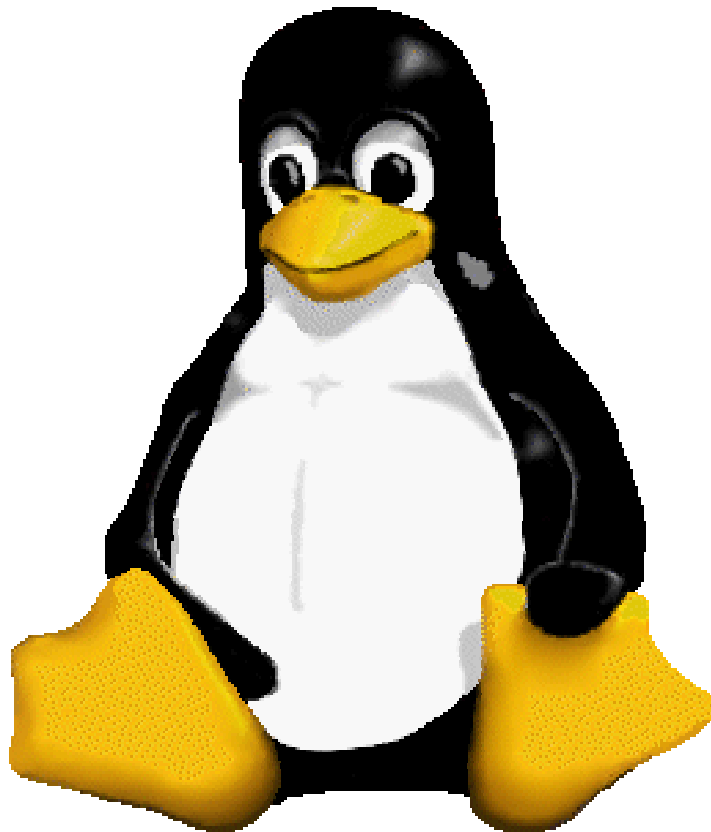
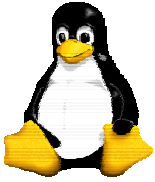


**I.U.T. Amiens**  
**Département Informatique**  
**Année Universitaire 2002/2003**



**Le système GNU/Linux**

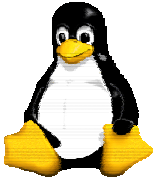
C. Drocourt – Màj le 30/09/2003



# Le système GNU/Linux

## Sommaire

1 - Introduction .....	6
1.1 - Historique .....	6
1.2 - Comment débiter.....	6
2 - Installation de linux .....	7
2.1 - Choisir sa distribution.....	7
2.2 - Les différentes possibilités d'installations .....	7
2.3 - Le début de l'installation .....	8
2.4 - Partitionnement du disque .....	8
2.5 - Que faut il installer ?.....	9
2.6 - Le reste de l'installation.....	9
2.7 - Problemes possibles lors du boot.....	9
2.7.1 - "LI" sans "LO".....	9
2.7.2 - message "Kernet panic unable to mount VFS system" .....	10
3 - Configuration du système .....	10
3.1 - Mon mot de passe, où est mon mot de passe ? .....	10
3.2 - Clavier Français .....	10
3.3 - LILO .....	11
3.4 - Les répertoires et les principaux fichiers de configuration.....	12
3.5 - Mettre une console externe comme console.....	12
3.6 - Num Lock au démarrage .....	13
3.7 - Imprimer !.....	13
4 - Le gestionnaire de packages : RPM.....	14
4.1 - Objectifs de RPM .....	14
4.2 - Installation .....	15
4.3 - Suppression.....	16
4.4 - Mise à jour .....	16
4.5 - Interrogation .....	17
4.6 - Vérification.....	18
5 - La gestion des disques et des partitions.....	19
5.1 - Montage manuel de partitions : mount .....	19
5.2 - Montage automatique de partitions : /etc/fstab.....	19
5.3 - Opérations sur disques et file systems (mke2fs, e2fsck, fdformat).....	20
5.4 - Comment générer une disquette de boot .....	20
5.5 - Partition de swap.....	21
5.5.1 - Swap : création de la partition .....	21
5.5.2 - Swap : fichier.....	21
5.6 - Optimisation des disques .....	21
6 - L'administration du système.....	22
6.1 - Ajouter un utilisateur, un groupe.....	22
6.2 - Enchainement des scripts de démarrage.....	23
6.3 - Ordonnancement de travaux : la crontab .....	23
6.4 - Le fichier /etc/syslog.conf .....	24
6.5 - La rotation des fichiers de logs : logrotate.....	24

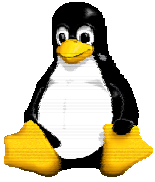


# Le système GNU/Linux

Version 9  
30/09/2003

---

6.6 - Mise en place des quota.....	24
6.6.1 - Logiciels nécessaires .....	24
6.6.2 - Compilation des utilitaires.....	24
6.6.3 - Activation de la gestion des quota dans le noyau.....	25
6.6.4 - Activation des quota sur les systèmes de fichiers.....	25
6.6.5 - Attribution d'un quota à un utilisateur .....	26
6.6.6 - Statistiques sur les quota.....	26
6.6.7 - Autres documentations sur les quota .....	26
6.7 - Lancer des " R "(emote ) commandes .....	26
6.8 - Les fichiers temporaires et la commande tmpwatch.....	27
6.9 - La commande strace .....	27
6.10 - La commande lsof.....	27
6.11 - La commande stat .....	27
6.12 - locate.....	27
6.13 - Les fichiers d'initialisation des interpréteurs de commandes .....	28
6.14 - Les sauvegardes sous Linux .....	28
6.14.1 - dump et restore .....	28
6.14.2 - tar .....	29
6.15 - Les fichiers core.....	30
6.16 - Consoles virtuelles.....	30
6.16.1 - Aspects pratiques .....	30
6.16.2 - Aspects théoriques .....	30
7 - X Window.....	31
7.1 - XF86Config.....	31
7.2 - Clavier français sous X... utiliser xkb.....	34
7.3 - Clavier français sous X, ancienne méthode : Xmodmap.....	35
7.4 - xdm .....	35
7.5 - Sécurité : le fichier .Xauthority .....	35
8 - Recompiler le noyau .....	36
8.1 - Cyrix .....	36
8.2 - Pentium.....	36
8.3 - Compilation .....	36
8.4 - Modules Chargeables.....	37
8.4.1 - Structure du noyau.....	37
8.4.2 - Qu'est-ce qu'un module chargeable ?.....	37
8.4.3 - Compilation du noyau .....	37
8.4.4 - Chargement : méthode manuelle .....	38
8.4.5 - Chargement automatique : kerneld.....	39
8.5 - SMP, autres architectures .....	42
9 - Configuration du réseau.....	43
9.1 - Installation ou changement de carte ethernet (3c509,3C590) .....	43
9.2 - Support des interfaces réseau.....	43
9.3 - Ajout des routes .....	44
9.4 - Spécifier le nom de la machine, puis des hôtes de votre réseau.....	44
9.5 - Regrouper des machines par sous-réseau .....	44
9.6 - Le fichier /etc/resolv.conf.....	45
10 - De Windows à Linux .....	45

