avez un système fonctionnant avec Debian 7 « Wheezy » et ne voulez pas le mettre à jour quand Debian Jessie sortira, utilisez ‘wheezy’ à la place de ‘stable’.
2. Si vous voulez sans cesse aider à tester la distribution ‘testing’, utilisez ‘testing’.
3. Si vous voulez suivre Jessie et voulez rester avec elle de testing à la fin de sa vie, utilisez ‘jessie’.

✓ Composants

Main	La section main comprend l'ensemble des paquets qui se conforment aux DFSG – Directives Debian pour le logiciel libre et qui n'ont pas besoin de programmes en dehors de ce périmètre pour fonctionner. Ce sont les seuls paquets considérés comme faisant partie de la distribution Debian.
Contrib	La section contrib comprend l'ensemble des paquets qui se conforment aux DFSG, mais qui ont des dépendances en dehors de main (qui peuvent être empaquetées pour Debian dans non-free).
Non-free	La section non-free contient des logiciels qui ne se conforment pas aux DFSG.

N.B. Si vous désirez disposer aussi des composants contrib et non-free, ajoutez contrib non-free après main.

Autrement, on peut utiliser un outil GNOME pour modifier le fichier **sources.list**. (Menu Système>Administration>Sources logicielles).

```
#gksu --desktop /usr/share/applications/software-properties.desktop /usr/bin/software-properties-gtk
```

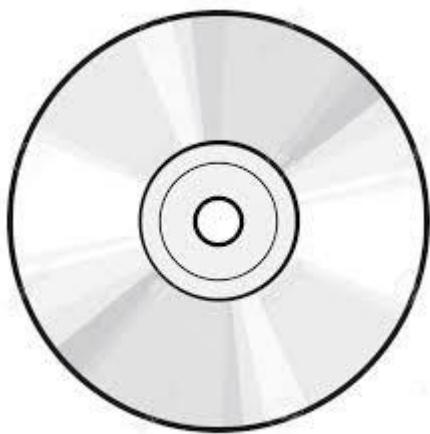
Exemple de fichier source sous Debian wheezy :

```
deb http://http.debian.net/debian wheezy main
deb-src http://http.debian.net/debian wheezy main

deb http://http.debian.net/debian wheezy-updates main
deb-src http://http.debian.net/debian wheezy-updates main

deb http://security.debian.org/ wheezy/updates main
deb-src http://security.debian.org/ wheezy/updates main
```

■ CD-ROM



Si vous préférez utiliser un **CD-ROM** pour installer des paquets ou mettre à jour votre système automatiquement avec APT, vous pouvez le mettre dans votre **/etc/apt/sources.list**.

Pour le faire, vous pouvez utiliser le programme **apt-cdrom** (avec le CD-ROM Debian dans le lecteur) :

```
# apt-cdrom add
```

Vous pouvez utiliser **-d** pour le répertoire où est monté votre CD-ROM ou ajouter un point de montage qui n'est pas un CD-ROM (comme une clé USB).

■ Gestion des paquets avec APT

Pour bien gérer vos paquets, vous disposez des commandes suivantes utilisables en ligne de commande :

#apt-get elle permet d'effectuer l'installation et la désinstallation facile de paquets en provenance d'un dépôt APT. Cette commande nécessite les droits d'administration. Elle s'utilise avec une sous-commande telle que **install**, **update**, **remove** ...qui décrit le type d'action demandé à APT.

A stylized, lowercase logo for "apt-get". The letters are rounded and have a hand-drawn feel. The letter "g" has a small circle with a dot in the center, resembling an eye or a target.

#apt-cache elle permet d'effectuer quelques manipulations basiques sur les paquets, installés ou non, disponibles dans la liste mise en cache des paquets des dépôts APT configurés. Cette commande ne nécessite pas les droits d'administration. Elle s'utilise avec une sous-commande telle que **show**, **policy**, **search**... qui décrit le type d'action demandé à APT.

#apt-file elle permet d'interroger la liste des fichiers contenus dans tous les paquets, qu'ils soient installés ou non. Cette commande doit être installée.

#apt-show-versions elle permet d'analyser le fichier status de dpkg et les listes d'APT pour trouver les versions des paquets installés, les versions des paquets disponibles et la distribution. Cette commande doit être installée.

#apt-rdepends elle permet de chercher à travers le cache APT pour trouver les dépendances d'un paquet. Cette commande doit être installée.

- Installation d'un paquet (exemple)

Il faut utiliser la commande **apt-get install** :

```
# apt-get install openssh-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Paquets suggérés :
  ssh-askpass rssh molly-guard ufw monkeysphere
Les NOUVEAUX paquets suivants seront installés :
  openssh-server
0 mis à jour, 1 nouvellement installés, 0 à enlever et 0 non mis à
jour.
Il est nécessaire de prendre 339 ko dans les archives.
Après cette opération, 725 ko d'espace disque supplémentaires
seront utilisés.
Réception de : 1 http://debian.proxad.net/debian/ wheezy/main
  openssh-server amd64 1:6.0p1-4 [339 kB]
  339 ko réceptionnés en 0s (394 ko/s)
Préconfiguration des paquets...
Sélection du paquet openssh-server précédemment désélectionné.
(Lecture de la base de données... 160570 fichiers et répertoires
déjà installés.)
Dépaquetage de openssh-server (à partir de .../openssh-
server_1%3a6.0p1-4_amd64.deb) ...
Traitement des actions différées (« triggers ») pour « man-db »...
Paramétrage de openssh-server (1:6.0p1-4) ...
Creating SSH2 RSA key; this may take some time ...
Creating SSH2 DSA key; this may take some time ...
Creating SSH2 ECDSA key; this may take some time ...
[ ok ] Restarting OpenBSD Secure Shell server: sshd.
```

- Recherche des paquets (exemple)

Rechercher un paquet s'effectue avec **apt-cache search** :

```
# apt-cache search openssh-server
  openssh-server - secure shell (SSH) server, for secure access from
  remote machines
```

Utilisez des métacaractères pour trouver un paquet dont vous ne connaissez pas exactement le nom :
comme **#apt-cache search ^openj**

- Description d'un paquet (exemple)

La sous-commande **apt-cache show** permet d'afficher la description d'un paquet :

```
# apt-cache show openssh-server
Package: openssh-server
Source: openssh
Version: 1:6.0p1-4
Installed-Size: 708
Maintainer: Debian OpenSSH Maintainers <debian-ssh@lists.debian.org>
Architecture: amd64
Replaces: openssh-client (<< 1:3.8.1p1-11), ssh, ssh-krb5
Provides: ssh-server
Depends: libc6 (>= 2.8), libcomerr2 (>= 1.01), libgssapi-krb5-2
(>= 1.10+dfsg-), libkrb5-3 (>= 1.6.dfsg.2), libpam0g (>=
0.99.7.1), libselinux1 (>= 1.32), libssl1.0.0 (>= 1.0.1), libwrap0
(>= 7.6-4~), zlib1g (>= 1:1.1.4), debconf (>= 1.2.0) | debconf-
2.0, openssh-client (= 1:6.0p1-4), libpam-runtime (>= 0.76-14),
libpam-modules (>= 0.72-9), adduser (>= 3.9), dpkg (>= 1.9.0),
lsb-base (>= 3.2-13), procps
Recommends: xauth, ncurses-term, openssh-blacklist, openssh-
blacklist-extra
Suggests: ssh-askpass, rssh, molly-guard, ufw, monkeysphere
Conflicts: rsh-client (<< 0.16.1-1), sftp, ssh (<< 1:3.8.1p1-9),
ssh-krb5 (<< 1:4.3p2-7), ssh-nonfree (<< 2), ssh-socks, ssh2
Description-fr: serveur shell sécurisé (SSH), pour accéder à des
machines à distance
Il s'agit d'une version portable de OpenSSH, l'implémentation
libre du
protocole SSH tel qu'il a été défini par le groupe de travail
« secsh »
de l'IETF.
.
Ssh (Secure Shell) est un programme qui permet de se connecter à
une
machine distante et d'y exécuter des commandes. Il fournit des
communications sécurisées et chiffrées entre deux hôtes non fiables
à
travers un réseau non sécurisé. Les connexions X11 ainsi que tout
port
TCP/IP peuvent également être redirigés dans ce canal sécurisé.
Il peut être utilisé pour fournir des canaux de communication
sécurisés à des applications.
```

...

- Affichage des dépendances (exemple)

Nous avons plusieurs solutions pour identifier les dépendances d'un paquet.

La commande **apt-cache depends** liste toutes les dépendances nécessaires d'un paquet :

```
# apt-cache depends openssh-server
openssh-server
Dépend: libc6
Dépend: libcomerr2
Dépend: libgssapi-krb5-2
Dépend: libkrb5-3
Dépend: libpam0g
Dépend: libselinux1
Dépend: libssl1.0.0
Dépend: libwrap0
Dépend: zlib1g
|Dépend: debconf
  Dépend: <debconf-2.0>
    cdebcnf
    debconf
  Dépend: openssh-client
  Dépend: libpam-runtime
  Dépend: libpam-modules
  Dépend: adduser
  Dépend: dpkg
  Dépend: lsb-base
  Dépend: procps
  Suggère: ssh-askpass
    ksshaskpass
    kwallletcli
    ssh-askpass-fullscreen
    ssh-askpass-gnome
  Suggère: rssh
  Suggère: molly-guard
  Suggère: ufw
  Suggère: monkeysphere
  Recommande: xauth
  Recommande: ncurses-term
  Recommande: openssh-blacklist
  Recommande: openssh-blacklist-extra
  Est en conflit avec: rsh-client
  Est en conflit avec: <sftp>
  Est en conflit avec: ssh
  Est en conflit avec: ssh-krb5
  Est en conflit avec: <ssh-nonfree>
  Est en conflit avec: <ssh-socks>
  Est en conflit avec: <ssh2>
  Remplace: openssh-client
  Remplace: ssh
  Remplace: ssh-krb5
```

- Installer les dépendances (exemple)

L'utilisation de **apt-get build-dep** installe les dépendances nécessaires pour construire le paquet donné en argument.

Voici un exemple qui montre comment installer les dépendances de Qt :

```
# apt-get -y build-dep libqt4-dev
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Choix de « qt4-x11 » comme paquet source à la place de « libqt4-dev »
Note : sélection de « libjpeg8-dev » au lieu de « libjpeg-dev »
Les paquets suivants seront ENLEVÉS :
libqtwebkit-dev
Les NOUVEAUX paquets suivants seront installés :
comerr-dev firebird-dev firebird2.5-common firebird2.5-common-doc flex freetds-common freetds-dev krb5-multidev libasound2-dev libaudio-dev libct4
libcups2-dev libdbus-1-dev libfbclient2 libgcrypt11-dev
libgnutls-dev libgnutls-openssl27 libgnutlsxx27 libgpg-error-dev
libgssrpc4
libgstreamer-plugins-base0.10-dev libgstreamer0.10-dev libibutil libicu-dev libjbig-dev libjpeg8-dev libkadm5clnt-mit8
libkadm5srv-mit8 libkdb5-6
libkrb5-dev liblcms1-dev libltdl-dev libmng-dev libmysqlclient-dev libodbc1 libp11-kit-dev libpam0g-dev libpq-dev libpq5
libreadline-dev libreadline6-dev
libsqLite0 libsqLite0-dev libsqLite3-dev libssl-dev libsybdb5
libtasn1-3-dev libtiff4-dev libtiffxx0c2 libxmu-dev libxmu-headers
libxslt1-dev libxt-dev
libxtst-dev libxv-dev odbcinst odbcinstdebian2 pkg-kde-tools
unixodbc unixodbc-dev x11proto-record-dev x11proto-video-dev
0 mis à jour, 62 nouvellement installés, 1 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 23,4 Mo dans les archives.
...
...
```

- Identification du dépôt d'un paquet (exemple)

La commande **apt-cache madison** affiche le dépôt fournissant les paquets spécifiés :

```
# apt-cache madison openssh-server
openssh-server | 1:6.0p1-4 | http://debian.proxad.net/debian/
wheezy/main amd64 Packages
      openssh | 1:6.0p1-4 | http://debian.proxad.net/debian/
wheezy/main Sources
```

- Provecance d'un paquet (exemple)

Pour savoir dans quel paquet se trouve un fichier, nous pouvons utiliser **apt-file**. Comme cet outil n'est pas installé de base, nous devons l'installer :

```
# apt-get -y install apt-file
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus
nécessaire :
 libqtwebkit4
Veuillez utiliser « apt-get autoremove » pour le supprimer.
Les paquets supplémentaires suivants seront installés :
 curl libconfig-file-perl libcurl3 liblist-moreutils-perl
libregexp-assemble-perl
Les NOUVEAUX paquets suivants seront installés :
 apt-file curl libconfig-file-perl libcurl3 liblist-moreutils-
perl libregexp-assemble-perl
0 mis à jour, 6 nouvellement installés, 0 à enlever et 0 non mis à
jour.
Il est nécessaire de prendre 767 ko dans les archives.
Après cette opération, 1 439 ko d'espace disque supplémentaires ...
```

- Lister le contenu d'un paquet (exemple)

La commande **apt-file list** permet de voir le contenu d'un paquet. Cette fonctionnalité est très proche de la commande **dpkg -L** excepté que le paquet n'a pas besoin d'être installé.

```
# apt-file list openssh-server
openssh-server: /etc/default/ssh
openssh-server: /etc/init.d/ssh
openssh-server: /etc/network/if-up.d/openssh-server
openssh-server: /etc/pam.d/sshd
openssh-server: /etc/ufw/applications.d/openssh-server
openssh-server: /usr/lib/openssh/sftp-server
openssh-server: /usr/lib/sftp-server
openssh-server: /usr/sbin/sshd
openssh-server: /usr/share/doc/openssh-client/examples/sshd_config
openssh-server: /usr/share/doc/openssh-server
openssh-server: /usr/share/lintian/overrides/openssh-server
openssh-server: /usr/share/man/man5/authorized_keys.5.gz
openssh-server: /usr/share/man/man5/sshd_config.5.gz
openssh-server: /usr/share/man/man8/sftp-server.8.gz
openssh-server: /usr/share/man/man8/sshd.8.gz
```

- Versions des paquets disponibles (exemple)

La commande **apt-show-versions** permet d'analyser le fichier statut de **dpkg** et les listes d'apt pour trouver les versions des paquets installés, les versions des paquets disponibles mais aussi la distribution. Cette commande doit être installée :

```
# apt-get install apt-show-versions
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus
nécessaire :
  libqtwebkit4
Veuillez utiliser « apt-get autoremove » pour le supprimer.
Les NOUVEAUX paquets suivants seront installés :
  apt-show-versions
0 mis à jour, 1 nouvellement installés, 0 à enlever et 0 non mis à
jour.
Il est nécessaire de prendre 34,9 ko dans les archives.
...
...
```

- Mise à jour des listes de paquets (exemple)

Sur Internet, les paquets sont mis à jour en permanence dans les dépôts.

Sur un poste de travail ou bien un serveur Linux, seuls les dépôts mentionnés dans le fichier **/etc/apt/sources.list** sont utilisés.

Apt conserve une liste des paquets obtenus lors de la dernière mise à jour. Il est donc nécessaire de mettre à jour régulièrement la liste des paquets avec **apt-get update** :

```
# apt-get update
Réception de : 1 http://security.debian.org wheezy/updates
Release.gpg [836 B]
Réception de : 2 http://security.debian.org wheezy/updates Release
[102 kB]
Réception de : 3 http://security.debian.org wheezy/updates/main
Sources [57,7 kB]
Réception de : 4 http://security.debian.org wheezy/updates/contrib
Sources [14 B]
Réception de : 5 http://security.debian.org wheezy/updates/non-
free Sources [14 B]
Réception de : 6 http://security.debian.org wheezy/updates/main
amd64 Packages [106 kB]
Réception de : 7 http://security.debian.org wheezy/updates/contrib
amd64 Packages [14 B]
Réception de : 8 http://security.debian.org wheezy/updates/non-
free amd64 Packages [14 B]
Réception de : 9 http://security.debian.org wheezy/updates/contrib
Translation-en [14 B]
Réception de : 10 http://security.debian.org wheezy/updates/main
Translation-en [62,2 kB]
Réception de : 11 http://security.debian.org wheezy/updates/non-
free Translation-en [14 B]
Atteint http://debian.proxad.net wheezy Release.gpg
Réception de : 12 http://debian.proxad.net wheezy-updates
Release.gpg [1 571 B]
Atteint http://debian.proxad.net wheezy Release
Réception de : 13 http://debian.proxad.net wheezy-updates Release
[124 kB]
Atteint http://debian.proxad.net wheezy/main Sources
Atteint http://debian.proxad.net wheezy/non-free Sources
Atteint http://debian.proxad.net wheezy/contrib Sources
Atteint http://debian.proxad.net wheezy/main amd64 Packages
Atteint http://debian.proxad.net wheezy/non-free amd64 Packages
Atteint http://debian.proxad.net wheezy/contrib amd64 Packages
Atteint http://debian.proxad.net wheezy/contrib Translation-en
Atteint http://debian.proxad.net wheezy/main Translation-fr
Atteint http://debian.proxad.net wheezy/main Translation-en
Atteint http://debian.proxad.net wheezy/non-free Translation-en
Réception de : 14 http://debian.proxad.net wheezy-updates/main
Sources [1 995 B]
Réception de : 15 http://debian.proxad.net wheezy-updates/contrib
...
...
```

- Mise à jour des dépôts (exemple)

La commande **apt-get upgrade** met à jour tous les paquets installés sur le système vers les dernières versions :

```
# apt-get -y upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants seront mis à jour :
  icedtea-netx icedtea-netx-common
2 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à
jour.
Il est nécessaire de prendre 876 ko dans les archives.
Après cette opération, 0 o d'espace disque supplémentaire seront
utilisés.
Réception de : 1 http://security.debian.org/ wheezy/updates/main
icedtea-netx-common all 1.4-3~deb7u2 [855 kB]
Réception de : 2 http://security.debian.org/ wheezy/updates/main
icedtea-netx amd64 1.4-3~deb7u2 [20,7 kB]
876 ko réceptionnés en 1s (627 ko/s)
Lecture des fichiers de modifications (« changelog »)... Terminé
(Lecture de la base de données... 163482 fichiers et répertoires
déjà installés.)
Préparation du remplacement de icedtea-netx-common 1.4-3~deb7u1
(en utilisant .../icedtea-netx-common_1.4-3~deb7u2_all.deb) ...
Dépaquetage de la mise à jour de icedtea-netx-common ...
Préparation du remplacement de icedtea-netx:amd64 1.4-3~deb7u1 (en
utilisant .../icedtea-netx_1.4-3~deb7u2_amd64.deb) ...
Dépaquetage de la mise à jour de icedtea-netx:amd64 ...
Traitement des actions différées (« triggers ») pour « desktop-
file-utils »...
Traitement des actions différées (« triggers ») pour « gnome-
menus »...
Traitement des actions différées (« triggers ») pour « hicolor-
icon-theme »...
Traitement des actions différées (« triggers ») pour « man-db »...
Paramétrage de icedtea-netx-common (1.4-3~deb7u2) ...
Paramétrage de icedtea-netx:amd64 (1.4-3~deb7u2) ...
```

- Mise à jour de tous les paquets installés (exemple)

La commande **apt-get dist-upgrade** met à jour tous les paquets installés vers les dernières versions en installant de nouveaux paquets si nécessaire, par opposition à upgrade qui n'ajoute pas de nouveaux paquets.

```
# apt-get dist-upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à
jour.
```

Avant d'exécuter dist-upgrade, il est recommandé de mettre à jour d'abord la liste des paquets avec update comme `#sudo apt-get update && sudo apt-get upgrade`.

- Désinstallation d'un paquet (exemple)

La commande **apt-get remove** permet la désinstallation de paquets :

```
# apt-get -y remove openssh-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus
nécessaire :
  libqtwebkit4
Veuillez utiliser « apt-get autoremove » pour le supprimer.
Les paquets suivants seront ENLEVÉS :
  openssh-server
...
```

Cette commande peut également être accompagnée de l'option **-purge** pour supprimer les fichiers de configuration comme **#apt-get -y remove --purge openssh-server**

La commande **apt-get autoremove** permet de désinstaller le paquet indiqué avec ses dépendances logicielles comme **#apt-get -y autoremove openssh-server**

Donc, pour supprimer définitivement un paquet, on peut combiner les 2 options ci-dessus

Comme **#apt-get -y autoremove --purge openssh-server**

- Suppression des paquets d'installation (exemple)

Apt conserve une copie de chaque paquet installé et désinstallé dans le répertoire **/var/cache/apt/archives**.

La commande **apt-get clean** supprime les copies de paquets installés et permet ainsi de récupérer de l'espace disque :

```
#apt-get clean
```

La commande **#apt-get autoclean** permet de supprimer les copies des paquets désinstallés.

- Gérer des paquets deb avec dpkg

Créé par Ian Jackson en 1993, **dpkg** est une commande chargée de l'installation, la création, la suppression et la gestion des paquets .deb de Debian. Elle permet également la gestion de paquets en provenance de sources extérieures aux dépôts APT. Vous pouvez exécuter **#man dpkg** pour avoir plus d'information sur son utilisation.

Internet & Réseaux

« Linux sans internet c'est comme une viande sans sauce »

■ GÉNÉRALITÉ



Internet est un réseau mondial de télécommunication entre des ordinateurs (*le réseau des réseaux*).

Il permet à tous types de périphériques électroniques – ordinateurs personnels, ordinateurs portables, téléphones mobiles, supercalculateurs, voire même des appareils électroménagers tels les téléviseurs et les réfrigérateurs – de communiquer entre eux et s'échanger, dans le cadre de ce que l'on appelle des services, des informations diverses : pages d'information, courriers électroniques, musique, films, messages instantanés, appels téléphoniques et de vidéoconférence, etc.

Ce réseau, tentaculaire et planétaire, est composé d'une multitude de liens qui, si illustrés sur un graphique, s'apparenteraient à toile d'araignée. Cette image a donné son nom à l'un des plus populaires services d'Internet : le *World Wide Web* (littéralement "toile - d'araignée - de dimension mondiale"), un service de consultation des pages d'information (les "pages Web"). Cependant, le WWW est loin d'être le seul service offert par Internet. Pour n'en nommer que quelques-uns, on retrouve aussi des services de courrier électronique, de messagerie instantanée, de téléphonie, de diffusion en direct de musique et de vidéo, d'échange de fichiers, etc.

Ainsi, Internet n'est-il pas lui-même un programme que l'on peut installer dans son ordinateur. C'est un réseau (*network*, en anglais) auquel chacun peut accéder par une connexion – communément appelée une *connexion à Internet*. Une fois le réseau rejoint, tous les services proposés à travers ce réseau Internet sont disponibles à vous, sous réserve de disposer des fonctions "client" permettant de les utiliser.

✓ Comprendre Internet en 6 heures en vidéo (gratuit)

Voir <http://www.ilico.org/2010/08/comprendre-internet-en-6-heures-et-en-video/>

✓ Cours sur les réseaux / modélisation réseau / fonctionnement de TCP/IP

Voir <http://olivieraj.free.fr/fr/linux/information/firewall/>

Voir <https://inetdoc.developpez.com/tutoriels/modelisation-reseau/>

Voir <https://openclassrooms.com/fr/courses/857447-apprenez-le-fonctionnement-des-reseaux-tcp-ip>

✓ Cours avancé sur les réseaux

Voir <http://stielec.ac-aix-marseille.fr/electron/cours.htm#internet>

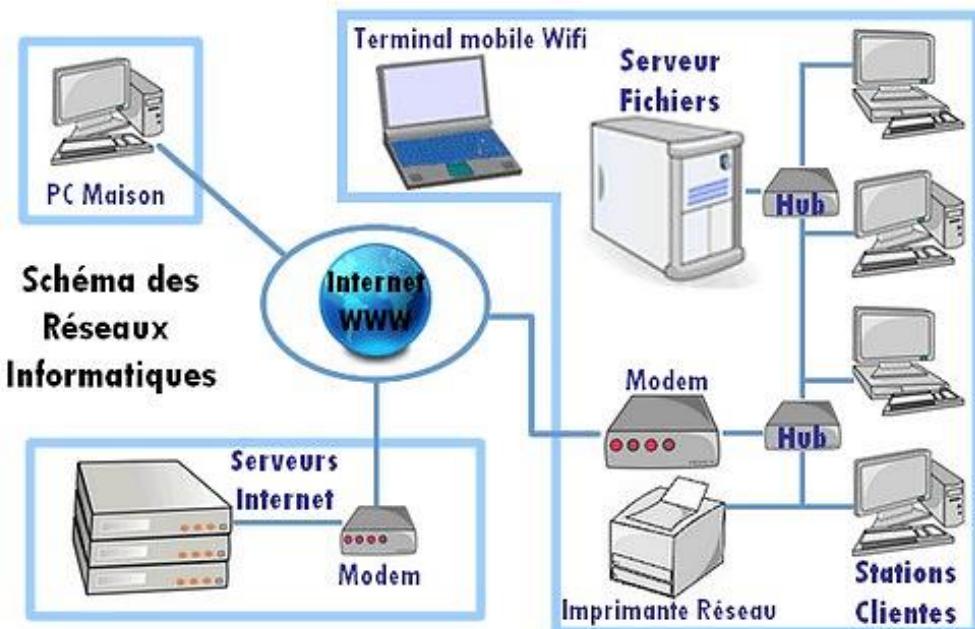


Un **réseau informatique** (en anglais, *data communication network* ou *DCN*) est un ensemble d'équipements reliés entre eux pour échanger des informations. Par analogie avec un filet (un réseau est un « petit rets », c'est-à-dire un petit filet), on appelle nœud l'extrémité d'une connexion, qui peut être une intersection de plusieurs connexions ou équipements (un ordinateur, un routeur, un concentrateur, un commutateur).

Indépendamment de la technologie sous-jacente, on porte généralement une vue matricielle sur ce qu'est un réseau.

De façon *horizontale*, un réseau est une strate de trois couches : les infrastructures, les fonctions de contrôle et de commande, les services rendus à l'utilisateur. De façon *verticale*, on utilise souvent un découpage géographique : réseau local, réseau d'accès et réseau d'interconnexion.

Notion de réseau



On utilise principalement 2 stratégies pour faire circuler une information sur le réseau.

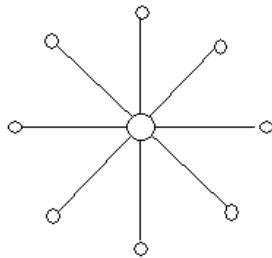
Soit :

- **Envoie complète de l'information** : très rarement utilisée car les risques d'erreurs sont importants.
- **Envoie par paquets** : décomposition des fichiers en petits morceaux, ce dernier consiste à envoyer séparément sur le réseau puis réassemblés par la machine destinataire.

■ Une histoire de mailles

Internet a été créé au départ pour une raison bien particulière. À l'époque, dans les années 1950, les communications étaient « point à point », c'est-à-dire qu'on ne pouvait communiquer qu'avec une seule machine à la fois. Les chercheurs qui devaient communiquer avec plusieurs autres chercheurs lors de réunions se sont rendu compte qu'il serait intéressant de pouvoir le faire en temps réel plutôt que de passer d'un interlocuteur à l'autre successivement.

Ils ont donc cherché à créer un nouveau moyen de communication qui ne serait alors plus centralisé, mais maillé.



Réseau de communication centralisé

Cela veut dire que toute information pourrait passer par différents points, et que si certains points disparaissaient, cela n'empêcherait pas l'information de circuler. Observez donc la figure suivante : vous voyez qu'avec un réseau de communication maillé, si un point de communication n'est plus en état de fonctionner, l'information peut passer par un chemin différent.

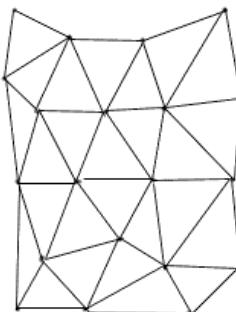


Schéma d'un réseau maillé

Mais maintenant que l'idée est posée, il reste à la mettre en œuvre !

Les chercheurs vont travailler et notamment mettre en place un réseau pour l'armée. C'est seulement au **début des années 1960** que l'on voit apparaître des textes décrivant les prémisses de ce que sera Internet.

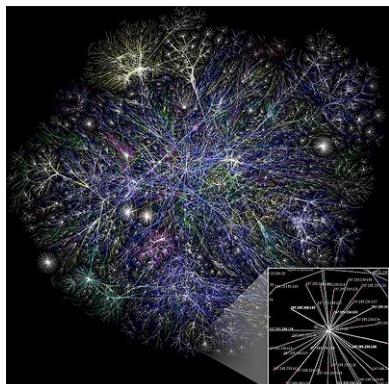
À la **fin des années 1960**, l'Arpanet, l'ancêtre d'Internet, ne comportait que quatre machines ! Les protocoles utilisés alors ne permettaient pas d'atteindre les buts fixés, à savoir de faire dialoguer des machines provenant de différents réseaux en utilisant différentes technologies de communication.

C'est alors que les chercheurs se sont orientés vers la création d'autres protocoles de communication, et notamment TCP/IP. Internet a continué de croître au fil des années, mais c'est **en 1990** qu'une révolution va permettre sa croissance réelle : le langage HTML et le protocole d'échange HTTP qui permet la création de pages web.

Tout va s'accélérer alors avec la création des premiers navigateurs capables d'afficher des images, et la libération de l'utilisation des noms de domaine. Nous pouvons voir sur la figure suivante la progression phénoménale d'Internet dans les **années 1990-2000**.

