

Améliorer sa Configuration | Linux

Michel Luc (cern91@worldnet.fr)

Table of Contents

Configuration de l'Environnement	1
L'environnement	4
LS en couleur	4
less mieux que more !?	4
Commandes et variables utiles	5
L'utilisateur non su	6
1–Les Manuel Français	9
2–Le Guide d'installation en Français	9
3–Le correcteur orthographique	10
4–Le Dico Français	10
5–Les Sites de distribution des Sources	10
6–Et quoi encore ?	12
1.1–Vérification	12
1.2–Installer les sources	13
1.3–Comment Pourquoi ?!	13
1.4–Nettoyage	14
2.1–Configurer le Noyau	14
2.2–Compiler le Noyau	15
3 –En bref , la procédure !	18
Précaution :	18
4 –Des adresses utiles :	25
Configuration du Système	32
Configuration des périphériques	33
Configuration des périphériques	33
Configuration de la carte son	34
Les cartes son reconnues sous Linux :	34
Vérification :	36
utilisation de sndconfig :	38
Configuration manuelle pour la Slackware :	40
Utilisation de pnpdump et isapnp :	44
Configuration des Editeurs	0
Récupération de ce qui peut encore l'être !	0
Adapter son environnement de travail	0
 Sudo	 0
 Du Français dans la Slackware	 0
 Pourquoi recompiler le Noyau ?	 0

Sudo

- [Description](#)
 - [Installation de Sudo](#)
 - [Configuration de Sudo](#)
 - [Utilisation](#)
-

Description

Sudo est un outils qui permet de configurer très facilement les droits d'exécution de certaines commandes pour que certains utilisateurs puissent utiliser ces commandes sans avoir de droits sur celles-ci, en clair sudo vous permet de définir quel utilisateur aura le droit de lancer des commandes en principe réservées à "root" comme, monter une partition, établir une connexion ppp pour récupérer les mails, les news, arrêter le système ...etc.

Installation

La dernière version à ce moment est la **1.6.3p5** :

Vous trouverez de l'info et de la doc sur le site : www.courtesan.com/sudo/

Vous pous télécharger les archives et packages RPM depuis un site FTP :

<ftp.tux.org/pub/security/sudo/>

<ftp://fr.rpmfind.net/linux/contrib/libc6/i386/>

Vous trouverez également des packages spécifiques pour Debian et Mandrake .

_ "Logez-vous" sous compte root (ou **su**)

Pour installaller un RPM : **rpm -ivh sudo-1.6.1-1.i386.rpm**

Pour décompresser l'archive : **tar xvzf sudo-1.6.3p5.tar.gz**

Je vais commenter l'installation de la version **sudo-1.6.3p5.tar.gz** :

_Placez vous dans répertoire ainsi créé **cd sudo-1.6.3p5** .

L'archive *.tar.gz utilise par défaut le répertoire d'installation **/usr/local/** , pour changer ce répertoire éditez le fichier **configure** et modifiez cette ligne **ac_default_prefix=/usr/local** au début du fichier en **ac_default_prefix=/usr** pour installer les exécutable(binaires) sous /usr/bin et /usr/sbin .

_Et ensuite entrez les commandes :

./configure

make

make check

make install

L'installation va créer l'exécutable **/usr/bin/sudo**, un fichier de configuration **/etc/sudoers** et un exécutable **/usr/sbin/visudo** pour éditer et modifier ce fichier .

Les droits de ces fichiers doivent ressembler à ceci :

```
tuxserv:/admin $ ls -l /etc/sudoers
-r--r----- 1 root    root          294 Dec 10  1999 /etc/sudoers
tuxserv:/admin $ ls -l /usr/bin/sudo
---s--x--x  1 root    root        81284 Dec 10  1999 /usr/bin/sudo
tuxserv:/admin $ ls -l /usr/sbin/visudo
---x--x--x  1 root    root        56776 Dec 10  1999 /usr/sbin/visudo
tuxserv:/admin $
```

Notez les droits particuliers de ces fichiers :

visudo est exécutable par tous, **sudoers** est seulement lisible par le propriétaire et le groupe et **sudo** est **suid** aussi pour cette raison vous ne devez installer sudo que sur une partition qui autorise le montage suid comme l'option "defaults" dans votre fichier /etc/fstab :

```
/dev/hda2          /          ext2      defaults    1 1
```

Configuration

Pour modifier le fichier de configuration /etc/**sudoers** (éditer et modifier), vous devez (en root) lancer **visudo** .

Pour que **visudo** fonctionne, l'éditeur **vi** doit être installé sur votre système !

Pour ma part, je n'utilise pas **vi** (mais emacs) aussi je modifie manuellement le fichier sudoers, mais attention aux erreurs de frappe et de syntaxe !

L'avantage de **visudo** est qu'il permet de détecter les erreurs de syntaxe et de faire prendre en compte les modifs par **sudo**.

Nous allons prendre l'exemple de deux utilisateurs, "mic" et "mac", aux quels nous voulont donner les droits pour arrêter la machine et se connecter avec PPP :

```
# sudoers file.
#
# This file MUST be edited with the 'visudo' command as root.
#
# See the sudoers man page for the details on how to write a sudoers file.
#

# Host alias specification

# User alias specification

# Cmnd alias specification
Cmnd_Alias    HALT=/sbin/shutdown
Cmnd_Alias    DIALER=/usr/local/bin/mdial

# User privilege specification
root    ALL=(ALL) ALL
mic    ALL=NOPASSWD:DIALER,NOPASSWD:HALT
mac    ALL=NOPASSWD:DIALER,NOPASSWD:HALT
```

mdial est le numéroteur internet que j'utilise.

NOPASSWD évite que sudo demande le mot de passe de l'utilisateur .

Utilisation

Pour arrêter la machine, il suffit de taper : **sudo shutdown -h now**

Pour se connecter à internet, tapez : **sudo mdial br> Si vous utilisez des scripts (perso) de connexion comme ppp-on et ppp-off, utilisez cette config :**

```
# sudoers file.
#
# This file MUST be edited with the 'visudo' command as root.
#
# See the sudoers man page for the details on how to write a sudoers file.
#

# Host alias specification

# User alias specification
```

```
# Cmnd alias specification
Cmnd_Alias      HALT=/sbin/shutdown
Cmnd_Alias      PPPON=/etc/ppp/ppp-on
Cmnd_Alias      PPPOFF=/etc/ppp/ppp-off
# User privilege specification
root    ALL=(ALL) ALL
mic    ALL=NOPASSWD:PPPON,NOPASSWD:PPPOFF,NOPASSWD:HALT
mac    ALL=NOPASSWD:PPPON,NOPASSWD:PPPOFF,NOPASSWD:HALT
```

Lorsque vous êtes sous un compte utilisateur, pour connaître les commandes (root) que vous pouvez utiliser, tapez **sudo -l**

```
tuxserv:mic $ sudo -l
User mic may run the following commands on this host:
  (root) NOPASSWD: /sbin/shutdown
  (root) NOPASSWD: /usr/local/bin/mdial
tuxserv:mic $
```

Pour en savoir plus sur sudo, consultez les pages de manuel : **man sudo**

Dernière mise à jour : **14-05-2000**

Du Français dans la Slackware

- [1 Les pages de manuel en Français.](#)
 - [2 Le Guide d'installation Français.](#)
 - [3 Le vérificateur orthographique Français.](#)
 - [4 Le dictionnaire Français.](#)
 - [5 Où récupérer les sources ?](#)
 - [6 Et Linux en Français ???](#)
-

1–Les Manuel Français

La Slackware et les autres distributions contiennent des "ManPage" qui vous explique comment faire .

Pour savoir par exemple quels paramètres utilise la commande "**ls**" et comment les passer : vous tapez : # **man ls** et vous avez la réponse en Anglais ?

En Français c'est possible en installant **man-fr** , la dernière version est la 0.8

Installation :

Vous pouvez essayer avec d'anciennes versions moins volumineuses mais incomplètes : man-fr-0.1 , man-fr-0.2 , ...

Copier **man-fr-0.8.tgz** dans **/usr/man/** puis décompresser :

```
# cd /usr/man
```

```
# tar -zxvf man-fr-0.8.tgz
```

Ensuite copiez manuellement les pages dans leurs répertoires respectif de /usr/man/**man1** ../**man2** ...

Si vous voulez consulter les pages avant de les installer :

```
# groff -T latin1 -mandoc "chown.2" | less
```

 pour voir la commande **chown** qui se trouve dans /usr/man/**man2** .

Pour Installer les pages de manuel en français , vous avez **/usr/man/Makefile** .

```
# make install
```

Vous aurez des pages de manuel en français qui seront en double avec celles en anglais ,vous pouvez utiliser ce [script](#) pour faire le nettoyage avant l'installation 'make install' : Vous devez le décompresser dans /usr/man avec 'gzip -d manclean.gz' .

Editez 'manclean' et lisez les commentaires d'utilisation.

Vous pouvez faire le nettoyage manuellement ? N'oubliez pas les répertoires ../**cat1** ../**cat2** ...

Pour pouvoir utiliser les possibilités de recherche de 'apropos' (whatis), il faut modifier le script /usr/sbin/**makewhatis** pour que celui-ci prenne en compte la section 'NOM' .

Si votre navigateur ne reconnaît pas la balise <pre>, il est possible que les lignes qui suivent, comportant beaucoup d'expressions (opérateurs pouvant être confondus avec des 'tags' html), ne s'affichent pas correctement: Vous les retrouverez dans le fichiers

LISEZ_MOI inclus dans les sources

Editez 'makewhatis' et recherchez la ligne :

```
if (($1~/^\.[Ss][Hh]/($2~/[Nn][Aa][Mm][Ee]/)|  
    (pages=="cat"$1~/^NAME/)){
```

Modifiez-la comme suit :

```
if (($1~/^\.[Ss][Hh]/($2~/[Nn][Aa][Mm][Ee]/| |$2~/[Nn][Oo][Mm]/))| |
```

```
(pages=="cat"$1~/^NAME/||$1~/^NOM/))){
```

*** **LISEZ MOI** ©Christophe Blaess™ (ccb@club-internet.fr) ***

Votre script ne comporte peut-être pas cette ligne , mais plutôt plusieurs autres du genre :

```
if ($line=~/^NAME/)
```

Que vous transformez en :

```
if ($line=~/^NAME/||~/^NOM/) l'opérateur '||' signifie 'OU' .
```

Une fois les modifications effectuées , sauvegarder et relancer le script : # **makewhatis**

Cela peut prendre plusieurs minutes .

2–Le Guide d'installation en Français

C'est beaucoup plus qu'un simple guide d'installation ,traduit en français par René Gougnenc .
 Décrivant bon nombre de commandes et de configurations ,il vous sera très utile ,et cela
 représente un excellent bouquin de référence pas cher !..

Installation :

Vous copier **install-guide-french-2.2.2.tar.gz** dans **/usr/local/info/** puis :

```
# cd /usr/local/info
```

```
# tar -zxpvf guide-french-2.2.2.tar.gz
```

Même si cette méthode peut paraître barbare : elle fonctionne .

Et vous obtiendrez un document au format dvi exploitable sous X . Pour être certain que les
 références croisées seront correctement établies , vous devez entrer la commande 'latex gs' 3 ,4
 ou 5 fois de suite :

```
# cd /usr/local/info/install-guide-2.2.2
```

```
# latex gs minimum 3 fois , ensuite :
```

```
# makeindex gs puis à nouveau :
```

```
# latex gs 2 fois
```

Vous avez votre document "**gs.dvi**" ,que vous pouvez consulter sous Xwindow avec xdvi . Vous
 pouvez avec Xfm installer une application pour le lancer automatiquement :lancer **Xfm** ,dans la
 fenêtre "**Applications**" faite une "**click droit**" et sélectionnez "**Install**" :

Name : Guide d'installation

icon : man.xpm

Push action : exec xdvi /usr/local/info/install-guide-french-2.2.2/gs.dvi

Drop action : exec xdvi \$* (si vous voulez utilisez l'application pour lire d'autre document dvi)

Les messages d'erreur.

Ignorer les messages "Underfull \hbox" et "Underfull \vbox" : .

Le premier message d'erreur vous indique que "french.sty" est introuvable ?!

Et pour cause , il n'est pas dans le source : mais cela fonctionne sans ?

Réponse de la part de **Philippe Petit** philippepetit@hotmail.com : Merci !

<<

```
Le fichier french.sty est un fichier > qui est utilisé sous
latex. Il sert a prendre en charge les règles typographiques francaises
et en particulier les accents. Ce fichier en principe se trouve dans la
librairie de Tex. Sinon il suffit de supprimer dans le source l'option
french dans la commande
\documentstyle[...french,...]{.....}. Si ce fichier est absent Latex
fonctionne normalement sans tenir compte de l'option.
```

>Si vous désirez plus de renseignement sur tex/latex vous pouvez

>consulter le site de l'association Gutenberg www.ens.fr/gut/ et
>découvrir les possibilités de ce superbe traitement de texte.

>>

3–Le correcteur orthographique .

Il est possible que vous ayez déjà installé ispell avec le "package" *ap2* .
en associant le dictionnaire **français–IREQ** vous obtiendrez un correcteur Français .
La dernière version de ispell est la 3.1.20 .

Installation :

Copier **ispell–3.1.20.tar.gz** dans **/usr/local/** puis :

```
# cd /usr/local
```

```
# tar -zxvf ispell–IREQ–3.1.20.tar.gz
```

Ispell est "portable" sur différente architecture , il vous faudra donc modifier certains paramètres d'installation :

Sur un système BSD ou équivalent il suffit de faire 'make all' pour terminer le décompactage , sur un system V USG ,il faut utiliser local.h .

Assurez–vous d'avoir au moins 10Mo de libre dans votre répertoire temporaire (/tmp).

Si non indiquer une variable d'environnement 'Set' 'TMPDIR' ayant un espace libre suffisant.

```
# cd /usr/local/ispell–3.1
```

```
# cp local.h.samp local .h
```

Vous allez utiliser ce fichier pour personnalisé votre installation de ispell–3.1 , les fichiers que vous pouvez éditer sont Makefile et config.X , pour lire les instructions .

Mais surtout **ne modifiez pas Makefile et config.X** .

Il y a beaucoup de paramètres qui entrent en jeu , regardez dans config.X , vous allez seulement vous occupez (et c'est déjà pas mal !) des suivants :

CC pour choisir gcc

de **BINDIR** ... à ... **MANDIR** pour indiquer vos répertoires

LANGUAGES pour sélectionner les dictionnaires

En fontion de votre distribution :

Editez **local.h**

vous devez enlever NO8BIT et ajouter USG : .

```
#undef NO8BIT
```

```
#define USG
```

BINDIR peut–être **/usr/bin** ou **/usr/local/bin**

LIBDIR peut–être **/usr/lib** ou **/usr/local/lib**

ELISPDIR peut–être **/usr/lib/emacs/site–lisp** ou **/usr/share/emacs/site–lisp**

TEXINFODIR peut–être **/usr/info** ou **/usr/local/info**

MAN1DIR peut–être **/usr/man/man1** ou **/usr/local/man/man1**

MAN4DIR peut–être **/usr/man/man4** ou **/usr/local/man/man4**

LANGUAGES pour cette section si vous installez 'françai–IREQ' regargez le chapitre suiant ,si non lisez le commentaire dans config.X .