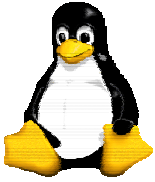


# Le système GNU/Linux

Version 9  
30/09/2003

---

10.1 - Montage de partitions dos/windows .....	45
10.2 - mtools .....	45
10.3 - Executer des applications Dos/Windows.....	46
10.3.1 - Dosemu et xdos.....	46
10.3.2 - Wine.....	46
10.4 - Samba et les réseaux MS-Windows .....	46
10.4.1 - Configuration de Samba sur la machine Linux .....	46
10.4.2 - Configuration de la machine MS-Windows .....	48
10.4.3 - Les outils smbclient et smbmount .....	48
11 - Les services reseaux .....	49
11.1 - NFS.....	49
11.1.1 - Coté client.....	49
11.1.2 - Coté serveur .....	49
11.2 - NIS.....	50
11.2.1 - Coté client.....	50
11.2.2 - Coté serveur .....	51
11.2.3 - Le démon yppasswd .....	52
11.3 - APACHE / httpd : un serveur Web .....	53
11.3.1 - Configuration d'Apache.....	53
11.3.2 - Accès aux fichiers et gestion des droits d'accès .....	58
11.3.3 - Configuration et utilisation des scripts CGI .....	58
11.3.4 - PHP et Mysql.....	59
12 - inetd, tcpd et xinetd.....	59
12.1 - Le super démon inetd.....	59
12.2 - Le wrapper tcpd.....	61
12.3 - xinetd .....	63
13 - Le DNS .....	64
13.1 - Le protocole DNS .....	65
13.2 - Hiérarchie des DNS .....	65
13.3 - Bases de données « whois ».....	66
13.4 - Tester un DNS .....	69
13.5 - Les objets DNS.....	72
13.6 - Contrôleurs primaire et secondaires .....	72
13.7 - Syntaxe d'un fichier de zone DNS .....	73
13.8 - Déploiement de Bind (ISC Software).....	74
13.8.1 - Compilation .....	74
13.8.2 - intégration système .....	74
13.8.3 - Configuration.....	74
13.9 - Exploitation du serveur DNS.....	77
14 - Le courrier électronique.....	78
14.1 - Introduction.....	78
14.2 - Les principes du courrier électronique .....	78
14.2.1 - Le principe de la messagerie.....	78
14.2.2 - Les composantes du système du courrier électronique.....	79
14.2.3 - Le Mail Transfer Agent (MTA).....	81
14.2.4 - Le Mail Deliver Agent (MDA).....	81
14.2.5 - Le Mail User Agent (MUA) .....	81

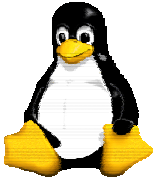


# Le système GNU/Linux

Version 9  
30/09/2003

---

14.2.6 - Format des adresses électroniques.....	82
14.2.7 - Structure d'un courrier.....	83
14.2.8 - L'enveloppe mail.....	85
14.3 - Le protocole SMTP.....	86
14.3.1 - Caractéristiques du protocole SMTP.....	86
14.3.2 - Caractéristiques du protocole ESMTP.....	88
14.4 - La réception du courrier.....	89
14.4.1 - Le mail local.....	89
14.4.2 - Utiliser une interface web.....	89
14.4.3 - Utiliser une connexion POP ou IMAP.....	89
14.5 - Interaction du mail avec le DNS.....	93
14.6 - Sendmail.....	94
14.6.1 - Les versions.....	94
14.6.2 - Sendmail et la sécurité.....	94
14.6.3 - Appel de sendmail.....	95
14.6.4 - Les fichiers et répertoires de sendmail.....	95
14.6.5 - Le fichier de configuration sendmail.cf.....	98
14.6.6 - Le fichier de configuration m4.....	98
15 - Index.....	102



# Le système GNU/Linux

## 1 - Introduction

### 1.1 - Historique

Linus Torvalds, étudiant à l'université d'Helsinki (Finlande) installe le système Minix sur son i386. Le système s'avérant trop limité pour Linus Torvalds, il décide d'aller plus loin sur la base de ce qui existe... Linux (Linus' Unix, l'Unix de Linus) 0.0.1 est né au mois d'août 1991. Linus lance alors un appel à contribution, et permet donc un libre accès au code source.

Dépendant au départ du système Minix, Linux devient autonome le 5 octobre, lors de la sortie officielle de la version 0.0.2. Cette version permet de faire tourner quelques applications GNU (logiciels libres) essentielles comme le compilateur gcc ou le shell bash.

Linus prend la décision de mettre le code source sous licence GPL : tout le monde peut alors participer au développement de Linux. Nous n'en mesurons qu'aujourd'hui les conséquences...