■ Internet aujourd'hui

Aujourd'hui, Internet c'est **3,4 milliards** d'internautes et **200 millions** de serveurs.



Parmi ces internautes, nous pouvons voir des disparités à travers le monde :

- 42 % des internautes viennent d'Asie !
- Le pays le plus internetisé est... la Corée du Sud ;
- Les internautes français représentent 1,6 % du total des internautes ;
- 78% des Américains ont Internet contre 10 % des Africains ;
- Une personne sur trois dans le monde a accès à Internet ;
- Le nombre d'internautes entre 2000 et 2010 a été multiplié par 4,5 ;
- La croissance de l'Internet en Afrique est de 2360 % entre 2000 et 2010

Je ne vais pas continuer à vous abreuver de chiffres, bien que certains soient étonnantes à connaître. Voyons plutôt le magnifique graphique présenté en figure suivante, qui représente les connexions entre machines d'Internet. Prenez votre loupe !

Cependant, n'oublions pas notre objectif premier : **comprendre le fonctionnement d'Internet**. Donc fini de rêvasser, passons aux choses sérieuses !

Maintenant que nous connaissons une partie de l'histoire d'Internet, il est grand temps de nous plonger dans son fonctionnement, notamment en étudiant sa création.

Nous sommes près de 1,8 milliard d'internautes aujourd'hui. Internet est une gigantesque toile d'araignée.

Comment est-ce possible de faire communiquer autant de machines ?

Comment ne pas s'y perdre dans ce dédale d'informations ?

Nous allons voir cela de ce pas, en essayant tout d'abord de comprendre comment Internet a été créé et quelles sont les normes qui ont été mises en œuvre pour orchestrer ce bal d'informations.

■ Comment communiquer ?



Imaginez que vous puissiez communiquer à chaque instant, quand vous le voulez, avec n'importe qui dans le monde ! C'est ce que nous propose Internet. Il n'est pas facile de s'exprimer lorsque nous sommes un petit groupe de 10 personnes, difficile lorsque nous sommes 100, et quasiment impossible quand nous sommes 1000. Internet se propose donc de relever le défi de pouvoir communiquer tous ensemble, en même temps, et ce, quand nous le souhaitons. Bien sûr pour arriver à cette prouesse, il a fallu créer un système de communication complexe permettant aux machines de parler entre elles.

➤ Mais comment ce modèle de communication a-t-il pu être créé ?

Eh bien le plus simple est de partir de ce que nous connaissons déjà de la communication. Et ça, tout le monde peut le faire !

Faisons un petit inventaire des moyens de communication :

- La parole ;
- Le téléphone ;
- Le courrier ;
- Le pigeon voyageur ;
- etc...

Essayons maintenant de comprendre, parmi ces moyens de communication, ce dont nous avons besoin pour communiquer.

Pour la parole, nous avons besoin :

- D'un émetteur ;
- D'un récepteur ;
- D'un support de transmission (l'air).

Pour le téléphone, c'est un peu pareil sauf que nous avons besoin d'un élément complémentaire qui est l'intermédiaire entre la parole et l'électronique. En effet, on transforme la parole en signaux électriques, ils arrivent côté récepteur, puis ils sont de nouveau transformés en paroles. Nous voyons qu'il y a une *encapsulation* de l'information.

Nous retrouvons ce système d'encapsulation dans le courrier, pour lequel nous avons besoin :

- D'un émetteur ;
- D'un récepteur ;
- D'un support de transmission (la lettre) ;
- D'un contenant (l'enveloppe) ;
- D'un intermédiaire (la poste).

Ainsi, nous commençons à comprendre ce qu'il nous faut pour communiquer.

- Maintenant, est-ce que cela va pouvoir s'appliquer aux ordinateurs ? Comment va-t-on faire pour parler tous en même temps ? Pourra-t-on communiquer avec l'autre bout du monde instantanément ?

Nous allons voir par la suite comment les chercheurs ont fait pour passer des principes de communication humains à des principes de communication pour ordinateurs.

Ils ont ainsi regroupé l'ensemble de leurs recherches et de leurs résultats dans une norme que devront respecter les personnes se connectant à Internet.

Il s'agit du **modèle OSI** !

■ Le modèle OSI



Open Systems Interconnection

Le modèle **OSI** est né en 1984. Les plus connaisseurs d'entre vous auront remarqué que celui-ci est né après la naissance d'Internet ! La raison est simple : le modèle OSI est né quand nous avons commencé à avoir une certaine expérience des communications entre ordinateurs. Il tient donc compte des communications existantes, mais aussi des communications futures et de leurs évolutions potentielles.

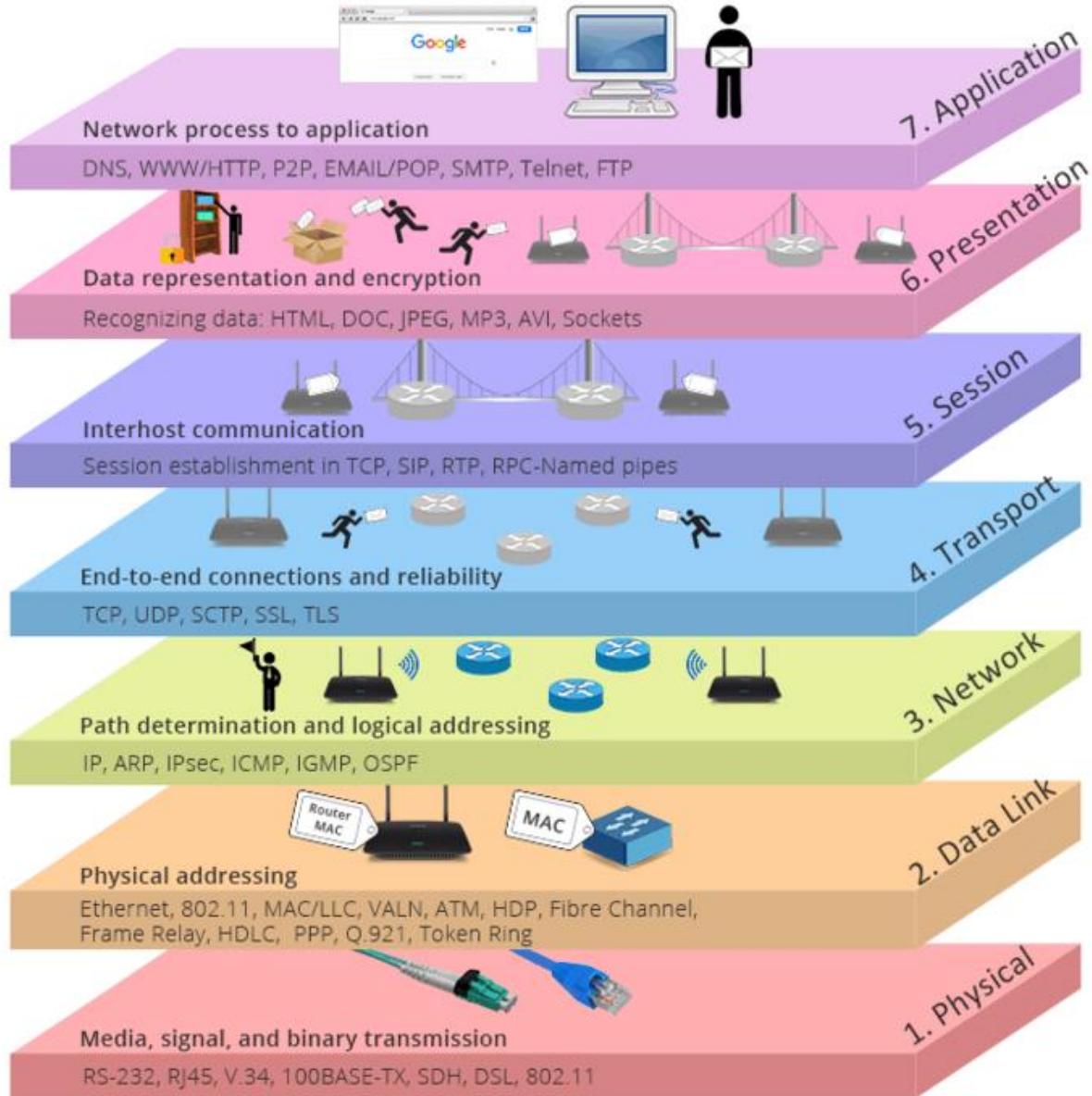
Son objectif est de normaliser les communications pour garantir un maximum d'évolutivité et d'interopérabilité entre les ordinateurs.

- Tout cela est fort sympathique, mais qu'est-ce que le modèle OSI ?

Le modèle OSI est une norme qui préconise comment les ordinateurs devraient communiquer entre eux. Ainsi, si vous voulez faire communiquer votre grille-pain avec votre lave-vaisselle, il faudra vous appuyer sur le modèle OSI, ou du moins vous en inspirer le plus possible. Cela impliquera notamment le respect de la communication par couches.

- Mais qu'est-ce que c'est que ces couches ?

Non, je ne suis pas devenu fou et ne suis pas sponsorisé par Pampers ! Le modèle OSI est un modèle en couches. Cela veut dire qu'il est découpé en plusieurs morceaux appelés couches, qui ont chacune un rôle défini, comme vous le montre le schéma de la figure suivante.



Nous voyons ici que le modèle OSI a sept couches. Chacune possède un nom différent.

➤ Mais pourquoi 7, et pas 14 ou 137 ?

Souvenez-vous du paragraphe précédent. Nous y avons vu que pour mettre en place une communication, il nous faudrait mettre en œuvre un certain nombre d'éléments, comme l'émetteur, le récepteur, le langage, etc. Eh bien les chercheurs ont imaginé combien d'éléments principaux il faudrait mettre en place pour communiquer. Et ils en ont trouvé 7 !

Chaque couche du modèle OSI va donc avoir un rôle à accomplir. Et l'ensemble de ces rôles va permettre de communiquer d'un ordinateur à un autre.

Examinons ces couches un peu plus en détail...

✓ **La couche 1 ou couche physique :**

- Nom : Physique.
- Rôle : offrir un support de transmission pour la communication.
- Rôle secondaire : RAS.
- Matériel associé : le *hub*, ou concentrateur en français.

✓ **La couche 2 ou couche liaison :**

- Nom : Liaison de données.
- Rôle : connecter les machines entre elles sur un *réseau local*.
- Rôle secondaire : détecter les erreurs de transmission.
- Matériel associé : le *switch*, ou commutateur.

✓ **La couche 3 ou couche réseau :**

- Nom : Réseau.
- Rôle : interconnecter les réseaux entre eux.
- Rôle secondaire : fragmenter les paquets.
- Matériel associé : le routeur.

✓ **La couche 4 ou couche transport :**

- Nom : Transport.
- Rôle : gérer les connexions applicatives.
- Rôle secondaire : garantir la connexion.
- Matériel associé : RAS.

✓ **La couche 5 ou couche session :** on s'en fiche !

Oui, vous m'avez bien lu, au-delà de la couche 4, on s'en fiche ! Bon, j'exagère un poil, mais pas tant que ça. La raison est simple : le modèle OSI est un modèle théorique. Le modèle sur lequel s'appuie Internet aujourd'hui est le modèle TCP/IP. Or, ce modèle n'utilise pas les couches 5 et 6, donc... on s'en fiche !

Bon, je crois que vous avez compris ! Par contre, la couche 7 existe bien. Et c'est pour elle que nous mettons tout cela en place, le grand manitou, le patron, l'**application** !

✓ **La couche 7 ou couche application :**

- Nom : Application.
- Rôle : RAS.
- Rôle secondaire : RAS.
- Matériel associé : le proxy.

➤ Quoi ? Une couche qui n'a pas de rôle ? Pourquoi est-elle là alors ?

Elle est là pour représenter les applications pour lesquelles nous allons mettre en œuvre des communications.

Ce n'est donc pas cette couche en elle-même que nous allons étudier, mais les couches qui sont là pour lui rendre service et acheminer les informations, les couches 1 à 4.

Les couches 1 à 4 sont appelées les couches "réseau". Ce sont elles qui ont la responsabilité d'acheminer les informations d'une machine à une autre, pour les applications qui le demandent.

Avant d'examiner plus en détail les couches, nous allons préciser le cadre d'utilisation du modèle OSI.

■ Règle d'or du modèle OSI

Le modèle OSI étant une norme, il doit indiquer, aux personnes voulant mettre en place des réseaux, comment travailler. Plus exactement, cela permet aux constructeurs de matériels de réseau de savoir comment fabriquer leurs matériels, et donc de garantir la compatibilité entre eux.

Si chacun respecte la norme, ça marche !

Nous avons vu que chaque couche avait un rôle qu'il faudra respecter. Ainsi, la couche 2 ne s'occupera jamais de la communication entre réseaux. De même que la couche 3 ne s'occupera pas de la communication sur un réseau local, etc.

Le modèle OSI ajoute deux règles plus générales entre les couches :

- Chaque couche est indépendante ;
- Chaque couche ne peut communiquer qu'avec une couche adjacente.

➤ Chaque couche est indépendante

L'impact sera que les informations utilisées par une couche ne pourront pas être utilisées par une autre couche.

Par exemple, pour ceux qui connaissent déjà un peu le réseau, l'adresse IP qui est une adresse de couche 3 ne pourra pas être utilisée par une autre couche, sous peine de ne pas respecter le modèle OSI.

Cela va permettre de garantir l'évolution des communications dans le temps.

Imaginez que vous utilisez Internet aujourd'hui. Sans le savoir, vous utilisez le protocole IPv4 pour la couche 3. Demain, nous allons passer en protocole IPv6 pour des raisons que nous expliciterons avec la couche 3. Si jamais nous utilisons des adresses IPv4 dans une autre couche, le jour où nous changerons le protocole de couche 3 qui utilise les adresses IPv4, nous devrons changer aussi le ou les protocoles qui utilisent cette adresse.

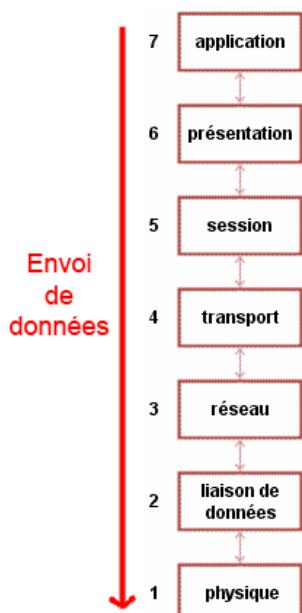
Rendre les couches indépendantes garantit qu'elles sont interchangeables.

Cela veut dire qu'on pourra changer un protocole associé à une couche sans avoir besoin de changer toutes les couches du modèle OSI.

C'est un peu comme si vous aviez une commode avec des tiroirs. Vous pouvez changer un tiroir cassé sans avoir à changer toute la commode !

Regardons la seconde règle...

➤ Chaque couche ne peut communiquer qu'avec une couche adjacente



Pour comprendre cette règle, vous allez devoir comprendre comment les machines se servent du modèle OSI pour communiquer.

Vous êtes devant votre ordinateur et votre navigateur préféré. Vous entrez l'adresse d'un site dans la barre d'adresses, et le site apparaît aussitôt.

Sans le savoir, vous avez utilisé le modèle OSI !

En gros, l'application (le navigateur) de couche 7, s'est adressée aux couches réseau pour que celles-ci transmettent l'information à l'application demandée sur la machine demandée (le serveur web sur la machine google.com par exemple).

Lors d'un envoi, nous parcourons donc les couches du modèle OSI de haut en bas, de la couche 7 à la couche 1, ainsi que vous pouvez le voir sur la figure suivante.

Ainsi, grâce à la seconde règle du modèle OSI, **nous garantissons que lors de l'envoi d'informations, toutes les couches du modèle OSI vont être parcourues**.

Ceci est garanti, car nous partons de la couche 7, et la règle nous dit qu'une couche ne peut communiquer qu'avec une couche adjacente. La couche 7 ne pourra donc communiquer qu'avec la couche directement sous elle, la couche 6.

En fait, c'est presque vrai, car comme vous le savez maintenant, le modèle OSI n'est qu'un modèle théorique, et la couche 7 s'adresse directement aux couches réseau pour communiquer, soit directement à la couche 4, qui s'adresse à la couche 3, qui s'adresse à la couche 2...

Nous pouvons ainsi garantir que tous les rôles associés à chaque couche, et donc nécessaires à la communication, vont être remplis !

Ce qu'il faut retenir

- Le modèle OSI est une norme précisant comment les machines doivent communiquer entre elles.
- C'est un modèle théorique, le modèle réellement utilisé étant le modèle TCP/IP.
- Le modèle OSI possède 7 couches.
- Chaque couche a un rôle particulier à accomplir.
- Les couches 1 à 4 sont les couches réseau.
- Les couches réseau offrent le service de communication à la couche applicative.
- Chaque couche est indépendante des autres.
- Chaque couche ne peut communiquer qu'avec une couche adjacente.
- Lors de l'envoi de données, on parcourt le modèle OSI de haut en bas, en traversant toutes les couches.

- vous connaissez et comprenez maintenant le modèle OSI ;
- il est composé de 7 couches chacune dédiée à un rôle particulier ;
- Il y a deux règles d'or associées à ce modèle qui permettent de garantir la bonne utilisation du modèle OSI.

■ Le succès de TCP/IP face à OSI



Plusieurs raisons expliquent le succès de TCP/IP :

Le modèle OSI était principalement utilisé dans la recherche, alors que TCP/IP était déjà dans une phase d'investissement et déjà utilisée avec succès par de nombreuses universités qui étaient à l'époque les principaux utilisateurs de réseaux, historiquement les gens ont donc eu tendance à utiliser le modèle TCP/IP.

De plus la première implémentation de TCP/IP dans l'Unix de l'université de Berkeley (BSD) était gratuite et relativement efficace ; les industriels et divers exploitants n'ont donc pas eu à beaucoup investir pour l'implémenter.

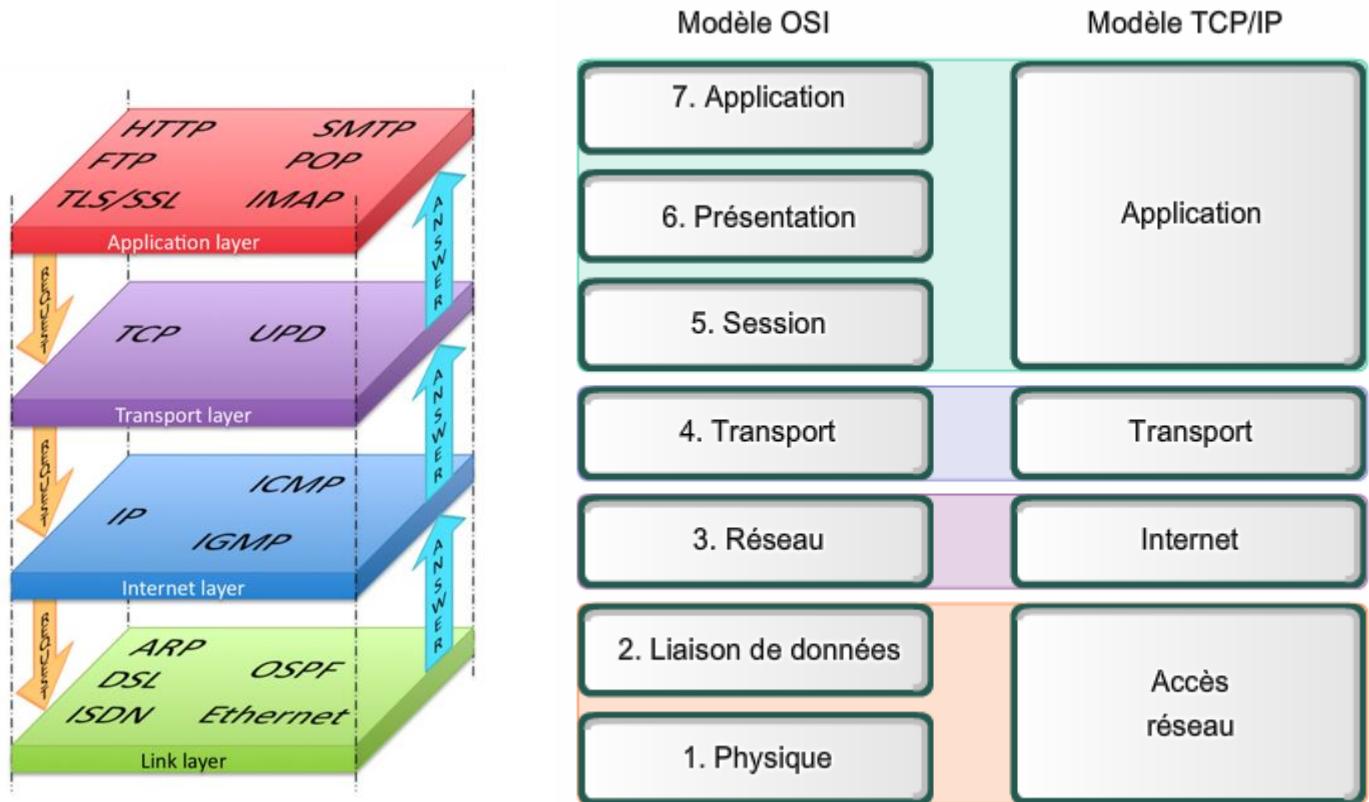
Le modèle OSI a finalement été jugé trop complet et trop complexe. La distance entre l'utilisation (l'implémentation) et le modèle étant parfois importante. Peu de programmes peuvent utiliser ou utilisent mal l'ensemble des 7 couches du modèle : les couches session et présentation sont fort peu utilisées et à l'inverse les couches liaison de données et réseau sont très souvent découpés en sous-couches tant elles sont complexes. Le comité rédacteur de la norme a même dû laisser de côté certains points techniques, comme la sécurité et le codage, pour conserver un rôle clair à chaque couche. Ce modèle est également redondant (le contrôle de flux et le contrôle d'erreur apparaissent pratiquement dans chaque couche).

Au niveau de l'implémentation, TCP/IP est beaucoup plus optimisé et efficace.

Tout ceci explique en partie le succès du modèle TCP/IP.

■ Le modèle TCP/IP

Le modèle **TCP/IP** est plus simple qu'**OSI**, avec seulement quatre couches : liaison, Internet, transport et application. La différence avec **OSI** est simplement que certaines couches ont été fusionnées. La couche liaison de **TCP/IP** regroupe notamment les couches physiques et liaison d'**OSI**. De même, la couche application de **TCP/IP** regroupe les couches session, application et présentation d'**OSI**.



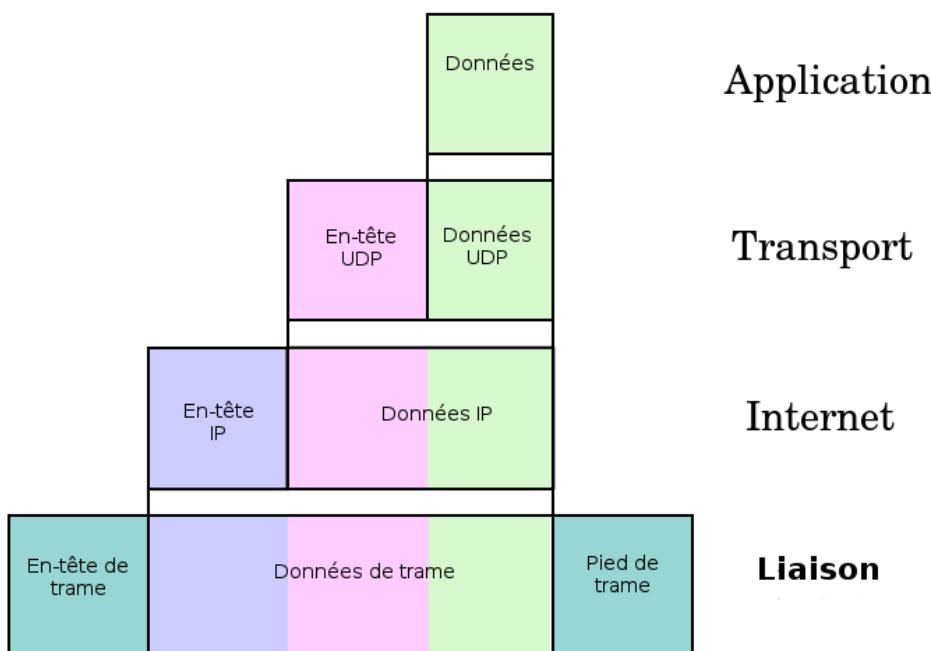
➤ L'encapsulation

Les données ne sont pas transmises telles quelles sur le réseau, car chaque protocole a besoin d'informations bien précises pour faire son travail. Par exemple, les protocoles de couche liaison ont besoin d'informations particulières, non présentes dans la donnée transmise, pour détecter les erreurs ou indiquer le récepteur. Même chose pour les protocoles TCP et UDP de la couche transport, qui ont besoin d'informations sur le processus émetteur et récepteur, qui ne sont pas dans la donnée transmise. Et ce problème nous amène à parler de l'**encapsulation**.

- **Les en-têtes des paquets**

Pour résoudre le problème précédent, chaque protocole ajoute les informations dont il a besoin à la donnée transmise. Ces informations sont regroupées dans un **en-tête**, placé au début des données à transmettre. Plus rarement, certains protocoles ajoutent leurs informations devant, mais aussi derrière la donnée à transmettre : l'en-tête est complété par un *pied* (le terme anglais est : *footer*). Lorsqu'un protocole prend en charge un paquet de données, il lui ajoute un en-tête à son début. Avec cette méthode, les en-têtes de

chaque couche sont séparés, placés les uns à côté des autres. Lors de la réception, cet en-tête sera enlevé : les couches supérieures n'ont pas besoin des en-têtes des couches inférieures.



- **Les Protocol Data Unit**

Si la couche physique s'occupe de la transmission de bits individuels, ce n'est pas le cas des autres couches, qui traitent des paquets (données + en-tête/pied) contenant plusieurs bits. Les couches réseau, liaisons et transports traitent des paquets de données de taille fixe (leur nombre de bits est fixé). Au-delà de la couche transport, les données ne sont pas structurées en paquets de taille fixe, mais en paquets de taille variable, qui dépendent du logiciel utilisé. Pour les couches liaison, réseau et transport, les paquets sont appelés des **protocol data unit**, (unités de données d'un protocole, en français). Ils portent des noms différents selon la couche, vu qu'ils contiennent des en-têtes différents.

- Pour la couche liaison, l'unité est la *trame*.
- Pour la couche réseau, l'unité est le *paquet*.
- Pour la couche transport, l'unité est le *segment* pour le protocole TCP et le *datagramme* pour le protocole UDP.

Couche OSI	Unité de transfert de données (PDU)	Taille du PDU
Physique	Bit ou Symbole	Bit/symbole unique
Liaison	Trame	
Réseau	Paquet	Taille fixe, plusieurs bits, dépendant du protocole utilisé.
Transport	Segment (protocole TCP)	Datagramme (protocole UDP)
Session		
Présentation	Paquets de taille variable	
Application		

■ Classification des réseaux

Historiquement, c'est la distance entre les équipements à connecter qui a constitué le premier critère de classement des réseaux de télécommunication.

Ce critère est fondé sur le mode de transport de l'information. Même si c'est de moins en moins vrai, on part du principe que l'on n'emploie pas les mêmes techniques pour véhiculer des données d'une pièce à l'autre ou d'un continent à l'autre.

Distance	Acronyme	Type de réseau
jusqu'à 25 mètres	PAN	Réseau local « domestique » : Personal Area Network
jusqu'à 10 km	LAN	Réseau local : Local Area Network
jusqu'à 50 km	MAN	Réseau métropolitain : Metropolitan Area Network
jusqu'à 1000 km	WAN	Réseau longue distance : Wide Area Network
jusqu'à 40 000 km	Internet	Réseau mondial

■ Installation et configuration du réseau

D'une manière générale, l'utilisation des paramètres automatiques (comme le DHCP) avec une connexion filaire (type Ethernet) permet d'être connecté sans aucune autre manipulation que celle de brancher le câble réseau.

Mais il existe de très nombreux cas de figure (Wifi, ADSL, Réglage réseau particulier, etc.) qui peuvent nécessiter une intervention particulière. Ils sont détaillés autant que possible dans cette documentation.

➤ Prise en charge du matériel

Indépendamment des paramétrages, pour accéder à un réseau (local ou internet) il faut que le matériel qui sert à se connecter puisse être utilisé par Ubuntu. Il existe plusieurs cas de figure en fonction du type de connexion :

- La connexion **Ethernet** (par fil) est normalement fonctionnelle et utilisable dès le premier démarrage.
- Pour une connexion en **Wifi** (que le module soit intégré ou en USB externe) c'est plus aléatoire. Elle est presque toujours utilisable, mais souvent au prix de quelques manipulations qui sont répertoriées sur les pages Wifi et partage (https://doc.ubuntu-fr.org/partage_de_connexion_internet).

Le choix de solutions Wifi vendues pour être compatibles avec Linux permet d'avoir une connexion fonctionnelle sans manipulation, comme dans le cas d'une connexion filaire.

Pour les connexions à Internet (avec ou sans réseau local) le matériel qui gère la connexion est un modem. (Voir ci-dessous)

- Pour une connexion **ADSL** (Modem connecté par USB & Autres) : voir le tutoriel pour le matériel réseau. (https://doc.ubuntu-fr.org/modem_adsl)

Les "BOX" (freebox, livebox, etc.) ont notamment ce rôle de modem, mais aussi souvent celui de pouvoir gérer le réseau local. Le fait de s'y connecter en "réseau local" (notamment par câble) permet généralement d'avoir accès à Internet depuis Ubuntu sans aucune autre manipulation.