Vérifiez encore une fois que tout est correct dans ce fichier (modifier en conséquence et sauvegarder) ,lisez le fichier **/usr/local/ispell–3.1/README** .

Et pour terminer :

```
# make all pour compiler les programmes
```

```
# make install pour installer ispell
```

```
# make clean pour nettoyer
```

```
# make realclean pour nettoyer encore plus
```

4–Le Dico Français

Il est possible d'installer un dictionnaire français qui fonctionne avec ispell .
la dernière version est la 1.4 .

Si vous désirez modifier la liste de mots du dictionnaire , vous pouvez installer **buildict**

Installation :

Copier **francais-IREQ-1.4.tar.gz** dans **/usr/local/ispell-3.1/langues/francais-IREQ/** :

Vous devez créer manuellement le répertoire **/usr/local/ispell-3.1/langues/francais-IREQ/**

```
# cd /usr/local/ispell-3.1/langues/francais-IREQ
```

```
# tar -zxvf francais-IREQ-1.4.tar.gz
```

Vous devez indiquer au dictionnaire les répertoires utilisés par ispell , pour cela :

Editer **/usr/local/ispell-3.1/langues/francais-IREQ/Makefile**

modifier le chemin de 'LIBDIR' et 'HASH' où se trouvent les fichiers '.hash' , '.aff' et 'buildhash' , cela peut-être :

```
LIBDIR=/usr/lib/ispell
```

```
HASH=/usr/bin/buildhash
```

Il est possible de compiler et installer **francais-IREQ** après l'installation de ispell :

```
# cd /usr/local/ispell-3.1/langues/francais-IREQ
```

```
# make création de francais.hash
```

```
# make install installation de francais.aff et francais.hash
```

Et pour construire manuellement le dictionnaire :

```
# buildhash francais.dico ./francais.aff francais.hash
```

Mais depuis la version 3.1 de ispell cette procédure est automatisée avec l'installation de ispell .

Pour cela vous devez Editer **/usr/local/ispell-3.1/local.h** pour ajouter (ou modifier) la ligne :

```
#define LANGUAGES
```

```
" {english,MASTERDICTS=english.med+,HASHFILES=med+.hash,VARIANTS=american,EXTRADICT=}
```

Cela doit être écrit sur une seule ligne !

Et suivant la version de ispell cette ligne peut ressembler à :

```
#define LANGUAGES
```

```
" {american,MASTERDICTS=american.med+,HASHFILES=americanmed+.hash,EXTRADICT=/usr
```

Vous pouvez installer également un dictionnaire Allemand ,Espagnol ..etc... Consultez les doc.(README).

Maintenant pour pouvoir utiliser le dictionnaire français :

Créer le fichier **~/ispell_francais** (dans votre répertoire \$HOME) , vérifiez également que 'francais.aff' et 'francais.hash' sont présents dans **/usr/lib/ispell/** .

Si l'installation du dictionnaire en lui-même est simple , son utilisation avec Emacs l'est un peu moins :

L'utilisation avec TeX , XEmacs et Emacs nécessite deux autres fichiers :

Vous devez créer les liens symboliques ,de **/usr/lib/ispell/francais-TeX.hash** vers

/usr/lib/ispell/francais.hash et de **~/ispell_francais-TeX** vers **~/ispell_francais** .

Rappel: Pour créer un lien symbolique .

Si les fichiers à lier ne se trouvent pas dans le même répertoire et si vous n'êtes pas dans ce répertoire : *Indiquez Toujours le chemin complet !!!*

```
# ln -s /usr/lib/ispell/francais.hash(fichier vers lequel pointe le lien,séparé par un espace)/usr/lib/ispell/francais-TeX.hash (fichier à créer).
```

Il ne vous reste plus que l'installation !?

```
# cd /usr/local/ispell-3.1
```

```
# make all si vous n'avez pas installé ispell
```

```
vous pouvez vérifier si vous n'aurez pas de surprise ? # make -n install
```

```
# make install
```

make clean
make realclean

Les messages d'erreur.

Si vous obtenez un message d'erreur avec la compilation de 'Term.c', probablement que vous essayez de compiler avec une configuration **BSD** sur un système **USG**, ou le **SIGNAL_TYPE** ne correspond pas à votre système.

Tester le dictionnaire avec Emacs :

1) Avec votre éditeur préféré vous allez créer un nouveau fichier :

Par exemple '**mon_fichier.texte**' dans votre répertoire \$HOME, une ligne suffit c'est juste une vérification, mettez dans cette ligne un ou deux noms propre et une ou deux fautes d'orthographe :
Sauvegarder .

Ensuite entrez la commande : # **ispell -d francais mon_fichier.texte** ispell doit vous proposer des mots pour remplacer ceux qui ne figurent pas dans le dictionnaire .

2) Lancer Emacs, la description est moins facile que l'utilisation qui s'effectue par menu .

emacs mon_fichier.texte, votre texte apparaît .

Changer de dictionnaire :

avec la touche **F10**, le menu s'affiche, soit vous utilisez 'PageUp', 'PageDown', les 'fleches' ou la lettre correspondante, pour sélectionner une option .

E Edition

2 Ispell

D Changer le dico

Barre d'espace et les dictionnaires disponibles s'affichent :

Tapez le nom du dictionnaire : **francais-tex**

Un message vous indique que le dictionnaire français est chargé .

Afficher à nouveau le menu **F10**

E

2

B Vérification du fichier

Si tout ce passe bien vous pouvez effectuer les corrections .

Si non Emacs vous indique qu'il ne trouve pas le fichier '.hash' :

Il n'accepte pas la syntaxe de 'francais-TeX.hash', vous devez alors noter le nom qu'il vous indique dans son message d'erreur, probablement 'francais-tex.hash', et créez un nouveau lien symbolique :

de /usr/lib/ispell/**francais-tex.hash** vers /usr/lib/ispell/francais.hash .

Et cela devrait fonctionner !

Et si vous avez installé les pages de manuel en français et modifier 'makewhatis', vous pouvez tester sous Emacs la recherche 'apropos' :

F10

H

M

entrez un nom de page (en français) '**chmod**' si 'makewhatis' a été correctement modifié la page correspondante s'affiche .

Vous pouvez installer **buildict** pour pouvoir modifier la liste de mot du dictionnaire .

Son installation est fort simple :

Créez manuellement /usr/local/ispell-3.1/addons/**buildict**/

Et décompressez les sources dans ce répertoire .

buildict remplace simplement le "shell script" **munchlist** de ispell par un "script perl" qui va permettre de compléter les mots du dictionnaire, mais 'buildict' est 2 fois plus lent que 'munchlist' et pour recompilier le dictionnaire cela peut prendre plus d'une heure et nécessiter plus de 100Mo d'espace disque de libre.

Voir le fichier README .

5–Les Sites de distribution des Sources

Dernières versions en date du 08 Février 2000 :

<ftp.lip6.fr/pub/TeX/GUTenberg/>

Les Pages de Manuel:

[man–fr–0.8.tgz](#) sur Lip6

[man–fr–0.8.tgz](#) sur Angers

<ftp.lip6.fr/pub/linux/french/docs/>

<ftp.u-bordeaux.fr/pub/Linux/docs/>

[man–fr–0.6.tgz](#) et [man–fr–0.8.tgz](#) ne sont peut-être plus distribués ,vérifiez sur :

<ftp.ibp.fr/pub/linux/docs/> ou <ftp.lip6.fr/pub/linux/french/docs/>

Le Guide d'Installation :

[install–guide–french–2.2.2.tar.gz](#)

au format dvi :[install–guide–french–2.2.2.dvi.gz](#)

au format postscript :[install–guide–french–2.2.2.ps.gz](#)

<ftp.lip6.fr/pub/linux/french/books/>

IsPELL le correcteur orthographique :

La dernière version [ispell–3.1.20.tar.gz](#)

<ftp.lip6.fr/pub/ispell/>

<ftp.cs.ucla.edu/pub/ispell-3.1/>

<ftp.math.orst.edu/pub/ispell-3.1/>

Le dictionnaire Français :

La dernière version [francais–IREQ–1.4.tar.gz](#)

L'éditeur pour le dictionnaire: [buildict.tar.gz](#)

<ftp.lip6.fr/pub/ispell/francais/>

<ftp.robot.ireq.ca/pub/ispell/>

<ftp.ibp.fr/pub/ispell/francais/>

Il existe un autre dictionnaire français pour ispell [ispell3.0–french.tar.z](#) sur <ftp.inria.fr/gnu>

6–Et quoi encore ?

En dehors des distributions récentes comme **RedHat, Mandrake, SuSE, Best Linux ...** :

Passer Linux en version française est possible avec les locale(s) nls ,mais cela demande une certaine gymnastique pour installer les libs ,gcc ,ELF

Je n'en dirais pas plus car je n'ai pas encore essayé : Si vous êtes curieux lisez la doc (en français) disponible sur : <ftp.lip6.fr/linux/french/docs/locale.fr> ou sur <ftp.ibp.fr/pub/french/docs/locale.fr>

Vous pouvez également récupérer le fichier [locale.fr.gz](#)

dernière mise à jour : 20–08–2000

Pourquoi recompiler le Noyau ?

A chaque fois que vous installez une nouvelle carte (Son ,Ethernet ,Scsi ..), un périphérique : Votre noyau n'est plus adapté . Soit vous reconfigurez avec les nouvelles options, soit vous changez de noyau pour une version plus récente : Dans les deux cas il faut RECOMPILER .!

Je vais commenter la recompilation effectuée sur une distribution **RedHat** et **Slackware**, mais cela s'applique aux autres distributions, c'est également valable pour un noyau version 2.0.0, 2.0.36, 2.2.14 ou 2.2.xx .

Depuis la version 2.2.15 il est possible, si l'anglais vous pose quelques problèmes de compréhension, d'obtenir une archive pour mettre à jour l'aide en ligne du menu de compilation du noyau vers une version Française :

Vous trouverez quelques adresses au bas de ce document .

En principe ce qui suit est suffisant pour vous permettre de recompiler le noyau, mais garder à portée de main le **Kernel HOWTO**.

- [1.1 Vérification .](#)
 - [1.2 Installer les sources .](#)
 - [1.3 Comment Pourquoi ?!](#)
 - [1.4 Nettoyage .](#)
 - [2.1 Configurer le Noyau .](#)
 - [2.2 Compiler le Noyau .](#)
 - [3. OUFfff ! Résumé .](#)
 - [4. Quelques adresses utiles.](#)
-

1.1–Vérification

Avant de commencer , nous allons vérifier la présence de certains fichiers pour essayer de mieux comprendre ce que vous allez faire .

Ces vérifications peuvent être effectuées sous Xwindow ,cela évite les commandes à taper :

Sous X :

vous lancez votre gestionnaire de fichiers XFM, KFM ou un autre, dans la barre de chemin s'affichera le chemin complet du répertoire .

Répertoires :

Vous devez trouver dans **/usr/src/linux/** :

les répertoires suivants : **arch** , **include** , **lib** , **modules** , **scripts**

mais aussi (certainement?) **documentation/** , **drivers/** , **fs/** , **init/** , **ipc/** , **kernel/** , **net/** , **mm/** ,

Les fichiers suivants :

Makefile et votre noyau **vmlinux** ou **vmlinuz** que vous pouvez sauvegarder .

Vous devez avoir dans **/usr/src/linux/scripts/** les fichiers : **Makefile** , **Configure** , **Menuconfig** et **patch-kernel** (entre autre) .

En fait **/usr/src/linux/** est un lien vers le répertoire de la version du noyau installé : Pour moi c'est **/usr/src/linux-2.2.14/** .

Si vous avez une version antérieure à la 2.0.0 , vous devriez la mettre à jour en récupérant les sources sur un site FTP (voir chapitre suivant) .

Vérifions les liens symboliques des répertoires suivants :

/usr/src/linux est un lien de **/usr/src/linux-2.2.14**

Ce lien se construit avec la commande : **ln -s /usr/src/linux-2.2.14 /usr/src/linux**

/usr/include/linux est un lien de **/usr/src/linux/include/linux**

/usr/include/asm est un lien de **/usr/src/linux/include/asm-i386**

Et si vous avez une carte SCSI :

`/usr/include/scsi` est un lien de `/usr/src/linux/include/scsi`

C'est fini ,vous pouvez quitter Xwindow .

Evidemment tout ça peut ce faire à partir d'une console en tapant les commandes appropriées !??

1.2–Installer les sources

3 possibilités :

1)vous garder les sources déjà installées car votre noyau correspond à vos options .

2)vous le mettez à jour avec un patch ou des modules .

3)vous installez de nouvelles sources pour un noyau tout neuf .

A partir de là logez vous en "root" .

1)vous ne changez rien vous allez utiliser les mêmes sources :

 Passez [au chapitre suivant](#)

2)vous installez un patch:

 copier le fichier .gz dans /usr/src/ puis :

```
# cd /usr/src
```

```
# gzip -cd patchxx.gz | patch -p0
```

(remplacer les x par les chiffres de votre version)

2)vous installez des modules :

 Copier le(s) module(s) dans /lib/modules/

 Et suivez les instructions de recompilation du noyau en effectuant après make zImage un :

```
# make modules
```

```
# make modules_install
```

3)Vous installez les sources :

Vous pouvez récupérer les sources (Modules et Patches) sur un site FTP . En principe les sources se trouve dans le répertoire /pub/linux/kernel/sources/ . Ne prenez pas la version avec le plus grand chiffre ,cela peut-être une version en cours de développement , prenez celle qui à un chiffre paire en deuxième position celui là est une version stable.

A l'installation les sources vont créer un répertoire /usr/src/linux : Donc avant l'installation il ne doit pas exister !... Il va falloir le supprimer (pas exactement) s'il existe .

Il serait téméraire (voir imprudent) de supprimer simplement ce répertoire "linux" :

Nous allons le déplacer en lui donnant le nom du numéro de sa version : Si vous avez déjà recompilé votre noyau vous avez un répertoire /linux-x.x.x/ , dans le cas contraire il vous faut récupérer les sources pour compiler votre noyau .

Il est possible que vous ayez déjà un répertoire portant ce nom(avec numéro) ...Alors !

La vérification que vous avez effectuée en premier vous dit qu'un répertoire "linux" et

"linux-2.2.13" existe dans /usr/src/ ?

_vérifiez la version de votre noyau actuel : # **uname -r**

cela donne par exemple : # **2.2.13**

_Votre répertoire /usr/src/linux-2.2.13/ contient bien les sources du noyau actuel !

_Si vous n'avez pas de répertoire /usr/src/linux-2.2.13/ : Alors déplacez "linux" vers

"linux-2.2.13" :

```
# cd /usr/src
```

```
# mv linux linux-2.2.13
```

Vérifiez qu'il n'y a plus de répertoire "linux" avant de décompresser les sources .

copier le fichier source .tar.gz dans /usr/src/ puis :

```
# cd /usr/src
```

```
# gzip -cd linux-2.x.x.tar.gz | tar xfv-
```

ou avec `# tar xzpvf linux-2.x.x.tar.gz`

si vous n'avez pas l'extension `.gz` : `# tar xvf linux-2.x.x.tar`

_Vérifiez que vous avez (à nouveau) un répertoire "**linux**" contenant les sources .

_Et surtout si vous installez des sources d'une autre distribution que la Slackware , allez dans ce répertoire `/linux/` et lisez la section *Installing the Kernel* du fichier README : Suivez les instructions.

1.3–Comment Pourquoi ?!