Une maîtrise originale de l'ensemble du développement en permet l'accès au plus grand nombre, sans aucune restriction de contenu. Ainsi, nul cahier des charges, nulle équipe isolée (et restreinte) de développeurs : ce sont les échanges d'idées et de bouts de codes qui vont améliorer le système. C'est de là qu'est né le succès actuel de Linux. En effet, l'apport constant des développeurs toujours plus nombreux, avec leurs améliorations ou leurs idées nouvelles, permet à Linux d'être toujours plus proche des besoins de ses utilisateurs.

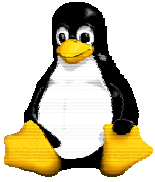
Linux est une libre implantation totalement gratuite des spécifications POSIX, avec des extensions System V et Berkeley (ce qui signifie qu'il ressemble à Unix, mais ne provient pas du tout des mêmes sources), qui est disponible à la fois sous forme de code source et de binaires. Il est propriété de Linus B. Torvalds (torvalds@transmeta.com) ainsi que des autres contributeurs à ce projet et est librement diffusable selon les termes de la "GNU Public Licence".

Linux ne relève ni du domaine public, ni du *partagiciel* mais du "*logiciel libre*", appelé couramment "*graticiel*" ; vous pouvez en distribuer des copies mais vous devez fournir le code source correspondant ou au moins le rendre disponible de la même façon. Si vous y faites des modifications que vous diffusez, vous êtes légalement tenu de distribuer le code source de vos modifications. Consultez la "*licence publique générale GNU*" (GPL) pour plus de détails. Vous en trouverez une copie dans l'arborescence des sources de Linux (fichier nommé "COPYING") ou par FTP sur le site prep.ai.mit.edu.

### 1.2 - Comment débiter

La meilleure méthode pour appréhender un système GNU/Linux est de rechercher des informations sur Internet. En effet, le développement du noyau Linux et de bons nombres de logiciels GNU ayant été réalisés à travers le réseau mondial, on peut y trouver toutes les informations nécessaires pour débiter, et même beaucoup plus.

Il faut commencer par récupérer et LIRE la liste des "Frequently Asked Questions" (FAQ) pris sur l'un des sites archivant les FAQ'S Usenet en général (par exemple, rtfm.mit.edu ou l'un de ses miroirs français comme ftp.lip6.fr dans le répertoire /pub/doc/faqs). Ces documents contiennent des centaines de renseignements permettant de résoudre la plupart de vos problèmes.



# Le système GNU/Linux

Version 9  
30/09/2003

Les "HOWTO", petits guides traitant chacun d'un domaine de Linux, sont une lecture indispensable. La plupart d'entre eux est disponible en version française dans le répertoire /pub/linux/french/docs/HOWTO sur le site ftp.lip6.fr.

Si vous avez accès au Web vous pouvez également trouver de précieux renseignements sur les URL :

- <http://www.freenix.org/linux>
- <http://echo-linux.alienor.fr/>
- <http://www.linux-france.com/article/>
- <http://www.linux-kheops.com/line/>

Si le monde UNIX est tout nouveau pour vous, et que Linux est votre premier contact avec ce domaine passionnant, suivez absolument les conseils suivants :

- Ne vous ruez pas sur les forums de discussion Linux pour poser mille questions auxquelles tout le monde est lassé de répondre. Lisez les quelques semaines auparavant, voyez ce qui s'y passe, consultez-en les archives mises à disposition sur certains sites, et il y a fort à parier que vous trouverez ce que vous cherchez sans déranger personne.
- Linux est un système UNIX parmi d'autres, et par conséquent toute la littérature spécifique à UNIX s'y applique. On n'apprend pas à maîtriser un nouveau système d'exploitation sans un minimum d'effort de documentation, souvenez-vous du temps qu'il vous a fallu pour maîtriser celui avec lequel vous travailliez auparavant, et des nombreux ouvrages que vous avez du consulter pour arriver à vos fins.
- Procurez-vous et lisez attentivement tous les ouvrages gratuits rédigés sur Linux. Ils contiennent les bases indispensables à votre apprentissage du système. Il ne vous sera pas pardonné de poser une question dont la réponse est contenue dans l'un de ces livres, ou dans l'un des documents complémentaires "HOWTO" disponibles tout aussi librement sur les sites diffusant Linux.
- Expérimentez, tentez de comprendre et de résoudre votre problème vous-même : c'est le meilleur apprentissage. Vous n'arriverez jamais à rien si vous faites tout faire par les autres. Linux et ses différentes documentations représentent des milliers d'heures de travail bénévole de centaines de développeurs, rédacteurs, traducteurs, testeurs ; ne condamnez pas le système si vous n'arrivez pas à en tirer parti parce que vous refusez de faire à votre tour un tout petit effort.

## 2 - Installation de linux

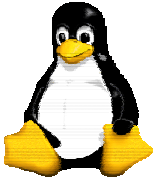
### 2.1 - Choisir sa distribution

Comme mentionné précédemment, le terme Linux ne désigne pas un système d'exploitation mais uniquement son noyau auquel il faut ajouter un ensemble de programmes et de logiciels pour obtenir un système complet. Pour cette raison, ce système d'exploitation est appelé GNU/Linux. Toutefois, les utilitaires compris dans cette distribution, leurs emplacements, les fichiers de configurations du système, ... sont autant de paramètres qu'il faut définir qu'elles donneront des systèmes différents.

Ainsi de tels systèmes sont appelés distributions comme : Redhat, Mandrake, Suse, Debian, ... Il n'y a pas *a priori* de meilleure distribution, c'est à chacun de les essayer, de lire les opinions de chacun, et du contexte d'utilisation de la machine sur laquelle on installe cette distribution.

### 2.2 - Les différentes possibilités d'installations

La solution la plus simple pour installer un système GNU/Linux est bien sur soit :



# Le système GNU/Linux

- D'avoir une machine uniquement dédiée à cela,
- D'avoir un disque vide,
- D'avoir prévu de l'espace libre.

Bien sur, il est toujours de redimensionner un disque pour récupérer de l'espace vide en utilisant l'utilitaire FIPS par exemple.