- Pour une connexion par un modem classique (appelé aussi RTC ou bas débit). Le matériel est généralement pris en charge sans problème (voir la page [modem](#)).

La section matérielle réseau du site (<https://doc.ubuntu-fr.org/materiel#reseaux>) contient également des informations sur l'installation de matériels réseaux spécifiques.

- Pour tout ce qui est sur la configuration de son réseau local, consultez la page https://doc.ubuntu-fr.org/tutoriel/comment_configurer_son_reseau_local

➤ Attention virus sur routeur

Les virus sont aussi sur vos routeurs et BOX ADSL :

- Un ver se constitue un botnet de routeurs équipés Linux (psyb0t)

Voir <http://www.generation-nt.com/psyb0t-botnet-routeurs-linux-mipsel-dronebl-actualite-252981.html>

- Chuck Norris Botnet Karate-chops Routers Hard (Chuck Norris)

Voir

http://www.pcworld.com/businesscenter/article/189868/chuck_norris_botnet_karatechops_routers_hard.html?tk=nl_bdx_h_crawl

Solution à ce jour :

1. Rebooter le routeur pour le remettre en configuration de Zéro.
2. Ou installer un driver libre que vous pourrez mettre à jour et surveiller l'activité (https://doc.ubuntu-fr.org/fournisseurs_d_acces_a_internet_locaux) : c'est d'ailleurs le seul moyen car vous n'avez pas la possibilité de voir l'activité avec les drivers par défaut (qui sont comme beaucoup de choses propriétaires pas mis à jour pour la sécurité, ou l'industriel a fermé).

■ CONNEXION À INTERNET (configuration du réseau sous Linux)



L'intégration à Internet se passe par l'intermédiaire d'un fournisseur d'accès à Internet (FAI), qui vous connecte à Internet. Selon la ou les méthodes que vous propose votre FAI, Internet est rejoint à travers de nombreux moyens de télécommunication : soit filaire (réseau téléphonique à bas débit ADSL ou xDSL, câble coaxial, fibre optique, courant électrique porteur CPL), soit sans fil (Wifi, Wimax, Internet par satellite, 3G...).

Les connexions à Internet de type filaire à haut débit ("box", ADSL ou câble coaxial) sont reconnues automatiquement par Ubuntu ; dans les autres cas, vous devrez probablement paramétrier manuellement votre connexion. Ubuntu intègre plusieurs outils pour vous faciliter la tâche.

Plus d'information sur la connexion internet : <https://doc.ubuntu-fr.org/reseau>

Network Manager

Network-Manager est l'outil de gestion des connexions réseau d'Ubuntu. Son utilité est la création et la configuration des accès à divers types de réseaux (Internet, réseau local [LAN], réseau privé virtuel [VPN]...). Network-Manager peut prendre en charge de nombreux types de connexions (réseau filaire, réseau sans-fil [Wi-Fi], modem téléphonique, réseau mobile, réseau privé virtuel...). Network-Manager peut aussi être utilisé pour partager une connexion réseau (généralement Internet) à d'autres ordinateurs de votre réseau local.

■ Installation / réinstallation

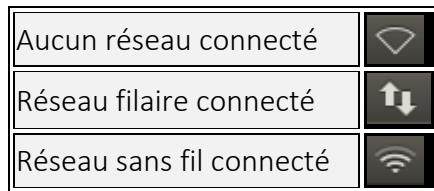
Network-Manager est installé par défaut. Dans Ubuntu et dans Kubuntu, il prend la forme d'une applet, une petite icône située dans le tableau de bord.

Il est possible que Network-Manager se soit désinstallé dans votre système ; c'est le cas, entre autres, lorsque vous installez un autre gestionnaire de connexions réseau (comme Wicd). Pour réinstaller Network-Manager, installez l'un des paquets suivants :

- Pour Ubuntu : <apt://network-manager-gnome/> ;
- Pour Kubuntu : <apt://plasma-widget-networkmanagement/> ;
- Pour Kubuntu (à partir de 14.04) : <apt://plasma-nm/>.

■ Utilisation (sous Ubuntu)

L'applet de Network-Manager se trouve automatiquement dans la zone de notification de votre tableau de bord. C'est à travers cette applet que vous pouvez gérer les connexions réseau de votre ordinateur. L'applet prend différents états en fonction de votre connexion (connecté / non connecté, réseau filaire / réseau sans fil, etc.)



Si l'applet n'apparaît pas dans votre tableau de bord ou si elle vous indique le message d'erreur *Network-Manager n'est pas lancé...*, vérifiez que le service Network-Manager est en fonction.

Sous **Gnome classique** et **Mate** si l'applet n'apparaît plus suite à une suppression par erreur :

1. Clic droit sur le tableau de bord,
2. Sélectionner "Ajouter au tableau de bord",
3. Chercher "Zone de notification" puis cliquer sur "Ajouter".

➤ Se connecter/déconnecter à un réseau détecté déjà configuré

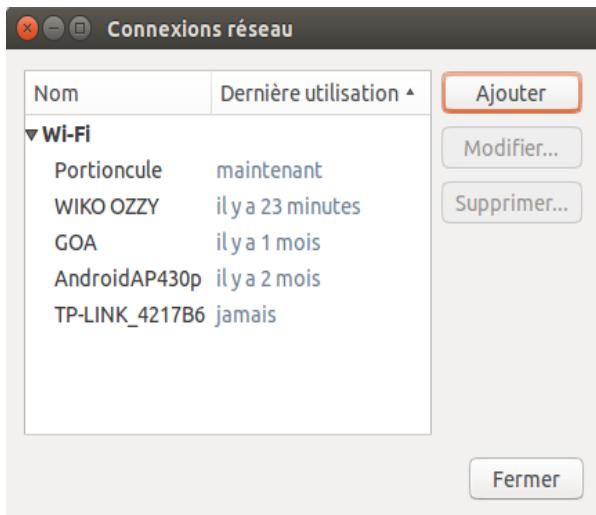


Un clic sur l'applet permet d'afficher les réseaux disponibles (notamment les réseaux sans fil détectés à porter, si votre ordinateur est équipé d'un adaptateur sans fil activé). Sélectionnez le réseau auquel vous désirez vous connecter. Ce menu vous offre aussi la possibilité :

- D'établir une connexion à un réseau sans fil invisible (qui ne diffuse pas son existence), lorsque votre ordinateur est doté d'un émetteur-récepteur de réseaux sans fil ;
- De configurer une connexion mobile à large bande (par réseau cellulaire 3G+, 3G, Edge, EV-DO), si votre ordinateur est relié à votre téléphone portable (GSM) ou à une clé mobile fournie par votre fournisseur d'accès à Internet ;
- De paramétriser des connexions réseaux privés virtuels (VPN), afin d'accéder à votre réseau d'entreprise depuis votre ordinateur à la maison ou depuis votre ordinateur portable lorsque vous êtes en déplacement.

Dans ce même menu, vous apercevez une ou des entrées **Se déconnecter**. Ces entrées permettent de couper la connexion désignée. Votre ordinateur ne sera ainsi plus relié au réseau mentionné.

➤ Configurer de nouveaux réseaux ou modifier des réseaux existants



Pour configurer des réseaux, que ce soit l'ajout de nouveaux réseaux ou la modification de réseaux existants, faites un clic sur l'applet de Network-Manager et sélectionnez l'entrée de menu **Modification des connexions....**

- Pour ajouter une nouvelle connexion :
 - Cliquez sur **Ajouter** : une fenêtre s'ouvre qui vous demande de choisir le type de connexion à créer (filaire, sans-fil, connexion mobile, VPN ou DSL...),
 - Une autre fenêtre s'ouvre, saisissez les informations relatives à votre connexion réseau, puis appuyez sur le bouton **Enregistrer** pour ajouter votre nouvelle connexion à la liste de celles disponibles ;
- Pour modifier une connexion existante, sélectionnez la connexion que vous souhaitez modifier, cliquez sur le bouton **Modifier** : une interface identique à celle de l'ajout d'une nouvelle connexion s'ouvre, mais avec certains champs d'informations pré-remplis. Modifiez les informations de votre choix, puis appuyez sur le bouton **Appliquer** pour que les changements soient pris en compte.
- Pour supprimer une connexion, sélectionnez la connexion à supprimer et cliquez sur le bouton **Supprimer**.

➤ Lier une connexion à une interface réseau précise

La suite nécessite quelques précisions. Par défaut, une connexion d'un certain type est disponible pour l'ensemble des interfaces compatibles. Par exemple, si votre ordinateur dispose de deux cartes réseau filaires Ethernet, alors l'ensemble des connexions filaires que vous avez paramétrées sont disponibles pour ces deux cartes. Vous pouvez vouloir limiter certaines connexions à une seule interface (ce qui est utile, par exemple, pour paramétriser le partage de connexion réseau). Pour limiter la disponibilité d'une connexion à une seule interface de réseau, précisez l'adresse MAC de l'interface à laquelle elle doit être liée. Pour ce faire, vous devez d'abord découvrir l'adresse MAC de vos interfaces réseau.

1. Ouvrez un terminal et exécutez la commande suivante :

```
#ifconfig
```

L'ensemble de vos interfaces réseau connues et actives sont alors listées ;

2. Localisez l'interface réseau à laquelle vous voulez lier une connexion. Notez le champ **HWaddr** pour cette connexion : c'est son adresse MAC ;
3. Rendez-vous ensuite dans l'interface d'ajout ou de modification d'une connexion. Inscrivez l'adresse MAC de l'interface correspondante dans le champ *Adresse MAC* ;
4. Appuyez ensuite sur le bouton **Appliquer** pour prendre en compte la modification.

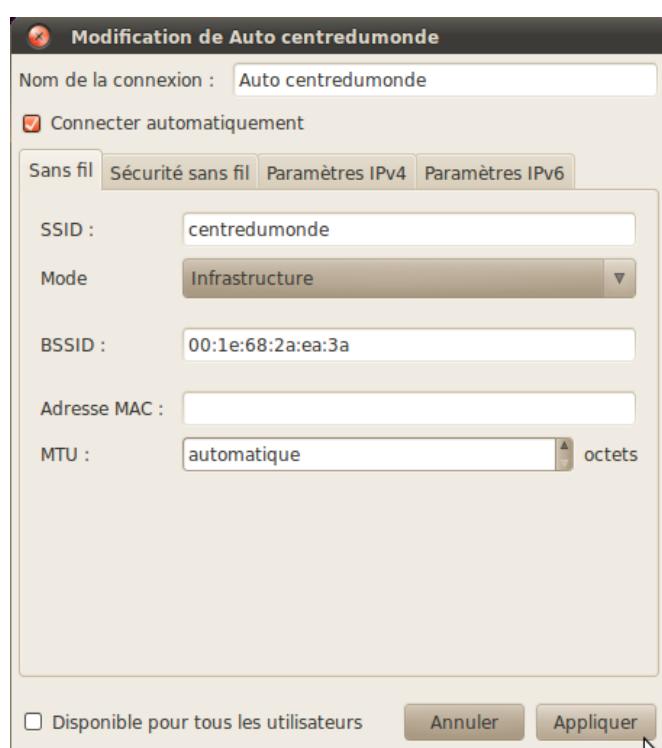
alexandre@Alex-DisqueMobile: ~

Fichier Édition Affichage Terminal Aide

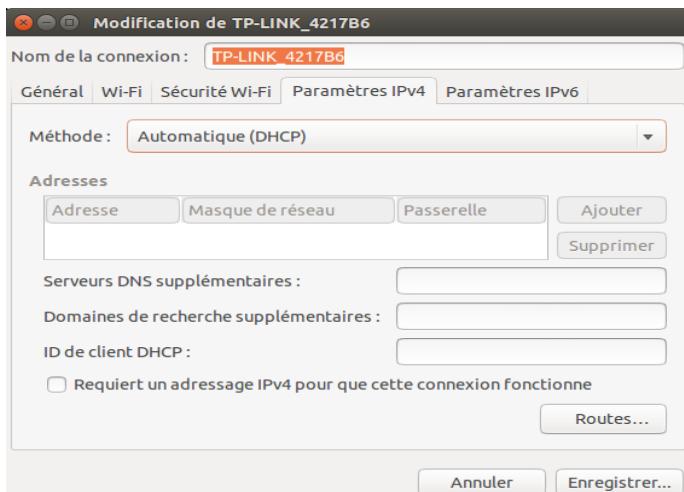
```
alexandre@Alex-DisqueMobile:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 00:1e:68:2a:ea:3a
          inet adr:10.42.43.1 Bcast:10.42.43.255 Masque:255.255.255.0
          adr inet6: fe80::21e:68ff:fea3a/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          Packets reçus:1869 erreurs:0 :0 overrun:0 frame:0
          TX packets:2200 errors:0 dropped:0 overrun:0 carrier:0
          collisions:0 lg file transmission:1000
          Octets reçus:287010 (287.0 KB) Octets transmis:2268756 (2.2 MB)
          Interruption:28 Adresse de base:0x8000

lo        Link encap:Boucle locale
          inet adr:127.0.0.1 Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          Packets reçus:99393 erreurs:0 :0 overrun:0 frame:0
          TX packets:99393 errors:0 dropped:0 overrun:0 carrier:0
          collisions:0 lg file transmission:0
          Octets reçus:31484523 (31.4 MB) Octets transmis:31484523 (31.4 MB)

wlan0     Link encap:Ethernet HWaddr 00:1f:3b:2c:ff:09
          inet adr:192.168.0.103 Bcast:192.168.0.255 Masque:255.255.255.0
```



➤ Gérer les adresses IP



Pour vous connecter à votre réseau local ou à internet, votre ordinateur doit être doté d'une adresse IP. L'adresse IP est un identifiant numérique unique qui identifie votre ordinateur parmi tous les autres ordinateurs de votre réseau ou d'internet. L'attribution de cette adresse IP unique peut se produire de plusieurs façons ; au moment d'établir la connexion au réseau désiré, un serveur envoie à votre ordinateur son adresse IP unique. Par défaut c'est la configuration automatique qui est activée. Si vous avez changé ces paramètres et que vous souhaitez revenir à une adresse ip automatique :

- Rendez-vous dans l'onglet *Paramètres IPv4* de l'interface de gestion de votre connexion ;
- Dans le champ *Nom de la connexion*, entrez un nom unique significatif pour votre connexion ;
- Dans le menu déroulant *Méthode*, choisissez la méthode **Automatique (DHCP)** ;
- Appuyez sur le bouton **Appliquer...** pour valider la modification.

Vous pouvez aussi paramétrier manuellement l'adresse IP attribuée à votre ordinateur. Ceci est particulièrement intéressant pour les serveurs, qui doivent sans cesse rester joignables à la même adresse pour tous les clients. Une configuration manuelle des adresses IP peut aussi accroître la sécurité de votre réseau local. Cependant, ce paramétrage est plus complexe, difficile à maintenir et long à appliquer à une grande infrastructure. Vous devriez laisser cette tâche à un administrateur réseau.



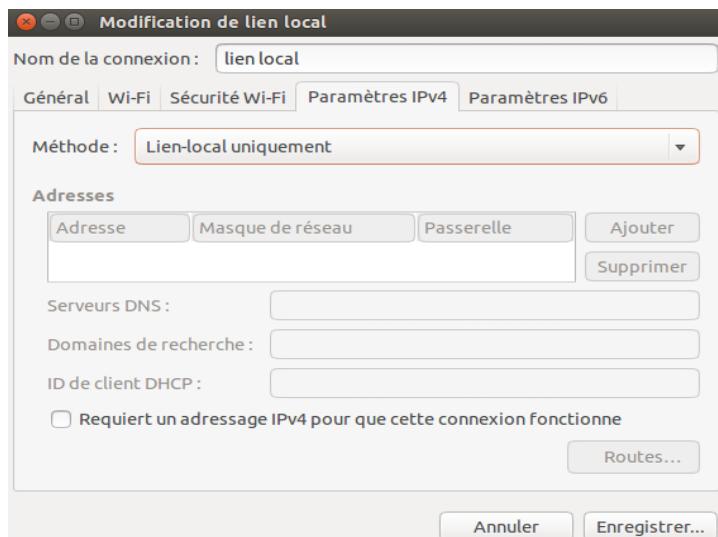
Pour paramétrier une connexion devant avoir une adresse IP fixe (au moment de créer une nouvelle connexion ou en modifiant une connexion existante) :

- Rendez-vous dans l'onglet *Paramètres IPv4* de l'interface de gestion de votre connexion ;
- Dans le champ *Nom de la connexion*, entrez un nom unique significatif pour votre connexion ;
- Dans le menu déroulant *Méthode*, choisissez la méthode **Manuel** ;
- À la droite du cadre *Adresses*, appuyez sur le bouton **Ajouter** ;
- Dans le cadre *Adresse*, inscrivez l'adresse IP : 192.168.1.23, le masque de sous-réseau : 255.255.255.0 et (accessoirement) la passerelle : 192.168.1.1 par défaut que doit utiliser votre connexion ;

- Dans le champ *Serveurs DNS*, inscrivez-la ou les adresses des serveurs DNS que doit utiliser votre connexion, par exemple 192.168.1.1. Séparez les adresses multiples par une virgule ;
1. Dans le champ *Domaine de recherche*, inscrivez le domaine dans lequel votre connexion doit rechercher automatiquement des adresses, si tel est le cas ;
 2. Appuyez sur le bouton **Appliquer...** pour appliquer les changements.
 3. Cochez la case *Requiert un adressage ipv4 pour que cette connexion fonctionne*.

Avec une AliceBox, il faut en plus rajouter dans l'onglet 'Paramètres Ipv4' l'IP du DNS principal soit 213.36.80.1. La machine apparaît bien dans l'interface Web du routeur menu 'ARP' mais pas dans le menu 'DHCP'.

➤ Modifier les paramètres IP pour la liaison de deux ordinateurs seulement



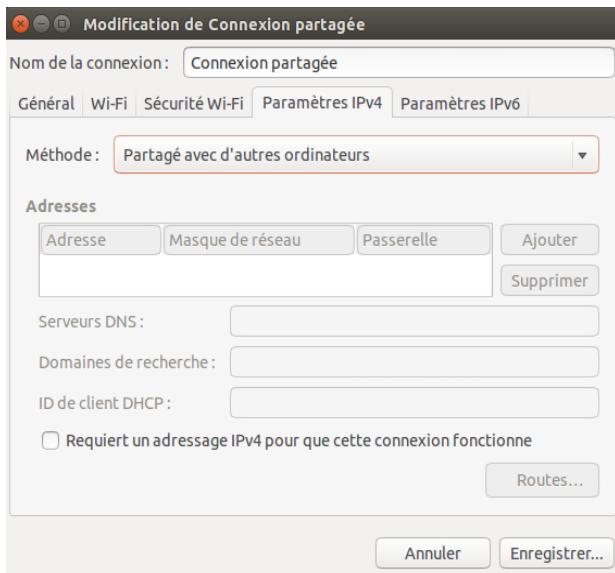
Ne pas oublier que la connexion directe de deux périphériques Ethernet (par exemple la connexion de deux ordinateurs) nécessite le recours à un câble croisé. Si l'on n'en dispose pas, le recours à un commutateur (Ethernet Switch) ou à un routeur (Router) est nécessaire, car ces appareils gèrent automatiquement la reconnaissance des câbles droits ou croisés et s'y adaptent. Dans ce cas, il sera probablement possible de faire bénéficier plusieurs périphériques du partage de la connexion.

Vous pouvez paramétriser votre connexion pour qu'elle utilise une adresse réservée à une connexion locale temporaire uniquement. Ceci est particulièrement utile lorsque vous reliez temporairement deux ordinateurs l'un à l'autre ou lorsque vous reliez votre ordinateur directement à un disque de stockage réseau (NAS). Des adresses temporaires sont automatiquement assignées à vos ordinateurs. Ce type de connexion est appelée "lien local".

Pour paramétriser une connexion de type lien local (au moment de créer une nouvelle connexion ou en modifiant une connexion existante) :

1. Rendez-vous dans l'onglet *Paramètres IPv4* de l'interface de gestion de votre connexion (Fenêtre Connexions réseau, sélectionner la connexion et cliquer sur modifier) ;
2. Dans le champ *Nom de la connexion*, entrez un nom unique significatif pour votre connexion ;
3. Dans le menu déroulant *Méthode*, choisissez la méthode **Lien-local** ;
4. Appuyez sur le bouton **Appliquer...** pour appliquer la modification.

➤ Modifier les paramètres IP pour le partage d'une connexion Internet



Vous pouvez paramétriser votre connexion pour qu'elle serve de pont entre plusieurs réseaux. Le plus fréquemment, cette configuration se rencontre lorsque l'ordinateur sous Ubuntu est relié directement au modem du fournisseur d'accès à Internet, et que l'on désire partager cet accès à Internet avec un autre ordinateur.

Ce genre de configuration nécessite que votre ordinateur dispose d'au moins deux cartes réseau (par exemple, une connexion filaire vers votre modem et une connexion sans fil vers votre réseau local). La connexion vers un réseau extérieur doit être de type automatique (DHCP), manuel (IP statique) ou lien-local. La connexion vers votre réseau local doit être de type partagé avec d'autres ordinateurs ; c'est celle-ci que nous configurons ici.

Pour paramétriser une connexion partagée avec d'autres ordinateurs de votre réseau local (au moment de créer une nouvelle connexion ou en modifiant une connexion existante) :

1. Dans le champ *Nom de la connexion*, entrez un nom unique significatif pour votre connexion ;
2. Dans l'onglet *Wi-Fi* dans le champ *Mode* choisissez **Ad hoc** ;
3. Rendez-vous dans l'onglet *Paramètres IPv4* de l'interface de gestion de votre connexion (Fenêtre Connexions réseau, sélectionner la connexion et cliquer sur modifier) ;
4. Dans le menu déroulant *Méthode*, choisissez la méthode **partagée avec d'autres ordinateurs** ;
5. Allez ensuite dans l'onglet *Paramètres IPv6* et mettez la *Méthode* en *Ignorer*.
6. Appuyez sur **Appliquer...** pour activer le partage.

Il se peut que la connexion se connecte puis se déconnecte puis se reconnecte, en boucle. Si cela vous arrive, il suffit :

1. D'aller dans les paramètres de la seconde connexion (celle qui est reliée à internet) et faites comme pour la connexion partagée, ignorez IPv6. (*Paramètres IPv6* → *Méthode* → *Ignorer*)
2. D'appuyer sur le bouton **Appliquer...** pour valider la modification.

Vous aurez peut-être besoin de redémarrer si la connexion fait encore des siennes.

Du côté des autres ordinateurs qui accèdent à votre connexion partagée, vous devez vous assurer que leur mode de connexion est paramétrée au type automatique (DHCP). Dans ce type de configuration, votre ordinateur sous Ubuntu se charge d'attribuer des adresses IP uniques à vos autres ordinateurs de votre réseau local.

Il se peut qu'après avoir tout bien paramétré, Internet ne soit toujours pas accessible sur les ordinateurs avec lesquels vous souhaitez partager votre connexion. Ceci est probablement dû à un problème de configuration DNS.

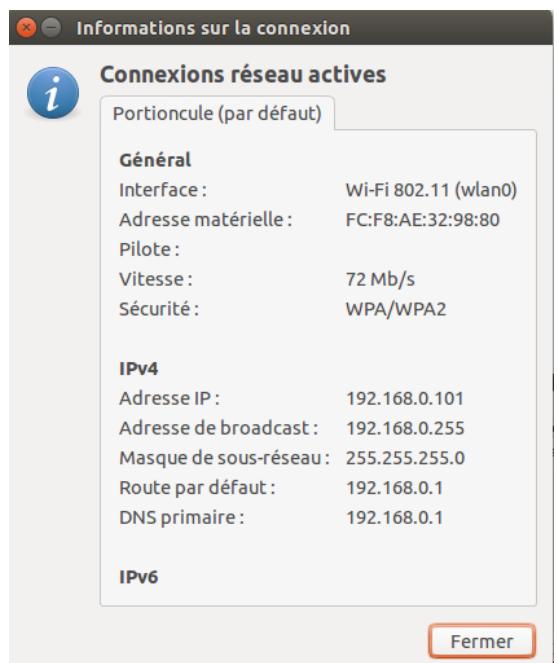
Pour corriger ce problème, il suffit d'indiquer les paramètres correspondants sur les clients concernés.

- Allez dans l'onglet **Paramètres IPv4**,
 - Dans **Méthode** choisissez **Adresse automatique uniquement (DHCP)**,
 - Dans la case **Serveur DNS**, indiquez le serveur DNS de votre choix (par exemple celui de votre Fournisseur d'accès Internet ou celui de Google - 8.8.8.8),
 - Dans la case **Domaines de recherche**, indiquez par exemple celui de Google - 8.8.4.4.

■ Utilisation avancée (Network-Manager)

➤ Voir l'état des réseaux actuellement connectés

Lorsque vous êtes connecté à un ou plusieurs réseaux, Network-Manager peut vous en indiquer les caractéristiques. Pour ce faire, faites un clic sur l'applet de Network-Manager et sélectionnez l'entrée de menu **Informations sur la connexion** (*voir l'image ci-contre*).



➤ Modifier les fichiers de configurations d'une connexion

Essayer d'abord la méthode décrite ci-dessus, avant de modifier manuellement ces fichiers.

Le fichier de configuration d'une connexion se trouve dans le dossier **/etc/NetworkManager/system-connections**. Elles sont utiles, entre autres, pour récupérer les paramètres d'une ancienne Ubuntu sur une autre partition.

Après importation, si les connexions automatiques ne fonctionnent pas, indiquez le nom de la carte réseau à utiliser dans *Connexions réseaux → Modification des connexions... → nom_de_la_connexion_concernée → Adresse MAC du périphérique*.

Dans certains cas, il peut être intéressant de modifier directement les configurations des cartes réseau, sans passer par l'interface graphique (problèmes graphiques, applet ne fonctionnant pas etc.)

Le fichier porte le nom de la connexion qui est affichée dans le Network-manager. Il est possible de modifier directement le fichier de configuration. Pour cela, un simple éditeur de texte suffira avec les droits d'administration. Pour commenter une ligne, il faut mettre le symbole # en début de ligne :

```
#sudo vim Connexion_filaire_1
```

Exemple de fichier :

```
[802-3-ethernet]
mac-address=74:D4:35:84:C0:AE
[connection]
id=Connexion filaire 1
uuid=622e166d-56e9-4a4a-9ed4-6cd5456482f0
type=802-3-ethernet

[ipv6]
method=auto

[ipv4]
method=auto
#dns=212.27.40.241;212.27.40.240; #serveur dns de Free
ignore-auto-dns=false
```

Après les modifications, il est nécessaire de relancer la carte réseau. Pour cela il faut lancer ces 2 commandes :

```
#sudo ifconfig [ nom_de_la_carte] down #désactive la carte réseau
#sudo ifconfig [ nom_de_la_carte] up #active la carte réseau
```

Pour obtenir le [nom de la carte], un 'ifconfig' sans paramètre vous donnera les informations : la mac-address du fichier correspond à HWaddr de la commande ifconfig:

```
#ifconfig
```

➤ **Network-Manager en ligne de commande**

Pour démarrer le service :

```
#sudo service network-manager start
```

Pour arrêter le service :

```
#sudo service network-manager stop
```

Network-Manager peut aussi se contrôler depuis un terminal (si l'on a plus accès à X par exemple) avec la commande :

```
#nmcli
```

Voici quelques commandes utiles :

- **nm-tool** affiche les informations sur vos connexions actives plus clairement que la commande **ifconfig**
- **nmcli con show** liste de toutes les connections configurées via NetworkManager.
- **nmcli con show uuid "UUID de la connexion"** ou **nmcli con show id "ID de la connexion"** permettent d'obtenir tous les détails liés à une connexion en particulier.
- **nmcli con down id "nom de la connexion"** désactive la connexion.
- **nmcli con up id "nom de la connexion"** active la connexion.
- **nmcli con delete id "nom de la connexion"** supprime la configuration et la connexion.

➤ Lancer automatiquement une connexion ou non

Clic sur l'applet de Network-Manager puis "Modification des connexions", choisir le réseau à modifier, clic sur *Modifier...* Dans l'onglet *Général* cochez ou décochez la case *Se connecter automatiquement à ce réseau si disponible*.

➤ Désactiver Network-Manager pour laisser agir ifup

Debian et Ubuntu diffèrent sur la manière de gérer leur réseau :

- Debian utilise **ifup/ifdown**
- Ubuntu utilise **NetworkManager**

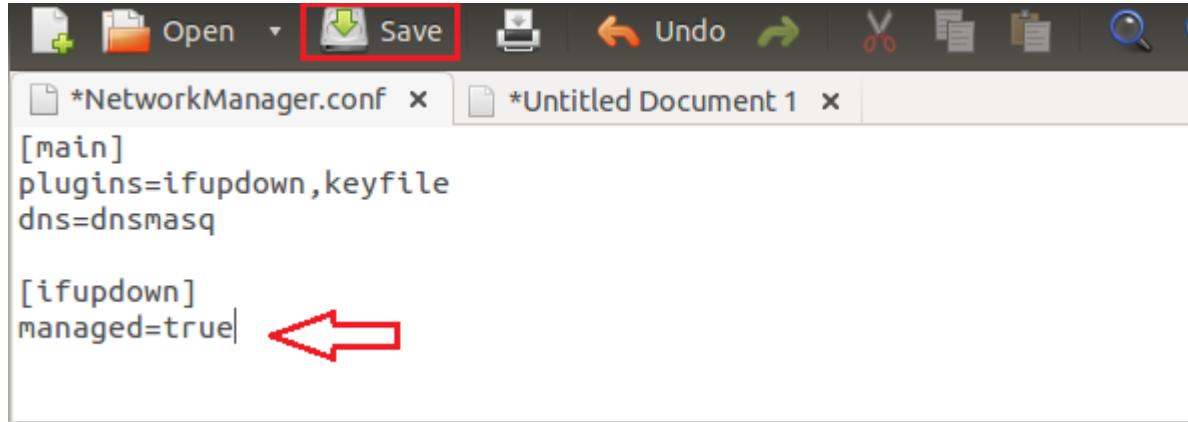
Avec Debian, le système gère sa configuration avec l'utilitaire ifup et les données sont présentes dans le fichier de configuration situé à **/etc/network/interfaces**, qui permet des configurations extrêmement complexes de mode serveur ou très avancées, dont nous n'avons généralement pas besoin sur un poste client, ou sur un ordinateur personnel.