Avant de recompiler votre noyau , lisez attentivement ce qui suit cela vous permettra peut-être d'obtenir un noyau super adapté à vos besoin.

Attention aux commandes , vous devez toujours savoir ce que doit faire l'instruction que vous passez : si non en cas de problème il vous sera difficile d'y remédier !!!

Le noyau évolue constamment et intègre des options permettant l'utilisation d'une plus grande variété de périphériques : Le numéro indiquant sa version change en tenant compte de la maturité du noyau : une version 2.1.0 peut-être une version Alpha , 2.1.2 peut-être une Beta mais la version stable de cette mouture sera 2.2.x .

La version 2.0.x a modifié l'installation du noyau par rapport aux versions 1.0.x précédentes aussi , dans `/linux/documentation/changes` , vous trouverez des indications sur cette mise à jour .

La version 2.2.x apporte une prise en charge supplémentaire, des périphériques et pilotes, fort appréciable :

Alors lisez la doc qui l'accompagne !

Les Modules et les Patches suivent l'évolution du noyau :

_Vous les trouverez sur un site FTP dans `/v2.0` pour un noyau 2.0.x

Regardez cette page pour [plus d'info](#) sur l'installation des patches .

Il est possible à fin de réduire le volume du noyau d'utiliser des modules chargeables .

Les modules sont des fichiers objets se terminant par "`.o`" que l'on peut charger ou retirer à volonté .

Pour savoir si vous devez utiliser les modules regardez cette [introduction aux modules](#) .

Pour une meilleur compréhension je vous conseille en plus de **Kernel HOWTO**(section Modules) , de lire (c'est d'une limpidité) l'article de Eric Dumas : [Les Modules chargeables](#) , avant de prendre une décision sur la version du noyau et le type de gestion des modules que vous allez installer .

Pour indiquer ces Options au noyau il faut le configurer ..!

Trois possibilités (encore) :

make config est un script, en mode console, très simple mais efficace .

make menuconfig c'est celui que j'utilise est un menu en couleur très facile à utiliser .

make xconfig c'est un menu sous **X** à condition d'avoir installé "Tcl/Tk"

.

1.4–Nettoyage

Commence dans un premier temps , avec les liens symboliques et avant de configurer le noyau .

```
# cd /usr/include
```

```
# rm -rf linux
```

```
# rm -rf asm
```

```
# ln -s /usr/src/linux/include/linux linux
```

```
# ln -s /usr/src/linux/include/asm-i386 asm
```

Si ces liens étaient inexistantes ou différents .

suivi de :

```
# cd /usr/src/linux
# make mrproper
Et après avoir créer les dépendances, faites : # make clean
.
```

2.1–Configurer le Noyau

Attachez une attention toute particulière à cette phase .

C'est certainement la plus délicate si vous ne comprenez pas les questions qui y sont posées ! Vous devez pour la plupart répondre par "y" ou "n" ,dans le doute laisser la réponse par défaut . Je ne vais pas détailler ici les options du menu , aidez-vous de **Kernel HOWTO**.

Pour connaître les options de votre noyau actuel :

```
# dmesg > conf_noyau
```

Editez ce fichier conf_noyau .

Vous pouvez l'imprimer et y noter les modifications que vous allez y apporter .

Vous également imprimer /boot/**config** ce fichier est le résultat de make menuconfig : Surtout ne l'éditez pas pour éviter les erreurs , mais copiez le sous un autre nom et éditez la copie .

Le menu va nécessité des réponse du genre "y" pour activer et "n" pour enlever et "m" pour mettre en module .

Par exemple dans la section *Network device support* , pour le gestionnaire permettant d'attribuer 2 adresses à la même machine :

CONFIG_DUMMY = j'écris "n" si je n'en veut pas , "y" si je veux l'avtiver et "m" si je l'utilise comme module

dans la section *Kernel hacking* , **CONFIG_PROFILE** n'a d'intéret que pour une version Alpha ou Beta du noyau , si vous répondez "y" vous aurez un noyau plus gros et plus lent voir instable .

2.2–Compiler le Noyau

Après avoir créer les dépendances avec "make dep" ,vous allez réaliser une image "zImage" Vous tapez :

```
# make zImage
```

Si vous voulez seulement tester le nouveau noyau en le créant sur une disquette , à la place de make zImage ,tapez :

```
# make Zdisk
```

Suivant la puissance de votre processeur le temps de compilation peut varier de 10 mn à + 1 heure : D'où l'importance de passer un peu de temps sur la configuration du noyau pour vous éviter de devoir recommencer la recompilation en cas d'erreur et surtout si vous avez un 386 sx16 ?!.

Bon ! normalement cette image doit être créée dans /usr/src/linux/arch/i386/**zImage**

Mais votre noyau ? :

```
# make zlilo
```

Si vous utilisez LInux LOader , cette commande installera le noyau sous le nom de **vmlinuz** ou **vmlinux** (suivant la version) dans le répertoire /usr/src/linux/ ou à la racine ou encore pour RedHat dans /boot/

Pour une image de "boot" sur disquette, servez vous de /usr/src/linux/arch/i386/boot/zImage , que vous pouvez utiliser en la copiant en vmlinuz, à la racine ou dans /boot/ suivant votre distribution, au lieu d'utiliser make zlilo ou make boot pour RedHat .

Ensuite suivant, votre distribution, la version du noyau et surtout si vous avez installé une carte SCSI, il vous faudra recréer une image **initrd** avec la commande(pour RedHat) : **mkinitrd /boot/initrd.img 2.0.36** , 2.0.36 étant la version du noyau et le nom du répertoire contenant les modules(dans /lib/modules/) :Pour cela le périphérique "loopback" doit être disponible !

3 –En bref , la procédure !

```

_Récupérer les sources .
_Installer les sources .
_Vérifier les liens .
_Et on commence :
# cd /usr/src/linux
# make mrproper
# make menuconfig
# make dep
# make clean
# make zImage
# make zliilo
# make modules
# make modules_install

```

Avec les versions 2.2.xx , le noyau est plus volumineux et vous pouvez recevoir un message d'erreur vous signalant de réduire votre noyau en utilisant **make bzImage** à la place de **make zImage**, suivez ce conseil .

Précaution :

Avant de relancer le système , il est recommandé de modifier `/etc/lilo.conf` pour pouvoir accéder à l'ancien noyau :

Renommer votre ancien noyau (que vous avez sauvegardé) par exemple en `vmlinuz-2.2.xx` (ou `vmlinuz-2.0.xx`) et éditez `lilo.conf` pour que celui-ci vous donne la possibilité de démarrer avec l'ancien noyau (avec la touche 'tab') .

Voici un exemple de ce que peut contenir `/etc/lilo.conf` :

```

#/etc/lilo.conf
#
#remplacer /dev/hda2 par votre partition linux ,
# le repertoire et le nom du noyau /boot/vmlinuz .

# boot=/dev/hda #si LILO dans le MBR
boot=/dev/hda2 # LILO sur la partition Linux
map=/boot/map
install=/boot/boot.b
keytable=/boot/fr-latin1.ktl
prompt
timeout=50
image=/boot/vmlinuz
    label=linux
    initrd=/boot/initrd
    root=/dev/hda2
    read-only
image=/boot/vmlinuz-2.2.xx
    label=old
    initrd=/boot/initrd
    root=/dev/hda2
    read-only
other=/dev/hda1
    table=/dev/hda
    label=dos

```

En cas de problème avec le nouveau noyau , au démarrage de LILO , appuyez sur [tab] et ceci apparait :

```
linux old dos
```

tapez **old** et vous retrouverez votre ancien système.

Pour ceux qui utilise une carte SCSI , il vous faudra recréer un fichier initrd.img avec la commande :

mkinitrd /boot/initrd.img 2.2.xx

Consultez la doc de votre distribution et des sources /usr/src/linux/drivers/scsi/

4 –Des adresses utiles :

www.kernel.org les sources du noyau, site officiel.

[ftp.kernel.org](ftp://ftp.kernel.org) les sources du noyau sur FTP

[ftp.lip6.fr/pub/linux/kernel/sources/](ftp://ftp.lip6.fr/pub/linux/kernel/sources/) les sources sur un miroir FTP

www.traduc.org/kernelfr Documentation FR pour le menu de compilation du noyau.

Les Modules Chargeables

Avec les anciennes version du noyau ,les modules devaient être chargés et déchargés

manuellement par les commandes :

```
# insmod  Insertion du module dans le noyau
# rmmod   Supression du module dans le noyau
# lsmod   Affichage des modules présents dans le noyau
```

La gestion des modules depuis la version 2 du noyau est automatique , le noyau charge les modules qui sont appelés et les décharge lorsqu'ils ne sont plus utilisés .

Notion : [A quoi ça sert les modules ?](#)

Considérons le noyau comme un bloc(objet) monolithique (en un seul morceau) : Ce bloc est statique et peut devenir volumineux . Comme vous l'avez vu , à chaque fois que vous ajoutez un périphérique , il vous faut recompiler ce bloc monolithique pour lui ajouter les gestionnaire des périphérique concernés : Cela présente l'inconvénient d'y inclure des gestionnaires pour des périphériques fort peu utilisés et qui seront chargés au démarrage du système diminuant ainsi les ressources disponibles . Et votre noyau risque de devenir très volumineux et lent .

[Pourquoi des modules chargeables ?](#)

Imaginez maintenant que pour réduire notre noyau monolithique nous le configurons avec les modules . Notre noyau ne sera plus monolithique mais aura une partie statique et une partie dynamique .

La partie dynamique va gérer les modules et la partie statique contiendra le strique nécessaire pour lancer notre système . Il faut évidemment que cette dernière contiennent un minimum d'options ! ... comme le support pour le disque à parir duquel "Boot" vôte système IDE ou SCSI et le type de fichiers ext2fs pour Linux et également NFS si votre machine utilise un répertoire /usr2/ situé sur un autre disque dur du réseau (local) .

Installation :

Les Modules sont situés dans le répertoire /lib/modules/x.x.x/ (numéro de la version)

Ce répertoire contient un fichier README ,cela peut aider !

Si vous avez récupéré les modules pour une ancienne version du noyau vous devez décompresser le fichier en faisant :

```
# tar zxvf modules-x.x.x.tar.gz puis :
# cd /lib/modules/x.x.x et:
# jed README par exemple ...
```

Avec la version 2. du noyau , les modules sont récupérable séparément depuis un site FTP dans :

/modules/2.0.34/cdrom/**Sony535.o** pour votre lecteur deCDRom Sony 535 .

Avec la version 2.0.30 un grand nombre de modules sont inclus .

La configuration du noyau :

Pour activer la gestion des modules , ces options sont indispensables :

Votre menu (en couleur) apparait avec # **make menuconfig**

Vous devez tapez "y" "n" ou "m" ,m passe le gestionnaire en module .

Dans la section :

```
# Loadable module Support
```

CONFIG_MODULES = répondez "y" ,Active les modules chargeables

CONFIG_MODVERSION = personnellement je répons "n" ,cela n'autorise pas d'autres modules que ceux compilés avec le noyau et implique la recompilation des modules à chaque mise à jour du noyau . Si vous répondez "y" ,Vous autorisez toutes les versions des modules présents .?

CONFIG_KERNELD = répondez "y" ,Active le chargemet Automatique des modules .

dans la section :

```
# General Setup
```

CONFIG_SYSVIPC = répondez "y" ,c'est le système de communication

Inter-processus utilisé par le noyau pour communiquer avec le démon qui automatise

le chargement des modules .

Otez-moi d'un doute ?!!

Pour connaître les options des différentes sections du menuconfig ,regardez le chapitre *recompilation du noyau* dans le **Guide du Rootard** qui détaille ce que vous pouvez y faire en fonction de votre configuration : Disque E-IDE ATAPI SCSI ,carte Ethernt Audio

Pour connaître les options de votre noyau actuel : vous enregistrez les messages de lancement dans un fichier :

dmesg > Options_du_Noyau et vous éditez ce fichier .

Les Modules :

Ils sont situés dans **/lib/modules/2.0.x/** ou **/lib/modules/current/** si vous avez installé différentes versions .

Ils seront mis en place dans un certain ordre par le démon **kerneld** à l'aide du fichier **/etc/rc.d/rc.modules** et "kerneld" est lancé par le fichier **/etc/rc.d/rc.s** , encore un , **/etc/conf.modules** indique chemin (PATH) des modules .

Installation des Paches

Je vais faire très court ,car à cet instant précis où j' écris ces lignes, je n'ai pas du tout envie de commenter l'installation et surtout la résolution des problèmes qui risque de survenir .

Ces explications nécessitent plusieurs pages et cela a déjà été écrit dans le **Kernel HOWTO**

Si non l'installation c'est très simple :

Vous obtiendrez une nouvelle version du noyau en appliquant un patch .

Si votre version est la 2.0.33 , vous allez sur un site de distribution FTP dans : **/pub/linux/kernel/sources/v2.0/** et vous récupérez l'archive **patch-2.0.34.gz** pour passer de la version 2.0.33 à la version 2.0.34 du noyau .

les patches sont réalisés pour la version précédente à laquelle ils s'appliquent :

patch-2.0.34.gz ne peut pas s'appliquer sur une version 2.0.32 .

Pour passer d'une version 2.0.23 à une version 2.0.34, il faudra récupérer **tous les patches** applicables entre ces deux versions.

Note : Si vous venez de découvrir Linux et que vous n'etes pas encore décidé sur le choix d'une distribution : Je vous conseille d'installer une distribution à base d'un noyau 2.2.x . Car cette version apporte des modifications importantes dans la configuration du noyau ,les bibliothèques et la gestion des systèmes de fichiers . Et si vous voulez mettre à jour une ancienne version 1.2.x vers 2.0.x ou 2.0.x vers 2.2.x vous aurez beaucoup à faire ! ..

_Comme je le disais , installer un patch c'est très simple : ça marche ou ça marche pas !? Mais le plus gros du travail est du *nettoyage* ...n'oubliez pas "**make clean**" .

vous copier votre patch dans **/usr/src/** là vous le détarez et vous verrez (certainement que non) une liste afficher les messages des modifications effectuées ,mais cela va trop vite alors ajouter "**-s**" pour afficher les erreurs ou(et) "**-p1**" si vos sources ne se trouvent pas dans **/usr/src/** :

```
# cd /usr/src
```

```
# zcat patch34.gz | patch -sp0 ou # zcat patch34.gz | patch -sp1
```

si tout se passe bien , vous faites :

```
# make clean
```

```
# make config
```

```
# make dep
# make clean
# make zImage
```

Des Problèmes ?!

Cela se situe certainement au niveau de "config.in" :
rechercher le fichier "config.in.rej" # find . -name "*.in.rej"
Editez le pour voir les modifications apportées par le patch : un "+" ou un "-" vous indique si cette option a été ajoutée ou enlevée : Si cela ne correspond pas à votre configuration éditez config.in et changez les "y" et les "n" si vous le jugez utile .
Il existe un site miroir de ftp.kernel.org ou vous pouvez récupérer les sources :
ftp.lip6.fr/pub/linux/kernel/sources/

Dernière mise à jour: **12-09-2000**

Configuration du Système

**Certainement la première chose à faire, est de compiler un noyau adapté à votre système .
Ce qui suit, n'est qu'un petit plus qui ne peut en aucun cas rivaliser avec un noyau trop lent
ou ne reconnaissant pas vos périphériques !**

Configuration des périphériques

- [Pilotes et périphériques](#)
 - [Liste des CDs et graveurs reconnus](#)
 - [Liste des cartes SCSI reconnues](#)
 - [Montage Auto des partitions](#)
-

Périphériques et pilotes :

Il est important que vos périphériques soit correctement identifiés .
Et pour qu'il vous soit plus simple de les nommer lors d'une configuration , vous allez créer un lien symbolique du périphérique vers un autre plus "parlant" ,par exemple votre modem:

Si votre modem utilise le port COM2 sous Dos ,sous Linux ce sera /dev/cua1 pour les appels sortants et /dev/ttyS0 pour les appels entrants .Et vous allez établir le lien suivant :

In -s /dev/cua1 /dev/modem

Il vous sera plus facile de vous souvenir de /dev/modem ?!!

dev = abréviation de device (périphérique).