## 2.3 - Le début de l'installation

Avec la majorité des distributions, il est possible de booter directement sur le CD. Il suffit donc simplement d'insérer le CD puis de redémarrer la machine. Si votre bios est assez ancien ou si le CD que vous possédez n'est pas bootable, il faudra passer par une phase de création de disquette(s) de boot. Pour cela consultez le manuel de la distribution que vous avez choisit. Sous DOS, il faudra par exemple utiliser l'utilitaire RAWRITE.

D'autres cas nécessitent également la création d'une disquette de boot, comme l'installation en réseau par ftp par exemple qui est utile pour des déploiements sur un grand nombre de machines.

## 2.4 - Partitionnement du disque

Vous devez pendant l'installation spécifier sur quel disque vous voulez installer le système. Si votre machine ne contient pas d'autre système aucun problème, de la même manière si vous avez un disque complet pour l'installation.

Dans le cas contraire il va falloir vous familiariser rapidement avec la gestion des disques sous Linux. En effet le premier disque IDE est appelé /dev/hda, la première partition de ce disque est /dev/hda1, de la même manière le second disque porte le nom /dev/hdb, ... Pour un disque SCSI il s'agit de /dev/sda.

Vous devez à la fin de cette étape être sur de posséder au moins une partition Linux (Linux native) et une partition de swap utilisée lorsque le système n'a plus de mémoire disponible (Linux swap) dont la taille peut varier en général de une à deux fois celle de la mémoire vive.

Beaucoup de distribution comporte une interface d'installation graphique qui facilite cette manipulation de partition. Toutefois il peut être intéressant de connaître l'outil utilisé en ligne de commande (qui est de toute façon souvent disponible dans tous les processus d'installation), il s'agit de fdisk, sa syntaxe est : fdisk /dev/disque.

Par exemple, si vous souhaitez installer Linux sur le second disque IDE, tapez fdisk /dev/hdb.

Les commandes de fdisk (dans l'ordre d'utilité) sont :

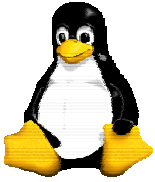
- m : la liste de toutes les commandes ;
- p : affiche les partitions ;
- n : ajouter une nouvelle partition ;
- d : détruire une partition ;
- t : changer le type de la partition ;
- q : quitter sans sauvegarder les changements ;
- w : sauvegarder et quitter.

Pour ajouter une partition, tapez n puis spécifiez le numéro de la partition (principale ou étendue), le numéro du bloc de début et sa taille (une taille paire est conseillée). Ensuite, donnez-lui son type (avec t) :

- 83 : Linux
- 82 : Swap.

Et sauvegardez (w).





# Le système GNU/Linux

Version 9  
30/09/2003

La partition est créée. Il est toutefois possible que vous ayez besoin de rebooter Linux pour qu'elle soit prise en compte par le noyau.

Le processus d'installation s'occupe en général de formater les partitions, vous pouvez toutefois être amenés à le faire vous-même avec la commande `mke2fs` (voir section correspondante).

## 2.5 - Que faut il installer ?

Les distributions sont en général fournies avec un ensemble de packages fournissant des ensembles d'utilitaires, de logiciels, ... certaines proposent une installation par défaut ou un mode de configuration graphique. Cette étape étant fortement dépendante de la distribution choisie nous ne rentrerons pas dans les détails ici. De toute façon vous pourrez installer d'autres logiciels par la suite si vous en avez oublié.

## 2.6 - Le reste de l'installation

Le reste de l'installation concerne en général la configuration de votre machine, à savoir la souris, la carte graphique, ... Les distributions GNU/Linux supportant de plus en plus tout type de matériel différent vous ne devriez pas avoir de problèmes ici (mais cela peut arriver...).

Il reste ensuite à configurer le chargeur du noyau que permettra de booter le système (et même d'autres), le plus utilisé est LILO mais il en existe d'autres comme GRUB.

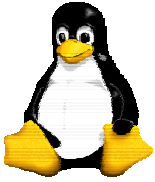
Ce logiciel doit en général être installé sur le MBR (master boot record) du disque, il permettra ainsi de booter d'autres systèmes d'exploitation. Si vous possédez déjà un autre gestionnaire de boot (comme XOSL par exemple), vous devez installer LILO sur la partition linux (`/dev/hda1` par exemple). Pour plus d'informations sur LILO on consultera la partie correspondante.

## 2.7 - Problemes possibles lors du boot

### 2.7.1 - "LI" sans "LO"

LILO a été installé sur le MBR, mais lors du boot, seul "LI" apparaît à l'écran (au lieu de LILO) et le PC se bloque! Ce problème provient du fait que seule la première partie de LILO commence à s'exécuter puis lilo ne parvient pas à charger l'image du noyau linux sur le disque, ceci parce qu'il ne le trouve pas. On s'en tirera dans 95% des cas en mettant l'option "linear" dans le fichier de conf de lilo: `/etc/lilo.conf` cette option permet de s'affranchir de la géométrie du disque (CHS) (cylinder/head/sector) en faisant un adressage linéaire du disque ou encore l'option "lba32".

```
boot=/dev/hda
delay=900
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
linear          <---- là...
other=/dev/hda1
    label=dos
    table=/dev/hda
image=/boot/vmlinuz
    label=linux
    root=/dev/hdc3
    read-only
```



---

```
append="mem=80M"
```

## 2.7.2 - message "Kernet panic unable to mount VFS system"

(ou message "unable to open initial console" )

Dans ce cas LILO a pu s'exécuter et a chargé le noyau linux, le noyau commence à s'exécuter et reconnaître tous les périphériques du PC, mais le noyau n'arrive pas à accéder au disque qui contient la partition système "/" (root). Le noyau doit d'une façon ou d'une autre savoir quelle partition de quel disque utiliser comme racine. On peut fournir cette information par la commande rdev. rdev permet de "patcher" (modifier) le noyau afin qu'il pointe sur le bon périphérique root "/" .  
ex: rdev /vmlinuz /dev/hda6

## 3 - Configuration du système

### 3.1 - Mon mot de passe, où est mon mot de passe ?

En bref : vous avez perdu le mot de passe de root... Bon, il y a plusieurs solutions.