On peut avoir à gérer une carte réseau à l'aide du fichier situé dans **/etc/network/interfaces**, cependant, dans les dernières versions d'Ubuntu, il est nécessaire de passer par le démon système NetworkManager afin de gérer la configuration réseau. C'est la manière standard de procéder désormais.

Voici donc comment désactiver le service système NetworkManager qui gère le réseau pour une carte ou pour une interface réseau, et pouvoir pour celle-ci uniquement revenir à l'utilisation du fichier **/etc/network/interfaces** sous Ubuntu. D'abord, éditer la configuration de network-manager :

```
#sudo nano /etc/NetworkManager/NetworkManager.conf
```

Et procéder aux modifications comme suit :



Wi-Fi

Le **WIFI** est la technologie sans fil utilisée pour se connecter à un réseau informatique (internet - intranet - local).

L'avantage de ce type de connexion est de s'affranchir du câble souvent contraignant. Le WIFI permet donc un meilleur confort de l'utilisation du réseau.

La prise en charge du matériel WIFI par le système d'exploitation Ubuntu est relativement bonne et elle s'améliore à chaque nouvelle version semestrielle.

En règle générale, vous n'aurez rien de spécial à faire, pour vous connecter à un réseau WIFI, votre ordinateur devrait vous signaler automatiquement les réseaux disponibles au démarrage de la machine. Si c'est le cas, vous pouvez passer directement au paragraphe qui traite de la connexion au WIFI. Si par contre vous rencontrez des problèmes pour vous connecter, passez au paragraphe qui suit.

■ Identifier sa carte réseau WIFI

Si votre ordinateur n'affiche aucun réseau WIFI alors que ces derniers sont disponibles, il est possible que le pilote de votre carte ne soit pas installé.

Pour savoir quel pilote est nécessaire, il faut d'abord connaître son matériel :

*Normalement la ligne recherchée contient le mot **wireless** ou **wlan** et souvent le **chipset** entre crochets.*

- Si votre carte WIFI est reliée par USB à l'ordinateur, dans un terminal, saisissez la commande suivante :

```
#lsusb
```

Regardez le nom exact de votre carte WIFI et notez aussi le chipset qu'elle utilise.

Par exemple pour un adaptateur USB de marque *D-Link* avec un chipset *Realtek* modèle *RTL8192SU* :

```
Bus 003 Device 076: ID 07d1:3303 D-Link System DWA-131 802.11n Wireless N Nano Adapter(rev.A1) [Realtek  
RTL8192SU]
```

- Sinon, dans un terminal, saisissez la commande suivante :

```
#lspci | grep -i net
```

Regardez le nom exact de votre carte WIFI, et notez aussi le chipset qu'elle utilise.

Par exemple pour une carte PCI de marque *Qualcomm* avec un chipset *Atheros* modèle *AR9565* :

```
02:00.0 Network controller: Qualcomm Atheros QCA9565 / AR9565 Wireless Network Adapter (rev 01)
```

- Si les commandes **lssusb** et **lspci** ne donnent aucune information sur le chipset (cas par exemple de la clé USB Wi-Fi de marque LG distribué avec certains enregistreur numérique TNT), essayez la commande suivante :

```
#sudo lshw
```

Vous obtiendrez plus d'informations, notamment le nom du chipset.

N.B. Consultez https://doc.ubuntu-fr.org/wifi_liste_carte pour savoir ce qui est recommandé au sujet de votre matériel.

En règle générale, 3 possibilités existent pour faire fonctionner les cartes WIFI.

Soit :

- Il vous manque un **micro-logiciel** propriétaire que vous pouvez installer par le gestionnaire de pilotes propriétaires,
- Votre version d'Ubuntu est actuellement trop ancienne et son noyau ne prend pas encore en charge votre matériel, dans ce cas migrer vers une version plus récente ou bien installez un noyau plus récent,
- Si vous avez mis à jour ou conservé votre système Windows 10 d'origine, activer la couche logicielle « **Windows Linux Subsystem** » dans Windows même pour rendre le pilote Wifi disponible sous Ubuntu.
Pour de plus amples informations ainsi que des précisions sur son activation, veuillez consulter la page Korben : Installer le Shell Bash sous Windows 10
- La dernière possibilité consiste à utiliser les pilotes existants pour Windows grâce à l'application **Ndiswrapper**.

■ Utilisation graphique du WIFI

Il existe plusieurs possibilités pour qu'Ubuntu puisse se connecter au WIFI. La plus simple est l'utilisation de l'application par défaut, Network-Manager.

Vous pouvez y accéder en cliquant sur le logo en forme d'antenne WIFI en haut à droite de votre écran — sous Gnome ou Unity — et en bas à droite sous Cinnamon ou Kde (Plasma).

En dehors de Network-Manager il existe encore d'autres utilitaires de configuration, en voici quelques-uns :

- Network-admin, (<https://doc.ubuntu-fr.org/network-admin>)
- WiFi-radar : possible après l'ajout du paquet correspondant, (<https://doc.ubuntu-fr.org/wifi-radar>)
- Wicd : une alternative à NetworkManager. (<https://doc.ubuntu-fr.org/wicd>)

■ Utilisation en ligne de commande du WIFI

Il existe des outils graphiques qui permettent de se faciliter grandement la tâche pour se connecter à un réseau sans fil. Mais il est tout à fait possible de le configurer à la main, directement dans les fichiers de configuration réseaux, ou avec l'outil wpa-supplicant.

Au sujet de la sécurité des réseaux sachez qu'il existe :

- Le WEP est un système dépassé en termes de sécurité. Il n'est pas conseillé d'utilisation.
- Le WPA (Wi-Fi Protected Access) est un mécanisme pour sécuriser les réseaux sans-fil Wi-Fi. Il est utilisé en remplacement de l'ancien système WEP, il existe actuellement le WPA et le WPA2 :
 - Le WPA ne se trouve plus que sur les anciens systèmes et n'est pas considéré comme complètement sécurisé même s'il offre déjà une très bonne protection.
 - Le WPA2 est présent sur tous les nouveaux matériels certifiés Wi-Fi. C'est le seul à être considéré comme complètement sécurisé et est donc privilégié d'utilisation.

Il est possible de configurer son réseau Wi-Fi en modifiant directement le fichier interfaces ou en utilisant Par wpa-supplicant.

■ Modification du fichier interfaces

On peut modifier avec les droits d'administration le fichier **/etc/network/interfaces** : Exemple du contenu du fichier interfaces :

```
auto lo
iface lo inet loopback

iface eth1 inet static
    wireless-keymode open
    wireless-mode managed
    wireless-nick Laptop
    address 192.168.1.4
    netmask 255.255.255.0
    gateway 192.168.1.1

    wireless-essid 9T_B800A
    wireless-key XXXXXXXXXXXXXXXXXXXXXXXX
    wireless-channel 11

auto eth1
wireless-essid 9T_B800A
wireless-key XXXXXXXXXXXXXXXXXXXXXXXX
wireless-channel 11

auto eth1
```

ou :

```
auto wlan0
iface wlan0 inet dhcp
    wpa-ssid nom_de_votre_connexion_wifi
    wpa-psk mot_de_passe_de_votre_connexion
```

➤ Les paramètres possibles

- `wpa-ssid <nom de votre connexion>` : Définit le nom du réseau Wi-Fi auquel se connecter, exemple :

```
#wpa-ssid Livebox-3818
```

- `wpa-bssid <adresse MAC de votre point d'accès>` : Définit l'adresse Ethernet de votre point d'accès auquel se connecter, exemple :

```
#wpa-bssid 01:2a:3b:4c:5d:6e
```

- `wpa-psk <mot de passe de votre connexion>` : Votre mot de passe WPA (peut être votre mot de passe directement en ASCII ou en Hexadécimal), exemple 1 :

```
#wpa-psk totomaison
```

Exemple 2 :

```
#wpa-psk 2f38e12c698a
```

- `wpa-key-mgmt <protocole>` : Définit les protocoles de connexion autorisés. Les valeurs possibles sont NONE, WPA-PSK, WPA-EAP et IEEE8021X, exemple WPA/WPA2 classique autorisé :

```
#wpa-key-mgmt WPA-PSK
```

- `wpa-group <chiffrage>` : Définit les modes de chiffrage autorisés. Les valeurs possibles sont CCMP, TKIP, WEP104 et WEP40. CCMP correspond à AES, exemple AES autorisé :

```
#wpa-group CCMP
```

Exemple AES et TKIP autorisés :

```
#wpa-group CCMP TKIP
```

- `wpa-pairwise <chiffrage par paires>` : Définit les modes de chiffrage par paires autorisés pour le WPA/WPA2. Les valeurs possibles sont CCMP, TKIP et NONE, exemple AES et TKIP autorisés :

```
#wpa-pairwise CCMP TKIP
```

- **wpa-auth-alg <système d'authentification>** : Définit les systèmes d'authentification autorisés. Les valeurs possibles sont OPEN (pour le WPA/WPA2), SHARED (pour le WEP) et LEAP, exemple WPA/WPA2 :
- ```
#wpa-auth-alg OPEN
```
- **wpa-proto <norme WPA autorisée>** : Définit les normes WPA autorisées. Les valeurs possibles sont WPA et RSN. RSN correspond à WPA2, exemple WPA2 seulement :
- ```
#wpa-proto RSN
```

Exemple WPA et WPA2 :

- ```
#wpa-proto WPA RSN
```
- **wpa-scan-ssid <0 ou 1>** : Autorise ou non le scan des requêtes (normalement autorisé), exemple :
- ```
#wpa-scan-ssid 1
```
- **wpa-ap-scan <0 ou 1 ou 2>** : Définit le mode de sélection du point d'accès. 0 : IEEE8021X, 1 : classique, 2 : sécurisé (pour le mode ad-hoc notamment), exemple point d'accès classique :
- ```
#wpa-ap-scan 1
```

### Exemple ad-hoc (PC à PC) :

```
#wpa-ap-scan 2
```

**wpa-identity** et **wpa-password** sont pour les utilisateurs avancés réalisant une connexion d'entreprise EAP :

- **wpa-identity <nom d'utilisateur>** : Définit le nom d'utilisateur de la connexion EAP
- **wpa-password <mot de passe>** : Définit le mot de passe utilisateur de la connexion EAP

## ➤ Quelques exemples

Connexion à un point d'accès en WPA2 AES :

```
auto wlan0
iface wlan0 inet dhcp
 wpa-ssid Livebox-3818
 wpa-ap-scan 1
 wpa-scan-ssid 1
 wpa-key-mgmt WPA-PSK
 wpa-group CCMP
 wpa-proto RSN
 wpa-psk totomaison
```

Connexion à un point d'accès en WPA/WPA2 TKIP/AES :

```
auto wlan0
iface wlan0 inet dhcp
wpa-ssid Livebox-3818
wpa-ap-scan 1
wpa-scan-ssid 1
wpa-key-mgmt WPA-PSK
wpa-group CCMP TKIP
wpa-proto RSN WPA
wpa-psk totomaison
```

Connexion Ad-Hoc (PC à PC) WPA TKIP :

```
auto wlan0
iface wlan0 inet static
address 192.168.0.12
netmask 255.255.255.0
gateway 192.168.0.1

wpa-ssid ReseauJP
wpa-ap-scan 2
wpa-scan-ssid 1
wpa-key-mgmt WPA-NONE
wpa-group TKIP
wpa-pairwise NONE
wpa-proto WPA
wpa-psk totomaison
```

Pour relancer le réseau sans fil, effectuez la commande suivante :

```
#sudo /etc/init.d/networking restart
```

Pour la syntaxe du fichier utiliser l'aide disponible

- Dans un terminal **man interfaces**,
- Dans les configurations des différentes cartes.

## ➤ Configuration en IP automatique (DHCP)

- Pour obtenir une adresse IP, si elle est automatique il suffit d'une commande après s'être connecté au réseau:

```
#sudo dhclient wlan0
```

- S'il y a une ligne comme DHCPOFFER 192.168.1.100 par exemple, cela veut dire que l'adresse IP a bien été donnée et donc la carte Wi-Fi et le routeur peuvent communiquer : vous pouvez ainsi surfer sur internet.
- Depuis la version feisty du système Ubuntu et l'introduction du démon d'avahi-daemon, la commande dhclient peut marcher mais avec certains ordinateurs il faut exécuter cette commande pour rétablir la connexion au net :

```
#sudo /etc/init.d/avahi-daemon stop
```

- Il faut ensuite refaire la commande **dhclient**.

Pour vérifier qu'une adresse IP a bien été donnée, taper `ifconfig wlan0` et vérifier qu'il y a une adresse IP à côté de l'interface wlan0.

En revanche s'il n'y a que des échecs (répétition des lignes DHCPDISCOVER on 255.255.255.255 ...), cela est peut-être dû à un problème au niveau de la connexion au réseau (première commande) ou alors le routeur n'est pas en IP automatique (DHCP). Dans ce cas il faut configurer le DHCP sur le routeur ou une configuration en IP statique sur le client (ordinateur).

## ➤ Configuration en IP statique

En IP static la configuration se fait avec plusieurs commandes. Il est recommandé tout d'abord :

- De changer de service de gestion réseau en "arrêtant" Network-manager pour ré-activer networking:

```
#sudo service network-manager stop && sudo service networking start
#sudo update-rc.d network-manager remove
#sudo update-rc.d networking defaults\
```

- Puis avec une première commande de donner l'adresse IP "**IPdeLaCarteWifi**" à la carte Wi-Fi et avec une seconde de régler la passerelle par défaut (le routeur) :

```
#sudo ifconfig wlan0 IPdeLaCarteWiFi (192.168.0.1 par exemple)
#sudo route add default gw IPDuRouteur (192.168.0.254 par exemple)
```

- Il faut ensuite rajouter les adresses des serveurs DNS (qui traduisent google.fr par exemple en adresse IP exploitable par l'ordinateur). Pour cela il faut modifier avec les droits d'administration le fichier **/etc/resolv.conf** et y ajouter la ligne suivante :

```
nameserver IPduServeurDNS
```

Un exemple ici pour les personnes ayant pour FAI Orange :

```
nameserver 80.10.246.2
```

## ➤ Se connecter à un réseau en mode console

Voici la commande pour se connecter au réseau en supposant que la carte Wi-Fi est wlan0 :

```
#sudo iwconfig wlan0 essid MonSSID
```

Faites attention aux majuscules dans le SSID il faut respecter exactement le nom du SSID du routeur. Par exemple si votre SSID est Livebox-56G alors la commande sera **sudo iwconfig wlan0 essid Livebox-56G** et non ... essid livebox-56g

- Pour vérifier que la carte Wi-Fi s'est connectée au réseau, taper **iwconfig** et il devrait y avoir marqué dans SSID :"MonSSID", Dans accès point il doit y avoir : Access Point : 00:C0:49:53:7C:7C ( l'adresse 00... ici n'est qu'un exemple c'est l'adresse MAC du routeur ).
- Si vous avez une clé WEP, il faut taper une commande supplémentaire :

```
#sudo iwconfig wlan0 key LaCLEWEP
```

- Si vous avez une sécurité WPA il faut suivre la méthode Par wpa supplicant,
- Si vous n'avez aucune protection, il n'y a pas besoin de cette ligne il faut passer à l'obtention ou la configuration d'une adresse IP.

## ■ WPA supplicant

WPA Supplicant est un service utilisé pour réaliser votre connexion WPA.

### ➤ Pre-requis

- WPA Supplicant gère une majeure partie des systèmes Wi-Fi, néanmoins vous pouvez vérifier si votre carte est compatible sur leur site [https://hostap.epitest.fi/wpa\\_supplicant/](https://hostap.epitest.fi/wpa_supplicant/)
- WPA Supplicant est normalement installé par défaut sur votre système, si ce n'est pas le cas installez le paquet [apt://wpa\\_supplicant](apt://wpa_supplicant)

La méthode qui suit est d'usage général pour la configuration d'une ou plusieurs connexions WPA.

### ➤ Configuration

Modifiez avec les droits d'administration le fichier **/etc/network/interfaces**. Chacune de vos interfaces réseau et leurs configurations apparaissent dans ce fichier :

- À la suite de la configuration réseau de votre interface Wi-Fi (en général nommée "wlan0"), indiquer le fichier de configuration de WPA Supplicant comme suit :

```
auto wlan0
iface wlan0 inet dhcp
 wpa-conf /etc/wpa_supplicant.conf
```

- Éditez ou créez avec les droits d'administration le fichier **/etc/wpa\_supplicant.conf** pour qu'il ressemble à ceci en pensant à adapter les paramètres utiles, notamment le nom de votre connexion et votre mot de passe :

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
network={
 ssid=<nom de votre connexion>
 scan_ssid=1 # nécessaire quand le ssid n'est pas diffusé
 proto=WPA RSN
 key_mgmt=WPA-PSK
 psk=<mot de passe de votre connexion>
}
```

Vous pouvez ajouter autant de configuration de points d'accès différents en ajoutant des paramètres `network={}` à votre fichier.

D'autres paramètres plus avancés peuvent être utilisés voir ce qui suit.

## Paramètres

Voici comment remplir votre fichier '**wpa\_supplicant.conf**'.

```
#On indique tout d'abord où se situe le programme (socket avec wpa_cli...)
ctrl_interface=/var/run/wpa_supplicant

#On protège l'accès à wpa_supplicant en l'attribuant à un groupe
#Le groupe peut être un identifiant (gid) ou un nom de groupe
#Par défaut le gid est à 0 pour signifier que c'est le root qui protège l'accès.
#ctrl_interface_group=wheel #Un exemple de nom de groupe
ctrl_interface_group=0

#On définit la version d'EAPOL (un moyen de transport de données EAP) >> Version : IEEE
#802.1X/EAPOL
#Pour info, wpa_supplicant gère la version 2 (IEEE 802-1X-REV-d8) mais il se peut qu'il y ait
#des soucis avec certaines cartes.
#On recommande donc l'utilisation de la version 1
eapol_version=1

#On définit le mode de sélection du point d'accès (AP : Access Point)
#1 : Configuration classique
#0 : Utilisé pour le protocole IEEE 802.1X (et les réseaux établis).
#2 : Associe les points d'accès dans un mode sécurisé. !!!Utilisé pour le mode Ad-Hoc!!!
Associe le SSID (nom de votre réseau) mais pas le BSSID (n° attribué à votre carte réseau).
A utiliser si vous avez une clef installée sous NDIS(WRAPPER) pour utiliser pleinement
#votre réseau :
dans ce cas le block network devra comprendre les informations 'key_mgmt', 'pairwise',
#'group' et les variables de protocole.
ap_scan=1

#On configure la ré-authentification rapide pour protocoles EAP
#Par défaut, déjà initialisé puisque sa désactivation n'a pas lieu d'être apparemment.
fast_reauth=1

#On peut configurer un support de moteur OpenSSL
#Non-abordé

#On peut indiquer le pilote utilisé mais ce n'est pas forcément recommandé
#driver_param="nom=valeur"

#On configure son réseau :

#Chaque réseau différent utilise un bloc 'network' propre
#Contenu des blocs :
```

```

#
#disabled:
=0 : réseau autorisé (par défaut)
=1 : réseau interdit
#
#
#ssid:
"nom de votre réseau" : entre guillemets si c'est une chaîne de caractères
1f25e68 : directement si c'est un nom de réseau en hexadecimal
#
#
#scan_ssid:
=0 : ne scanne pas les requêtes des réseaux (par défaut)
=1 : scanne les requêtes (nécessaire quand le ssid n'est pas diffusé)
#
#
#bssid: 3f:58:56:fe:c9:68 : N° attribué à votre carte réseau (optionnel et non-recommandé
surtout si votre carte change de BSSID)
#
#
#priority: 0 : Indique la priorité du réseau.
#
#
#mode:
=0 : Infrastructure ('Managed') (par défaut)
=1 : IBSS (ad-hoc, peer-to-peer)
#Le mode ad-hoc est utilisé pour un réseau de dongle :
- il n'accepte que les modes key_mgmt=NONE (avec possibilité de clef WEP) ou WPA-NONE (avec
groupe TKIP/CCMP)
-plus important, ap_scan=2 obligatoirement.
-enfin le bloc devra comporter : proto=WPA, key_mgmt=WPA-NONE, pairwise=None, group=TKIP (ou
CCMP(AES), mais pas les 2), et la clé psk devra être donnée.
#
#
#proto: protocole
=WPA : WPA(1) (WPA/IEEE 802.11i/D3.0)
=RSN : WPA2 (IEEE 802.11i) (par défaut)
#
#
#key_mgmt: encryption
=WPA-PSK : WPA 'pre-shared key' (requiert une clef PSK)
=WPA-EAP : WPA utilisant une authentification EAP (peut utiliser un programme externe
(Xsupplicant)).
=IEEE8021X : IEEE 802.1X utilisant une authentification EAP et, optionnellement la
génération de clefs WEP dynamiques.
=NONE : Pas de WPA : clef WEP ou connexion directe.
=WPA-NONE : Pour réseaux Ad-Hoc (possibilité de groupage TKIP ou CCMP(AES))
#Par défaut : WPA-PSK WPA-EAP
#
#
#auth_alg: système d'authentification du réseau
=OPEN : pour WPA et WPA2
=SHARED : pour WEP
=LEAP : pour LEAP sur réseau EAP

#Par défaut : Sélection automatique
#
#
#pairwise: !A ne pas confondre avec le 'group' qui est plus connu!
=CCMP : AES (CBC-MAC : RFC 3610, IEEE 802.11i/D7.0)
=TKIP : (IEEE 802.11i/D7.0)
=NONE : Utilisé pour le mode ad-hoc principalement
#Par défaut : CCMP TKIP
#
#
#group: Chiffrage (courant)
=CCMP : AES (RFC 3610, IEEE 802.11i/D7.0)
=TKIP : (IEEE 802.11i/D7.0)
=WEP104 : WEP avec clef 104bits
=WEP40 : WEP avec clef 40bits (IEEE 802.11)
#Par défaut : CCMP TKIP WEP104 WEP40
#
#

```

```

#psk: clef WPA (256bits)
="mon mot de passe" : entre guillemets pour un mot de passe 'texte' (entre 8 et 63caractères
compris)
=2effca35695fe31c3b : pour une clef hexadécimale (peut être générée avec la commande
'wpa_passphrase "ssid" "mot de passe"')
#Non-utilisé en mode EAP.
This field is not needed, if WPA-EAP is used.
#
#
#Paramétrage EAP
#Non-Rédigé
#

#Exemple de blocs:

#WPA-PSK simple
network={
 ssid="simple"
 psk="very secret passphrase"
 priority=5
}

#EAP simple :
network={
 ssid="example"
 proto=RSN
 key_mgmt=WPA-EAP
 pairwise=CCMP TKIP
 group=CCMP TKIP
 eap=TLS
 identity="user@example.com"
 ca_cert="/etc/cert/ca.pem"
 client_cert="/etc/cert/user.pem"
 private_key="/etc/cert/user.prv"
 private_key_passwd="password"
 priority=1
}

#WPA-PSK et -EAP simple :
network={
 ssid="example"
 bssid=00:11:22:33:44:55
 proto=WPA RSN
 key_mgmt=WPA-PSK WPA-EAP
 pairwise=CCMP
 group=CCMP
 psk=06b4be19da289f475aa46a33cb793029d4ab3db7a23ee92382eb0106c72ac7bb
}

#EAP-PSK simple :
network={
 ssid="eap-psk-test"
 key_mgmt=WPA-EAP
 eap=PSK
 identity="eap_psk_user"
 eappsk=06b4be19da289f475aa46a33cb793029
 nai="eap_psk_user@example.com"
}

#IEEE 802.1X avec clefs WEP dynamiques
network={
 ssid="1x-test"
 key_mgmt=IEEE8021X
 eap=TLS
 identity="user@example.com"
 ca_cert="/etc/cert/ca.pem"
 client_cert="/etc/cert/user.pem"
 private_key="/etc/cert/user.prv"
 private_key_passwd="password"
 eapol_flags=3
}

```

```

#LEAP avec clefs WEP dynamiques
network={
 ssid="leap-example"
 key_mgmt=IEEE8021X
 eap=LEAP
 identity="user"
 password="foobar"
}

#Connexion directe (sans chiffrage)
network={
 ssid="plaintext-test"
 key_mgmt=NONE
}

#Clef WEP partagée :
network={
 ssid="static-wep-test2"
 key_mgmt=NONE
 wep_key0="abcde"
 wep_key1=0102030405
 wep_key2="1234567890123"
 wep_tx_keyidx=0
 priority=5
 auth_alg=SHARED
}

#Ad-Hoc (IBSS) avec chiffrage TKIP
network={
 ssid="test adhoc"
 mode=1
 proto=WPA
 key_mgmt=WPA-NONE
 pairwise=None
 group=TKIP
 psk="secret passphrase"
}

```

## ***Exemples***

Dans tous les cas votre fichier /etc/network/interfaces ressemblera à ça :

```

auto wlan0
iface wlan0 inet dhcp
 wpa-conf /etc/wpa_supplicant.conf

```

OU :

```

auto wlan0
iface wlan0 inet static
 address 192.168.0.12
 netmask 255.255.255.0
 gateway 192.168.0.1

 wpa-conf /etc/wpa_supplicant.conf

```

et seul votre fichier de configuration  
**/etc/wpa\_supplicant.conf** change :

- Connexion à un point d'accès en WPA2 AES :

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
network={
 ssid="Livebox-3818"
 proto=RSN
 key_mgmt=WPA-PSK
 group=CCMP
 psk="totomaison"
}
```

- Connexion à un point d'accès en WPA/WPA2 TKIP/AES :

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
network={
 ssid="Livebox-3818"
 proto=WPA RSN
 key_mgmt=WPA-PSK
 group=CCMP TKIP
 psk="totomaison"
}
```

- Connexion Ad-Hoc (PC à PC) WPA TKIP :

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=2
network={
 ssid="Livebox-3818"
 mode=1
 proto=WPA
 key_mgmt=WPA-NONE
 pairwise=None
 group=TKIP
 psk="totomaison"
}
```

## ➤ Plusieurs cartes à configurer en WPA

Si vous configurez plusieurs connexions en WPA il est recommandé d'indiquer à chaque connexion le driver à WPA Supplicant.

## ➤ Utilisation de Bridges/Ponts

Si vous utilisez un pont pour votre connexion WPA, il faut appliquer le fichier de configuration WPA Supplicant aussi bien à l'interface de connexion qu'au pont.

## ➤ Conflits avec logiciels de connexion Wi-Fi

Il se peut que WPA Suplicant entre en conflit avec vos logiciels de configuration réseau, dans ce cas désactiver les applications concernées comme indiqué plus haut.

## ➤ Configuration propre à certains chipsets

### *WPA avec une carte rt2x00*

Vous avez essayé les méthodes ci-dessus avec votre carte rt2x00 sans succès, voici une autre solution. En effet, les modules rt2x00 gèrent eux-mêmes la cryptographie WPA, ce n'est donc pas possible d'utiliser les utilitaires ci-dessus. Ce qu'il faut savoir avant de s'y lancer :

- Tout se passe avec les commandes **iwpriv** et **iwconfig**, il faut donc que la configuration Wi-Fi soit complète.
- Ce module a un défaut, il a besoin d'un certain temps pour se synchroniser avant de fonctionner, donc ne pas s'étonner des "sleep 3" dans le script ci-dessous.

Avec bien entendu \$IFACE le nom de votre interface réseau, \$ESSID le nom du réseau wifi et \$WPAPSK la clé wpa en clair.. (il y a un moyen de la chiffrer mais celui-ci ne sera pas détaillé. Les commandes importantes sont donc : "iwpriv \$IFACE enc 3" qui active la cryptographie, et "iwpriv \$IFACE auth 3" qui dit que ce sera du WPA-PSK

L'intérêt est d'en faire un petit script qui se lancera en pre-up de la configuration de la carte Wi-Fi Voici un exemple de script :

```
IFACE="rausb0" # a changer en fonction de l'interface
ESSID="*****"
WPAPSK="*****" # en clair

ifconfig $IFACE down
ifconfig $IFACE up
ifconfig $IFACE down
ifconfig $IFACE up
sleep 3
iwconfig $IFACE mode Managed
iwpriv $IFACE enc 3
iwpriv $IFACE auth 3
iwconfig $IFACE essid "$ESSID"
sleep 3
iwpriv $IFACE wpapsk "$WPAPSK"
sleep 3
iwconfig $IFACE essid "$ESSID"
sleep 3
```

et ensuite, dans le fichier **/etc/network/interfaces**, vérifier que l'on a les notes suivantes :

```
auto rausb0 # à adapter
iface rausb0 inet dhcp # à adapter si on est en static
 pre-up /usr/local/bin/wpa-config.sh # emplacement du script ci-dessus
```

Et voilà, votre carte Wi-Fi ira se connecter automatiquement au démarrage de la machine sur votre réseau Wi-Fi en WPA !