Répéter la commande pour :

In -s /dev/cua0 /dev/mouse si votre souris utilise le port COM1

In -s /dev/hdc /dev/cdrom si votre CD est en Master sur le second controlleur IDE

In -s /dev/sg0 /dev/scanner pour votre scanner SCSI

Une carte **RNIS** (numéris) utilise **/dev/ttyl0** .

In -s /dev/ttyl0 /dev/modem

Pour les distributions les plus récentes `/dev/cuax` n'est plus utilisé et `/dev/ttySx` le remplace. Si vous utilisez une souris **PS/2** et que `/dev/psaux` n'existe pas, vous pouvez le construire de cette façon :

MAKEDEV busmice

Vérifier que `/dev/psmouse` a bien été créé, puis faire :

In -s /dev/psmouse /dev/psaux

Vous devez appliquer un **chmod 666** sur ces périphériques pour que tout le monde puisse les utiliser .

chmod 666 /dev/cua0 /dev/cua1 /dev/hdc /dev/sg0

De la même façon vous pouvez rendre utilisable pour les autres utilisateurs, le mixeur et périphériques audio :

/dev/mixer , /dev/dsp et /dev/audio

Vous trouverez la liste des périphériques dans "Les fichiers Spéciaux "(section 12) du [Guide-du-ROOTard](#) .

Les pilotes:

D'une part, sous Linux vous sélectionnez ([X] [M]) votre périphérique à la compilation du noyau, intégré au noyau [X] (ou en module [M]) qui gèrera (via un module) votre CD ou votre Ctrl SCSI par exemple .

D'autre part pour des périphériques comme la carte graphique ou le scanner entre autres ,ce sera XFree ou l'application qui installera les bibliothèques et le pilote approprié : Pour votre scanner "Mon-Scan-Scsi" de marque "Mon-Scan", Sane (Scanner Access Now Easy) fournira "libsane-Mon-Scan.so" ...etc.

Une parenthèse à propos des cartes graphiques :

XFree86 installe le serveur **X** contenant un certain nombre de pilotes pour différents circuits "chips"

Pourquoi ma carte n'affiche-t-elle pas les 1024x768 en 24 bits sous Linux alors qu'elle le fait sans problème sous Windows ???

Si vous avez installé Linux à partir d'une distribution comme la RedHat, vous n'avez probablement pas remarqué le choix des différents pilotes (serveur X) disponibles ! Linux classe ces serveurs par "chipset" et non par fabricant car ceux-ci peuvent très bien utiliser un S3 86C pour un modèle et un ET4000 sur un autre : Il est donc important, pour que les développeurs Linuxiens et bénévoles puissent réaliser un pilote pour votre carte, que le fabricant communique la doc technique indispensable à la gestion du chipset qui équipe celle-ci !!?

Mais comme certains fabricants refusent tout support de leur carte 3D pour Unix/Linux !!!...?

Pour avoir la liste du matériel reconnu par Linux et obtenir la liste des cartes graphiques disposant d'un pilote pour Xwindow, *Vous devez absolument consulter le [Hardware-HOWTO](#) et le [XFree86-HOWTO](#) .*

Les CDs et graveurs reconnus sous Linux :

- SCSI CD-ROMS
- ATAPI CDROMS
 - Aztech/Orchid/Okano/Wearnes/TXC/CyDROM
 - Conrad TXC
 - CyCDROM CR520ie/CR540ie/CR940ie
 - Goldstar R420
 - Matsushita/Panasonic/Creative, Longshine, TEAC
 - Mitsumi (standard) [no XA/Multisession]
 - Mitsumi [XA/MultiSession]
 - NEC CDR-260 - Optics(Lasermate CR328 A, Storage DOLPHIN 8000AT)
 - Philips/LMS CM206
 - Sanyo CDR-H94A
 - Sony (CDU-510, CDU-515, CDU-535, CDU-531, CDU31A/CDU33A)
 - ISP16/MAD16/Mozart

Améliorer sa Configuration | Linux

GRAVEURS CD:

- All SCSI-3/mmc compliant drives
- All ATAPI/mmc compliant drives ("It seems that all drives that have been in 1999 or later are MMC compliant. If a recent drive does not work with likely found a firmware bug.")
- Philips CDD 521
- COMPRO/Matsushita/Panasonic CW-7502
- ARTEC 4X/4X RW/24X
- Philips CDD-2000
- TEAC CD-R55S
- Yamaha CDR-400
- ACER CR-1420C
- ACER CDRW 6206A
- BTC BCE 621E
- COMPRO CW-7502
- Dysan CR-622 ???? See Wearnes 622
- Dysan CR-1622
- DynaTec CDM-240J (see Pinnacle RCD-4x4)
- DynaTec CDM-240 (use cdrecord driver=yamaha_cdr100)
- DynaTec CDM-400 (use cdrecord driver=yamaha_cdr100)
- Grundig CDR-100
- Hewlett Packard 4020i
- Hewlett Packard 6020i
- HP C4324/C4325 (HP SureStore 4020i/6020i)
- HP 7100
- HP 7110
- HP 7200
- HP 8100
- Hewlett Packard 9100 Series
- Hi-Val CD-R (see Pinnacle RCD-4x4)
- JVC XR-W2001 (uses TEAC code - see below - audio not working)
- JVC XR-W2010 (uses TEAC code - see below - audio not working)
- JVC XR-W2020 (uses TEAC code - see below - audio not working)
- JVC XR-W2080 (first MMC drive from JVC)
- Kodak PCD-200 or Kodak PCD-200 Plus
- Kodak PCD-225
- Kodak PCD-240
- Kodak PCD-600
- Matsushita CW-7502
- Matsushita CW-7582
- Memorex CR-622 ???? See Wearnes 622
- Memorex CR-1622
- Memorex CRW-1622
- Microboards PlayWrite 2000 (use cdrecord driver=sony_cdu924)
- Microboards PlayWrite 4000 (use cdrecord driver=yamaha_cdr100)
- Microboards PlayWrite 4001RW
- MicroNet MasterCD Plus 4x4 (use cdrecord driver=yamaha_cdr100)
- MicroNet MasterCD Plus 4x6
- Mitsubishi MCA-CDRW 226
- Mitsumi CR-2401-TS
- Mitsumi CR-2600-TE
- Mitsumi CR-2801
- Mitsumi CR-4801
- Nomai 680.rw
- Olympus CDS615E
- Olympus CDS620E (use cdrecord driver=sony_cdu924)
- Olympus CD-R2x6 (use cdrecord driver=sony_cdu924)
- Optima Dis Koverly 650 CD-R (use cdrecord driver=sony_cdu924)
- OTI CDRW 965
- Panasonic CW-7502
- Panasonic CW-7582
- Philips CDD 521 (CDD521/02 Revision: 2.06 has bad firmware)
- Philips CDD 521 (upgraded units only: ID: CDD521/10 Revision: 2.07)
- Philips CDD 522

Améliorer sa Configuration | Linux

- Philips CDD 2000
- Philips CDD 2600
- Philips CDD 3600
- Philips CDD 3610
- Philips Omniwriter 26
- Philips Omniwriter 26A
- Pinnacle Micro RCD-1000 (see TEAC/JVC): Need to upgrade firmware to 2.3)
- Pinnacle Micro RCD-5020 (see TEAC/JVC - audio not working)
- Pinnacle Micro RCD-5040 (see TEAC/JVC - audio not working)
- Pinnacle Micro RCD-4x4
- Pioneer DW-S114X
- Plasmon CDR 4220 (not tested)
- Plasmon RF-4100
- Plasmon RF-4102
- Plasmon CDR 4400 (use cdrecord driver=yamaha_cdr100)
- Plasmon CDR 480
- Plextor PX-R24CS (use cdrecord driver=ricoh_ro1420c)
- Plextor PX-R412C
- Plextor PX-R820C
- Procom PCDR 4 (use cdrecord driver=yamaha_cdr100)
- Ricoh RO-1420C
- Ricoh MP-6200
- Ricoh MP-6200I
- Ricoh MP-6201
- Smart Friendly CD-R1002 (use cdrecord driver=sony_cdu924)
- Smart Friendly CD-R1004 (use cdrecord driver=yamaha_cdr100)
- Smart Friendly CD-R2004 (use cdrecord driver=sony_cdu924)
- Smart Friendly CD-R2006 PLUS
- Smart Friendly CD-R2006 PRO
- Smart Friendly CD-R4000 (use cdrecord driver=yamaha_cdr100)
- Smart Friendly CD-R4006
- Smart Friendly CD-R4012
- Smart Friendly CD-RW226
- Smart Friendly CD-R8020
- Sony CDU920S
- Sony CDU924S
- Sony CDU926S
- Sony CDU928
- Sony CDU940S
- Sony CDU948S
- Sony CRX-100
- Taiyo Yuden CD-WO EW-50
- TEAC CD-R50S
- TEAC CD-R55S
- TEAC CD-R56S
- That's Write CDR-N820S
- Traxdata CDRW2260+
- Traxdata CDR-4120
- Traxdata CDRW-4260
- Turtle Beach 2040R (use cdrecord driver=ricoh_ro1420c)
- Wearnes CD-R622
- Wearnes CD-R632P
- Yamaha CDR-100
- Yamaha CDR-102
- Yamaha CDR-200
- Yamaha CDR-400 (Firmware revision 1.0d and up otherwise upgrade)
- Yamaha CDR-401
- Yamaha CRW-4001
- Yamaha CRW-2260
- Yamaha CRW-4260
- Yamaha CRW-4416

Les cartes SCSI reconnues sous Linux :

- 7000FASST SCSI

- ACARD
- Adaptec(AHA 152x, AHA-174x, AHA-294x(W/UW/D/WD), AHA-398x(W), AHA-274x EISA, AHA-284x VLB; Adaptec, AIC-777x, AIC-785x, AIC-787x, AIC-788x, AHA PCI, Adaptec AHA-2940(AU/U2W/U2/U2B/U2BOEM), AHA-274xT, AHA-2842, AHA-2910B, AHA-2920C, AHA-2930(U/U2), AHA-2944(D/WD/UD/UWD), AHA-3940(U/UW/AUW/U2W/U2B), AHA-3950U2D, AHA-3985(U/UW), AIC-786x, AIC-789x, AIC-3860, Adaptec 1520B, Adaptec 1522B, Adaptec 1542)
- AdvanSys(ABP510, ABP5150, ABP5140, ABP5142, ABP902, ABP915, ABP920, ABP930, ABP930U, ABP930UA, ABP960, ABP960U, ABP542, ABP742, ABP842, ABP940U, ABP940UA, ABP970, ABP970U, ABP940UW, ABP970UW, ABP752, ABP852, ABP950, ABP950UW, ABP980, ABP980U, ABP980UA)
- Always IN2000
- AM53/79C974 PCI
- BusLogic(Mylex, MultiMaster, FlashPoint)
- Data Technology Corp DTC3180/3280
- DTC3180/3280
- EATA-DMA (DPT, NEC, ATT SNI, AST, Olivetti, Alphasatronics)
- EATA-PIO (old DPT PM2001, PM2012A)
- EATA ISA/EISA/PCI (DPT and generic EATA/DMA-compliant boards)
- Future Domain 16xx
- GDT SCSI Disk Array Controller
- Generic NCR5380/53c400
- Initio 9100U(W)
- Inition INI-A1000U2W
- NCR(5380, 53c810, 53c810A, 53c815, 53c825, 53c825A, 53c860, 53c875, 53c875, 53c895 53c400, NCR5380, 53c810, 53c810A, 53c815, 53c820, 53c825, 53c810, 53c720, 53c700, 53c710, 53c700-66, 53c406A)
- IOMEGA Parallel Port ZIP drive
- PAS16
- Perceptive Solutions PSI-240I
- PCI2000, PCI2220i
- PSI240i
- Qlogic FAS
- Qlogic ISP(FC)
- Seagate ST-01/ST-02 and Future Domain TMC-8xx
- SYM83C8XX
- Tekram DC-390(T)
- Trantor T128/T128F/T228
- UltraStor 14F/24F/34F
- Western Digital (WD7000-FAST, WD7000-FAST2, WD7000-ASC, WD7000-AX/MX/EX)

Montage Auto des partitions :

Vous avez un lecteur de disquette **/dev/fd0** à monter dans **/mnt/floppy**

et un CDRom **/dev/cdrom** à monter dans **/mnt/cdrom**

si votre cdrom est en master sur le 2eme controleur IDE, **/dev/cdrom** sera un lien symbolique de **/dev/hdc** :

ln -s /dev/hdc /dev/cdrom

Montage Auto avec fstab

Pour monter par exemple votre CDRom, **/dev/cdrom** (lien symbolique de **/dev/hdc**), dans le répertoire de montage **/mnt/cdrom** : Vous devez taper cette ligne

mount -t iso9660 -o ro /dev/cdrom /mnt/cdrom ou **mount -t iso9660 -o ro /dev/hdc /mnt/cdrom** ce qui revient au même

Pour automatiser cette manipulation , il faut ajouter dans **/etc/fstab** :

/dev/cdrom /mnt/cdrom iso9660 noauto,ro,user 0 0

Et taper la commande **mount /dev/cdrom**

noauto : Ne pas monter automatiquement la partition au démarrage .

ro : Monter en lecture seulement (Read Only).

user : Autorise un utilisateur non su à monter la partition

Ensuite les chiffres indiquent le nombre de jours entre deux traitements de la partition (commande dump pour ext2fs), suivi de l'ordre de tests des partitions (fsck): 0 = aucune vérification automatique n'est effectuée lors du démarrage.

Il existe d'autres options possible ,vous pourrez trouver par exemple pour le montage d'un CDRom SCSI une ligne comme:

```
/dev/scd0 /mnt/cd iso9660 user,exec,dev,nosuid,ro,noauto 0 0
```

[Des explications plus détaillées se trouvent dans le chapitre Les Lecteurs de CDRom du SCSI-HOWTO .](#)

Soit vous tapez la commande **mount /dev/cdrom**,soit vous utilisez un utilitaire de montage sous X comme **xmount** ou **usermount** .

Pour le montage Auto des partitions, les infos sont contenues dans le fichier **/etc/fstab** ,puis mis à jour dans le fichier **/etc/mtab**.

Le fichier mtab contient les partitions montées :Surtout ne modifiez pas ce fichier, vous risquez d'endommager vos partitions car à l'arrêt du système le fichier mtab permet de démonter les partitions montées !!!

Vous trouverez un exemple du fichier fstab dans [config.tgz](#)

Automontage avec autofs

Pour autoriser les utilisateurs à monter les périphériques amovibles (CDRom et Disquette), vous utilisez dans ce cas l'option "user" dans la ligne de montage du fichier /etc/fstab : Mais cela enlève les droits d'exécution sur le CD ainsi monté .