La première :

1. bootez avec les disquettes boot et root de votre distribution.
2. loguez-vous en root (là, pas de mot de passe).
3. montez la racine du disque dur : `mount -t ext2 /dev/hda1 /mnt`
4. allez dans le répertoire /mnt. Il ne reste plus qu'à éditer le fichier /mnt/etc/passwd, y supprimer le mot de passe de root : `root::0:0:root:/root:/bin/bash`.

Vous rebootez Linux normalement, et en principe, vous n'avez plus besoin de mot de passe pour vous loguer en root.

Bon, passons à d'autres solutions : rebooter le pc en mode dit *single-user*. Pour cela, lors du boot avec LILO, fournir LILO: linux single (remplacer ici "linux" par le nom sous lequel LILO connaît votre noyau). Un shell root va apparaître. Attention : le clavier est en qwerty et la partition en lecture seule. Pour y remédier :

```
loadkeys /usr/lib/kbd/keytables/fr.map  
mount -w -n -o remount /
```

Soit vous utilisez la commande passwd, soit vous éditez le fichier /etc/passwd

Si vous n'avez pas de disquette de boot, vous pouvez souvent vous en sortir en passant `init=/bin/sh` lors de l'amorçage. Ensuite, monter la racine et éditer à la main le fichier /etc/passwd.

### 3.2 - Clavier Français

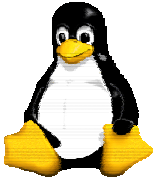
Bien, commençons par la chose utile : le clavier français. Normalement, les distributions configurent ça tout seul... mais on ne sait jamais !

Une fois loggé en *root*, tapez la commande suivante :

```
/usr/bin/loadkeys /usr/lib/kbd/keytables/fr-latin1.map
```

Si vous avez un clavier suisse-romand, utilisez `sf-latin1.map`.

Maintenant, le clavier est français ! Attention, cela reste temporaire. La solution la plus simple est de le rajouter dans le fichier /etc/rc.local avec la Slackware mais vous pouvez aussi utiliser le programme /sbin/setup ou directement /usr/lib/setup/SeTkeymap. Pour la Red Hat, réaliser l'opération avec le panneau de configuration kbdconfig.



# Le système GNU/Linux

Version 9  
30/09/2003

Avec ces deux distributions, la procédure d'installation aura normalement procédé à cette configuration et vous n'aurez probablement rien à faire, sauf si vous voulez changer de type de clavier...

## 3.3 - LILO

*LILO* (Linux LOader) permet d'avoir un système multiple d'amorçage : MS-Windows, MS-Dos, Linux, etc. Ce programme va se loger dans le secteur de boot de votre disque pour vous permettre de sélectionner la partition sur laquelle booter.

Si vous avez installé *LILO* sur le disque dur et que cela plante, bootez avec une disquette MS-DOS et tapez `FDISK /MBR` pour restaurer les paramètres de boot du DOS, et enlever *LILO* par la même occasion. Il suffira de relancer *LILO* en ayant rebooté sous Linux avec une disquette de boot.

Le fichier de configuration LILO se trouve généralement dans le fichier `/etc/lilo.conf`. Les distributions permettent la génération d'un fichier d'une manière automatique.

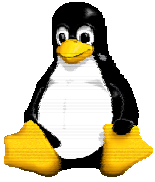
Voici un exemple de fichier :

```
boot = /dev/hda # disque sur lequel on met le fichier lilo:
delay = 300    # temps d'attente avant le lancement du boot.
                # permet d'avoir le temps de selectionner la partition
                # a amorcer.
vga = normal   # mode de l'écran a choisir
ramdisk = 0    # paranoia setting
# Linux : dernière version : le Pc boote ici par défaut
image = /vmlinuz # Noyau Linux 1 (chemin complet et nom du fichier)
root = /dev/hda2
append="no-hlt, aha1542=0x230"
label = linux
read-only
other = /dev/hda1 # Partition 1 :
label = dos      # texte à taper pour booter dessus : dos
table = /dev/hda # disque : /dev/hda
image = /zimage  # Noyau Linux 2
root = /dev/hda2 # Racine (Partition 2 de mon disque)
append="aha1542=0x230" # append... voir explication plus bas
label = old     # Texte a taper : old
read-only      #
```

Plusieurs questions peuvent vous effleurer l'esprit :

1. `read-only` : lorsque vous amorcez Linux, il va faire une vérification de la cohérence des informations des partitions Linux. C'est une vérification des systèmes de fichiers. Pour réaliser cette opération, il est indispensable que la partition soit en lecture seule. En effet, vérifier un disque en cours d'utilisation pourrait tout détruire. Cette vérification n'a pas lieu à tous les coups si vous éteignez bien la machine (appuyez sur les touches *Ctrl-Alt-Del* ou utilisez les commandes `halt`, `reboot` et `shutdown`). Cette vérification n'est de toute manière lancée que sur les partitions Linux. C'est le programme `fsck` qui effectue cette opération. Je vous déconseille de le lancer "pour voir" : une utilisation erronée peut provoquer des pertes de données sur le disque !

2. pourquoi deux noyaux ? C'est assez simple : Linux change, bouge. Les noyaux sont modifiés assez souvent. Par exemple, on peut amorcer sa machine soit avec un noyau stable, soit avec un noyau de développement.



# Le système GNU/Linux

Version 9  
30/09/2003

3. append : Cette option est très importante : elle permet de donner un ou plusieurs paramètres au noyau Linux. Par exemple, ici on lui indique qu'une carte SCSI se trouve à l'adresse 230. Dans le cas contraire, elle ne serait pas reconnue. Cette option est utilisée également si vous avez plusieurs interfaces réseau, SCSI, etc.

Pour faire prendre en compte au noyau une augmentation de mémoire centrale, il faut rajouter la ligne d'option `append=80M` dans le fichier `/etc/lilo.conf`, puis relancer le programme `lilo`.