**Attention : le script ci-dessus dépend fortement de votre version d'iwpriv.** Ainsi les commandes auth et enc peuvent être refusées. Les trois lignes concernant iwpriv devront être remplacées par :

```
iwpriv $IFACE set EncrypType=TKIP
iwpriv $IFACE set AuthMode=WPAPSK
iwpriv $IFACE set WPAPSK="$WPAPSK"
```

Une variante peut être de ne pas se connecter automatiquement au boot et d'ajouter la commande "dhclient \$IFACE" à la fin du script et de lancer le script à la main dès que l'on veut se connecter, utile pour un portable qui voyage beaucoup.

## ➤ Utilisation

### ○ *Tester le service wpa-supplicant*

```
#sudo wpa_supplicant -c/etc/wpa_supplicant.conf -w -D<pilote> -i<interface>
```

- Le « pilote » générique est *wext*. Si vous utilisez un pilote Wi-Fi Windows via **Ndiswrapper**, remplacez <pilote> par **Ndiswrapper**.  
Vous pouvez exécuter *wpa\_supplicant* pour avoir la liste de tous les types d'interface du pilote supportés.
- L'interface la plus courante est *wlan0*.  
Vous pouvez vérifier quelle est votre carte avec la commande **iwconfig** pour peu que le module correspondant soit inséré dans le noyau.
- Vous pouvez utiliser l'option **-d** pour avoir plus d'informations.  
L'option **-B** n'a pas été mise pour ne pas lancer le processus en service pour ce premier test.

Si les informations à l'écran vous permettent de conclure que l'interface est connectée au point d'accès, vous pouvez passer à la suite. (Arrêtez le processus *wpa\_supplicant* avec les touches **<CTRL> + C**).

### ○ *Lancer le service wpa\_supplicant au démarrage*

Éditez le fichier **/etc/default/wpa\_supplicant** avec les droits d'administration pour y mettre ce qui suit :

```
ENABLED=1
OPTIONS="-w -D <pilote> -i <interface> -c /etc/wpa_supplicant.conf"
```

- *Pour restaurer le réseau après hibernation*

Pour restaurer le réseau après hibernation enregistrez le script **networking** qui suit dans **/etc/pm/sleep.d/networking** :

```
#!/bin/sh
/etc/pm/sleep.d/networking

case "$1" in
 hibernate|suspend)
 # /etc/init.d/networking stop
 ;;
 thaw|resume)
 (sleep 10 ; /etc/init.d/networking restart) &
 (sleep 20 ; /etc/init.d/networking restart) &
 ;;
 *)
 ;;
esac
```

Rendez-le exécutable

## ■ Le script Wificheck

**Wificheck** est un script bash créé pour faciliter la récolte des informations nécessaires à la résolution d'une majorité des problèmes rencontrés avec les connexions sans-fil.

Ces informations sont celles demandées expressément sur le sujet

Forum : <http://forum.ubuntu-fr.org/viewtopic.php?id=141183> du forum

### ➤ Pre requis

Le script nécessite des droits administrateurs pour l'exécution de certaines commandes, votre mot de passe vous sera donc demandé.

### ➤ Installation et lancement

**Étape 1** : Copiez et collez cette ligne dans un terminal, et validez par la touche entrée :

```
#wget -N -t 5 -T 10 https://framagit.org/cracolinux/wificheck/raw/master/wificheck && chmod +x wificheck
&& ./wificheck
```

Cette commande crée un fichier wificheck.log dans le dossier personnel. Ce fichier est un recueil d'informations sur les cartes wifi, les pilotes utilisés, le chiffrement de la box utilisé, ...

**Étape 2 :** Il ne reste plus qu'à copier le contenu du fichier wifichck.log (et non le fichier wifichck qui est le programme qui a généré le fichier wifichck.log) situé dans le dossier personnel et le copier sur le forum.

**N.B.** les balises code sont incluses dans le fichier wifichck.log, inutile de les ajouter.

## ➤ Désinstallation

Pour désinstaller le script il suffit de supprimer le fichier **wifichck**.

## ■ Utilisation avancée (WIFI)

Veuillez consulter la page [https://doc.ubuntu-fr.org/wifi#utilisation\\_avancee](https://doc.ubuntu-fr.org/wifi#utilisation_avancee) pour vous aider dans l'utilisation avancée d'une carte réseau WIFI.

## ■ Optimisation du WIFI

Voir la page [https://doc.ubuntu-fr.org/wifi\\_optimisation](https://doc.ubuntu-fr.org/wifi_optimisation)

## ■ Problèmes et cas particuliers

*Exemple : pour faire fonctionner le wifi avec une Livebox2, il n'y a en fait rien à changer sur le PC : il faut juste mettre la Livebox en mode association en appuyant sur le bouton N° 1 sur le côté de la box. Puis rentrer la clé WEP ou WPA quand Ubuntu vous le demande.*

Des problèmes qui semblent particuliers aux cartes Realtek sont également décrits sur cette page. Notamment la désactivation du double chiffrement et l'utilisation d'une autre antenne.

## ➤ Réseau sans fil détecté mais impossible de s'y connecter

Avec la version 1.4.0 de network-manager, a été introduit une nouvelle fonctionnalité consistant à modifier l'adresse MAC de la carte Ethernet ou WI-FI.

Pour le WI-FI cela peut créer le problème suivant : les réseaux sont bien visibles mais il est impossible de s'y connecter.

Pour remédier à cela, il suffit d'éditer un simple fichier.

Modifiez le fichier **/etc/NetworkManager/NetworkManager.conf** avec les droits d'administration et à la fin de ce fichier copier/coller les lignes suivantes :

```
[device]
wifi.scan-rand-mac-address=no
```

Le fichier devrait ressembler à cela :

```
[main]
plugins=ifupdown,keyfile

[ifupdown]
managed=false

[device]
wifi.scan-rand-mac-address=no
```

Enregistrer le fichier et fermer l'éditeur. Pour que la modification soit prise en compte, relancer le service network-manager.

Dans un terminal :

```
#sudo service network-manager restart
```

La connexion au réseau WIFI devrait maintenant s'effectuer normalement !

### ➤ UEFI Secure Boot

Certaines cartes Wifi sont bloquées via le **Secure Boot** {Secteur d'amorçage sécurisé}

**Par exemple:** Les cartes Broadcom bcm43xx, dont le code est la propriété de Broadcom Inc ™

### ➤ Clé WPA redemandée à chaque démarrage

Si votre clé d'accès à la borne est redemandée sans cesse et que vous ne voulez pas avoir de trousseau de clé à débloquer toujours :

1. Supprimez tous les fichiers du dossier caché `~/.local/share/keyrings/` de votre **Dossier Personnel**,
2. Avec Seahorse, créez un nouveau trousseau appelé `login` auquel vous ne donnez pas de mot de passe (laissez vide)
3. Reconnectez-vous à votre box.

Désormais la connexion est automatique à chaque démarrage sans avoir à saisir de mot de passe. Pour plus d'informations, veuillez consulter la section « Solution avec perte de sécurité » de la page « Désactiver les mots de passe ».

## ➤ Voir en clair une clé WIFI enregistrée

Sur les versions récentes d'Ubuntu, au moins depuis 16.04, il est facile de retrouver une clé WIFI que l'on a entré si on en a besoin (perdue, nécessaire pour un autre appareil...).

- Avec l'interface GNOME shell (17.04 et supérieures), aller dans Paramètres / WIFI ; à droite, sélectionner le réseau voulu et cliquer sur la roue dentée. Une nouvelle fenêtre apparaît ; dans l'onglet sécurité, vous pouvez faire apparaître le mot de passe.
- Avec Unity, cliquer sur l'icône réseau en haut à droite de l'écran (dans les indicateurs) et choisissez dans le menu "Modification des connexions". Une nouvelle fenêtre s'ouvre. Dans les réseaux sans fil, cherchez le réseau qui vous intéresse, sélectionnez-le puis cliquez le bouton "modifier". Dans la nouvelle fenêtre, choisissez l'onglet sécurité pour voir le mot de passe.
- Pour les versions anciennes d'Ubuntu, il faut rechercher l'application "Mots de passe et clés". Dans la fenêtre qui s'affiche, dans le menu de gauche, dans les mots de passe, vous trouvez les mots de passe WIFI

## ➤ Réseau sans fil détecté mais impossible d'obtenir l'adresse IP

Le logiciel de configuration du réseau (**Network-Manager**, **Wicd**, etc) arrive à détecter les réseaux sans fil disponibles, mais quand vous cliquez pour vous connecter, la connexion n'aboutit pas.

Dans le cas de **Network-manager**, dès que vous cliquez sur le SSID, le logiciel tente de se connecter, puis s'arrête au bout de quelques secondes.

Pour ce qui est de **Wicd**, le logiciel gèle sur l'étape **d'Obtention de l'adresse IP...** avec la barre de progression qui oscille de gauche à droite continuellement.

Pour régler cela, suivez ces instructions :

- Changez de service de gestion réseau en "arrêtant" Network-Manager pour réactiver networking :

```
#sudo service network-manager stop && sudo service networking start
#sudo update-rc.d network-manager remove
#sudo update-rc.d networking defaults
```

Puis dans un terminal rechercher le nom de votre carte réseau sans fil en tapant ce qui suit :

```
#iwconfig
```

Et ça devrait afficher quelque chose comme cela :

```
lo no wireless extensions.

eth0 no wireless extensions.

wlan0 IEEE 802.11bg ESSID:off/any
 Mode:Managed Frequency:2.422 GHz Access Point: Not-Associated
 Tx-Power=20 dBm
 Retry long limit:7 RTS thr:off Fragment thr:off
 Power Management:off
```

En l'occurrence, **wlan0** est le nom de la carte réseau *sans fil* de votre ordinateur. Notez-le car on en aura besoin plus tard !

- Modifiez avec les droits d'administration le fichier **/etc/network/interfaces** :

```
auto lo
iface lo inet loopback
```

Ajouter dans ce fichier texte le paragraphe suivant, où **wlan0** est le nom de votre carte réseau *sans fil*.

```
auto wlan0
iface wlan0 inet dhcp
```

- Redémarrez le réseau avec cette commande :

```
#sudo /etc/init.d/networking restart
```

Enfin, réessayez de vous connecter ça devrait fonctionner !

## ➤ Connexion lente

Si vous remarquez que votre connexion Wi-Fi est anormalement lente et que cela ne vient pas de votre matériel (carte ou AP) et avant de réinstaller, recompiler...d'autres pilotes :

- Vérifiez le Bit Rate de la carte :

```
iwconfig wlan0 | iw dev wlan0 link
```

- Si vous voyez Bit Rate=1Mb/s alors tapez :

```
#sudo ifdown wlan0
#sudo iwconfig wlan0 rate 54M
#sudo ifup wlan0
```

Enfin, faites "iwconfig" pour vérifier qu'il vous met bien Bit Rate=54Mb/s. Voilà qui devrait aller mieux !

pour que ça reste permanent : Ajoutez dans **/etc/rc.local** (avant la ligne exit 0)

```
ifconfig wlan0 up
iwconfig wlan0 rate 54M
```

On redémarre et le tour est joué !

## ➤ Problèmes de résolution DNS et configuration en IP Fixe

Il peut arriver qu'une configuration en IP fixe, via `/etc/network/interfaces`, ne vous permette plus d'effectuer des "résolution DNS".

En clair cela signifie que votre ordinateur ne sera plus en mesure de traduire un nom de domaine (ex: [www.google.com](http://www.google.com)) en adresse IP.

Plusieurs solutions traînent sur la toile, avec plus ou moins d'efficacité et plus ou moins de « bonne pratique ».

Dans ce cas et uniquement si vous utilisez une configuration fixe définie dans `/etc/network/interfaces`. Il est recommandé de :

- Désactiver le service Network-Manager, de
- Réactiver le service networking et de
- Configurer votre/vos interfaces en renseignant DNS et Domaine de recherche.

Comme mentionné sur le fil de discussion Ubuntu 12.04 Server: Configuration DNS et IP fixe — source : [sorrodje.alter-it.org](http://sorrodje.alter-it.org) ; titre : « Ubuntu 12.04 Server: Configuration DNS et IP fixe » ; auteur : Sorrodje ; date : 30 mai 2012.

- Ajoutez ceci à la configuration de votre/vos interfaces :

```
auto eth1
iface eth1 inet static
[...]
dns-nameservers 8.8.8.8 # ip de votre router ou du dns à utiliser
dns-search localdomain # nom de domaine local de recherche si exploité
```

Relancez vos interfaces ou redémarrez votre ordinateur !

## ➤ La connexion réseau ne se réactive pas en sortie de veille ou d'hibernation

Une solution est de lister en terminal le module utilisé par le wifi :

```
#sudo lshw -class network
```

- Noter le nom du **driver** WIFI utilisé spécifié par la commande **lshw -class network**.

Par exemple :

```
lshw -class network
ATTENTION: ce programme devrait être lancé en tant que super-utilisateur
*-network
 description: Interface réseau sans fil
 driver=r8712u
```

- o vérifier que la commande **lsmod** affiche le même nom du driver.

Par exemple :

```
lsmod
r8712u 142855 0
```

En tout état de cause c'est bien le nom du module listé par la commande **lsmod** qu'il faudra insérer dans le fichier **/etc/pm/config.d/load**

- o Taper cette commande pour créer (ou modifier) le fichier **/etc/pm/config.d/load**

```
echo "SUSPEND_MODULES=r8712u" | sudo tee -a /etc/pm/config.d/load
```

- o Rendre le fichier exécutable :

```
#sudo chmod +x /etc/pm/config.d/load
```



**Il semble qu'à partir de la version 16.04 et du noyau 4.4.0-66 il faille en plus relancer NetworkManager**

```
sudo service network-manager restart
```

Si cette commande fonctionne, il est possible d'automatiser l'opération pour éviter de le faire après chaque mise en veille :

1. Créer un script dans le répertoire **/etc/pm/sleep.d/NWM.sh** :

```
echo -e '#!/bin/sh\n/etc/init.d/network-manager restart\nexit 0' | sudo tee /etc/pm/sleep.d/NWM.sh
```

2. Rendre le script exécutable:

```
sudo chmod +x /etc/pm/sleep.d/NWM.sh
```



**A partir d'Ubuntu 15.04** Ubuntu étant passé à systemd depuis la version 15.04, la solution ci-dessus ne fonctionne plus. Une solution consiste à créer le fichier suivant:

1. Ouvrez un **terminal** et exécutez la commande suivante :

```
sudo nano /lib/systemd/system/retablir_wifi_apres_veille.service
```

Entrez votre mot de passe.

2. Collez le contenu suivant:

```
[Unit]
Description=Local system resume actions
After=suspend.target

[Service]
Type=oneshot
ExecStart=/sbin/rmmod r8712u ; /sbin/modprobe r8712u

[Install]
WantedBy=suspend.target
```

► en remplaçant **r8712u** par le nom du pilote concerné en se référant à la méthode indiquée ci-dessus pour Ubuntu 14.10 pour le trouver. Faites Ctrl+X pour sauvegarder puis répondre O pour enregistrer les changements.

3. Lancer enfin les 2 commandes suivantes afin que le système prenne en compte le fichier et démarre le service:

```
sudo systemctl enable retablir_wifi_apres_veille
sudo systemctl start retablir_wifi_apres_veille
```

# Modems

Cette page a pour but de vous décrire chaque type de modem et de vous aiguiller vers la page de la documentation correspondante.

## ■ Modems bas débit

Les modems Bas débit et ADSL (PPPoE, PPPoA) peuvent parfois poser problème à la configuration, ou demander des pilotes ou configurations spécifiques. Les modems bas débit (souvent appelés 56k ou analogiques), ont pour la plupart besoin de pilotes pour fonctionner... Il vous faudra utiliser soit Wvdial, soit gnome-ppp, pour pouvoir utiliser ce type de modem.

### ➤ Connexion avec Wvdial (ligne de commande)

Reportez-vous à la page sur <https://doc.ubuntu-fr.org/wvdial>

### ➤ Configuration avec une interface graphique

Pour Ubuntu ou Xubuntu ou Lubuntu, on pourra utiliser gnome-ppp qui est une interface graphique à Wvdial. Cette application permet de vérifier que le modem est reconnu, de le paramétriser et d'établir les connexions. Pour Kubuntu, l'application équivalente est [kPPP](#).

### ➤ Installation des pilotes

Si votre modem est branché via un câble **Série**, vous n'aurez pas besoin d'installer de pilote dans 99% des cas ! Si votre modem est usb, c'est 50/50... Par contre, si votre modem est interne, vous avez très peu de chance de pouvoir l'utiliser sans pilote...

*Si cette procédure ne marche pas voici donc des méthodes pour installer les pilotes de votre modem :*

[https://doc.ubuntu-fr.org/modem\\_56k-proprio](https://doc.ubuntu-fr.org/modem_56k-proprio)

### ➤ Modem ISDN

Voir [https://doc.ubuntu-fr.org/modem\\_isdn](https://doc.ubuntu-fr.org/modem_isdn)

### ➤ Modem 3G / 3G+ / LTE

Les modems UMTS ou GPRS (connu communément avec le nom d'Internet Mobile) permettent une connexion à internet en utilisant les mêmes réseaux qu'utilisent les téléphones mobiles (GSM, GPRS et CDMA). Et la plupart de ces modems sont USB ou PCMCIA.

## *Configuration*

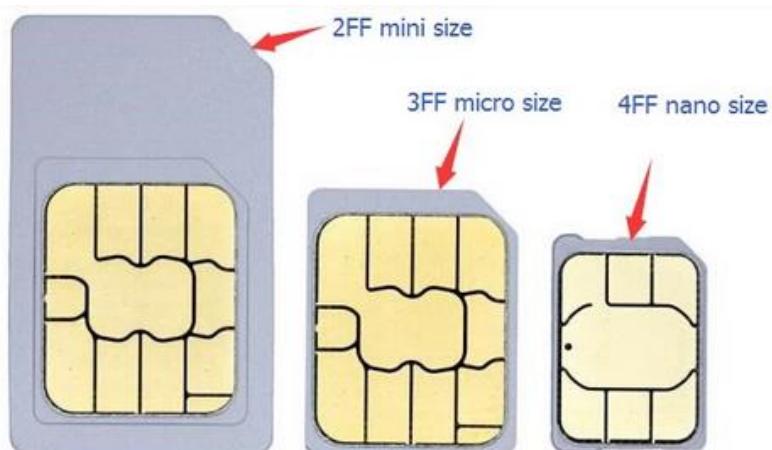
Le paramétrage avec Network-manager est extrêmement simple. Suivez les indications [https://doc.ubuntu-fr.org/telephone\\_modem](https://doc.ubuntu-fr.org/telephone_modem). Il s'agit, une fois votre clé branchée, de cliquer sur l'applet de Network-manager, vous y verrez affiché : " Nouvelle connexion mobile à large bande". Cliquez dessus et suivez les instructions. Si le fournisseur, ou votre pays n'est pas dans la liste, vous devez remplir manuellement. Pour cela il vous suffit généralement de connaître l'APN de votre opérateur. Vous pouvez modifier les paramètres en cliquant sur "Modifications des connexions" toujours dans l'applet de Network-manager.

## *Gérer sa SIM*

Pour gérer la recharge USSD, via l'envoi d'un code (ex \*100#), de SMS, etc, installer **modem-manager-gui** ou encore **prepaid-manager-applet**.

Si avec prepaid-manager vous avez l'erreur suivante :

```
"Modem error: ussd_initiate failed: Timeout was reached"
```



Ou encore :

```
"Modem error: ussd_initiate failed:
GDBus.Error:org.freedesktop.ModemManager.Modem.General: USSD session already active."
```

Vous pouvez éditez le fichier **/usr/share/mobile-broadband-provider-info/serviceproviders.xml** avec les droits d'administration et y intégrer les paramètres propres à votre opérateur comme suit, si celui-ci n'est pas encore dans la liste. S'il est dans la liste alors vous avez juste à compléter la ligne **ussd** en y mettant votre code USSD :

```
<apn value="aircelgprs.pr">
 <plan type="prepaid"/>
 <usage type="internet"/>
 <name>GPRS (Prepaid)</name>
 <voicemail>120</voicemail>
 <balance-check>
 <ussd>*122*011#</ussd>
 </balance-check>
 <balance-top-up>
 <ussd replacement="CODE">*130*CODE#</ussd>
 </balance-top-up>
 </apn>
<apn value="aircelumts.pr">
 <plan type="prepaid"/>
 <usage type="internet"/>
 <name>GPRS (Prepaid)</name>
 <voicemail>120</voicemail>
 <balance-check>
 <ussd>*122*011#</ussd>
 </balance-check>
 <balance-top-up>
 <ussd replacement="CODE">*130*CODE#</ussd>
 </balance-top-up>
 </apn>
<apn value="aircelweb.pr">
 <plan type="prepaid"/>
 <usage type="internet"/>
 <name>GPRS (Prepaid)</name>
 <voicemail>120</voicemail>
 <balance-check>
 <ussd>*122*011#</ussd>
 </balance-check>
 <balance-top-up>
 <ussd replacement="CODE">*130*CODE#</ussd>
 </balance-top-up>
 </apn>
</apn>
```

Vous pouvez également le faire en ligne de commande avec ModemManager. Commencez par lister vos périphériques :

```
#mmcli -L
```

Cette commande vous donnera un retour du type :

```
/org/freedesktop/ModemManager1/Modem/12
```

Ce qui vous permet de connaître le chiffre à utiliser pour les commandes suivantes. Activer votre modem :

```
#mmcli -m 12 -e
```

Enfin entrez votre requête USSD :

```
#mmcli -m 12 --3gpp-ussd-initiate="*100#"
```

Le terminal devrait vous répondre quelque chose du type :

```
USSD session initiated; new reply from network: 'Votre forfait mensuel Internet est de
2GB 298MB et 446KB valable jusqu au 14/12/2016 17:33:24.'
```

Adaptez le 12 au chiffre reçu après la première commande, et le code USSD au vôtre.

Voir ces pages pour plus d'informations: Clé USB Orange ([https://doc.ubuntu-fr.org/orange\\_3g](https://doc.ubuntu-fr.org/orange_3g)) et Carte PCMCIA SFR ([https://doc.ubuntu-fr.org/pcmcia\\_3gplus\\_sfr](https://doc.ubuntu-fr.org/pcmcia_3gplus_sfr)) ainsi que celle-ci pour d'autres alternatives.

### ➤ Utiliser votre téléphone 3G pour vous connecter à internet

Dans la plupart des cas il n'y a aucunement besoin de vous procurer une clé 3G pour avoir accès à internet avec le réseau mobile, vous pouvez directement utiliser votre téléphone. Voir cette page [https://doc.ubuntu-fr.org/telephone\\_modem](https://doc.ubuntu-fr.org/telephone_modem) pour plus d'informations.

## ■ Modems haut débit

Les modems Haut débit existent en de multiples modèles.

- \* Il y a les modems routeurs, ou les box, (connectés en Ethernet : Câble réseau) qui s'ils sont bien configurés, fonctionneront immédiatement sous Ubuntu.
- \* Il y a les anciens modems PPPoE et PPPoA en usb, ou en Ethernet...
- \* Il y a aussi les modems hybrides, qui sont des modems PPPoE en Ethernet, qui contiennent un routeur, mais ce routeur est désactivé par défaut !

### ➤ Si votre modem est Ethernet - Câble réseau et connexion wifi

Si votre modem est en Ethernet, et que malgré tout, il ne marche pas directement, cette page vous expliquera comment la configurer, ainsi que l'activation du wifi, et de la configuration des ports : [modem\\_adsl\\_routeur\\_et\\_box](#)

Si votre modem se révèle ne pas avoir de routeur... passez à la prochaine étape.

## ➤ Modem USB, ou Ethernet sans routeur

Si votre modem est en Ethernet, et que malgré tout, il ne marche pas directement, cette page vous expliquera comment la configurer, ainsi que l'activation du wifi, et de la configuration des ports : [modem\\_adsl\\_routeur\\_et\\_box](#)

Voici la page qui vous permettra de configurer votre modem sans trop de douleurs : [https://doc.ubuntu-fr.org/modem\\_adsl](https://doc.ubuntu-fr.org/modem_adsl)

## ■ Modems câble

Certains fournisseurs d'accès à Internet, comme Vidéotron (Québec), Cogeco (Québec), Wanadoo, Bravo Telecom (Québec) et Numéricâble (Europe), proposent un accès à Internet par le câble.

Marque	Modèle	Type de branchement	Reconnaissance
Scientific Atlanta	Webstar DPX 100	Ethernet	Automatique
Scientific Atlanta	Webstar DPX 100	USB	Automatique
Motorola	SURFboard SB4200E	Ethernet	Automatique
RCA	DCM425	Ethernet et USB	Automatique
Thomson, Inc	DCM245 Cable Modem	Ethernet et USB (Ethernet over USB)	Automatique

**Remarque :** Le modem Thomson établit la connexion de manière autonome. Il agit comme un routeur : vous devriez avoir internet sans rien faire, sinon suivez le guide du réseau (vous êtes dans le cas 1).

## ■ NAVIGATION SUR LE WEB (sous Linux)



Le **Web** est l'ensemble des documents et des services disponibles sur Internet accessibles via le protocole HTTP(S). Une caractéristique essentielle du Web est la capacité de passer d'un service à un autre par des liens hypertexte. Ces liens forment le réseau d'où le Web (*toile* en anglais) tire son nom.

Un site Web est quant à lui un ensemble de pages Web liées entre elles, traitant d'un même sujet et visibles sur un même nom de domaine.

Plus d'informations sur le World Wide Web : <https://fr.wikipedia.org/wiki/Web>

## HTTP et navigation

Pour accéder à un site Web, il faut utiliser un client Web, communément appelé *navigateur*, par exemple : Firefox, Chromium, Google Chrome, Opera...

### ■ URL

On passe généralement par un moteur de recherche pour trouver un site, mais on peut aussi saisir directement l'adresse d'une page web pour l'afficher. On parle alors d'**URL**.

Une URL est donc l'adresse d'un contenu Web spécifique.

Une URL se compose ainsi (sans crochet ni espace) :

**[protocole] :// [hôte] / [chemin] / [nom de la page web] ? [chaîne de requête] # [signet]**

- Dans le domaine du Web, les protocoles sont **HTTP** ou **HTTPS**. Généralement il n'est pas nécessaire de saisir le protocole dans la barre d'adresses, le navigateur web l'ajoute automatiquement si il est manquant. Le protocole est même masqué par défaut dans la barre d'adresse de certains navigateurs.
- L'hôte peut-être au choix une **adresse IP** ou un **nom de domaine** avec ou sans sous-domaine. Un nom de domaine est de la forme : **[sous-domaine] . [SLD] . [TLD]**

Pour un client web il est seulement obligatoire de spécifier le protocole et l'hôte. C'est généralement l'adresse de la page d'accueil des sites qui est dans ce cas affichée par défaut.

## ■ HTTP

En anglais, HTTP est l'acronyme de *Protocole de Transfert HyperTexte*.

Un document hypertexte est un document contenant des hyperliens. Il s'agit généralement de documents HTML. Les hyperliens sont des liens vers d'autres URL. Ils permettent d'établir un réseau entre les documents, les services, et avec d'autres formes de contenu (images, documents, multimédia...).

HTTPS est une version sécurisée du HTTP, qui implique que les informations échangées entre le navigateur et le serveur web soient chiffrées.

HTTP utilise par défaut le port **80**, et HTTPS le port **443**.

Il existe une extension du protocole HTTP appelée WebDAV qui permet de récupérer, envoyer et synchroniser des fichiers via le protocole HTTP. WebDAV a lui-même été étendu aux protocoles CalDAV et CardDAV qui permettent de gérer des agendas et des carnets d'adresses via le web.

## Clients

Les clients web les plus couramment utilisés sont les navigateurs : **Firefox**, **Chromium**, **Google Chrome**, **Opera**, etc.

Les autres clients HTTP sont par exemple, **les robots d'indexation**, **les aspirateurs de site** ou **les agrégateurs web**.

Pour l'envoi, récupération et synchronisation de fichiers avec **WebDAV**, le client est souvent le gestionnaire de fichiers : **Dolphin**, **Nautilus**,... et utilise parfois des noms de protocoles non standard tels que *davs* (GNOME) ou *webdav* (KDE) (le nom standard du protocole est *dav*).

Pour la synchronisation des contacts et agenda, les clients sont généralement les applications de gestion d'informations personnelles (PIM) de l'environnement de bureau, ou des clients de messagerie qui implémentent ce type de synchronisation : **Evolution**, **Kontact**,...

## Serveurs

Un serveur web est une machine fournissant l'ensemble des services permettant de mettre à disposition du contenu web. Son centre névralgique est donc le **serveur HTTP**.

Les serveurs HTTP les plus utilisés sur Internet sont **Apache** et **Nginx**. En plus de ces deux serveurs, le serveur **Lighttpd** est également disponible sous Ubuntu.

Les applications et les langages de développement pour le web possèdent aussi souvent leur propre serveur HTTP intégré, c'est le cas pour **PHP**, **Python**, **NodeJS**, etc.

Un serveur web est rarement constitué uniquement d'un service HTTP. C'est le plus souvent une pile logicielle comprenant un serveur HTTP, un interpréteur de scripts et un système de gestion de bases de données.