L'administrateur (root) peut utiliser quelques subtilités pour [corriger ce problème](#), mais une autre solution plus élégante consiste à utiliser **autofs** = l'Automonteur de SUN .

_Conditions àremplir :

Installer autofs

compiler le noyau, section FileSystems, avec l'option Kernel automounter support

.... autofs est inclus dans le noyau pour les version 2.0.31 et +

pour vérifier si autofs est actif, tapez la commande :

```
cat /dev/filesystems
```

```
ext2
nodev  proc
      iso9660
nodev  autofs
```

Configuration :

ATTENTION ! IMPORTANT :

_L'automontateur n'autorise pas le montage directement à la racine du disque, mais uniquement dans un sous-répertoire .

_Les répertoires d'aumontage doivent exister mais être vide jusqu'au montage, car l'automonteur en masquera le contenu .

_Chacun de ces répertoires correspondra à une liste de montage déclarées dans le fichier /etc/auto.master .

Vous allez "automonter" vos lecteurs de disquettes, de CDs et unités de sauvegarde dans le répertoire d'automontage **/misc** , en créant les sous répertoires :

```
cd /
```

```
mkdir misc
```

```
mkdir misc/floppy
```

```
mkdir misc/cdrom
```

```
mkdir misc/jaz
```

Ces répertoires doivent être vides !

Pour plus d'info, consultez les pages de manuel : **man autofs**

Maintenant vous allez définir les options de montage dans les fichiers de configuration de l'automonteur **autofs** :

/etc/auto.master

```
# /etc/auto.master fichier de configuration de autofs
#
# $Id: auto.master,v 1.2 1997/10/06 21:52:03 hpa Exp $
# Sample auto.master file
# Format of this file:
# mountpoint map options
# For details of the format look at autofs(8).

/misc /etc/auto.misc --timeout 60
```

/etc/auto.misc

```
# /etc/auto.misc automontage des unités
#
# $Id: auto.misc,v 1.2 1997/10/06 21:52:04 hpa Exp $
# This is an automounter map and it has the following format
# key [ -mount-options-separated-by-comma ] location
# Details may be found in the autofs(5) manpage

floppy          -fstype=auto           :/dev/fd0
cdrom           -fstype=iso9660,ro    :/dev/cdrom
jaz            -fstype=ext2           :/dev/sdcl
```

Pour relancer l'automonteur, tapez la commande :

/etc/rc.d/init.d/autofs restart

La commande **ls /misc/cdrom** , provoquera automatiquement le montage du CDRom sous **/misc/cdrom** avant d'afficher son contenu

Les répertoires **/misc/floppy** , **/misc/cdrom** et **misc/jaz** ne seront visibles que lorsque l'automontage aura été effectué .

L'automonteur permet bien plus que le simple montage des unités disques et partitions :

Utiliser en réseau avec NFS et NIS il devient un puissant outils de montage pour les répertoires partagés et exportés .

Brièvement prenons un petit exemple, considérons :

_D'une part votre machine avec laquelle vous travaillez sur des docs Linux .

_D'autre part un serveur "**booboo**" sur votre réseau, qui utilise une base de données (PostgreSQL), pour stocker ces importantes docs Linux, classée en deux grandes catégories : doc en Français et doc en Anglais respectivement dans un répertoire **../pgsql/france** et **../pgsql/usa** sur ce serveur .

1 :Vous ne voulez pas monter ces docs trop volumineuses systématiquement au démarrage de votre machine :

+ 2 :Le partage va être réalisé par **NFS** et le montage des volumes de nfs par **autofs** .

= Voilà du travail pour l'automonteur :

Créez un répertoire **/travail** . ;-))

Puis ajouter au fichier **/etc/auto.master** :

```
# /etc/auto.master fichier de configuration de autofs
#
/misc /etc/auto.misc --timeout 60
/travail /etc/auto.travail
```

Et créer un fichier **/etc/auto.travail** :

```
# /etc/auto.travail automontage des bases de données
#
db-france      -rsize=8192,wsiz=8192,bg,soft,intr    booboo:/opt/db/pgsql/france
db-usa        -rsize=8192,wsiz=8192,bg,soft,intr    booboo:/opt/db/pgsql/usa
```

Remarquez que le nom de machine précédant les deux points ":" suffit pour déterminer que le type de fichier est nfs

Pour plus d'info sur les options utilisables, consultez les pages de manuel : [man autofs](#)

Bien évidemment les répertoires **/travail/db-france** et **/travail/db-usa** doivent exister et être vides avant l'automontage .

L'utilisation est encore plus pratique avec **NIS** en exportant les deux fichiers **auto.master** et **auto.travail** pour les rendre accessibles à tous les utilisateurs du réseau . Dans ce cas vous (votre machine et les autres clients) n'aurez plus besoin du fichier **/etc/auto.travail**, c'est NIS qui chargera du travail :-))

Pour en savoir plus sur la configuration de ce type de montage, regardez la page [Configuration NIS et NFS](#).

Dernière mise à jour: **12-09-2000**

Configuration des périphériques

- [Le Clavier](#)
 - [La souris](#)
 - [L'imprimante](#)
 - [Liste des imprimantes supportées](#)
 - [Le DVD](#)
-

Le Clavier :

Avec la **Slackware** le menu setup vous permet de changer de table .

Si non pour choisir une table correspondant à votre type de clavier , allez dans **/usr/lib/kbd/keytables** et pour un clavier français prenez **fr-latin1.map** et éditez **/etc/sysconfig/keyboard** et modifiez la ligne :

```
KEYTABLE="/usr/lib/kbd/keytables/fr-latin1.map"
```

Sous X :

Pour un clavier FR azerty 101 touches, dans la section "**Keyboard**" de XF86Config :

```
Section "Keyboard"
  Protocol "Standard"
  XkbRules "xfree86"
  XkbModel "pc101"
  XkbLayout "fr"
EndSection
```

_Ajoutez aussi les touches ,fin et début de ligne,supprime et retour arrière, pour la saisie en mode console , regardez ci-dessous le fichier **\$HOME/.inputrc** .

```
# caractères accentués 8 bits
set meta-flag on
set convert-meta off
set output-meta on
# Touches d'édition
"\e[1~":beginning-of-line
"\e[4~":end-of-line
"\e[3~":delete-char
"\C-h": backward-delete-char
"\e\C-h": backward-kill-word
"\C-?": backward-delete-char
"\e\C-?": kill-word
"\e\e[3~": kill-word
"\e\[A": "info \C-m"
# pour rxvt
"\e[7~":beginning-of-line
"\e[8~":end-of-line
# pour xterm
"\e[H": beginning-of-line
"\e[F": end-of-line
```

_Pour avoir le pavé numérique au démarrage du système , éditez **/etc/rc.d/rc.local** et ajoutez ces lignes :

```
# Num Lock ON
echo "Vérouillage Num-lock"
for tty in /dev/tty[1-9]*;
do
setleds -D +num < $tty > /dev/null
done
# end
```

Sous X :

Pour Xterm :

Pour que les touches retour et supprime [BckSp] et [del] fonctionne dans une fenêtre xterm , ajouter ces lignes dans ~/.xinitrc :

```
usermodmap = $HOME/.Xmodmap
xmodmap $usermodmap
puis éditer ~/.Xmodmap pour ajouter :
Keycode 22 = BackSpace
Keycode 107 = Delete
```

et enfin modifier ~/.Xdefaults comme suit:

Pour utiliser nxterm ou un autre, commentez les lignes existantes (xterm*VT100) et en ajoutez quelque chose comme :

```
# Utilisation des touches début , fin et supprime :
# pour XTerm
#xterm*VT100
#
```

...Cela ne fonctionne pas forcément avec nxterm et kterm ! mais essayez?! :

```
# pour nxTerm
nxterm*VT100
```

La Souris :

En mode console il est pratique d'effectuer du copier-coller, pour cela vérifiez :

1) que gpm est installé :

type gpm

2) que gpm est bien lancé au démarrage du système , en promenant votre souris un curseur doit apparaitre à l'écran et vous devez pouvoir surligner du texte avec le bouton gauche.

Sur une **RedHat** un fichier /etc/rc.d/init.d/gpm initialise cette fonction et contient ces lignes :

```
. /etc/rc.d/init.d/functions
MOUSECFG=/etc/sysconfig/mouse
case "$1" in
  start)
    echo -n "Starting gpm mouse services: "
    if [ -f "$MOUSECFG" ]; then
      . "$MOUSECFG"
    else
      echo "(no mouse is configured)"
      exit 0
    fi

    if [ "$MOUSETYPE" = "none" ]; then
      echo "(no mouse is configured)"
      exit 0
    fi

    if [ "$MOUSETYPE" = "Microsoft" ]; then
      MOUSETYPE=ms
    fi

    if [ -n "$MOUSETYPE" ]; then
      daemon gpm -t $MOUSETYPE
    else
      daemon gpm
    fi
    echo
    touch /var/lock/subsys/gpm
  ;;
```

si gpm n'est pas opérationnel, éditez **/etc/rc.d/rc.local** , et ajoutez :

_pour une souris Microsoft :

Running gpm for microsoft mouse

echo "lancement de gpm"

gpm -t ms

end

_pour une souris USB :

Running gpm for USB mouse

echo "lancement de gpm"

gpm -m /dev/usbmouse -t imps2

end

_Vérifiez également que vous avez un fichier /etc/sysconfig/mouse , qui contient :

pour une souris Microsoft ,

MOUSETYPE="Microsoft"

XEMU3=yes # pour une souris 2 boutons

pour une souris PS/2 , cela peut ressembler à :

MOUSETYPE="ps/2"

XMOUSETYPE="GlidePointPS/2"

FULLNAME="ALPS GlidePoint (PS/2)"

XEMU3=yes

Sous X :

_Editez le fichier /etc/X11/XF86Config ou /etc/rc.*/XF86Config ou /etc/XF86Config

Dans la section "**Pointer**" :

Pour une souris Microsoft :

```
Protocol "Microsoft"
Device "/dev/mouse "
```

Pour une souris PS/2 :

```
Protocol "ps2"
Device "/dev/psaux "
```

Pour construire le port psaux : **MAKEDEV busmice** puis **In -s /dev/psmouse /dev/psaux**

Pour une souris Logitech MouseMan :

```
Protocol "MouseMan"
Device "/dev/mouse "
BaudRate 1200
Resolution 100
Buttons 3
```

Pour une souris à molette :

```
Protocol "Intellimouse"
Device "/dev/mouse "
BaudRate 1200
Resolution 100
ZaxisMapping 4 5
Buttons 3
```

Pour une souris USB :

```
Protocol "IMPS/2"
Device "/dev/usbmouse"
```

Pour construire le port usb : **mknod /dev/usbmouse c 10 32**

Pour MultiMouse :

```
Protocol "mousesystems"
Device "/dev/mumse"
```

L'utilitaire MultiMouse de Takashi Manabe ne reconnaît pas les tablettes graphiques Wacom .

Pour une tablette graphique :

```
Protocol "Microsoft"
Device "/dev/null"
```

L'utilisation des tablettes graphiques Wacom ou Summagraphics, nécessitent la présence du module /usr/X11R6/lib/modules/xf86Wacom.so (pour Wacom) et doit être déclaré dans la section "**Modules**" par la ligne :

Load "xf86Wacom.so" # (pour Wacom).

et la section "**Xinput**" (toujours pour Wacom) doit ressembler à :

Section "Xinput"

```
SubSection "WacomStylus"
```

```
Port "/dev/ttyS1"
```

```
DeviceName "Wacom"
```

```
AlwaysCore
```

```
EndSubSection
```

```
SubSection "WacomCursor"
```

```
Port "/dev/ttyS1"
```

```
AlwaysCore
```

```
EndSubSection
```

```
SubSection "WacomEraser"
```

```
Port "/dev/ttyS1"
```

```
AlwaysCore
```

```
EndSubSection
```

```
EndSection
```

#remplacez /dev/ttyS1 par /dev/mouse si vous n'avez pas d'autres pointeurs que la tablette

L'imprimante :

Sous **RedHat** vous pouvez utiliser **printtool** pour configurer votre imprimante, sous **SuSE** utilisez **Yast** .

Pour ceux qui utilise le gestionnaire de fenêtres **KDE** (version 1.1.2) il existe un assistant pour la configuration de votre imprimante, "Printer Wizard" qui vous aidera pour configurer votre imprimante.

Le paramétrage et la reconnaissance de votre imprimante va dépendre essentiellement de **GhostScript** (gs pour les familiers).

Par exemple sur une **RedHat 6.0** la version de ghostscript est la **5.10** , pour une **SuSE 6.3** c'est la version **5.10-51** : Au moment où j'écris ces lignes la dernière version est la **6.0** récupérable sur : <ftp.cs.wisc.edu/ghost/aladdin/>

```
La mise à jour de : ghostscript-x.x .
                   ghostscript-fonts-std-x.x.
                   ghostscript-fonts-other-x.x.
                   ghostscript-x.xlibpng
                   ghostscript-x.xjpeg
```

est indispensable pour ceux qui désirent imprimer des images depuis GIMP ou leurs photos de vacances depuis un CD Kodak.

Le modèle de votre imprimante ne figure pas dans la liste ?

_Pour imprimer du texte, une définition de 360x360 est suffisante, vous pouvez choisir le modèle précédant ou le modèle générique .

Choisissez un modèle plus ancien de la même marque, Surtout ne prenez pas un modèle HP pour une Epson , sous prétexte qu'elles sont capables d'imprimer toutes les deux en 360x720 !!! ? : Par exemple pour *Epson Stylus*, si le modèle *color 400* n'existe pas dans la version de ghostscript installé :

Je peux utiliser "Epson stylus Color (UP)" pour une *Stylus color 400* et sélectionner dans Les paramètres des couleurs " stancy, Stylus Color any type, 360x360Dpl "

Je pourrais alors imprimer du texte (avec la couleur) et des images avec une définition de 360x360 (Pas terrible !).

_Si l'application, StarOffice, WordPerfect...., ne reconnait pas votre imprimante, sélectionnez une imprimante générique PostScript et utilisez le port et l'imprimante configurés par votre distribution (lp sur /dev/lp0 par défaut) : Les filtres installés se chargeront de faire la conversion .

_Mieux encore , vous pouvez imprimer dans un fichier à la place de sélectionner une imprimante: ce fichier sera au format PostScript (ps) et le "Viewer" **GhostView** (gv) pourra être utilisé pour afficher et imprimer votre document : ATTENTION! tous les dégradés de couleurs ne sont pas toujours correctement pris en compte par GhostView .