Ex de syntaxe : `append="mem=80M aha152x=0x340,10,5"` ou bien (plus ennuyeux) au prompt de `lilo`, taper la commande `linux mem=32M`

4. Pour installer le fichier, exécutez `/sbin/lilo` et c'est tout !

En cas de problème avec LILO il faut d'abord essayer de supprimer l'option "compact" qui peut se trouver dans son fichier de configuration.

Ne surtout pas oublier de relancer LILO après chaque modification de son fichier de configuration !

## 3.4 - Les répertoires et les principaux fichiers de configuration

Les répertoires

<code>/etc</code>	Fichiers de configuration
<code>/dev</code>	Périphériques (devices)
<code>/home</code>	Répertoire des utilisateurs du système
<code>/sbin</code>	Binaires de configuration
<code>/bin</code>	Binaires de base
<code>/tmp</code>	Répertoire temporaire
<code>/usr</code>	Répertoire user (avec <code>bin</code> , <code>lib</code> , ...)
<code>/var</code>	Répertoire variable, <code>spool</code> , <code>mail</code> , ...

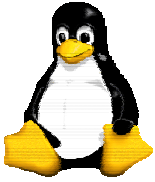
Les principaux fichiers de configurations sont :

<code>/etc/hosts</code>	Liste des IP des machines, des noms et des noms symboliques
<code>/etc/passwd</code>	Liste des utilisateurs et des configs de ces utilisateurs
<code>/etc/group</code>	Liste des groupes d'utilisateurs
<code>/etc/resolv.conf</code>	Liste des machines serveur de noms (DNS)
<code>/etc/host.conf</code>	Configuration de l'ordre de recherche des noms
<code>/etc/hosts.allow</code>	Fichier config du tcp wrapper <code>tcpd</code> , liste des autorisations
<code>/etc/hosts.deny</code>	Fichier config du tcp wrapper <code>tcpd</code> , liste des interdictions
<code>/etc/services</code>	Liste des démons, leur protocoles et leur port d'attachement
<code>/etc/inetd.conf</code>	Listes des démons gérés par le super serveur <code>inetd</code>
<code>/etc/protocols</code>	Liste des protocoles avec leurs identifiants
<code>/etc/profile</code>	Fichier de configuration de tous les utilisateurs
<code>/etc/fstab</code>	Liste des disques montés au démarrage.
<code>/etc/inittab</code>	Fichier de config du run level au démarrage.
<code>/etc/rc.d/</code>	Répertoire des scripts de démarrage
<code>/etc/termcap</code>	Fichier de configuration des terminaux

## 3.5 - Mettre une console externe comme console

Certains utilisateurs peuvent vouloir mettre comme console un minitel ou un vieux vt100 qui ne passe pas sur la carte vidéo mais sur le port série. Il suffit d'ajouter dans le fichier `lilo.conf` la ligne suivante :

```
append = "scon=0x03f8 serial=0,9600n8"
```



# Le système GNU/Linux

## 3.6 - Num Lock au démarrage

Pour configurer la touche Num Lock au démarrage (par exemple pour qu'elle soit positionnée), il suffit de mettre dans `/etc/rc.d/rc.local`

```
for tty in /dev/tty[1-9]*;
do
    setleds -D +num < $tty > /dev/null
done
```

## 3.7 - Imprimer !

Maintenant que la machine fonctionne, voyons comment imprimer. Les imprimantes sont gérées par le programme `lpc` et par le démon `lpd`.

Le démon `lpd` doit être lancé au boot. Il est lancé soit dans `rc.M` soit dans `rc.inetd2` (vous devez le voir lorsqu'il lance tous les démons de la machine. Si ce n'est pas le cas, il faut décommenter quelques lignes dans l'un des fichiers cités).

Supposons que l'imprimante s'appelle `betty`. Dans un premier temps, dans le fichier `/etc/rc.d/rc.local`, ajoutez

```
echo "Setting up printer betty"
/usr/sbin/lpc up betty
```

Passons maintenant aux choses sérieuses. Le fichier dans lequel va être décrit l'imprimante est le fichier `/etc/printcap`. Exemple :

```
betty|lp:lp=/dev/lp1:sd=/var/spool/lp1:sh:\
lf=/var/adm/lpd-errs:of=/etc/start-dj500:
```

La première colonne représente le nom de l'imprimante. J'ai mis `betty` et `lp` car de cette manière, c'est l'imprimante par défaut. Ensuite, la configuration se fait via des mots-clefs. Pour plus d'information, faites `man printcap`. Voici quand même une description de ceux utilisés.

- `lp` : device sur lequel est branché l'imprimante (LPT1 ici) ;
- `sd` : répertoire du spool d'impression ;
- `sh` : supprime la page d'entête ;
- `lf` : fichier d'erreurs ;
- `of` : programme de filtrage.
- `rm` : remote machine qui gère l'imprimante
- `rp` : nom de l'imprimante sur la machine distante

Comment ça marche ? Bon, supposons que j'ai un fichier `toto.ps`. Je fais donc `lpr toto.ps` (ou `lpr -Pbetty toto.ps`). Le fichier va être déposé dans le répertoire `/var/spool/lp1`.

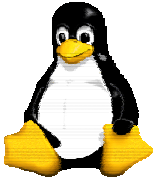
Le démon va ensuite envoyer le fichier sur le programme de filtrage. Ce système permet d'imprimer des fichiers ayant des formats assez variés. Utiliser `lprm` pour retirer une requête d'impression active.