On utilise couramment la pile **LAMP** : Linux + Apache + MySQL + PHP, mais d'autres combinaisons existent :

- **LEMP** = Linux + Nginx + MySQL + PHP
- Linux + Apache + MySQL + Python
- Linux + Apache + PostgreSQL + PHP
- Linux + NodeJS + PostgreSQL
- etc.

## Les navigateurs web

Ce portail a pour but de recenser un maximum de **navigateurs web** fonctionnant sous Ubuntu, et de les présenter brièvement.

L'utilisation d'un bloqueur de publicité est très fortement recommandée.

Installez l'extension **AdBlock Plus**, **uBlock Origin** (ou une extension similaire), afin de masquer les bandeaux de publicités ainsi que les popups sponsorisés qui ralentissent votre navigation.

### ■ Les principaux navigateurs

- La colonne *Interface* indique avec quels outils graphiques sont développés l'interface, ou le cas échéant avec quels environnements le navigateur se marie bien (notamment en termes d'aspect, en fonction de votre thème graphique).

C'est un paramètre dépendant de votre environnement :

- GNOME Shell, Xfce, Budgie, Cinnamon et Unity correspondent à **GTK+** ;
- KDE Plasma et LXQt correspondent à **Qt**

- La colonne *Format* indique si le code du navigateur est open-source. Les licences respectives de ces navigateurs sont à chercher sur leurs sites web respectifs.

- La colonne *HTML5* indique si le navigateur a un support correct des technologies web modernes. Les scores ont tous été relevés durant la même période (le 23/07/2017) et sur le même système (Ubuntu MATE 17.04), afin que les navigateurs soient comparables, **merci donc de ne pas éditer individuellement les scores**.

Un score supérieur à 450/555 est normalement largement suffisant pour un usage complet du web ; un usage normal du web nécessite un score minimum de l'ordre de 350. La présence d'une croix rouge signifie que le navigateur n'avait même pas les technologies suffisantes pour exécuter le test pratiqué pour relever ces scores.

NOM	Icon	Free	Moteur de rendu	HTML5	Interface	Description
<u><a href="#">FIREFOX</a></u>			Gecko	474/555	Bonne intégration à Unity.	Navigateur web multiplateforme de référence, libre, extensible, personnalisable, avec des performances inégalées. C'est le navigateur par défaut de la plupart des variantes d'Ubuntu.
<u><a href="#">CHROMIUM</a></u>			Blink	516/555	intégration GTK+ correcte	Navigateur open-source de Google, servant de base à <i>Google Chrome</i> . Navigateur par défaut d'Ubuntu Budgie. Il est compatible avec les extensions du Chrome Web Store.
<u><a href="#">QUPZILLA</a></u>			QtWebEngine (basé sur WebKit)	320/555	Qt	Navigateur extrêmement léger <sup>2)</sup> basé sur QtWebEngine (qui remplace QtWebKit). Très adapté à KDE ou LXQt, il est très portable (et peut même être utilisé sur Windows). Le développement de QupZilla a été abandonné au profit de Falkon
<u><a href="#">FALKON</a></u>			QtWebEngine (basé sur WebKit)	516/555	Qt	Successeur de QupZilla, intégré aux projets KDE.
<u><a href="#">GNOME WEB</a></u>			WebKitGTK+	397/555	GTK+ 3	Navigateur web épuré du projet GNOME, intégré et adapté à l'ergonomie de GNOME Shell.
<u><a href="#">EOLIE</a></u>			WebKitGTK+	397/555	GTK+ 3	Navigateur épuré et léger, mais complet et innovant, particulièrement adapté à un environnement GNOME 3.
<u><a href="#">MIDORI</a></u>			WebKitGTK+	319/555	GTK+ 3	Navigateur ultra-léger récemment intégré au projet Xfce. C'est aussi le navigateur par défaut dans Elementary OS.
<u><a href="#">VIVALDI</a></u>			Blink	516/555	Thème très personnalisable	Navigateur multi-plateforme, basé sur Chromium, qui a pour but d'implémenter les fonctionnalités avancées d'Opera 12. Il est compatible avec les extensions du Chrome Web Store.
<u><a href="#">KONQUEROR</a></u>			KHTML	487/555	Qt	Couteau suisse du bureau KDE Plasma, c'est entre autres un navigateur web très moderne et performant, et un gestionnaire de fichiers riche en fonctionnalités pour KDE.

NOM	Icon	Free	Moteur de rendu	HTML5	Interface	Description
<u><a href="#">MIN</a></u>			Electron	516/555		Navigateur simple, léger et multiplateforme, Min est développé avec la technologie Electron.
<u><a href="#">BRAVE</a></u>			Blink	526/555	intégration GTK+ correcte	Navigateur open-source basé sur Chromium, mais plus rapide. Il est orienté vers la confidentialité de l'utilisateur. Il permet en outre d'accéder au réseau Tor. Suivez les instructions de la page d'explication de l'installation sur le site du projet
<u><a href="#">PALE MOON</a></u>			Goanna	397/555		Navigateur léger, rapide et personnalisable, basé sur une ancienne version de Firefox.
<u><a href="#">OPERA</a></u>			Blink	518/555		Navigateur gratuit, rapide extensible et sécurisé, basé sur Chromium.
<u><a href="#">NAVIGATEUR WEB UBUNTU</a></u>			Blink	516/555	Ubuntu Phone	Le "Navigateur Web Ubuntu" (webbrowser-app), basé sur Chromium et développé par Canonical pour la convergence desktop, smartphone et tablette, est le navigateur de la tablette BQ Aquaris M10 vendue avec Ubuntu.
<u><a href="#">IRIDIUM BROWSER</a></u>			Blink	516/555	intégration GTK+ correcte	Basé sur Chromium, un navigateur stable et open-source, "dégooglisé" et sans intrusion dans votre vie privée. Hélas, il n'est pas disponible en français pour Ubuntu !
<u><a href="#">GOOGLE CHROME</a></u>			Blink	518/555	intégration GTK+ correcte	Version non-libre de Chromium, avec des améliorations mais davantage d'immiscions dans votre vie privée.
<u><a href="#">TOR BROWSER</a></u>			Gecko	362/555		Navigateur basé sur Firefox, qui protège la vie privée, et permet de surfer anonymement sous certaines conditions. Il permet aussi d'accéder au réseau anonyme TOR.
<u><a href="#">SLIMJET</a></u>			Blink	516/555	intégration GTK+ correcte	Navigateur (propriétaire ?) basé sur Chromium avec des fonctionnalités pré-intégrées comme le blocage de publicités et le téléchargement de vidéos depuis le net

Nom	Icon	Free	Moteur de rendu	HTML5	Interface	Description
<u>REKONQ</u>			Webkit	279/555	Qt	Un navigateur ultra-léger, bien intégré à KDE, mais dont le développement s'est arrêté en 2014 <sup>3)</sup> .
<u>UZBL</u>			Webkit	308/555		Navigateur graphique très léger, avec interface en lignes de commande.
<u>NETSURF</u>			Netsurf		GTK+	Fourni par le paquet netsurf-gtk, c'est un navigateur graphique basique, à l'interface facile à maîtriser.
<u>DILLO</u>			?		FLTK	Un navigateur extraordinairement léger (utilise à peine plus d'1Mo de mémoire RAM !), mais utilisant les standards du web des années 2000.

## ■ Navigateur en ligne de commande

- elinks
- lynx
- w3m (commandes au clavier ou à la souris)
- links2

La plupart de ces navigateurs en ligne de commande proposent (parfois via un paquet à part) un mode graphique permettant d'afficher certaines images et couleurs.

## ■ COURRIER ELECTRONIQUE (sous Linux)



Le **courrier électronique** (ou courriel) désigne le service de transfert de messages envoyés à des destinataires par un système de messagerie électronique à travers Internet. Tout comme pour le courrier postal, l'expéditeur d'un message électronique précise à quelle adresse celui-ci doit être envoyé. Internet joue alors le rôle de postier et achemine le message au bon destinataire.

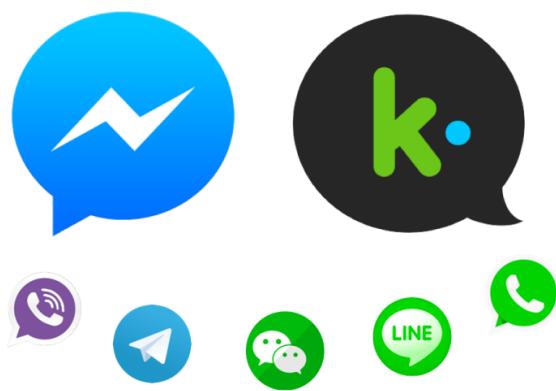
Pour recevoir et envoyer des courriels, il est indispensable de disposer soi-même d'une adresse électronique (une boîte à lettres) et d'un client de messagerie électronique (l'outil vous servant à consulter votre boîte à lettres). Ubuntu n'est pas un fournisseur de services de courrier électronique : votre FAI met généralement à votre disposition une adresse électronique à votre nom, et de nombreux fournisseurs de services courriels gratuits existent dans Internet

Plus d'information sur le courrier électronique : [https://fr.wikipedia.org/wiki/Courrier\\_%C3%A9lectronique](https://fr.wikipedia.org/wiki/Courrier_%C3%A9lectronique)

Ubuntu inclut de base plusieurs logiciels de messagerie électronique pour vous permettre de consulter votre boîte.

- **Thunderbird** (Ubuntu et Xubuntu)
- **Evolution** (Ubuntu GNOME)
- **Kontact** (Kubuntu)

## ■ MESSAGERIE INSTANTANEE (sous Linux)



La **messagerie instantanée** (ou *instant messaging*, IM) est un terme qui désigne l'échange instantané de messages textuels entre plusieurs ordinateurs connectés au même réseau de communication. Contrairement au courrier électronique, ce qui distingue la messagerie instantanée est que les messages sont envoyés en temps réel, ce qui permet un dialogue interactif entre les acteurs d'une communication par ce système.

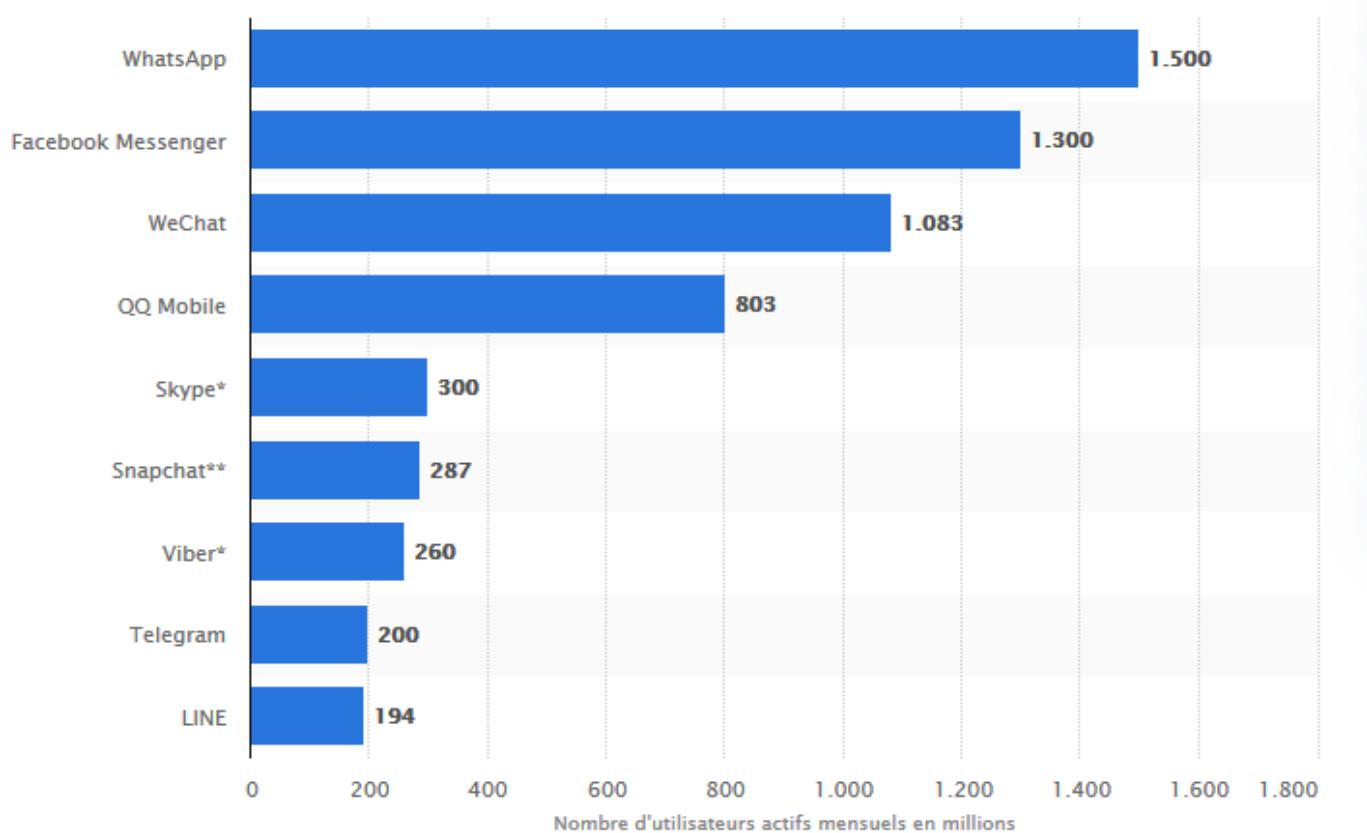
Afin de participer à des discussions instantanées, il est indispensable de rejoindre un réseau de messagerie

instantanée. Ceux-ci sont nombreux dans Internet : parmi les plus mondialement connus, nommons .NET Messenger Service (MSN), Yahoo! et Jabber. Contrairement au courrier électronique, la plupart des réseaux de messagerie instantanée ne peuvent pas communiquer entre eux. Avant de rejoindre un réseau IM, consultez vos correspondants pour savoir lequel ils utilisent.

Une fois que vous avez rejoint un réseau IM, pour recevoir et envoyer des messages instantanés, vous devez disposer d'un client de messagerie instantanée. Ubuntu inclut de base des logiciels de messagerie instantanée, qui n'attendent que vous leur fournissiez vos informations d'adhésion à un réseau.

Plus d'information sur la messagerie instantanée : [https://doc.ubuntu-fr.org/messagerie\\_instantanee](https://doc.ubuntu-fr.org/messagerie_instantanee)

## Statistic d'utilisation d'app populaire en IM (2020) :



## ➤ TELEPHONIE et VIDÉOPHONIE



Grâce à l'augmentation des débits de transfert, de nouvelles possibilités technologiques se généralisent. C'est le cas de la **téléphonie sur IP** (ou *voix sur IP*, VoIP), qui permet de converser vocalement entre deux ordinateurs, et de la **vidéophonie sur IP** (ou *vidéoconférence*), qui permet de communiquer vocalement et visuellement par la transmission instantanée de votre voix et de votre image.

Afin de participer à des discussions instantanées par téléphonie ou par vidéophonie, vous devez disposer d'un microphone (pour la voix) et d'une caméra (ou *webcam*, pour votre image). Pour une communication fluide, il est vivement conseillé de disposer d'une connexion haut débit à Internet. Enfin, il est indispensable de rejoindre un réseau de communication par IP. Ceux-ci sont nombreux dans Internet : parmi les plus mondialement connus, nommons Skype, Gtalk et SIP.

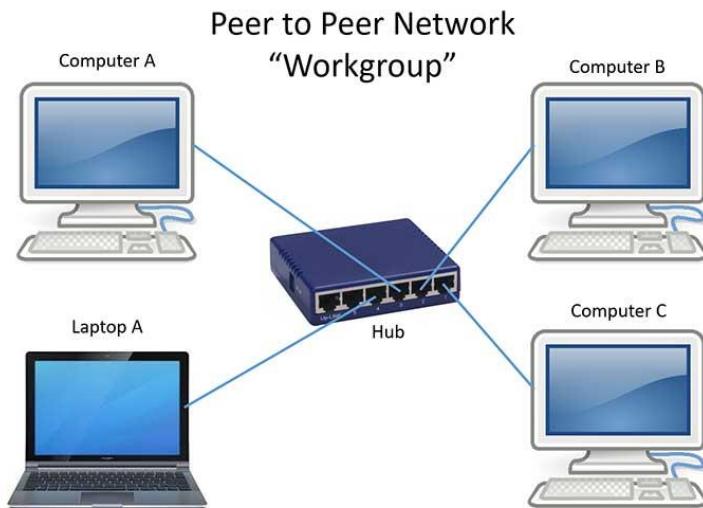
Chaque réseau de communication par IP ne peut pas communiquer avec les autres réseaux. Avant de rejoindre un réseau de téléphonie ou de vidéophonie sur IP, consultez vos correspondants pour savoir lequel ils utilisent.

Une fois que vous avez rejoint un réseau de communication sur IP, pour recevoir et envoyer des appels, vous devez disposer d'un client de téléphonie et de vidéophonie. Vous pouvez installer très facilement de tels logiciels sur Ubuntu, il suffira alors de leur fournir vos informations d'identification.

*Plus d'informations sur la téléphonie et la vidéoconférence sur Ubuntu : <https://doc.ubuntu-fr.org/visioconference>*

- Empathy
- Skype
- Ekiga
- Linphone
- Jitsi

## ■ ECHANGE DE FICHIER P2P (sous Linux)



Le terme **pair-à-pair** est la traduction de l'anglais *peer-to-peer*, laquelle est souvent abrégée P2P. Les systèmes pair-à-pair permettent à plusieurs ordinateurs d'échanger entre eux très facilement des fichiers informatiques par Internet. Puisque ces échanges ne passent pas par un serveur, il est assez difficile de les contrôler, de les réguler. Beaucoup de systèmes d'échange de fichiers musicaux et vidéo s'appuient sur ces principes.

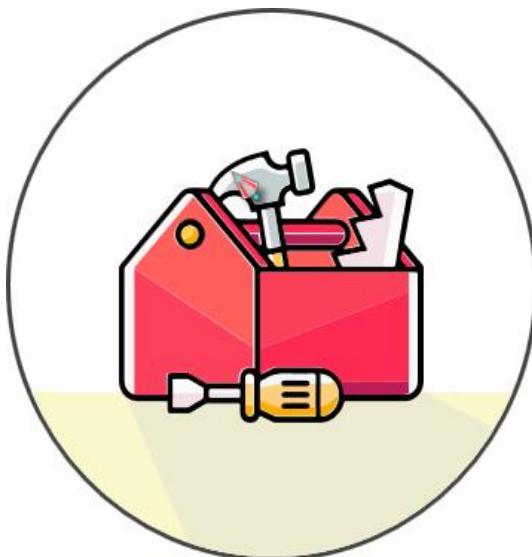
Les réseaux pair-à-pair utilisent Internet pour échanger des données. Il existe plusieurs types de réseaux P2P et chacun utilise un protocole de dialogue spécifique ; chaque réseau est donc

incompatible avec les autres. L'accès à ces réseaux se fait à l'aide d'un logiciel de partage de fichiers (ou *client P2P*). Ubuntu inclut de base des logiciels de P2P, et de nombreux autres sont facilement installables.

*Plus d'informations sur le pair-à-pair et les clients de partage de fichiers : <https://doc.ubuntu-fr.org/p2p>*

- Transmission (Ubuntu, Xubuntu)
- KTorrent (Kubuntu)

## ■ LES OUTILS



Sécurité dans les réseaux : <https://doc.ubuntu-fr.org/securite#reseaux>

Partages de connexion, d'imprimante, de fichiers, etc. : <https://doc.ubuntu-fr.org/partage>

Proxy : <https://doc.ubuntu-fr.org/proxy>

wakeonlan - Utiliser le Wake-On-Lan sur Linux

connexion\_internet\_via\_mobile\_bluetooth - avoir internet sur son ordinateur via son téléphone portable. ([https://doc.ubuntu-fr.org/tutoriel/connexion\\_internet\\_via\\_mobile\\_bluetooth](https://doc.ubuntu-fr.org/tutoriel/connexion_internet_via_mobile_bluetooth))

ZeroConf - auto configuration des réseaux grâce à zeroconf (<https://doc.ubuntu-fr.org/zeroconf>)

Autres :

- **Nagios** - Nagios, le moniteur de supervision : installation et configuration. (<https://doc.ubuntu-fr.org/nagios>)
- **W3C Markup Validator** - contrôler ses pages avec un validateur (x)html. (<https://doc.ubuntu-fr.org/w3c-markup-validator>)
- **Zenmap** : interface graphique destinée à la surveillance réseau (<https://nmap.org/zenmap/>)

## TELECHARGEMENT avec WGET

A la place de télécharger des fichiers de votre navigateur, vous pouvez juste copier et coller leur url et les télécharger avec **wget** !

Principales caractéristiques de wget :

- Supporte http et ftp
- Peut reprendre les téléchargements interrompus
- Peut télécharger des sites entiers ou au moins repérer les liens morts
- Très utile dans les scripts ou quand aucun graphique n'est disponible (Administration de système, système embarqué)
- Supporte les proxy (variables d'environnement http\_proxy et ftp\_proxy)

Exemple d'utilisation de wget :

```
#wget -c \ http://microsoft.com/customers/dogs/winxp4dogs.zip
```

Continue un téléchargement interrompu

```
#wget -m http://lwn.net/
```

Fait un site miroir

```
#wget -r -np http://www.xml.com/ldd/chapter/book/
```

Téléchargement récursif d'un livre en ligne pour les accès hors ligne.

**-np** : "no-parent". Suit uniquement les liens dans le répertoire courant.

## VERIFICATION DES FICHIERS avec MD5

Solution bon marché pour vérifier l'intégrité des fichiers

```
#md5sum FC3-i386-disk*.iso > MD5SUM
```

Calcule un checksum MD5 (Message Digest Algorithm 5) 128 bits d'un fichier donné. Généralement redirigé vers un fichier.

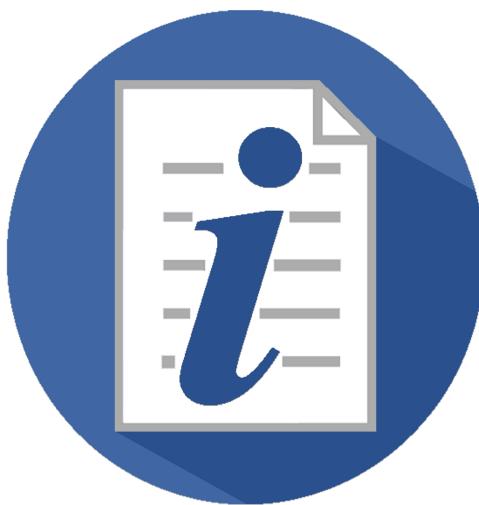
**Exemple de sortie :**

```
db8c7254beeb4f6b891d1ed3f689b412 FC3-i386-disc1.iso
2c11674cf429fe570445afd9d5ff564e FC3-i386-disc2.iso
f88f6ab5947ca41f3cf31db04487279b FC3-i386-disc3.iso
6331c00aa3e8c088cc365eeb7ef230ea FC3-i386-disc4.iso
```

```
#md5sum -c MD5SUM
```

Vérifie l'intégrité des fichiers MD5SUM en comparant leur somme de contrôle MD5 actuelle avec celle d'origine.

## ■ LEXIQUE



Il peut être utile de connaître les termes suivants :

- **DHCP** : C'est le protocole qui assigne automatiquement une adresse IP (locale) à une machine qui se connecte au réseau. Dans ce cas, une machine paramétrée pour récupérer son adresse IP en mode DHCP est la première à se connecter au réseau, elle récupère la première adresse IP disponible dans la plage d'attribution du système allouant ces adresses (modem routeur ou serveur), par exemple elle se verra attribuer l'adresse IP "192.168.0.1". Mais si votre machine est la seconde à se connecter au réseau elle pourra récupérer par exemple l'adresse IP "192.168.0.2". Cette

assignation aléatoire des adresses IP peut poser des problèmes si vous souhaitez partager une imprimante en réseau. En effet pour partager une imprimante en réseau il va falloir dire à la machine qui la cherche où elle se trouve EXACTEMENT sur le réseau. Or si cette adresse IP change vous comprenez que votre machine désirant imprimer ne pourra plus localiser l'imprimante en question.

*Remarque :*

À ce sujet il existe deux types d'adresse IP :

- Les adresses IP locales, qui sont de la forme : 192.168.X.X, 10.X.X.X ou 172.Y.X.X (avec X entre 0 et 255 et Y entre 16 et 31) dites non-routables et utilisables sur un réseau local,
  - Les adresses IP publiques dites routables qui sont toutes les autres.
- 
- **CUPS** : (*Common Unix Printer Service*) Ce que les utilisateurs d'UNIX utilisent pour partager une imprimante et en particulier Linux qui fait partie de la famille Unix. Dans la suite on suppose que CUPS est installé sur la machine serveur d'impression.
  - **Samba** : est le service de base pour partager des dossiers entre machines Linux et/ou avec des machines sous Windows ou encore d'autres systèmes d'exploitation. Cf. (<https://doc.ubuntu-fr.org/samba>)
  - **DNS** : (*Domain Name System*) C'est le mécanisme qui permet d'obtenir l'adresse IP (212.43.230.226 par exemple) correspondant à un nom de machine tel que **doc.ubuntu-fr.org**.
  - **Serveur DNS** : Ce sont les serveurs que votre FAI utilise pour faire cette conversion. Vous pouvez créer un serveur DNS si vous souhaitez donner un nom à votre machine au lieu de lui assigner une adresse IP.
  - **FAI** : Fournisseur d'Accès à Internet.

# Impression

Sous Unix / Linux, les commandes d'impression n'impriment pas vraiment. Elles envoient des tâches à des queues d'impression, soit sur la machine, locale, soit sur des serveurs d'impression ou sur des imprimantes réseau.

## Système robuste :

Redémarrez un système, il continuera à imprimer les travaux en attente.

## ■ COMMANDE D'IMPRESSION

Variable d'environnement utile : PRINTER

Définit l'imprimante par défaut sur le système.

Exemple : export PRINTER=lp

```
#lpr [-P<queue>] <fichiers>
```

Envoie les fichiers à la queue d'impression spécifiée. Les fichiers doivent être en format texte ou PostScript. Sinon, vous n'imprimerez que des déchets.

```
#a2ps [-P<queue>] <fichiers>
```

“Any to PostScript” convertit de nombreux formats vers PostScript et l'envoie le résultat vers la queue spécifiée. Fonctionnalités utiles : plusieurs pages / feuille, numérotation des pages, cadre d'informations.

## ■ CONTRÔLE DE TRAVAUX D'IMPRESSION

```
#lpq [-P<queue>]
```

Affiche tous les travaux d'impression de la queue par défaut ou de la queue donnée

lp is not ready			
Rank	Owner	Job	File(s)
1st	asloane	84	nsa_windows_backdoors.ps
2nd	amoore	85	gw_bush_iraq_mistakes.ps

```
#cancel <job#> [<queue>]
```

Retire la tâche spécifiée de la queue d'impression

# Grep & Find

« L'UN DES OUTILS DE RECHERCHES AVEC UN ALGORITHME PLUS PERFORMANT QUE JAMAIS. »

## [[GREP]] \_ Recherche de contenu

---

Grep est un **outil de recherche** de contenu d'un fichier.

### ■ LES DIFFERENTS EXPRESSIONS RATIONNELS de GREP utiles :

- Trouver les lignes qui commence par [qlq\_chose] : `#grep '^[qlq_chose]" file`
- Trouver les lignes qui contient exactement [qlq\_chose] : `#grep '^([qlq_chose])$" file` ou en ajoutant l'option **-iw**
- Trouver les lignes (entières) qui contient [qlq\_chose] : `#grep "[qlq_chose]" file`
- Trouver les lignes (entières) qui ne contient pas [qlq\_chose] : `#grep -iv "[qlq_chose]" file`
- L'option **-c** (count) permet à grep de retourner un nombre (total) de ligne en occurrence avec [qlq\_chose].
- L'option **-n** permet à grep de retourner que la position (nombre) de la ligne en occurrence avec [qlq\_chose].

**P.S :** A caractère générique, on peut mentionner à grep une **recherche multiple** en ajoutant à la fin de commande le symbole "\*" (étoile) (ex : `#grep -iwr "[qlq_chose]" *`), **-r** permet également de réaliser une **recherche récursive**.

## [[FIND]] \_ Recherche d'un fichier

---

On peut faire une recherche avec différente expression de critère avec Find.



### ■ SYNTAX :

-**a** recherche par rapport au temps.

-**c** recherche par rapport au statut.

-**size** recherche par rapport à la taille, on peut ajouter un singe **+/-[n]** pour une recherche plus précise.

-**name** recherche par rapport au nom.

[**n**] entier.

[**unit**] k(kilo) / b(bloc) / c(octet).

### ■ LES DIFFERENTS EXPRESSIONS RATIONNELS de FIND utiles :

> Trouver un/tous les fichier(s) qui ont été accéder depuis [n] jour : #find [répertoire] -atime [n] -print

> Trouver un/tous les fichier(s) qui ont été accéder depuis [n] minute : #find [répertoire] -amin [n] -print

> Trouver un/tous les fichier(s) qui comportent [n][unit] taille : #find [répertoire] -size [n][unit] -print

> Trouver un/tous les fichier(s) qui ont une taille vide : #find [répertoire] -empty -print

> Trouver un/tous les fichier(s) qui se nomment [name] : #find [répertoire] -name [name]\* -print, souvent accompagné du symbole "\*" pour une recherche par occurrence

**N.B.** find est plus adapté aux recherches sur les fichiers créés récemment.

## Alternative de la commande find avec locate

---

Outil de recherche à base d'expressions régulières, une alternative à find beaucoup plus rapide.