Liste des imprimantes reconnues en mode graphique depuis la version 5.10 de GhostScript :

- Apple Dot Matrix Printer (should also work with Imagewriter)
- Apple Imagewriter in high-resolution mode
- Apple Imagewriter in low-resolution mode
- Apple Imagewriter LQ in 320 x 216 dpi mode
- Apple StyleWriter 2x00
- C.Itoh M8510 printer
- Canon BubbleJet BJ10e
- Canon BubbleJet BJ-100/200/210/240
- Canon LBP-8II
- Canon LIPS III
- Canon Color BubbleJet BJC-600, BJC-4000 and BJC-70
- Canon Color BubbleJet BJC-800/2000
- Canon LBP-8II laser printer

Améliorer sa Configuration | Linux

- Canon LIPS III laser printer in English (CaPSL) mode
- CalComp Raster format
- DEC LA50 printer
- DEC LA70 printer
- DEC LA70 printer with low-resolution text enhancement
- DEC LA75 printer
- DEC LA75plus printer
- DEC LN03 printer
- DEC LJ250 Companion color printer
- DEC LJ250 driver
- Epson AP3250 printer
- Epson-compatible dot matrix printers (9- or 24-pin)
- Epson LQ-2550 and Fujitsu 3400/2400/1200 color printers
- Epson LP-8000 laser printer
- Epson LQ850 printer at 360 x 360 DPI resolution
- Epson Stylus 800 printer
- Epson Stylus Color
- HP DeskJet 500C with 1 bit/pixel color
- HP DeskJet 500C with 24 bit/pixel color and gh-quality color (Floyd-Steinberg)
- HP DeskJet 540C and Citizen Projec IIc (-r200x300)
- HP DeskJet 500C printing black only
- HP DeskJet 510, 520, and 540C (black only)
- HP DeskJet 500C (same as cdjcolor)
- HP DeskJet 550C/560C
- HP DeskJet and DeskJet Plus
- HP DeskJet 500
- HP DeskJet 500C alternate driver
- HP DesignJet 650C
- HP LaserJet
- HP LaserJet IID/IIP/III* with TIFF compression
- HP LaserJet III* with Delta Row compression
- HP LaserJet IIID with duplex capability
- HP LaserJet 4 (defaults to 600 dpi)
- HP LaserJet 4 with Floyd-Steinberg dithering
- HP LaserJet Plus
- HP LaserJet 5 6 family (PCL XL), bitmap
- HP LaserJet 5 6 family, gray-scale bitmap
- HP 2563B line printer
- HP PaintJet color printer
- HP PaintJet XL driver
- HP PaintJet XL driver
- HP PaintJet XL color printer
- HP PaintJet XL300 color printer
- HP PaintJet 1200C
- HP black-and-white PCL XL printers (LaserJet 5 and 6 family)
- HP color PCL XL printers (not available yet)
- Imagen ImPress printers
- IBM 9-pin Proprinter
- IBM Jetprinter ink-jet color printer (Model #3852)
- NEC P6/P6+/P60 printers at 360 x 360 DPI resolution
- Mitsubishi CP50 color printer
- OCE 9050 printer
- Okidata MicroLine 182
- Okidata MicroLine IBM-compatible printers
- Ricoh 4081 laser printer
- StarJet 48 inkjet printer
- Sony Microsystems NWP533 laser printer [Sony only]
- Tektronix 4693d color printer, 2 bits per R/G/B component
- Tektronix 4693d color printer, 4 bits per R/G/B component
- Tektronix 4693d color printer, 8 bits per R/G/B component
- Tektronix 4695/4696 inkjet plotter
- Xerox XES printers (2700, 3700, 4045, etc.)

Le DVD :

Ce paragraphe est daté car il risque (je l'espère) de devenir rapidement obsolète: Le **02-03-2000** Ici des solutions ne sont pas vraiment expliquées, mais sont simplement exposés les problèmes .

Les problèmes juridiques subsistent : " Comment développer un logiciel de décryptage en GPL, alors que les clefs de décodage sont commercialisées ?!!"

_Tout d'abord, un soutien aux développeurs Norvégiens, qui ont réalisé "DeCSS" un utilitaire de décodage de DVD-video (cryptés) à la suite de la commercialisation d'un logiciel de lecture dont les clefs de décodages n'étaient pas cryptées ? , et qui maintenant se retrouvent, poursuivis en justice et en infraction vis à vis des lois sur les brevets des algorithmes et de l'industrie du DVD .
... Avec un peu de bonne volonté ?... une solution est certainement envisageable !

On ne peut que constater le résultat néfaste de l'application des Brevets, tant pour ces ingénieurs Norvégiens que pour les utilisateurs de Linux .

Vous trouverez de l'info sur le sujet (Royalties reversés à: DOLBY, MEPLA, MPEG2 et DVD FORUM .) sur le Web et dans différents magazines comme Linux Pratique, Planète Linux, Linux Magazine

DVD-Rom : du type encyclopédie (base de données), accessible et relativement facile à utiliser .

Les choses évoluent tout de même

DVD-Video : du type Le Titanic (film), reste plus complexe à mettre en oeuvre pour les vidéos cryptées :

Les DVD-video non cryptés, cela existe mais c'est une minorité.

Pour la vidéo, le système de fichier **UDF** est utilisé et devra être supporté par le noyau .

Pour plusieurs raisons (le décodage MPEG CSS entre autre), la lecture de DVD-video et le format UDF sont encore en développement : Il vous faudra appliquer des patches au noyau pour utiliser votre DVD-vidéo .

Des Solutions ?

1) Vous utilisez une carte MPEG-2 :

vous avez une carte de décompression(vidéo) MPEG (Il faut évidemment qu'elle est un pilote), dans ce cas si la carte se charge du décryptage c'est le fabricant qui s'acquitte des "royalties" : donc cela fonctionne !

2) Vous utilisez une décompression par logiciel :

La carte ne prend pas en charge le décryptage et il faut passer par un logiciel de décompression MPEG qui réalise le décryptage ???

Et là cela nous renvoie aux problèmes juridiques cités plus haut !

Sous Linux pour lire le MPEG à condition qu'il soit décrypté, vous pouvez utiliser **xMovie** ou **mpeg2play** .

3) Vous attendez :

Peut-être pas la plus téméraire des solutions mais certainement la plus objective .

Un jour peut-être les problèmes juridiques trouveront une solution.

Bientôt les projets en cours aboutiront .

Demain une version stable du noyau 2.4.x incluant UDF sortira .

Des Adresses !

Le patch pour lire les DVD-video : www.kernel.dk

Pour les utilisateurs d'un noyau stable 2.2.x :

Appliquer un patch au noyau : www.csc.calpoly.edu/~bfennema puis vous recompilez .

Utiliser le module (binaire) : trylinux.com/projects/udf puis vous placez le module dans

/lib/modules/2.2.x/fs/ et vous le charger dans le noyau avec la commande **insmod**

Pour les utilisateurs d'un noyau en développement 2.3.x :

Il suffit d'utiliser une version 2.3.17 ou plus .

Logiciels de lecture vidéos (flux MPEG) :

xMovie : heroine.linuxbox.com

mpeg2play : home.germany.net/100-5083

Pilotes pour les cartes de décompression MPEG :

Cartes à base de EM8400 , série NetStream : www.sigmadesigns.com

Cartes d'extension MPEG-2 à base de Zoran 37610 pour Matrox série G200 et Productiva G100 :

livid.on.openprojects.net

Carte Dxr-2 de Creative Labs : opensource.creative.com

The Linux MPEG2 Project : mpeg.openprojects.net

Le projet du groupe Linux TV : inuxtv.org/dvd

Et pour vous tenir informé :

roadrunner.swansea.uk/linux.ork/v4l.html

www.rpi.edu/~veliaa/linux-dvd

www.linuxdvd.org/dvd

www.opendvd.org

www.linuxvideo.org

trylinux.com/projects/udf/index.html

livid.on.openprojects.net/howto.html

Il existe une autre solution, dont je ne parlerais pas ici, qui consiste à monter le DVD à la norme iso 9660 mais alors là vous avez intérêt à posséder un PIII 500 minimum et au moins 20 Go de libre ???

Les "bidouilleurs" et futurs experts trouveront de la doc sur le Web et dans les différents magazines dédiés à Linux .

Dernière mise à jour: 12-09-2000

Configuration de la carte son

Pour une **RedHat** vous pouvez utiliser "sndconfig" , en mode console ou dans un xterm, tapez **sndconfig** pour lancer l'utilitaire de configuration de la carte son .

Avec la **Slackware** nous verrons comment configurer manuellement la carte son, car l'utilitaire "sndconfig" n'est pas inclus dans cette distribution .

Pour configurer une carte **ISA PnP** il faut utiliser les outils "**pnpdump**" et "**isapnp**", nous verrons comment dans la configuration manuelle de la **Slackware** : c'est identique sous **RedHat**.

- [Liste des cartes son supportées](#)
- [Vérification](#)
- [Configurer avec sndconfig](#)
- [Configuration manuelle](#)

- [Configurer une carte ISA PnP](#)

Les cartes son reconnues sous Linux :

- Aced Notebook Sound
- AdLib
- Advance Logic ALS-001, ALS-007, ALS-100, ALS-110, ALS 120
- Aztech Sound Galaxy
- AD1816(A) based cards
- Compaq Deskpro XL sound
- Creative/Ensoniq AudioPCI 97(ES1371)
- Crystal CS4232.
- Ensoniq AudioPCI 1370(SoundBlaster 64/128 PCI)
- Ensoniq SoundScape
- ESS 688/1688/1868 AudioDrive, Tech Maestro
- Gravis UltraSound MAX/PnP
- Logitech SoundMan Games(not SM16 or SM Wave)
- MediaTrix AudioTrix Pro
- MediaVision Jazz16 (ProSonic, SoundMan Wave)
- Microsoft Sound System(D1848/CS4248/CS4231)
- Mozart/MAD16 (OPTi 82C928)
- MAD16 Pro (OPTi 82C929/82C930)
- miroSOUND PCML2
- OPL3-SA1 sound chip
- OPL3-SA2/3/x sound chip
- Pro Audio Spectrum/Studio 16, Logitech SoundMan 16
- PSS (AD1848, ADSP-2115, ESC614)
- S3 Sonic Vibes
- SoundBlaster (SB 16/32/64, DS, ESS, Jazz16, Live, or SB clone)
- Turtle Beach MultiSound Classiq, Tahiti, Monterey, Pinnacle, Fiji
- Turtle Beach WaveFront(Tropez Plus, Tropez, Maui)
- VIA 82C686 Audio
- Yamaha OPL3-SA1, OPL3-SA2, SA3 and SAx

Vérification :

Pour vérifier si votre carte est installée, tapez :

`cat /dev/sndstat`

```
OSS/Free:3.8s2+-971130
Load type: Driver loaded as a module
Kernel: Linux lmiche.lmsoft.fr 2.2.5-22smp #1 SMP Wed Jun 2 09:11:51 EDT 1999 i686
Config options: 0

Installed drivers:

Card config:

Audio devices:
0: Sound Blaster 16 (4.13) (DUPLEX)

Synth devices:
0: Yamaha OPL3

Midi devices:
0: Sound Blaster 16

Timers:
0: System clock
```

```
Mixers:
0: Sound Blaster
```

Et si vous obtenez quelque chose de similaire, alors vous pouvez appliquer un "chmod 666" sur /dev/dsp, /dev/audio, /dev/mixer ... pour que tous les utilisateurs puissent accéder à ces périphériques et se servir des utilitaires audio et multimédia .

Et votre carte son doit fonctionner !

Pour tester votre carte, vous pouvez jouer un fichier wav, pour savoir où trouver un fichier audio au format wav sur votre système, tapez la commande "**find -name *.wav**", puis par exemple si vous avez un fichier test.wav dans le répertoire /usr/share/sounds, tapez :

play /usr/share/sounds/test.wav

Pour vérifiez si /dev/audio existe:

ls -l /dev/audio

```
crw-rw-rw- 1 root sys 14, 4 mai 5 1998 /dev/audio
```

Pour vérifier si les modules audio sont chargés :

lsmod

Module	Pages	Used by
opl3	3	0
sb	6	0
uart401	2	[sb] 0
sound	15	[opl3 sb uart401] 0

utilisation de sndconfig :

En mode console ou dans un xterm :

Après avoir tapez **sndconfig** , une fenêtre apparaît vous indiquant qu'une détection de la carte va être lancée, appuyez sur [**Enter**] et ensuite suivez les instructions .

Si vous possédez une carte PnP et que l'installation avec "sndconfig" échoue, alors utilisez "pnpdump" pour configurer votre carte et "isapnp" pour initialiser la carte ([voir ci-dessous](#)), puis lancez à nouveau "sndconfig" avec la commande :

sndconfig --noprobe

Configuration manuelle pour la Slackware :

Rappel :

Avant de configurer la carte son, pour que vous compreniez mieux la gestion des modules et le fonctionnement des cartes PnP sur votre [Slackware](#), nous allons voir comment s'initialise le système

je vais essayer de vous en faire une description simple et rapide .

Démarrage du système

Le premier fichier qui va initialiser votre Slackware est /etc/**inittab** qui à son tour va lancer d'autres scripts d'initialisation contenus dans le répertoire /etc/rc.d/

utilisation de sndconfig :

inittab par défaut lance le mode "Multi Users" (niveau 3)

rc.S initialise le système

rc.M lance les différents scripts

REGARDEZ le CONTENU de ces 3 SCRIPTS , cela vous apprendra beaucoup sur le fonctionnement de votre [Slackware](#) :

TOUT COMMENCE PAR LA !!!

Le noyau Linux est modulaire ce qui permet de lui ajouter des modules en fonction des besoins comme par exemple le pilote de la carte audio .

Pour la gestion de ces modules vous pouvez utiliser le démon "**kerneld**" qui va se charger d'ajouter le module au noyau lorsqu'il est nécessaire et l'enlever lorsqu'il est inutilisé .

Une autre méthode pour ajouter le module au noyau est d'utiliser `/sbin/modprobe` mais dans ce cas le module reste en permanence dans le noyau et occupe des ressources même s'il n'est plus utilisé.

La commande `/sbin/lsmmod` permet de voir les modules ajoutés au noyau .

Pour que "kerneld" charge le module qui gère la carte son, il a besoin du nom des modules et de leurs paramètres (options), contenus dans le fichier par défaut `/etc/modules.conf` , ou `/etc/conf.module` .

_Le fichier `/etc/rc.d/rc.modules`

Pour que les modules soient chargés au démarrage, il faut que "**kerneld**" soit lancé, cela se fait avec le script `/etc/rc.d/rc.S` ou le script `/etc/rc.d/rc.modules`.

Vous devez donc éditer le fichier `/etc/rc.d/rc.modules` et enlever les # pour obtenir :

```
if [ -x /sbin/kerneld ]; then
/sbin/kerneld
fi
```

...ATTENTION ! Ces lignes peuvent déjà être actives dans `/etc/rc.d/rc.S`, dans ce cas vous n'avez pas besoin de les ajouter dans `/etc/rc.d/rc.modules` .

Ce fichier **rc.modules** offre la possibilité de ne pas utiliser "kerneld" mais de charger vos modules avec `/sbin/modprobe` :

N'effacez aucune ligne, ajoutez simplement un # au début de la ligne à désactiver et insérez une nouvelle ligne avec vos paramètres .

Ne modifiez pas non plus l'ordre de chargement des modules c'est très important car par exemple le module du JoyStick ne peut être chargé avant celui de la carte son dont il dépend !

LISEZ ATTENTIVEMENT les COMMENTAIRES de ce fichier .

LIRE aussi la DOC dans `/usr/doc/modules`

Au "boot" de votre machine une ligne comme : "**Remounting the root device ...**" , vous indique que rc.S est lancé .

Ensuite il ne reste plus qu'à taper la commande `/sbin/lsmmod` pour vérifier si vos modules sont chargés .