Exemples :

- accès à des imprimantes distantes accessibles en réseau par d'autres stations Unix.

```
##PRINTTOOL1## REMOTE
```

```
lp1|deskjet:\                <== nom local de l'imprimante (coté linux)
:sd=/var/spool/lpd/lp1:\     <== zone de spool local pour cette imprimante
:mx#0:\                      <== maximum file size [0=illimité]
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
:rm=lpserve.com.univ-plm.fr:\          <== remote machine qui gère l'imprimante
:rp=deskjet:          <== nom de l'imprimante sur la machine distante
##PRINTER1## REMOTE
lp|laser4m:\
:sd=/var/spool/lpd/lp:\
:mx#0:\
:rm=lpserve.com.univ-plm.fr:\
:rp=laser4m:
```

- accès sur des imprimantes réseau (qui ont une interface ethernet) et postscript. Dans ce cas on peut directement adresser l'imprimante, sans passer par une machine gérant un spool distant, pour soumettre les travaux d'impression

```
lp|laserlum5eme:\
:lp=/dev/null:sh:\    <=== nom du device d'impression local (/dev/lp normalement)
:sd=/var/spool/lpd/lp:\    <=== zone de spool locale pour cette imprimante
:mx#0:\
:rm=impr5.com.univ-plm.fr:rp=raw:    <== nom reseau IP de la machine distante gérant son
spool
# :rp=laserlum5eme:\    <=== remote printer : nom de l'imprimante
# :if=/var/spool/lpd/lp/filter:    <=== filtre éventuel de conversion des données
```

- Accès à une imprimante sur port parallèle : "personnelle"

On peut utiliser le programme de configuration printtool de la distribution RedHat. IL n'y a pas de problème pour faire reconnaître l'imprimante sur un port parallèle.

## 4 - Le gestionnaire de packages : RPM

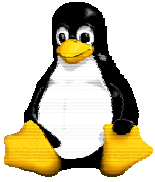
### 4.1 - Objectifs de RPM

Avant de chercher à utiliser RPM, il est préférable de savoir à quoi il peut servir.

**Mise à jour** : vous pouvez avec RPM effectuer une mise à jour des différents composants de votre système sans avoir à tout réinstaller. Lorsque vous avez à votre disposition une nouvelle version d'un système d'exploitation basée sur RPM (ex : Red Hat Linux), vous n'avez pas besoin de réinstaller votre machine (comme vous le feriez pour d'autres systèmes). RPM effectue une mise à jour automatisée et intelligente de votre système. Vos fichiers de configuration seront conservés lors de la procédure de mise à jour.

**Interrogation** : RPM possède des fonctions d'interrogation très poussées. Vous pouvez effectuer des recherches, grâce à la base de données, sur des packages ou sur un simple fichier. Ainsi, vous pouvez facilement savoir à quel package appartient tel ou tel fichier et donc savoir d'où il vient. Les fichiers contenus dans un package RPM se trouvent dans une archive compressée, avec un en-tête binaire contenant des informations sur le package et son contenu. Vous pouvez ainsi interroger rapidement et individuellement les packages.

**Vérification du système** : RPM possède aussi des fonctions de vérification des packages. Si vous pensez avoir effacé un fichier contenu dans un package, il vous suffit de vérifier ce package et toute



# Le système GNU/Linux

Version 9  
30/09/2003

anomalie vous sera rapportée. Si nécessaire, vous pouvez alors réinstaller le package, sans que vos fichiers de configuration soient supprimés.

**Sources premières** : un des objectifs est de permettre l'utilisation des sources premières, telles qu'elles sont distribuées par les auteurs du logiciel. Avec RPM, vous aurez à votre disposition les sources premières, les mises à jour (patch) utilisés et les instructions nécessaires à la compilation. C'est un grand avantage, et pour plusieurs raisons. Par exemple, vous pouvez facilement évoluer vers une nouvelle version de package. Il vous suffit de jeter un simple coup d'oeil aux mises à jour (patch) pour savoir ce qu'il vous suffit de faire.

Tout ceci est d'une grande importance pour les programmeurs, mais au final les utilisateurs en seront aussi très satisfaits. La distribution BOGUS sont à l'origine de ce concept.

RPM possède donc cinq fonctions différentes (sans compter la création de package) : installation, suppression, mise à jour, interrogation et vérification. Voici une présentation détaillée de ces cinq fonctions. Pour plus d'informations et pour une description des options, utilisez la commande `rpm --help` ou consultez les pages du man (ainsi que l'ouvrage consacré à RPM).

## 4.2 - Installation

Les packages RPM ont généralement des noms du type `foo-1.0-1.i386.rpm`, (`foo`) est le nom du package, (`1.0`) la version, (`1`) le numéro de mise à jour, et (`i386`) l'architecture. Pour installer un package, utilisez simplement la commande :

```
$ rpm -ivh foo-1.0-1.i386.rpm
foo #####
```

Comme vous pouvez le constater, RPM affiche le nom du package (qui ne correspond pas forcément au nom du fichier, qui aurait pu être `1.rpm`) et une succession de caractères `#` qui représentent une jauge de progression.

L'installation de package est très simple, mais vous pouvez tout de même rencontrer quelques erreurs :

### Package déjà installé

Si le package est déjà installé, vous verrez apparaître :

```
$ rpm -ivh foo-1.0-1.i386.rpm
foo package foo-1.0-1 is already installed
error : foo-1.0-1.i386.rpm cannot be installed
```

Si vous voulez vraiment installer ce package, utilisez l'option `--replacepks` qui permet d'ignorer l'erreur.

### Fichiers en conflit

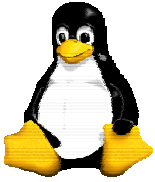
Si vous tentez d'installer un package qui contient un fichier déjà installé sur le système, vous verrez apparaître :

```
$ rpm -ivh foo-1.0-1.i386.rpm
foo /usr/bin/foo conflicts with file from bar-1.0-1
error : foo-1.0-1.i386.rpm cannot be installed
```

Pour ignorer l'erreur, utilisez l'option `--replacefiles`.

### Dépendances





# Le système GNU/Linux

Version 9  
30/09/2003

Des packages RPM peuvent dépendre d'autres packages, ce qui signifie qu'ils doivent au préalable être installés pour que tout fonctionne correctement. Si vous tentez d'installer un package dépendant d'un autre package, vous verrez apparaître :

```
$ rpm -ivh bar-1.0-1.i386.rpm
failed dependencies :
foo is needed by bar-1.0-1
```

Pour éviter ce problème, vous devez d'abord installer les packages requis. Vous pouvez tout de même forcer l'installation (mais le package risque fort de ne pas fonctionner) en utilisant l'option `--nodeps`.