- Affiche tous les fichiers sur votre système contenant clé dans leur nom : `#locate clé`
- Affiche tous les fichiers \*.pdf existant sur votre système : `#locate "*.pdf"`
- Affiche tous les fichiers \*mousse\* dans le répertoire indiqué (chemin absolu) :  
`#locate "/home/frigo/*mousse*"`

**N.B.** locate est bien plus rapide grâce à l'indexation de tous les fichiers dans une base de données dédiée, qui est mise à jour régulièrement.

# Redirection

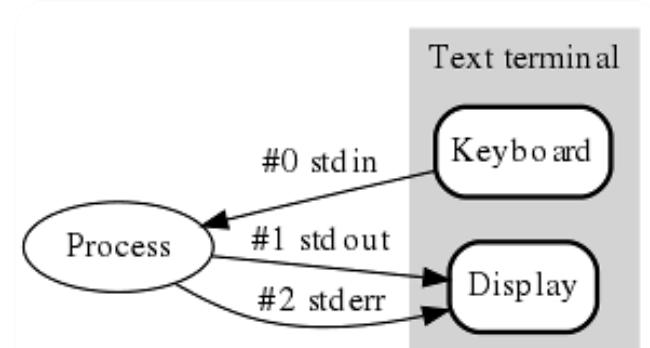
Il est parfois utile de conserver le résultat d'une commande, par exemple le résultat de **dmidecode** qui répertorie votre matériel est un texte relativement long. vous pouvez, bien entendu, copier le texte, créer un fichier et y coller ce que vous venez de copier. il existe pourtant une méthode qui vous fera gagner un peu de temps, **la redirection**.

En tapant : # [command] > file (ex : #dmidecode > listing\_matériel).

## Sortie standard

Toutes les commandes qui sortent du texte sur votre terminal le font en écrivant sur leur sortie standard.

La sortie standard peut être écrite (redirigée) dans un fichier en utilisant le symbole >



## L'erreur standard

Les messages d'erreur sont normalement envoyés (si le programme est bien écrit) vers l'erreur standard au lieu de la sortie standard.

L'erreur standard peut être redirigée par 2> ou 2>> (ex : #cat f1 f2 > file0 2> file\_err).

- 
- ✓ Il existe ce qu'on appelle la redirection inverse, utilisable avec certaines commandes. Prenons un exemple :

Vous venez de créer un fichier disques.txt qui contient, dans le désordre, la liste des centaines de CD que comporte votre discothèque. Vous aimeriez bien trier cette liste par ordre alphabétique.

- ✓ Vous avez de la chance ! La commande sort réalise cela. Il vous suffira de faire :

```
#sort < disques.txt
```

- ✓ 1 est le descripteur de la sortie standard, donc 1> est équivalent à >

- ✓ On peut rediriger à la fois la sortie et l'erreur standard vers le même fichier en utilisant `&>` (ex : `#cat f1 f2 &> all_file`).
- ✓ On peut également envoyer un texte dans un fichier en écrasant son contenu.

En tapant: `#echo Hello world ! > text.txt`

\* Pour "imprimer" le contenu des plusieurs fichiers dans un seul fichier : `#cat file0 file1 file2 > file_all`

### ✓ REDIRECTION 2.0 !

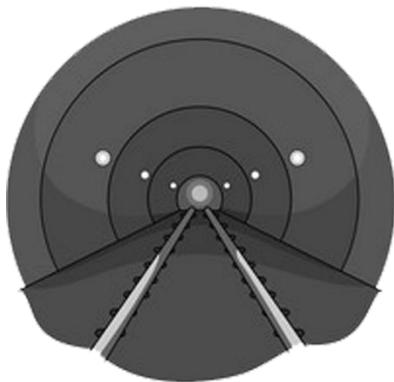
**Double flèche (>>)** : permet à la redirection de ne pas écraser les anciens contenus du fichier mais de juste ajouter la sortie de commande.

---

# Pipes

« LE MARIAGE D'UNE COMMANDE. »

Les **pipes** sont une sorte de **redirection** sur lequel il prend le **std-out** d'une commande vers le **std-in** d'une autre via un tunnel (ex : `#ls | sort -r`) ce qui veut dire littéralement lister le répertoire courant en triant par ordre alphabétique et inversement (descendant).



- ◆ **STD-IN** (standard-input) : l'entrée d'un standard... signifie envoyer qlq chose dans l'entrée (la commande en générale).
- ◆ **STD-OUT** (standard-output) : sortie d'un standard... les résultats d'une commande.

- 
- ✓ La commande "`#xargs`" sert à convertir les std-out en une suite d'argument pour les std-in d'une autre commande après l'utilisation d'un pipe (ex : `#find ~ -type f ! -name '.*' -print0 | xargs -0 grep -il 'term' | xargs rm -r`) i-e, rechercher tous les fichiers dont le nom ne commence pas par un point "." (les fichiers cachés) et contient le caractère "term" et puis les supprimer récursivement si ils existent.
-

# RSYNC

## « COPIE INTELLIGENTE AVEC RSYNC. »

**rsync** («remote sync»: sync. à distance) a été conçu pour synchroniser des répertoires sur 2 machines reliées par un lien à faible débit.

- o Ne copie que les fichiers qui ont changé. Les fichiers de taille identique sont comparés au moyen de sommes de contrôle.
- o Ne transfère que les blocs qui diffèrent au sein d'un fichier !
- o Peut compresser les blocs transférés.
- o Conserve les liens symboliques et les permissions sur les fichiers : également très pratique pour les copies sur la même machine.
- o Peut fonctionner à travers ssh (shell sécurisé). Très pratique pour mettre à jour le contenu d'un site Internet, par exemple.

### ■ EXEMPLE D'UTILISATION :

```
#rsync -a /home/arvin/agents_sd6/ /home/sydney/vrac/
```

**-a** : mode archive. Équivalent à -rlptgoD... Un moyen facile de dire que vous voulez de la récursion et souhaitez préserver presque tout.

```
#rsync -Pav --delete /home/steve/idées/ /home/bill/mes_idées/
```

**-P** : --partial (garder les fichiers partiellement transférés) et  
--progress (afficher la progression du transfert)  
--delete : effacer les fichiers à l'arrivée qui n'existent plus à la source.

**Attention** : les noms de répertoires doivent finir par /. Sinon, vous obtenez un répertoire mes\_idées/idées/ à la destination.

### ■ COPIE VERS UNE MACHINE DISTANTE :

```
#rsync -Pav /home/bill/legal/arguments/ \bill@www.sco.com:/home/legal/arguments/
```

Un mot de passe sera demandé à l'utilisateur bill.

## ■ COPIE DEPUIS UNE MACHINE DISTANTE A TRAVERS SSH :

```
#rsync -Pav -e ssh
homer@cuve.duff.com:/prod/bière/ \frigo/homer/bière/
```

On demandera à l'utilisateur homer le mot de passe de sa clé ssh.

# Archivage & compression

« STOCKER VOS DONNEES GRACE A TAR.GZ. »

## ■ Archivage

La commande "#tar" permet d'**archiver** ou **désarchiver** un ou plusieurs fichiers avec ou sans **algorithme de compression** (c'est à vous de choisir).

## ■ LES DIFFERENTES OPTIONS pour TAR :



**-C** pour créer l'archive.

**-X** pour extraire une archive.

**-Z** pour utiliser l'algorithme de compression gzip.

**-V** pour obtenir un résultat visuel.

**-f** pour dire dans quel fichier on va créer l'archive.

**-t** pour visualiser le contenu d'une archive sans le décompresser.

## ■ SYNTAX :

◆ Pour créer une archive avec compression gzip.

```
#tar -czvf [rép.à_convertir] [rep.archive_name].tar.gz
```

◆ Pour afficher juste le contenu d'une archive sans le décompressé.

```
#tar -tzvf [rep.archive_name].tar.gz
```

◆ Pour extraire une archive.

```
#tar -xzvf [rep.archive_name].tar.gz [rép. où décompressé]
```

## ■ Compression

Très utile pour **compacter** de gros fichiers et **économiser** de la place.

```
#[un]compress <fichier>
```

Utilitaire de compression traditionnel d'Unix. Crée des fichiers.Z.

Seulement gardé pour raisons de compatibilité. Performance moyenne.

```
#g[un]zip <fichier>
```

Utilitaire de compression GNU zip. Crée des fichiers .gz. Assez bonne performance (semblable à celle de Zip)

```
#b[un]zip2 <fichier>
```

Le plus récent et le plus performant des utilitaires de compression. Crée des fichiers .bz2. La plupart du temps 20-25% meilleur que gzip. Utilisez celui-ci

! Maintenant disponible sur tous les systèmes Unix.

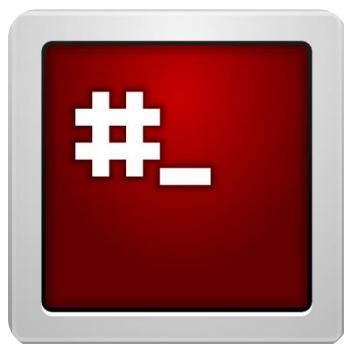
#LE\_SUPER-UTILISATEUR  
#PERMISSION & PROPRIETAIRE  
#GESTIONS DES COMPTES UTILISATEURS  
#GESTIONS DES GROUPES UTILISATEURS

« Prenez le contrôle de votre système GNU/Linux »

# Le super utilisateur

« ROOT »

➤ "#su" (avec le mot de passe root) vous permet de devenir un "super user" avec un **prompt** de type : "**root@machine : ~#**", comme son nom l'indique, **on possède tous les droits** sur tout le système de fichier et peut faire tout ce qu'on veut dans le bash pour **contrôler l'intégralité du noyau linux**. La plupart des distributions récentes désactivent le compte « root » pour une mesure de sécurité.



➤ Par contre, "#sudo" ou **Super User DO** (avec le mot de passe personnel de l'utilisateur) permet de lancer une commande administrative comme "#chmod" ou "#chown" en restant dans son compte personnel (à condition que l'utilisateur est dans le groupe appelé "sudoers") avec un **prompt** de type : "**user@machine : ~\$**". C'est l'intérêt de l'utilisation de cette commande pour éviter des catastrophes. Pour éviter la reconnexion automatique (dans un lapse de temps), on peut vider le cache qui enregistre le mot de passe en ajoutant l'option -k (ex : #sudo -k).

**ATTENTION ! :** Vous pouvez quand même devenir un administrateur avec votre mot de passe personnel en utilisant la commande "#sudo su", qui signifie littéralement "exécuter la commande #su en tant que membre du groupe sudoers".

> En règle générale, on utilise « sudo » que temporairement :

- Pour des tâches administratives (exécuter des commandes qui demande des autorisations root).
- Pour accéder à l'ensemble du système de fichiers.
- Pour gérer les utilisateurs.



> En étant un super-utilisateur, on peut usurper l'identité d'un utilisateur classique :

Quand un super-utilisateur crée un fichier, ce fichier lui appartient et selon les droits qui s'appliquent, les autres utilisateurs ne peuvent pas y accéder. Se faire momentanément passer pour un autre utilisateur est alors nécessaire pour que l'utilisateur lui-même devient le propriétaire du fichier.

```
root@ubuntu:~# su gen Je me fais passer pour « gen »
gen@ubuntu:~$ touch toto.txt
gen@ubuntu:~$ ls -l Le fichier créé appartient à « gen »
total 32
drwxr-xr-x 2 gen gen 4096 2009-08-22 14:49 Modeles
-rw-r--r-- 1 gen gen 0 2009-08-31 05:47 toto.txt
drwxr-xr-x 2 gen gen 4096 2009-08-22 14:49 vidéos
gen@ubuntu:~$ exit Je reprends mon rôle initial
exit
root@ubuntu:~#
```

# Permission et propriétaire

« REPRENEZ LE CONTRÔLE GRÂCE A CHMOD, CHOWN ET UMASK. »



GNU/Linux est un **système multi-utilisateur**, les fichiers et répertoires d'un utilisateur ne doivent pas être accessibles par les autres, les fichiers de configuration du système doivent être protégés. En exécutant la commande "#ls" dans un répertoire, on peut montrer les différentes informations sur son contenu, y compris les permissions et les propriétaires du fichier.

```
root@fredon:/home/paul/Documents# ls -l
total 20
-rw-r--r-- 1 paul paul 0 2008-08-15 14:42 projet.txt
-rw-rw-r-x 1 paul comptta 7406 2008-08-15 14:44 rapport2006.ods
-rw-rw-r-- 1 paul paul 7363 2008-08-15 14:44 rapport-activite.odt
-rw-rwxr-x 1 paul comptta 255 2008-08-15 14:52 script.sh
```

## ■ LES 4 TYPES D'ACCÈS

r w x

- pas d'autorisation.

✓ Un **fichier/répertoire** peut prendre les trois types d'autorisation [---] ou [--x] ou [-w-] ou [-wx] ou [r--] ou [r-x] ou [rw-] ou [rwx] (**C'est un système octal**).

✓ Les droits sont responsables d'un grand nombre d'erreurs de configuration.

✓ Ces droits n'ont pas la même signification pour **un fichier** que pour **un répertoire**.

○ *Pour un fichier :*

- r lecture (afficher).*
- w écriture (modification).*
- x exécution (exécution d'un script).*

○ *Pour un répertoire :*

- r lire le contenu (lister les fichiers).*
- w modifier le contenu (créer et supprimer des fichiers).*
- x accéder aux fichiers (répertoire ou sous-répertoires).*



- ✓ Le droit « **w** » accordé à un **répertoires** permet :
    - *D'y effacer des fichiers quels que soient le propriétaire et les droits qui s'appliquent à ces fichiers.*
    - *Quand il est donné à un groupe, n'importe quel utilisateur de ce groupe peut supprimer des fichiers (dangereux).*
  
  - ✓ Le droit « **x** » accordé à un répertoire est un préalable indispensable pour exercer des droits sur les **fichiers** contenus.
  
  - ✓ L'utilisateur qui crée un fichier en **devient le propriétaire**.
  
  - ✓ Les droits ne s'appliquent pas au « **super utilisateur** » :
    - *Il a tous les droits sur tout le système de fichiers.*
    - *C'est une très grande responsabilité puisque sous **GNU/Linux** tout est presque repose sur les fichiers.*
- 

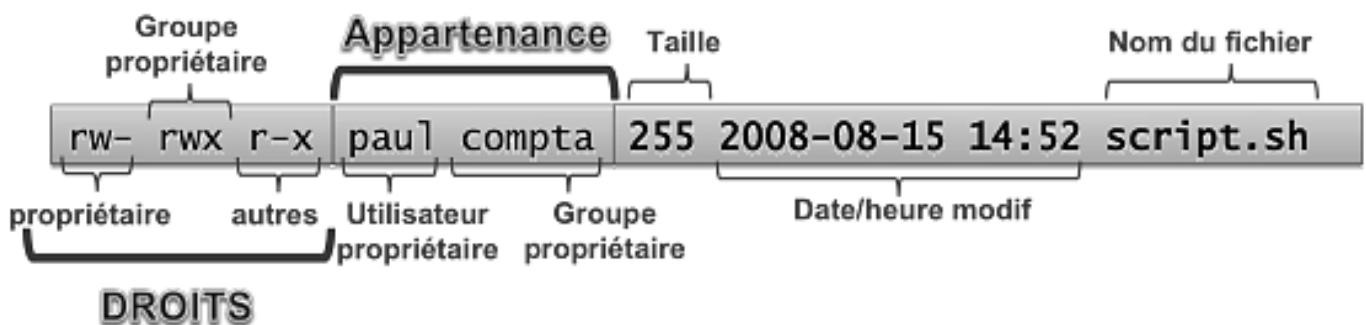
## ■ LES 3 GROUPES D'UTILISATEURS

**USER** le propriétaire du fichier/répertoire.

**GROUP** le groupe propriétaire (tous les utilisateurs membre du groupe).

**OTHERS** les autres qui désigne tous les utilisateurs non membre des 2 précédents.

## ■ LE SCHEMA GENERAL



Propriétaire : Lecture, écriture

Groupe : Lecture, écriture et exécution

Autres : Lecture et exécution

## ■ MANIPULATION avec CHMOD, CHOWN et UMASK

### [[CHMOD]] – Permission

En étant utilisateur **root** (admin), on peut modifier le **droit d'accès** sur un “**file/doc**” en tapant la commande suivante :

```
#chmod [option_spécifique] file/doc
```

#### >> LES 2 MODES DE CHMOD

Avec **[variable]** = r, w, x

- Mode symbolique :

Ce mode est basé sur les symboles (**u, g, o** et **a**) et des opérateurs (**+, -** et **=**)

**u** (user), **g** (group), **o** (others), **a** (all users)

**+** (ajouter le droit), **-** (enlever un droit). **=** (ajouter le droit et retirer tous les autres)

> **a+/-[variable]** pour changer les droits d'accès d'un file/doc sur tout les utilisateurs, i-e les trois positions (all).

> **u+/-[variable]** pour changer les droits d'accès d'un file/doc que sur le propriétaire.

> **g+/-[variable]** pour changer les droits d'accès d'un file/doc que sur le groupe propriétaire.

> **o+/-[variable]** pour changer les droits d'accès d'un file/doc que sur les autres utilisateurs.

Exemple :

```
#chmod a-r file0
#chmod u+rwx file1
#chmod g+rw file2
```

- Mode octal :

Basé sur les nombres octaux de **0** à **7** et chaque bit de la traduction binaire correspond un droit

OCTAL (base 8)	BINAIRE (base 2)
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Exemple :

```
#chmod 664 file0 (rw- rw- r--)
```

6 = 110 = rw-

5 = 101 = r-w

4 = 100 = r--

**Propriétaire :** Lecture, écriture

**Groupe :** Lecture et exécution

**Autres :** Lecture seulement

- 
- ✓ **Attention !** ce mode permet de modifier tous les droits en même temps, à utiliser avec précaution.
  - ✓ **Très efficace !** pour s'assurer que tous les fichiers ont les mêmes droits.
  - ✓ Utilisé pour **sécuriser** les accès des utilisateurs aux fichiers.
- 



## >> LES DROITS ETENDUS SUID, SGID et Sticky Bit

### [SUID] \_ Set User ID

- > Dans le cadre des permissions UNIX, Setuid ou SUID abréviation de « Set User ID », est un moyen de transférer des droits **aux utilisateurs**. En effet, permet de bénéficier de droits supplémentaires lors de l'exécution d'une commande. Un utilisateur quelconque peut alors avoir des droits supplémentaires seulement s'il execute la commande ayant le SUID.

PRINCIPE D'UTILISATION	Pour ajouter un droit <b>SUID</b> à un programme, utiliser la commande « <b>chmod</b> » de la même manière que vous procéderiez pour un droit normal.
NOTATION SYMBOLIQUE	Il est symbolisé par la lettre <b>s</b> et est affiché à la place du droit d'exécution de l'utilisateur. ➤ <b>Syntaxe :</b> #chmod u+s file ➤ <b>Exemple :</b> #chmod u+s data/hotprog
NOTATION NUMERIQUE	La valeur numérique d'un <b>SUID</b> est <b>4000</b> . ➤ <b>Syntaxe :</b> #chmod 4+[droit_octal] file ➤ <b>Exemple :</b> #chmod 4755 monprog (donne les droits rws r-x r-x au fichier monprog). ➤ <b>N.B :</b> la présence du droit <b>SUID</b> suppose la présence du droit en exécution qui permet de lancer le fichier exécutable.

- > Exemple de la commande « **passwd** » qui permet de modifier son mot de passe, « **passwd** » doit alors écrire dans le fichier **/etc/shadow** et pourtant :

```
linux:~# ls -l /etc/shadow
-rw-r----- 1 root shadow 700 2007-12-04 18:39 /etc/shadow
```

Aucune permission d'écriture sur ce fichier

```
linux:~# ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 28480 2007-02-27 08:53 /usr/bin/passwd
```

La commande aura les droits du super-utilisateur même si n'importe quel autre utilisateur lance son exécution

- > En clair, un utilisateur de type **/home/toto** pourrait lui-même redéfinir son mot de passe une fois loggé.

➤ **Syntaxe :** #passwd [mot\_de\_passe].

## [SGID] – Set Group ID

► Identique à celle du SUID mais appliquée **au groupe propriétaire**. La commande obtiendra les droits du groupe propriétaire si elle est exécutée par un autre utilisateur.

✓ Attention ! appliqué à un répertoire, le SGID :

- Modifie le groupe propriétaire d'un fichier créé dans le répertoire.
- Ce ne sera plus le groupe primaire du propriétaire mais plutôt le groupe propriétaire du répertoire.
- Il y a donc un mécanisme d'héritage entre le répertoire et les fichiers nouvellement créés qu'il contient.

<b>PRINCIPE D'UTILISATION</b>	Pour ajouter un droit <b>SGID</b> à un programme/répertoire, utiliser la commande « <b>chmod</b> » de la même manière que vous procéderiez pour un droit normal.
<b>NOTATION SYMBOLIQUE</b>	Il est symbolisé par la lettre <b>s</b> et est affiché à la place du droit d'exécution du <b>groupe</b> . ➤ <b>Syntaxe</b> : #chmod g+s file/répertoire ➤ <b>Exemple</b> : #chmod g+s data/
<b>NOTATION NUMERIQUE</b>	La valeur numérique du droit <b>SGID</b> est <b>2000</b> . ➤ <b>Syntaxe</b> : #chmod 2+[droit_octal] file/répertoire ➤ <b>Exemple</b> : #chmod 2755 monprog (donne les droits rwx r-s r-x au fichier monprog). ➤ <b>N.B</b> : Ne pas oublier que le droit <b>SGID</b> s'affiche à la place du droit en exécution du groupe sans que ce dernier soit supprimé !

► Voici un exemple :

drwxrws--- 2 root compt 4096 2008-08-24 13:05 docs-compta

SGID positionné sur « docs-compta »

-rw-r--r-- 1 paul compt 0 2008-08-24 13:09 nouveau.txt

Le fichier nouvellement créé par paul appartient au groupe « compt »

## [Sticky Bit] \_ bit collant

➤ Le droit « Sticky Bit » est alloué à la catégorie autres d'un répertoire. Il permet d'interdire à tout utilisateur (sauf le root) de **supprimer** un fichier dont il n'est pas le propriétaire, quelque soient ses droits. Ce bit collant permet donc d'aller à l'encontre du droit en écriture d'un répertoire dont héritent les fichiers du répertoire.

PRINCIPE D'UTILISATION	Pour ajouter un droit <b>Sticky Bit</b> à un programme/répertoire, utiliser la commande « <b>chmod</b> » de la même manière que vous procéderiez pour un droit normal.
NOTATION SYMBOLIQUE	Il est symbolisé par la lettre <b>t</b> et est affiché à la place du droit d'exécution de la catégorie des <b>autres</b> .  ➤ <b>Syntaxe :</b> #chmod a+s file/répertoire ➤ <b>Exemple :</b> #chmod a+s tmp/
NOTATION NUMERIQUE	La valeur numérique du droit <b>Sticky Bit</b> est <b>1000</b> .  ➤ <b>Syntaxe :</b> #chmod 1+[droit_octal] file/répertoire ➤ <b>Exemple :</b> #chmod 1755 tmp/ (donne les droits rwx r-x r-t au répertoire /temp). ➤ <b>N.B :</b> Ce droit s'affiche en lieu et place du droit en exécution de la catégorie <b>autres</b> .

## [[UMASK]] \_ Le masque de protection

---

Le masque de protection est le fichier qui permet de définir les droits par défaut de tout fichier créé.

### > Principe de « umask » :

Ce masque comporte comme un filtre et utilise la notation binaire. On parle de filtre car il ne contient pas la série des 3 chiffres octaux correspondants aux droits à allouer aux fichiers, mais celle correspondant aux droits à ne pas allouer (complémentation de la vraie valeur en binaire).

Le système **UNIX** affecte à un fichier les droits globaux résultant de la soustraction des droits maximaux 777 par le masque de protection.

- **Exemple :** Si le masque de protection vaut **037** alors **740** (=777-037) seront les droits alloués à tout nouveau fichier.
- **Syntaxe :** #umask [droit\_octal] (ex : #umask 037)

777 = rwx rwx rwx = 111 111 111  
- 037 = --- -wx rwx = 000 011 111  
= 740 = rwx r-- --- = 111 100 000

- D'après cet exemple, tout nouveau fichier aura les droits **740 (rwx r-- ---)** car le masque de protection vaudra **370 (--- -wx rwx)**.
- Pour connaître la valeur du masque de protection, tapez « **umask** » sans attribut.

## [[CHOWN]] \_ Propriétaire

---

CHOWN comme « CHange OWNer ». En étant utilisateur **root** (admin), on peut modifier le propriétaire/groupe\_propriétaire d'un **file/doc** (à condition que ce dernier existe) en tapant la commande suivante :

- **Syntaxe\_1 :** #chown [propriétaire\_name : group\_propriétaire\_name] file/doc (ex : #chown jean:direction file0)
- **Syntaxe\_2 :** #chown [propriétaire\_name] file/doc (ex : #chown paul file0)
- **Syntaxe\_3 :** #chown :[group\_propriétaire\_name] file/doc (ex : #chown :compta file0)

**P.S :** On peut également ajouter une option récursive **-R** qui permet de changer tous les permissions/propriétaires de l'arborescence d'un répertoire/sous-répertoires (ex : #chmod -R a+rwx doc0 ou #chown -R user1.grp1 file0).

# Gestions des comptes utilisateurs



➤ **Créer un compte pour un nouvel utilisateur** signifie lui permettre d'être connu du poste local, s'y loguer, voir un accès complet sur son rép. Personnel. Mais aussi dans une configuration réseau, de réseau, de pouvoir se connecter à son compte par telnet et ftp, De pouvoir bénéficier de services réseau de partage distant (sous Linux par NFS et sous Windows 9x par SMB).

## Création de l'utilisateur

Tout d'abord, il faut se passer pour un super-utilisateur pour avoir le droit de créer un autre utilisateur. Pour ce faire, on utilise la commande « **useradd** »

- **Méthode n°1 :**

```
#useradd [user_name]
```

Ceci crée : le répertoire personnel **/home/paul**, pourtant par défaut le nom du compte une nouvelle entrée dans les 2 fichiers fondamentaux **/etc/passwd** et **/etc/group**.

Pour connaître les options de useradd (indispensable pour gérer les comptes à l'aide de scripts) faire un **#man useradd**.

- **Méthode n°2 :**

```
#useradd -d /home/paul -s /bin/bash -m paul
```

### Description des options

- d <dossier> : spécifie le dossier de l'utilisateur (son **\$HOME**).
- s <shell> : spécifie le shell qui sera utilisé au démarrage d'une session en console.
- m : crée-le **\$HOME** de l'utilisateur et y met les fichiers de configuration du shell de l'utilisateur.

## Création de mot de passe pour un utilisateur

```
#passwd [user_name] (puis entrer le password de paul à définir).
```

## Suppression d'un compte de l'utilisateur (non connecté)

**#userdel -r [user\_name]** (l'option **-r** supprime aussi le rép. Personnel et les fichiers de l'utilisateur, autrement dit, elle supprime toute trace de l'utilisateur dans le fichier de configuration : **etc/passwd** y compris dans les groupes d'utilisateurs).

## Modification d'un compte de l'utilisateur

**#usermod [option][user\_name]**

### Description des options

- l <nouveau\_login> <ancien\_login> : permet de changer le login d'un utilisateur.
- s <nouveau\_shell> <login> : pour changer le shell par défaut de l'utilisateur.
- u <nouveau\_uid> <login> : pour changer l'uid de l'utilisateur, attention : les permissions ne seront pas modifiées).
- d <nouveau\_dossier> <login> : pour changer le dossier personnel de l'utilisateur).
- m -d <nouveau\_dossier> <login> : pour changer le dossier, y copie le contenu de l'ancien et ajuste les permissions).

**N.B :** « **#usermod** » sans argument permet de voir la liste des options possibles.

## Autres commandes diverses sur les informations des utilisateurs

**#who** Indique tous les utilisateurs connectés au système.

**#whoami** Indique en tant que quel utilisateur je suis connecté.

**#finger [user\_name]** Fournit des détails (nom réel, etc) au sujet de <user\_name>

Désactivé sur certains systèmes (raisons de sécurité).

# Gestions des groupes utilisateurs



Un groupe est, aussi pour **GNU/Linux**, un ensemble d'utilisateur qui partagent les mêmes fichiers et répertoires. Nous verrons que les fichiers accordent des droits d'accès réglables à ces groupes.

Chaque utilisateur doit faire partie **au moins d'un groupe d'un groupe**, son **groupe primaire**. Celui-ci est défini au moment de la création du compte, et par défaut, l'utilisateur appartient à un nouveau groupe créé, **portant son nom**.

Ainsi, dans **/etc/passwd** chaque utilisateur possède un groupe par défaut, précisé par son identifiant GID dans ce fichier

L'appartenance au groupe primaire n'étant pas exclusive, tout utilisateur **peut faire partie de plusieurs autres groupes**, appelés ses **groupes secondaires**. Mais le rôle joué par le groupe primaire demeure prépondérant, comme nous le verrons dans le système de permissions des fichiers.