_Le fichier `/etc/modules.conf` (ou `conf.modules`)

```
#/etc/modules.conf
alias net-pf-4 off
alias net-pf-5 off
alias sound sb
pre-install sound insmod sound dmabuf=1
alias midi opl3
options opl3 io=0x388
options sb io=0x220 irq=5 dma=1 dma16=5 mpu_io=0x330
# fin
```

utilisation de `sndconfig` :

Pour résumer :

_kerneld utilise le fichier modules.conf ou conf.modules pour charger le module de votre carte son.
_pour activer kerneld enlever les # dans le fichier rc.modules ou dans le fichier rc.S
_pour ne pas utiliser kerneld, enlever les # dans rc.modules des lignes correspondantes à vos modules et y ajouter les paramètres (options) nécessaires .

... Voilà pour les modules ..!

Certaines versions (anciennes) du noyau incluent, à la fin du fichier README, un script pour la création des périphériques (/dev) : Ce fichier se trouve dans le sous-répertoire /drivers/sound/ des sources du noyau .

Utilisation de pnpdump et isapnp :

_ Vous avez une carte ISA PnP :

En mode console ou dans un xterm, tapez :

```
pnpdump > /etc/isapnp.conf
```

Cela va créer le fichier /etc/isapnp.conf qui sera utilisé pour initialiser votre carte PnP (Plug and Play).

Je vais prendre comme exemple une carte **ESS1868 PnP Audiodrive**, très souvent utilisée par les assembleurs pour remplacer une Sound Blaster .

Cette carte ess1868 pnp utilise un processeur (chip) ES1868 et dans notre exemple elle utilise le port 220 (io=0x220) l'interruption 5 (irq=5) et le canal DMA 1 (dma=1).

Le module pour gérer cette carte sera celui de la SoundBlaster, "sb.o" .

Vous devez maintenant éditer le fichier /etc/isapnp.conf pour définir les paramètres de la carte, dans notre cas : **io=0x220 irq=5 dma=1**

_Le fichier **isapnp.conf**

Pour initialiser correctement la carte, il faut donc que le fichier /etc/isapnp.conf contienne les mêmes valeurs de paramètres(options) que ceux du module définis dans le fichier /etc/modules.conf .

chaque configuration possible (choix) commence par la ligne :

```
# Start dependent functions: priority acceptable
```

et comme l'indique cette ligne :

```
# Multiple choice time, choose one only !
```

parmis ces choix vous ne devez en utiliser qu'un seul !

Ces choix sont déterminés en fonction de l'adresse I/O, ici 0x220

Il reste donc à déterminer si vous utilisez un seul canal DMA (DMA 0) ou 2 canaux (DMA 0) plus (DMA 1) , si vous voulez utiliser l'émulation (synthé) opl3 Yamaha (IO 2) et/ou mpu Roland (IO 1) .

Et pour terminer il faut enlever le # devant (ACT Y) active=yes !

Enlever les # en début des lignes correspondantes à vos paramètres , pour obtenir quelque chose comme :

```
#/etc/isapnp.conf
#
# $Id: pnpdump.c,v 1.18 1999/02/14 22:47:18 fox Exp $
# This is free software, see the sources for details.
# This software has NO WARRANTY, use at your OWN RISK
#
# For details of this file format, see isapnp.conf(5)
#
```

```
# For latest information and FAQ on isapnp and pnpdump see:
# http://www.roestock.demon.co.uk/isapnptools/
#
# Compiler flags: -DREALTIME -DNEEDSETSCHEDULER -DABORT_ONRESERR
#
# Trying port address 0203
# Board 1 has serial identifier 27 ff ff ff ff 68 18 73 16

# (DEBUG)
(READPORT 0x0203)
(ISOLATE PRESERVE)
(IDENTIFY *)
(VERBOSITY 2)
(CONFLICT (IO FATAL)(IRQ FATAL)(DMA FATAL)(MEM FATAL)) # or WARNING

# Card 1: (serial identifier 27 ff ff ff ff 68 18 73 16)
# Vendor Id ESS1868, No Serial Number (-1), checksum 0x27.
# Version 1.0, Vendor version 1.0
# ANSI string -->ESS ES1868 Plug and Play AudioDrive
(CONFIGURE ESS1868/-1 (LD 1

(DMA 0 (CHANNEL 1))
(INT 0 (IRQ 5 (MODE +E)))
(IO 0 (SIZE 16) (BASE 0x0220))
(IO 1 (SIZE 4) (BASE 0x0388))
(IO 2 (SIZE 2) (BASE 0x0330))

(NAME "ESS1868/-1[1]{ESS ES1868 Plug and Play AudioDrive}")
(ACT Y)
))
```

Le fichier `/etc/conf.modules` (ou `/etc/modules.conf`) doit définir les options du module, ici **sb.o** :

```
alias sound sb
pre-install sound insmod sound dmabuf=1
alias midi opl3
options opl3 io=0x388
options sb io=0x220 irq=5 dma=1 mpu_io=0x330
```

Lire le [Sound-HOWTO](#) et la doc du répertoire `/usr/src/linux/Documentation/` pour savoir quel module utiliser, dans le cas où **sb.o** ne fonctionne pas avec votre carte son.

vous trouverez un exemple de fichier `/etc/isapnp.conf` pour la [ES1868 PnP](#) et la [SB 16 PnP](#) .

Une carte Pnp à besoin d'être initialisée avec les valeurs (options) indiquées dans le fichier `/etc/conf.modules`, ou pour une Slackware dans le fichier `modules.conf` ou dans le script `rc.modules` si vous n'utilisez pas `kernel`, pour cela le système va utiliser `/etc/isapnp.conf` pour envoyer les paramètres d'initialisation à la carte avec la commande: **isapnp /etc/isapnp.conf**

Enfin, il faut que ces commandes soient lancées au démarrage du système :

Pour **Slackware** :

Le script `/etc/rc.d/rc.S` va se charger d'initialiser la carte et de charger le module :

_Soit vous enlevez les # des lignes correspondantes à vos paramètres dans le fichier `rc.S` (System Initialisation)

_Soit vous ajoutez de nouvelles lignes contenant vos valeurs dans le fichier `rc.S` si elles ne s'y trouvent pas .

La commande **isapnp /etc/isapnp.conf** doit précéder le lancement du script **rc.modules** .

Pour **RedHat** :

Vous avez uniquement à définir les valeurs des paramètres de votre carte dans le fichier `/etc/conf.modules` , si `sndconfig` ne l'a pas fait .

SI votre CARTE n'est PAS PNP il est INUTILE de CONFIGURER isapnp.conf

Vous pouvez initialiser votre carte avec la commande :

`isapnp /etc/isapnp.conf`

Vous pouvez charger le module avec la commande :

`/sbin/modprobe -k sb io=0x220 irq= 5 dma=1 mpu_io=0x330`

Il ne reste plus qu'à [Vérifier et tester](#) votre carte son

Note:

Il est important que les cartes PnP soient initialisées si vous utilisez Windows avec Linux :

Lorsque vous envoyez la commande : `/sbin/modprobe -k sb io=0x220 irq= 5 dma=1`

`mpu_io=0x330` , la carte utilisera ces valeurs uniquement si elle a été initialisée avec :

I/O 220

IRQ 5

DMA 1

Lorsque vous installez ou modifiez les valeurs des paramètres d'une carte sous Windows, ces valeurs seront différentes de celles utilisées par la carte sous Linux , il faut donc que le système au démarrage de Linux puisse utiliser le fichier `/etc/isapnp.conf` pour rétablir les valeurs .

Dernière mise à jour: **20-07-2000**

- [Emacs et xEmacs](#)
- [Jed et xJed](#)
- [Joe](#)
- [Vi et Vim](#)

Configuration des Editeurs

Emacs, xEmacs :

_Bien que Emacs est beaucoup plus qu'un simple éditeur, nous configurerons que le minimum utile à l'affichage .

Très souvent en standard Emacs ne prend pas en compte les caractères accentués, pour cela il faut :

D'une part créer un fichier `~/.emacs` et y ajouter une ligne pour que le standard européen soit pris en compte :

`(standard-display-european 1)`

Vous pouvez également ajouter la coloration syntaxique avec cette ligne : `(setq`

font-lock-maximum-decoration t)

Votre fichier ~/.emacs peut ressembler cela :

```
;; ~/.emacs
(global-font-lock-mode t)
(setq font-lock-maximum-decoration t)
(standard-display-european 1)
```

Jed, xJed :

_Voici un éditeur plus léger que Emacs et certainement plus simple à configurer et à utiliser, mais qui émule les autres éditeurs .

Création d'un fichier ~/.jedrc à partir du fichier /usr/lib/jed/lib/jed.rc

Localisez sur votre système ce fichier jed.rc et copiez le dans ~/.jedrc

Le chemin peut-être, sur la Slackware /usr/lib/jed/lib/ , sur la RedHat /usr/lib/jed/lib

Ensuite vérifiez que vous avez bien un fichier /usr/lib/jed/lib/defaults.sl qui contient cette ligne :

```
() = evalfile("linux");
```

si non il faut créer ce fichier contenant cette seule ligne .

Enfin éditez le fichier /usr/lib/jed/lib/linux.sl et décommentez ou ajoutez ces lignes :

```
Info_Directory = "/usr/info"
variable UCB_Mailer; UCB_Mailer = "/bin/mail";
```

Le répertoire info peut se trouver dans /usr/local/info/ ou ailleurs !

_Il ne vous reste plus qu'à éditer ~/.jedrc pour personnaliser la configuration de Jed

Pour L'émulation **EDT, Emacs** ... décommentez les lignes correspondantes :

```
() = evalfile("edt"); % EDT emulation
() = evalfile("emacs"); % Emacs-like bindings
() = evalfile("linux"); % Linux Emulation (Touches Clavier)
```

Pour activer la touche de Tabulation :

```
setkey("self_insert_cmd", "^I");
```

Pour obtenir la coloration syntaxique :

```
USE_ANSI_COLORS = 1;
```

et modifiez les couleurs (set_color("menu", "white", "blue");) à votre convenance ...etc ,lisez la doc car il est possible de modifier beaucoup de paramètres.

Sous X :

Vous pouvez configurer l'apparence de xJed en ajoutant dans ~/.Xdefaults :

```
! xjed
xjed*Background: Midnight blue
xjed*font: 10x20
xjed.geometry: 80x25+150+50
```

Ou par exemple pour utiliser le pavé numérique (émulation EDT), ajoutez dans ~/.Xmodmap :

```
! pour XJed Emulation EDT
keycode 77 = KP_F1
keycode 112 = KP_F2
keycode 63 = KP_F3
keycode 82 = KP_F4
keycode 86 = KP_Separator
```

Joe :

_Même principe que pour Jed :

Vérifiez l'existence du fichier /usr/lib/joe/joerc , et copiez le dans ~/.joerc

Ensuite vous éditez ce fichier ~/.joerc pour configurer Joe .

Vérifiez également que /usr/bin/jmacs , /usr/bin/jstar , et /usr/bin/jpico sont des liens symboliques de /usr/bin/joe : si non il faut les créer avec la commande :

```
cd /usr/bin
```

In -sf joe jmacs ; In -sf joe jstar ; In -sf joe jpico

Vi , Vim :

_Pour pouvoir utiliser Vim (vi amélioré) ou gVim, il suffit de récupérer les exemples de fichiers vimrc et gvimrc qui se trouvent dans le répertoire /usr/doc/vim-common-5.3 et les copier dans votre répertoire HOME :

```
cp /usr/doc/vim-common-5.3/vimrc_example ~/.vimrc
cp /usr/doc/vim-common-5.3/gvimrc_example ~/.gvimrc
```

Ensuite vous pouvez éditer le fichier ~/.vimrc pour configurer l'éditeur à votre convenance :

```
set guifont=8x13bold
set tabstop=4
set shiftwidth=4
set nowrapscan
set ignorecase
```

Ceci définit les fontes et la touche de tabulation .
Pour plus d'information, consultez le [Vim-HOWTO](#).

Dernière mise à jour: **12-09-2000**

Récupération de ce qui peut encore l'être !

Le plus courant est la perte du mot de passe (root) et l'utilisation inconsidéré de la commande init .

- [Disquette de Boot et de sécurité](#)
- [J'ai perdu mon mot de passe ?](#)
- [Manipulation des runlevel avec init !](#)
- [Erreur d'effacement de fichiers ?!](#)

Disquette de Boot et de sécurité :

Disquette de Boot:

Pour créer une disquette de démarrage du système, rien de plus facile, en mode console ou dans un xterm, il suffit d'utiliser le script **mkbootdisk** .

Après avoir insérez une disquette dans le lecteur ! Tapez :

```
mkbootdisk --device /dev/fd0 2.2.5-22 -v
```

--device indique le lecteur de disquette, ici /dev/fd0 , suivi de la version du noyau et -v (verbose) pour avoir l'affichage du suivi du déroulement .

Cette disquette disposera du mode "Rescue" il suffit, au démarrage (avec la disquette), à l'invite LILO : de taper rescue .

pour voir les options possibles, tapez "man mkbootdisk" ou " mkbootdisk " .

ou moins simple ?

pour une **Slackware** lancer le menu **setup** et sélectionnez "lilo" puis "make lilo bootdisk" avec **RedHat** et les autres distributions, tapez la commande :

```
cat /usr/src/linux-2.2.10/arch/i386/boot/zImage > /dev/fd0
```

ou

```
cat /boot/vmlinuz > /dev/fd0
```

suivi de

```
rdev /dev/fd0 /dev/hda2
```

```
rdev -R /dev/fd0 1
```

remplacer *hda2* par la partition racine de Linux et */usr/src/linux-2.2.10/arch/i386/boot* par le chemin ou se trouve *zImage* .

A la place de *cat /boot/vmlinuz > /dev/fd0*, pour être sûr que le noyau tiendra sur la disquette(3.5), tapez:

```
dd if=/boot/vmlinuz of =/dev/fd0 bs=1440K
```

cette commande peut être utilisée à partir du CD d'installation avec :

```
mount /dev/cdrom
```

```
cd /mnt/cdrom/images
```

```
dd if=/boot.img of =/dev/fd0 bs=1440K
```

Disquette de sécurité:

Si vous avez un CD d'installation, il contient certainement l'image *rescue.img* dans le répertoire *images*, si non :

La disquette de secours "bite et couteau"(rescue) est récupérable sur : www.clark.net/~toehser ou sur FTP sunsite.unc.edu

Vous pouvez utiliser ou créer une disquette Amorce/racine, utilisant l'option *RAMDISK* du noyau, pour démarrer le système afin de monter vos partitions dans un répertoire de montage (*/mnt/*) pour modifier/restorer les fichiers défectueux :

*Consulter les chapitres "Amorce/Racine " et "Scripts pour créer des disquettes " du **Boot-Disk-HOWTO***

Récupérer son mot de passe :

_ Utilisez la disquette d'installation ou de secours du système pour démarrer la machine.

_ Montez la partition root :

```
mkdir /mnt
```

```
mount -t ext2 /dev/hda2 /mnt
```

remplacez */dev/hda2* par votre partition racine Linux .

Vous pouvez, si vous ne disposez pas de disquette de boot, démarrer le système en mode Single User

_ à l'apparition de *LILO* , tapez :

```
LILO: linux single
```

remplacez *linux* par le label de votre noyau déclaré dans *lilo.conf*

ATTENTION ! votre partition est en lecture seule et votre clavier en qwerty :

pour remonter la partition en lecture/écriture : **mount -w -n -o remount /**

pour passer le clavier en azerty : **loadkeys /usr/lib/kbd/keytables/fr.map**

_ Editez */etc/passwd* ou */mnt/etc/passwd* et supprimez le mot de passe pour obtenir :

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

Et si vous utilisez les *Shadow-password* , vos fichiers doivent ressembler à :

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

```
/etc/shadow : root::10855:0:99999:7:-1:-1:134538444 (version RedHat/Linuxconf)
```

```
/etc/shadow : root::10422:0:::: (version Slackware)
```

_ Une fois ces fichiers modifiés et sauvegardés, il ne reste plus qu'à redémarrer normalement le système .

_ Se logger sous compte root(sans mot de passe) pour créer un nouveau mot de passe avec la commande *passwd* .

Les RunLevel et la commande init :

Comme le dit la page de manuel `man init` Init est le père de tous les processus .

Et si vous éditez `/etc/inittab` vous trouverez cette ligne :

`id:3:initdefault:`

Qui indique le niveau d'exécution (RunLevel) par défaut

RunLevel :

- 0** – **halt** arrêt du système (`shutdown -h now`)
- 1** – Mono utilisateur (Single user)
- 2** – Multi utilisateurs sans NFS (sans le réseau)
- 3** – Multi utilisateurs + réseau
- 4** – inutilisé
- 5** – lance le serveur X11 (`xdm`)
- 6** – **reboot** relance le système (`shutdown -r now`)

Pour savoir quel niveau d'exécution est actuellement utilisé, il suffit de taper la commande `runlevel`

Sur la **Slackware** le niveau **4** peut-être utilisé pour lancer X11 (**xdm**)

Evidemment le niveau **0** et **6** ne doivent pas être utilisés, si non vous créez une boucle sans fin : A chaque démarrage le système s'arrêtera ou redémarrera ?!!

Si la commande `init q` permet de relire le fichier `/etc/inittab`, il convient de manier en connaissances de causes les autres commandes comme : `init 5`, `init 4`, `init 2` ou `init s`.

En cas de doute, avant de relancer ou d'arrêter votre système, vérifiez que dans le fichier `/etc/inittab` vous avez toujours :

`id:3:initdefault:` ou `id:5:initdefault:`.

Si vous voulez pouvoir choisir entre le démarrage en mode Multi utilisateurs (3) et en mode `xdm` (5), il suffit de commenter la ligne `initdefault` :

`#id:3:initdefault:`

Et le système vous demandera d'indiquer le mode à utiliser .

ATTENTION ! Ce qui suit n'est pas sans risque et ne garantit pas la récupération intégrale de vos données .

Et comme sur tous les systèmes si depuis l'effacement accidentel vous avez manipulé des fichiers (écriture) il y a de fortes chances que vos fichiers soient irrémédiablement perdus !

La première chose est de FAIRE DES SAUVEGARDES le reste est du bidouillage de secours !

Effacement ou formatage accidentel :

Si vous avez formaté accidentellement une partition, vous pouvez utiliser un script perl pour récupérer vos données, ce script se trouve sur www-eleves.enst-bretagne.fr/~aubert/perl ou dans le [GUIDE_DU_ROOTARD](#) .

Une autre méthode plus brutale consiste à faire un `strings /dev/hda2 > recup`, et il ne vous reste plus qu'à faire le tri dans ce fichier pour y récupérer ce qui est encore utilisable.

La récupération de fichiers effacés est encore plus délicate :

Plus le système de fichiers (FS) concerné sera démonté rapidement après l'effacement des fichiers, plus vous aurez de chance de récupérer un maximum de données .

Vous trouverez le détail des opérations à effectuer (plutôt longues) dans le [Ext2fs-Undeletion mini-HOWTO](#) , Je vais seulement vous indiquer le principe .

- _Démontage de la partition
- _Remontage de la partition en lecture seule
- _Attribution d'un espace suffisant pour contenir les données récupérées
- _Recherche des blocs effacés (inodes)
- _Récupération des blocs de données .

En supposant que vos fichiers se trouvaient sur la partition **/dev/hdb1** montée en **/usr/local** :

_Pour remonter la partition en lecture seule `mount -o ro,remount /usr/local`

ou si vous préférez :

```
umount /usr/local
```

```
mount -t ext2 -o ro /dev/hdb1 /usr/local
```

Si un processus utilise **/usr/local/bin** par exemple, il faudra tuer (kill) ce processus avant de pouvoir démonter ou remonter ce système de fichiers .

_Si vous ne disposez pas de partitions ayant un espace libre suffisant pour y copier la récup, il faut utiliser un **ramdisk** et le formater, par exemple un ramdisk de 2 Mo monté en **/mnt/recup** :

```
mkdir /mnt/recup
```

```
dd if=/dev/zero of=/dev/ram0 bs=1K count 2048
```

```
mke2fs -v -m -0 /dev/ram0 2048
```

```
mount -t ext2 /dev/ram0 /mnt/recup
```

Attention à ne pas démonter votre ramdisk avant d'avoir copier vos données récupérées sur un support physique .

_La lecture des blocs, qui étaient attribués à vos fichiers, ce fera avec l'utilitaire **fsgrab** , que vous trouverez sur http://pobox.com/~aaronc/tech/fsgrab*.tgz

_Création de la liste des inodes libérés récemment :

```
echo lsdel | degugfs /dev/hdb1 > recup.txt
```

Il ne vous reste plus qu'à rechercher dans **recup.txt**, les inodes classés par date et heure si vos données effacées s'y trouvent .

_Récupérer les infos (stat) sur l'inode effacé :

supposons que dans le fichier recup.txt vous trouviez un **inode: 148003** dont la date et l'heure correspondent à votre fichier effacé :

```
degugfs: stat
```

si la dernière ligne d'info indique **TOTAL = 6** , alors ces blocs de données pourront-etre récupérés en tapant :

```
degugfs: dump /mnt/recup/recup.000
```

Ce fichier ainsi récupéré se termine par des données qu'il faut corriger avec la commande :

```
dd count=1 if=/mnt/recup/recup.000 of=/mnt/recup/fichier.000 bs=6065
```

bs=6065 est la taille du fichier indiqué par Size: 6065 de la commande stat de debugfs .

Voici l'exemple donné dans le [Ext2fs-Undeletion mini-HOWTO](#)

```
debugfs: stat
```

```
Inode: 148003 Type: regular Mode: 0644 Flags: 0x0 Version: 1
```

```
User: 503 Group: 100 Size: 6065
```

```
File ACL: 0 Directory ACL: 0
```

```
Links: 0 Blockcount: 12
```

```
Fragment: Address: 0 Number: 0 Size: 0
```

```
ctime: 0x31a9a574 -- Mon May 27 13:52:04 1996
```

```
atime: 0x31a21dd1 -- Tue May 21 20:47:29 1996
```

```
mtime: 0x313bf4d7 -- Tue Mar 5 08:01:27 1996
```

```
dtime: 0x31a9a574 -- Mon May 27 13:52:04 1996
```

```
BLOCKS:
```

```
594810 594811 594814 594815 594816 594817
```

```
TOTAL: 6
```

-IL ne vous reste plus qu'à copier ce fichier fichier.000 sur un support physique disque dur ou disquette .

Il vaut mieux patienter quelques minutes, le temps d'une sauvegarde, que de passer plusieurs heures voir plusieurs jours à essayer de récupérer des morceaux de fichiers ! Aussi lisez la partie consacrée à la [Sauvegarde des données](#)

Adapter son environnement de travail

- [Sauvegarde des données](#)
- [Prompt utilisateur personnalisé](#)
- [Déporter l'affichage sur une machine distante](#)
- [Lancer plusieurs sessions X](#)

Sauvegarde des données :

Si vous n'avez pas acheté la dernière version de ARKEIA, vous pouvez quand même effectuer des sauvegardes à l'aide d'utilitaires fournis avec votre distribution .

Dump et Restore :

L'utilisation de **dump** et **restore** est très simple, pour sauvegarder une partition **/dev/hda2** sur une unité à bande **rmt0** ou dans un répertoire **/backup** :

```
dump 0sfu 3600 /dev/rmt0 /dev/hda2
```

```
dump 0fu /backup /dev/hda2
```

si votre unité de sauvegarde est située sur une machine distante **orion** :

```
dump 0sfu orion:/dev/rmt0 /dev/hda2
```

Les paramètres:

0 = sauvegarde complète

s = taille de la bande

f = fichier

u = écriture de la date dans `/etc/dumpdates`

Utilisez `man dump` pour en savoir plus sur les options de dump .

Pour restaurer les fichiers et répertoires :

```
restore -if /dev/rmt0
```

```
restore -if orion:/dev/rmt0
```

_ Vous pouvez également automatiser vos sauvegardes en ajoutant une entrée dans la crontab : la commande `crontab` permet à chaque utilisateur de se créer une crontab.

fichier `/var/spool/cron/root` (pour le compte root)

```
01 2 * * * /usr/bin/dump 0sfu 3600 /dev/rmt0 /dev/hda2
```

Ce qui effectuera une sauvegarde de `/dev/hda2` sur votre Tape tous les premiers du mois à deux heures du matin.

Si la syntaxe vous semble trop barbare, vous pouvez utiliser VCRON, de Daniel Roche, qui fonctionne sous X .

Tar :

L'utilisation de **tar** permet la sélection des fichiers et répertoires à inclure ou non dans la sauvegarde .

pour sauvegarder vos fichiers contenus dans `/usr/local` sur votre unité de sauvegarde `rmt0` ou `backup` :

```
tar cvf /dev/rmt0 /usr/local
```

```
tar cvf /backup /usr/local
```

Tar peut-être très pratique (et rapide) pour restaurer des partitions, supposons que votre windows vient de planter et rien que de penser au temps que cela va vous prendre pour tout réinstaller cela vous donne la migraine, alors mieux que l'aspirine, utilisez tar :

si votre partition Windows est en `/dev/hda1` et votre partition de backup en `/dev/hdb4` montée en `/backup`

_ monter la partition windows en /mnt/win

```
mount -t vfat /dev/hda1 /mnt/win
```

_ ce placer dans la partition windows

```
cd /mnt/win
```

_ créer l'archive de sauvegarde

```
tar cf -* | gzip -v9 > /backup/win.tgz
```

... Consultez les pages de manuel, car suivant la version de tar et gzip, cette commande peut ne pas fonctionner: Celle qui suit devrait fonctionner dans la plupart des cas

```
cd /
```

```
tar cfvz /backup/win.tgz /mnt/win/*
```

Ensuite pour réinstaller Windows :

_ formater la partion VFAT (éventuellement)

```
mkdosfs /dev/hda1
```

_ réinstaller Windows

monter la partition windows, puis tapez depuis le répertoire /backup :

```
tar xvfz win.tgz -C /mnt/win/
```

ou si vous préférez cette méthode, en copiant d'abord l'archive dans /mnt/win :

```
cp /backup/win.tgz /mnt/win/
```

```
cd /mnt/win
```

```
gzip -dc win.tgz | tar xf -
```

Voilà ! votre windows est restaurer .

Evidemment il n'est pas interdit (c'est même recommandé) de combiner sauvegarde/restauration et crontab pour automatiser vos copies de sauvegarde .

Un prompt personnalisé pour chaque utilisateur :

Si vous utilisez plusieurs comptes différents pour la bureautique, le développement, l'administration système

il peut être pratique d'identifier rapidement le prompt à l'aide de la couleur de ce dernier !

Prenons l'exemple d'une machine **orion** ayant trois comptes **root**, **droopy**, **devel** :

```
PS1='\033[1;31m\u@\h\033[0m]:\W > '
```

ce qui donne le prompt =

```
[root@orion]:/root >
```

```
PS1='\033[1;36m\u@\h\033[0m]\W \$ '
```

ce qui donne le prompt =

```
[droopy@orion]/droopy $
```

```
PS1='\033[1;33m\u:\s\033[0m]:\W > '
```

ce qui donne le prompt =

```
[devel:bash]:/devel >
```

Vous placez cette ligne dans ~/.bashrc et vous pouvez également ajouter une ligne pour afficher le chemin dans la barre de titre d'un xterm :

```
# ~/.bashrc
```

```
# Definition du Prompt
```

```
if [ "$TERM" = "xterm" ]; then
```

```
    PS1="\033]2;\h - \w\007\h:\W \$ "
```

```
else
```

```
    PS1='\033[1;31m\u@\h\033[0m]:\W > '
```

```
fi
```

Options utilisables :

\t = heure

\d = date

\u = nom de l'utilisateur

\h = nom de la machine

\W = répertoire courant

\w = chemin complet

\s = nom du shell

\n = saut de ligne

Code des couleurs :

`[1;31m\ 1` = caractère gras , 31 = rouge :

30 =Noir
31 =Rouge
32 =Vert
33 =Jaune
34 =Bleu
35 =Magenta
36 =Blue clair
37 =Blanc

Il existe d'autres façons de configurer le prompt vous en trouverez la description dans le [Configuration-HOWTO](#) .

Déporter l'affichage sur une machine distante :

Considérons deux machines **orion** et **galiopé** d'un même réseau , l'utilisateur va lancer depuis **orion** un xterm qu'il va afficher sur **galiopé** :

`xterm -display galiopé:0.0`

Evidemment pour éviter que tout le monde puisse afficher n'importe quoi sur n'importe quelle machine, l'utilisateur de la machine distante qui accepte l'affichage devra, sous X et dans un xterm, taper la commande :

`xhost +orion`

Dans cet exemple (xterm) toutes les commandes tapées dans le xterm depuis galiopé seront traitées par orion .

Le serveur X utilise beaucoup de ressources, et ce principe de fonctionnement des terminaux X va permettre à **orion** depuis le mode console de lancer l'affichage d'une application X sur **galiopé** . Le processeur d' orion pourra donc être réservé au calcul et galiopé à l'affichage .

Il est possible aussi de lancer l'application simultanément sur plusieurs machines du réseau avec le multiplexeur **XXM** utilisé pour la télé-maintenance : Vous trouverez un article de Daniel Roche sur le sujet, paru dans Linux Magazine N°7 sur www.linuxmag-france.org.

Lancer plusieurs sessions X :

Si vous éditez le fichier `/etc/inittab` , vous trouverez ces lignes :

```
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

qui définissent les consoles tty utilisables, par défaut la console réservée au serveur **X** est donc **tty7**

Mais il est possible d'utiliser plusieurs consoles Xwindow .

_Dans `/usr/X11R6/bin/startx` , ajoutez ces lignes au script startx :

```
CONSOLE=`tty`
CONSOLE=`basename $CONSOLE`
CONSOLE=`echo $CONSOLE | sed S/tty//`
CONSOLE=`expr $CONSOLE -1`
echo "Lancement de X sur " $CONSOLE
exec xinit $clientargs -- $serverargs : $CONSOLE
```

ou si vous préférez, ce qui suit :

```
for AFFICHAGE in 0 1 2 3 4 5;
do
```

```
if [ ! -f /tmp/.X $AFFICHAGE -lock];  
then  
  exec xinit $clientargs -- $serverargs : $AFFICHAGE > /dev/console 2> exit  
fi  
done
```

_Une autre solution consiste à passer le display en argument à startx :
en tapant **startx** le display est **O** et la console **tty7** que vous pouvez rappeler avec la combinaison des touches **[Ctrl]+[Alt]+[F7]**

startx -- :1 [Ctrl]+[Alt]+[F8]

startx -- :2 [Ctrl]+[Alt]+[F9] ...etc .

ATTENTION ! Si vous ne possédez pas au minimum un Pentium II 200 et 64 Mo de RAM, il vaut mieux vous abstenir .