## Ajout d'un utilisateur à un groupe

Pour ajouter [user\_name] au groupe [group\_name], on va lancer la commande :

```
#adduser [user_name] [group_name]
```

Ou :

Editer le fichier **/etc/group** et ajouter une liste d'utilisateurs (séparés par des virgules) sur la ligne du groupe (ou utiliser Linuxconf).

## Lister tous les groupes ( primaire et secondaires d'un utilisateur)

```
#groups [user_name]
```

## Création d'un nouveau groupe

```
#groupadd [group_name]
```

## Suppression d'un groupe

```
#groupdel [group_name] (Le groupe est supprimé du fichier /etc/group).
```

## >> A RETENIR !

- ✓ Tout ce qui concerne la **gestion et l'authentification des utilisateurs** est inscrit dans un seul fichier **/etc/passwd**

Structure de **/etc/passwd** : (user\_name = toto)

toto:	x:	1001 :	1001 :	commentaire	/home/toto	/bin/bash	shell pour toto
<b>Le répertoire courant de toto</b>							
<b>(Optionnelle)</b> : ici on peut ajouter un commentaire concernant l'utilisateur toto							
numéro de groupe gid <u>exemple</u> : par défaut le gid de root est 0							
numéro d'utilisateur uid, sa valeur est le véritable identifiant pour le système Linux; <u>exemple</u> : par défaut l'uid de root est aussi 0							
mots de passe							
nom de connexion							

- ✓ La gestion des **groupes** est assurée par **/etc/group**

Structure de **/etc/group** : (user\_name = toto)

toto:	x:	1001 :	toto
<b>nom de l'utilisateur1 qui est membre de ce groupe toto</b>			
numéro de groupe gid <u>exemple</u> : par défaut le gid de root est 0			
mots de passe			
nom du group			

- ✓ Les mots de passe cryptés sont maintenant placés dans /etc/shadow (par sécurité, lisible seulement par root)

Structure de /etc/shadow : (user\_name = toto)

toto:	<b>QjiKLM7ie:</b>	<b>10795 :</b>	<b>0 :</b>	<b>99999 :</b>	<b>7 :</b>	<b>-1 :</b>	<b>-1 :</b>	<b>134537220 :</b>
est un indicateur réservé à un usage ultérieur.								
est le nombre de jours entre le 1er janvier 1970 et la date de désactivation du compte .								
est le nombre de jours après quoi le mot de passe expire de sorte que le compte est désactivé. -1 est utilisé pour indiquer un nombre infini de jours								
nombre de jours pour lequel un utilisateur est averti que son mot de passe expirera.								
correspond au nombre de jours après quoi le mot de passe doit être modifié. Ce champ est rarement utilisé. Par défaut, sa valeur est 99999								
c'est le nombre de jours avant que le mot de passe ne puisse être changé. Usuellement, la valeur est = 0. Ce champ n'est pas souvent utilisé								
c'est le nombre de jours entre le 1er janvier 1970 et la date de dernier changement du mot de passe.								
<b>mots de passe crypté</b>								
<b>nom de connexion</b>								

#### Autres commandes diverses sur les informations des groupes

#groups Indique à quels groupes j'appartiens.

#groups [user\_name] Indique à quels groupes <user\_name> appartient.

#PROCESSUS #VARIABLES\_D'ENVIRONNEMENT  
#SYSLOG & #GLOBBING\_BASH #TRUX #EXTRA

« Ce que vous devez savoir autrement »

# Processus

« TOUT DANS UNIX QUI N'EST PAS UN FICHIER EST UN PROCESSUS. »



Depuis le début, Unix prend en charge le vrai multitâche préemptif. Faculté de lancer de nombreuses tâches en parallèle, et de les interrompre même si elles ont corrompu leur propre état ou leur propre donnée.  
Faculté de choisir quels programmes précis vous lancez.  
Faculté de choisir les entrées utilisées par vos progs : rammes, et de choisir où vont leurs sorties.

## Processus

C'est une instance d'un programme en cours d'exécution dont plusieurs instances d'un même programme peuvent s'exécuter en même temps.

## Données associées aux processus

Ouvrir un fichier, mémoire allouée, pile, id processus, parent, priorité, état...

## Programmes en tâche de fond

Même mode d'utilisation dans tous les shells, particulièrement utile :

- Pour les tâches en ligne de commande dont les résultats peuvent être examinés plus tard, en particulier celles qui prennent beaucoup de temps.
- Pour lancer des applications graphiques depuis la ligne de commande et les utiliser ensuite à la souris.

Démarrer une tâche : ajouter & au bout de votre ligne

trouver\_prince\_charmant beau intelligent riche &

## Parfaite maîtrise des tâches grâce à #PS, #TOP et #KILL

### [PS] \_ Liste de tous les processus

#ps ux

Affiche tous les processus appartenant à l'utilisateur courant.

#ps aux (remarque : #ps edf sur systèmes System V)

Affiche tous les processus existant sur le système

```
ps -aux | grep bart | grep bash
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
bart 3039 0.0 0.2 5916 1380 pts/2 S 14:35 0:00 /bin/bash
bart 3134 0.0 0.2 5388 1380 pts/3 S 14:36 0:00 /bin/bash
bart 3190 0.0 0.2 6368 1360 pts/4 S 14:37 0:00 /bin/bash
bart 3416 0.0 0.0 0 0 pts/2 RW 15:07 0:00 [bash]
```

PID (Process ID)	Identifiant du processus.
VSZ (Virtual SiZe)	Taille virtuelle du processus (code + données + pile).
RSS (ReSident Size)	Nombre de Ko occupés en mémoire.
TTY (TeleTYpe)	Terminal.
STAT	<u>Statut :</u> <ul style="list-style-type: none"><li>▪ R (Runnable: exécutable)</li><li>▪ S (Sleep: endormi)</li><li>▪ W (paging en cours de pagination)</li><li>▪ Z (Zombie)...</li></ul>

## [TOP] \_ Activité en temps réel des processus

#top

Affiche les processus les plus actifs, triés par utilisation du proc.

L'ordre de tri peut être changé en tapant

- M : utilisation Mémoire

- P : %CPU

- T : Temps d'exécution.

```
top - 15:44:33 up 1:11, 5 users, load average: 0.98, 0.61, 0.59
Tasks: 81 total, 5 running, 76 sleeping, 0 stopped, 0 zombie
Cpu(s): 92.7% us, 5.3% sy, 0.0% ni, 0.0% id, 1.7% wa, 0.3% hi, 0.0% si
Mem: 515344k total, 512384k used, 2960k free, 20464k buffers
Swap: 1044184k total, 0k used, 1044184k free, 277660k cached

 PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
3809 jdoe 25 0 6256 3932 1312 R 93.8 0.8 0:21.49 bunzip2
2769 root 16 0 157m 80m 90m R 2.7 16.0 5:21.01 X
3006 jdoe 15 0 30928 15m 27m S 0.3 3.0 0:22.40 kdeinit
3008 jdoe 16 0 5624 892 4468 S 0.3 0.2 0:06.59 autorun
3034 jdoe 15 0 26764 12m 24m S 0.3 2.5 0:12.68 kscd
3810 jdoe 16 0 2892 916 1620 R 0.3 0.2 0:00.06 top
```

---

✓ L'ordre de tri peut être changé en tapant.

M utilisation Mémoire, P %CPU, T Temps d'exécution

---

✓ On peut arrêter une tâche en tapant **k** (kill) et son numéro.

---

## [KILL] \_ Arrêt de processus

**#kill [pids]**

Envoie un signal d'arrêt aux processus spécifiés. Cela permet aux processus de sauvegarder leurs données et s'arrêter eux-mêmes. A utiliser en premier recours.

**Exemple : #kill 3039 3134 3190 3416**

**#kill -9 [pids]**

Envoie un signal d'arrêt immédiat. Le système lui-même se charge d'arrêter les processus. Utile quand une tâche est vraiment bloquée (ne répond pas à kill 1).

**#kill -9 -1**

Arrête tous les processus de l'utilisateur courant. 1 tous les processus.

**#killall -[signal][command]**

Arrête toutes les tâches exécutant <commande>.

**Exemple : #killall bash**

**#xkill**

Vous laisse arrêter une application graphique en cliquant dessus ! Très rapide ! Utile quand vous ne connaissez pas le nom de commande de l'application.

### ASTUCE !

---

Si votre application graphique est plantée et que vous ne pouvez plus accéder à votre terminal, ne rebootez pas !

- Il est probable que votre système soit encore intact. Essayer d'accéder à une console texte en appuyant sur les touches **[Ctrl][Alt][F1]** (ou **[F2]**, **[F3]** pour davantage de consoles texte)
- Dans la console texte, vous pouvez arrêter l'application corrompue.
- Une fois fait, vous pouvez retourner à la session graphique en appuyant sur **[Ctrl][Alt][F5]** ou **[Ctrl][Alt][F7]** (suivant de votre distribution)
- Si vous ne pouvez pas identifier le programme corrompu, vous pouvez arrêter tous les processus : **#kill 9 1**  
Vous êtes ensuite redirigé vers l'écran de connexion.

- 
- ✓ **#jobs** Fournit la liste des tâches de fond issues du même shell.

Exemple :

```
[1] Running ~/bin/trouver_sens_vie sansdieu &
[2]+ Running make mistakes &
```

- ✓ **#fg %<n>** Faire de la dernière / nième tâche de fond la tâche courante.

Exemple :

```
make mistakes
```

- ✓ **#bg** ou [CTRL+Z] Mettre la tâche courante en arrière-plan.

Exemple :

```
[2]+ make mistakes &
```

- ✓ **#kill %<n>** Interrrompt la nième tâche.

Exemple :

```
kill %1
[1]+ Terminated ~/bin/trouver_sens_vie sansdieu
```

---



```
#!/bin/bash
```



```
#!/bin/bash
```

# Variables d'environnement



Les shells permettent à l'utilisateur de définir des variables.  
Celles-ci peuvent être réutilisées dans la commande shell.  
Convention : noms en minuscules.

Vous pouvez aussi définir des variables d'environnement :  
des variables qui sont aussi visibles depuis les scripts ou  
les exécutables appelés depuis le shell.

Convention : noms en majuscules

## env

Affiche toutes les variables d'environnement existantes ainsi que leur valeur.

### ▪ Exemples de variables de shell

#### ○ Variables de shell (bash)

projdir=/home/marshall/gadgets

ls \$projdir; cd \$projdir

#### ○ Variables d'environnement (bash)

cd \$HOME

export DEBUG=1

./chercher\_vie\_extraterrestre

(affiche des informations de mise au point si DEBUG est défini)

### ▪ Variables d'environnement standards

#### \$LD\_LIBRARY\_PATH

Chemin de recherche de bibliothèques partagées ;

#### \$DISPLAY

Écran sur lequel afficher les applications X (graphiques).

#### \$EDITOR

Éditeur par défaut (vi, emacs...).

#### \$HOME

Répertoire de l'utilisateur courant.

#### \$HOSTNAME

Nom de la machine locale.

#### \$MANPATH

Chemin de recherche des pages de manuel.

#### \$PATH

Chemin de recherche des commandes.
<b>\$PRINTER</b>
Nom de l'imprimante par défaut.
<b>\$SHELL</b>
Nom du shell courant.
<b>\$TERM</b>
Type du terminal courant.
<b>\$USER</b>
Nom de l'utilisateur courant.

## ■ Variables d'environnement \$PATH

Spécifications	Exemples
<b>\$PATH</b> Spécifie l'ordre de recherche de commandes pour le shell.	/ home/acox/bin:/usr/local/bin:/usr/kerberos /bin:/usr /bin:/bin:/usr/X11R6/bin:/bin:/usr/bin
<b>\$LD_LIBRARY_PATH</b> Spécifie l'ordre de recherche pour les bibliothèques partagées (codes binaires partagés par les applications, comme la bibliothèque C) pour ld	/usr/local/lib:/usr/lib:/lib:/usr/X11R6/lib
<b>\$MANPATH</b> Spécifie l'ordre de recherche pour les pages de manuel	/usr/local/man:/usr/share/man

---

Il est fortement recommandé de ne pas avoir le répertoire “.” dans votre variable d'environnement PATH, en particulier pas au début.

- ✓ Un intrus pourrait placer un fichier ls malveillant dans vos répertoires. Il serait exécuté à chaque appel de ls depuis ces répertoires et pourrait s'attaquer à vos données personnelles.
- ✓ Si vous avez un fichier exécutable de nom test dans un répertoire, il sera utilisé à la place du programme test par défaut et certains scripts ne fonctionneront plus correctement.
- ✓ Chaque fois que vous entrez dans un nouveau répertoire, le shell perdra du temps à mettre à jour sa liste de commandes disponibles.

Lancez vos propres commandes ainsi : ./test

---

## ■ ALIAS

Les **shells** vous permettent de définir des alias : des raccourcis pour des commandes que vous utilisez très souvent.

Exemples :

**alias ls='ls la'**

Utile pour toujours lancer des commandes avec certains paramètres.

**alias rm='rm i'**

Utile pour faire que rm demande toujours une confirmation.

**alias tor='trouver\_objet\_rambaldi vite risque'**

Utile pour remplacer de longues commandes utilisées régulièrement.

**alias cia='. /home/sydney/env/cia.sh'**

Utile pour initialiser rapidement un environnement

(. est une commande shell pour exécuter le contenu d'un script shell).

## ■ Fichier .bashrc

.**bashrc** est un script shell lu à chaque fois qu'un shell bash est lancé.

**Vous pouvez utiliser ce fichier pour définir**

- Vos variables d'environnement par défaut (PATH, EDITOR...)
- Vos alias
- Votre invite de shell ("prompt" : voir le manuel de bash pour plus de détails)
- Un message de bienvenue

# SYSLOG

« IL EXISTE UN FICHIER APPELÉ SYSLOG QUI ENREGISTRE TOUT L'ACTIVITÉ DU NOYAU LINUX. ON A SOUVENT BESOIN DE CONSULTER LE JOURNAL DU SYSTÈME. »

On peut la consulter

En tapant :



`#cat /var/log/syslog` // la commande "#tail" permet de consulter que les 10 dernières lignes (par défaut), avec l'option - [nombre] (ex : `tail -15 /var/log/syslog`) peut être pratique pour visualiser les 15 dernières lignes.

La commande sœur de "#tail" c'est "#head" qui permet de voir les dix premières lignes. Dans certain cas, l'option -f peut être pratique qui permet de visualiser en temps réel le fichier log.

# GLOBING BASH

« UNE PETITE PRÉCISION PEUT CHANGER VOTRE VIE. »



I  
Bash

- L'étoile (\*) qui remplace zéro ou plusieurs caractères (all) (ex : `#ls *`) veut dire littéralement lister tous les répertoires et sous-répertoires.
- ? (variable\_char) remplace 1 caractère, ?? remplace 2 caractères et ainsi de suite (ex: `#ls ?????`) lister tous les répertoires à 5 caractères.
- Le backslash (\) permet de préciser un caractère dans le même principe que si dessous (ex: `#ls \s??`), ce syntaxe veut dire lister tous les répertoires à 3 caractères qui commence par "s".
- [...] peut être utile pour dire au bash des caractères spécifiques (2 car. minimums) (ex : `#ls [sw]*`), veut dire lister tous les répertoires ayant commencé par un lettre "s" ou une lettre "w".

- 
- ✓ Precéder les commandes par "^" signifie au bash la **négation** de ce dernier (ex: `#ls [^sw]*`).
-

# TRUX

« LES TRICKS de LINUX POUR TRAVAILLER PLUS EFFICACEMENT. »



- **#exit** (déconnexion) [ctrl+d] Pour quitter le terminal.
- **#clear** [ctrl+l] Permet de nettoyer l'interface du terminal.
- **#reset** Réinitialisation du terminal.
- **#tree** Arborescence (Ex : #tree /home/user/).
- **#history** Historique (bash), -c pour effacer les historiques (Ex : #history 10) pour afficher les 10 dernières commandes.
- **#man** Permet de consulter les manuels de commande (Ex : #man top), on peut utiliser aussi l'option -help (Ex : #top --help).
- **#which** Avant de lancer une commande, which vous dit où elle est trouvée (Ex : #which ls).



- **#!!** Réutiliser la dernière commande.
- **#![command]** Réutiliser la dernière commande du [command] (en entière)\_(Ex: #!cp).
- **#?![key\_word]** Cherchera dans l'historique la dernière commande qui utilisait cette partie de commande et l'exécutera (Ex : #!?gra) pourrait lancer la commande aptitude full-upgrade.
- **#bc** Une calculatrice maniable et complète. Elle inclut même un langage de programmation ! Utiliser l'option l pour faire du calcul avec virgule flottante.
- **#date** Retourne la date courante. Utilisé dans les scripts pour indiquer quand la commande débute ou est terminé.



- MODE GRAPHIQUE > MODE CONSOLE (virtuelle) [ctrl+alt+F1, F2, F3, F4, F5, F6]
- MODE CONSOLE (virtuelle) > MODE GRAPHIQUE [ctrl+alt+F7]
- Auto-complétions [TAB] ↲
- Navigation dans l'historique des commandes Flèches [haut] et [bas]
- Rechercher une commande dans l'historique de bash [ctrl+r]
- BEGIN [ctrl+a] / END [ctrl+e]
- DEL\_word [ctrl+w] Supprime un mot qui précède le curseur.
- DEL\_all\_word [ctrl+u] Supprime tous les mots qui précède le curseur.
- UNDEL [ctrl+y] Annule une suppression.
- CANCEL [ctrl+c] Annule une commande en cours d'exécution.
- a<>b [ctrl+t] / aaa<>bbb [alt+t] Effectue une permutation de lettre.
- Paste [ctrl+shift+v] Colle une/plusieurs mot(s) dans un terminal.

- Pour la commande "#mkdir" ou "#touch", dans la mesure du possible, évitez de créer des espaces (surtout dans les répertoires ou fichiers sensibles), sinon il existe trois manières de le procéder :

```
#mkdir/touch test\ 1
#mkdir/touch "test 2"
#mkdir/touch [test 3]
```



## SEQUENCE DE COMMANDES

- Possibilité de taper la prochaine commande dans votre terminal même si la commande courante n'est pas terminée.

- Possibilité de séparer plusieurs commandes par le symbole ;  
`#echo "Vous êtes le plus beau" ; sleep 10 ; echo "des menteurs"`



- Conditions : utiliser || (ou) ou && (et) :

`#more Dieu || echo "Dieu existe !"` N'exécute echo que si la première commande échoue.

- `#ls ~sd6 && cat ~sd6/* > ~sydney/recettes.txt`  
 N'affiche le contenu des fichiers du répertoire que si la commande ls réussit (indique un accès en lecture).



## COMPARAISON DE FICHIER

- **#diff** fichier1 fichier2  
 Affiche les différences entre 2 fichiers, ou rien si les fichiers sont identiques.
- **#diff r rep1/ rep2/**  
 Affiche les différences entre fichiers de même nom dans les 2 répertoires.



### Autres outils graphiques pour comparer des fichiers :

- **Tkdiff** : Outil pratique pour comparer des fichiers et fusionner leurs différences.
- **Kompare** : même principe que tkdiff (pour Fedora Core).
- **Gvimdiff** ...

## SHUTDOWN (arrêt du système)

```
#shutdown h +5 (h: halt)
```

Éteint le système dans 5 minutes.

Les utilisateurs reçoivent un avertissement dans leur console.



```
#shutdown r now (r: redémarrer)
```

```
#init 0
```

Un autre moyen d'arrêter le système (utilisé par shutdown)

```
#init 6
```

Un autre moyen de redémarrer (utilisé par shutdown)

[Ctrl][Alt][Del]

Fonctionne aussi sur GNU/Linux (au moins sur PCs!)

## ASTUCES RESEAUX



### Configuration réseau

```
#ifconfig a
```

Affiche les informations sur toutes les interfaces réseau disponibles sur votre système.

```
#ifconfig eth0
```

Liste les détails de l'interface eth0

```
#ifconfig eth0 192.168.0.100
```

Assigne l'adresse IP 192.168.0.100 à eth0  
(1 adresse IP par interface)

```
#ifconfig eth0 down
```

Eteint l'interface eth0  
(libère son adresse IP)

```
#route add default gw 192.168.0.1
```

Configure la route par défaut pour les paquets à destination de l'extérieur du réseau local. La passerelle (ici 192.168.0.1) est responsable de l'envoyer à la prochaine passerelle, etc., jusqu'à la destination finale.

	<p><b>#route</b> Listes les routes existantes</p> <p><b>#route del default</b> route del &lt;IP&gt; Supprime les routes données Utile pour redéfinir une nouvelle route.</p>
	<p><b>#ping freshmeat.net</b> <b>#ping 192.168.1.1</b></p> <p>Essaye d'envoyer des paquets à la machine donnée et retourne un paquet en accusé de réception.</p> <pre>PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data. 64 bytes from 192.168.1.1: icmp_seq=0 ttl=150 time=2.51 ms 64 bytes from 192.168.1.1: icmp_seq=1 ttl=150 time=3.16 ms 64 bytes from 192.168.1.1: icmp_seq=2 ttl=150 time=2.71 ms 64 bytes from 192.168.1.1: icmp_seq=3 ttl=150 time=2.67 ms</pre>
Test réseau	<p>n.b :</p> <ul style="list-style-type: none"> <li>○ Quand vous pouvez pinger votre passerelle, votre interface réseau Fonctionne</li> <li>○ Quand vous pouvez pinger une adresse IP externe, vos paramètres réseau sont corrects!</li> </ul>
Résume de la configuration de réseau	<p>Uniquement pour les cas simples à interface unique, sans serveur dhcp...</p> <p>Connectez-vous au réseau (câble, carte wifi ou périphérique...)</p> <p>Identifier votre interface réseau :</p> <p><b>#ifconfig a</b></p> <p>Assigner une adresse IP à votre interface (supposons eth0)</p> <p><b>#ifconfig eth0 192.168.0.100</b> (exemple)</p> <p>Ajouter une route à votre passerelle (supposons 192.168.0.1) pour les paquet sortant du réseau:</p> <p><b>#route add default gw 192.168.0.1</b></p>
Résolution de noms	<p>Votre programme a besoin de savoir quelle adresse IP correspond à un nom de domaine donné (comme kernel.org)</p> <p>Un Domain Name Server (DNS) s'occupe de cette procédure.</p>

Vous devez juste spécifier l'adresse IP d'un ou plusieurs serveurs DNS dans votre fichier **/etc/resolv.conf** :

*nameserver 217.19.192.132*

*nameserver 212.27.32.177*

Les changements prennent effet immédiatement !



## Comment prononcer “Linux” ?

Bien sur, chaque pays et/ou langage peuvent avoir leur propre prononciation.

En anglais, il est difficile de deviner !

En fait, voici comment Linus Torvalds le prononce : <http://freeelectrons.com/pub/audio/torvaldssays-linux.ogg>

## Pages info

Sous GNU, les pages de manuel sont en voie de remplacement par les pages info. Certaines pages de manuel indiquent même de consulter plutôt les pages info.

```
#info <commande>
```

Fonctionnalités d'info :

- Documentation structurée en sections (“noeuds”) et soussections (“sousnoeuds”)
- Possibilité de parcourir cette structure : sommet, suivant, précédent, haut
- Pages info générées à partir des mêmes sources texinfo que la documentation en HTML.

## Recherche de ressources sur Internet

### Résolution de problèmes

- La plupart des forums et des archives de listes de discussions sont publics, et sont indexés très régulièrement par Google.
- Si vous cherchez la cause d'un message d'erreur, copiez-le tel quel dans le formulaire de recherche, entre des guillemets (“message d'erreur”). Il est très probable que quelqu'un d'autre ait déjà rencontré le même problème.
- Pensez bien à utiliser Google Groups : <http://groups.google.com/>  
Ce site indexe plus de 20 ans de groupes de discussion.

### Recherche de documentation

- Recherchez <outil> ou <outil> page pour trouver la page d'accueil de l'outil ou du projet et ensuite trouver les plus récentes

- Recherchez <outil> documentation ou <outil> manual (en anglais) dans votre moteur de recherche préféré.

### Recherche de documentation générique

WikiPedia : <http://fr.wikipedia.org> De nombreuses et utiles définitions en informatique. Une vraie encyclopédie. Ouverte aux contributions de chacun.

## Quelques applications de bureau

Faire une démonstration sur un écran avec un projecteur !

- **Mozilla** : navigateur Internet, client de courrier électronique et éditeur HTML
- **Firefox** : navigateur léger dérivé de Mozilla
- **OpenOffice** : suite bureautique complète compatible avec MS
- **Office** : traitement de texte, tableur, présentations, graphiques...
- Le **GIMP** : un éditeur graphique extrêmement puissant
- **Gqview** : afficheur de galerie de photos

**Evolution** : client de messagerie et calendrier semblable à Outlook.

## Alternatives aux outils sous Windows

- Internet Explorer > Mozilla Firefox
- IIS > Apache
- Money > GNU Cash
- MS Office > OpenOffice
- MS Outlook > Evolution
- MS Project > Mr Project (Planner)
- Nero > k3b
- Photoshop > The GIMP
- WinAmp > xmms
- W. Media Player > xine mplayer

Plus d' alternatives:

<http://linux.ie/newusers/alternatives.php>

## GNU / Linux à la maison

A la maison, GNU / Linux est aussi une alternative sérieuse à Windows

- **Sans virus**

La plupart des virus sont conçus pour tirer parti des failles de sécurité de Windows et n'ont aucun effet sur GNU / Linux.

- **A l'épreuve des virus**

Même si vous exécutez un virus compatible avec Linux, il n'aurait pas la permission de modifier le système.

- **A l'épreuve des erreurs**

Les autres membres de la famille ne peuvent ni toucher au système ni aux fichiers de quelqu'un d'autre. Ils ne peuvent endommager que leurs propres fichiers.

- **Décourage les pirates**

Même si vous êtes connecté en permanence à Internet, votre système attire moins les pirates.

- **Respect de la vie privée**

Votre système ne va pas discrètement recueillir des informations sur les films ou les sites internet que vous préférez.

- **Convivialité**

Vos programmes sont conçus pour des utilisateurs par des utilisateurs. Ils sont mieux susceptibles de satisfaire vos besoins.

Les développeurs peuvent facilement être contacter pour leur suggérer de nouvelles fonctionnalités.

- **Liberté**

Les données que vous créez vous appartiennent pour toujours. Elles ne sont pas prisonnières d'une application propriétaire à travers un format propriétaire (parfois breveté !).

Vous êtes libres d'aider votre entourage en partageant vos programmes avec lui.

Vous êtes libres d'utiliser les mêmes programmes au travail également !

- **Vous pouvez passer à GNU / Linux pour :**

La bureautique : traitement de texte, tableur, présentations.

Internet : navigation et courrier électronique.

Le multimédia : vidéo, son et graphiques (y compris appareils photo numériques).

Mieux connaître les ordinateurs et la programmation.

**Si vous possédez encore une copie de Windows, vous pouvez la garder (option au démarrage) pour :**

Les jeux. Nombre d'entre eux ne sont encore conçus que pour Windows ou Mac.

Utiliser des logiciels propriétaires spécifiques ou des cdroms éducatifs.

Utiliser du matériel non encore pris en charge sous GNU / Linux.

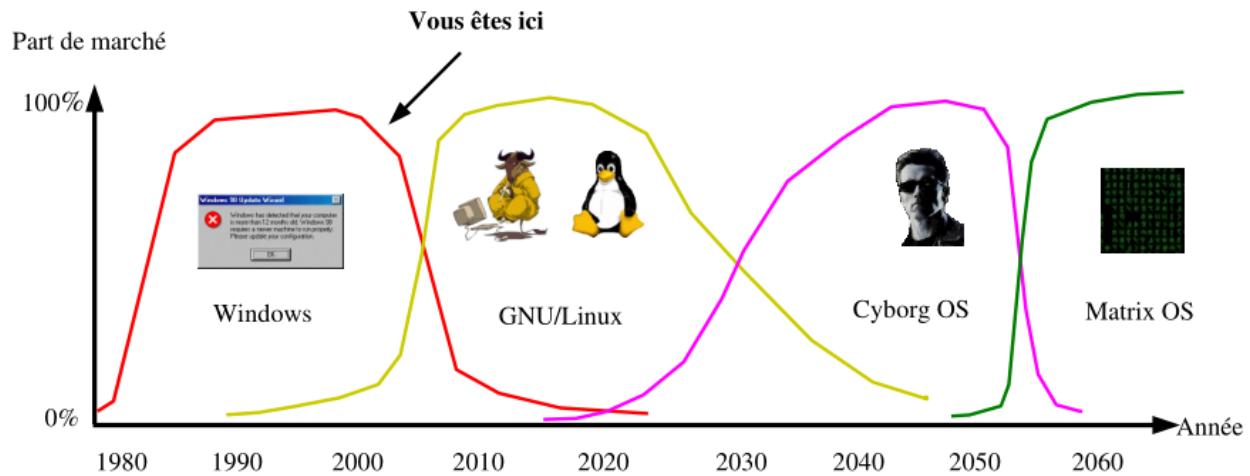
## Utiliser les distributions GNU / Linux

**Utilisation de distributions GNU / Linux :**

- Vous permettez d'installer GNU / Linux dans un emplacement libre sur votre disque dur, tout en gardant Windows ("double démarrage")
- Ont une interface très conviviale qui peut détecter automatiquement la plupart des matériels. Vous n'avez aucun pilote à installer !
- Vous permettez de choisir les types d'applications à installer
- Fournissent une interface de configuration conviviale
- Distributions recommandées pour les débutants : Ubuntu ou Linux Mint.

Ne ratez pas le prochain train !

## Histoire des systèmes d'exploitation



FIN ☺