## Sources

Pour installer un package `.src.rpm`, deux solutions :

- ```
rpm -ivh <packagename>.src.rpm
cd /usr/src/redhat/SPECS
rpm -bb <packagename>.spec
cd ../RPMS/i386 (ou l'architecture voulue)
rpm -ivh <packagename>.i386.rpm
```
- ```
rpm --rebuild *src.rpm
```

 va lancer la création du paquet binaire: si tout se passe bien, on trouvera le fichier `.i386.rpm` prêt à l'installation dans `/usr/src/packages/RPMS/i386/`

## 4.3 - Suppression

La suppression d'un package est aussi simple que son installation :

```
$ rpm -e foo
```

Note : nous utilisons ici le nom `foo`, qui n'est pas le nom original du package (ici `foo-1.0-1.i386.rpm`).

Lors d'une suppression, vous pouvez rencontrer des problèmes de dépendances dans le cas où le package que vous tentez de supprimer est requis par d'autres packages. Vous verrez apparaître :

```
$ rpm -e foo
removing these packages would break dependencies :
foo is needed by bar-1.0-1
```

Vous pouvez tout de même forcer la suppression (mais les packages dépendants risquent fort de ne pas fonctionner) en utilisant l'option `--nodeps`.

## 4.4 - Mise à jour

La mise à jour d'un package est aussi simple que son installation :

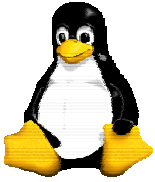
```
$ rpm -Uvh foo-2.0-1.i386.rpm
foo #####
```

Vous ne vous en apercevrez pas, mais RPM supprime automatiquement les anciennes versions du package `foo`. Vous pouvez toujours utiliser l'option `-U` pour installer les packages, puisqu'elle fonctionne même si le système ne possède pas d'ancienne version du package.

RPM effectue une mise à jour intelligente du package avec une sauvegarde des fichiers de configuration, vous verrez apparaître un message du type :

```
saving /etc/foo.conf as /etc/foo.conf.rpmsave
```





# Le système GNU/Linux

Version 9  
30/09/2003

Cela signifie que les modifications que vous avez apportées au fichier de configuration risquent de ne plus être compatibles avec le nouveau fichier de configuration du package, RPM effectue donc une sauvegarde du fichier d'origine et installe le nouveau. Observez les différences entre les deux fichiers, afin que le système puisse continuer à fonctionner correctement.

Une mise à jour combine en fait les fonctions d'installation et de suppression de package, vous pouvez donc rencontrer une des erreurs décrites plus haut, plus une. Si RPM pense que vous êtes en train d'effectuer une mise à jour vers une version plus ancienne, vous verrez apparaître :

```
$ rpm -Uvh foo-1.0-1.i386.rpm
foo package foo-2.0-1 (which is newer) is already installed
error : foo-1.0-1.i386.rpm cannot be installed
```

Pour ignorer cette erreur et donc pour forcer la mise à jour, utilisez l'option `--oldpackage`.

## 4.5 - Interrogation

Pour interroger la base de données sur les packages installés, utilisez la commande `rpm -q`. Utilisez simplement `rpm -q foo` pour afficher le nom, la version et le numéro de mise à jour du package `foo` :

```
$ rpm -q foo
rpm-2.0-1
```

Vous pouvez remplacer le nom du package par une des ces options (Package Specification Options) :

- \* `-a` interroge tous les packages installés
- \* `-f <file>` interroge tous les packages qui contiennent le fichier `<file>`
- \* `-p <packagefile>` interroge le package `<packagefile>`

Il existe plusieurs façons de spécifier les informations à afficher lors de l'interrogation des packages. Vous pouvez utiliser les options suivantes (Information Selection Options) :

- \* `-i` affiche un certain nombre d'informations sur le package (nom, description, numéro de mise à jour, taille, date de création, date d'installation, fournisseur, ...)
- \* `-l` affiche la liste de fichiers contenus dans le package
- \* `-s` affiche l'état de tous les fichiers du package (état : normal ou absent)
- \* `-d` affiche une liste des fichiers de documentation (pages du man, pages info, README, ...)
- \* `-c` affiche une liste des fichiers de configuration. Ces fichiers sont ceux que vous modifierez après l'installation pour les adapter à votre système (`sendmail.cf`, `passwd`, `inittab`, ...)

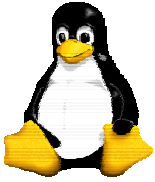
Pour les options qui affichent une liste de fichiers, vous pouvez ajouter `-v` pour un affichage avec un format similaire à celui de la commande `ls -l`.

Par exemple, pour savoir les paquetages installés sur votre machine, il suffit de faire :

```
gandalf # rpm -q -a
mailcap-1.0-3
rpm-devel-2.2.5-1
groff-1.10-6
```

On peut aussi grouper les options : `-qa` est équivalent à `-q -a`. Pour avoir plus de renseignements sur les différents paquetages installés (numéro de version, quelle distribution, ...), il suffit de rajouter en plus l'option `-l` et `-i`. Par exemple :

```
gandalf # rpm -qil rpm-devel
Name      : rpm-devel          Distribution: Red Hat Linux Colgate
Version   : 2.2.5            Vendor: Red Hat Software
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

Release : 1 Build Date: Thu Sep 05 23:14:10 1996  
Install date: Wed Nov 13 11:30:50 1996 Build Host: porky.redhat.com  
Group : Development/Libraries Source RPM: rpm-2.2.5-1.src.rpm  
Size : 136142  
Summary : Header files and libraries for programs that manipulate rpm packages

## Description :

The RPM packaging system includes a C library that makes it easy to manipulate RPM packages and databases. It is intended to ease the creation of graphical package managers and other tools that need intimate knowledge of RPM packages.

/usr/include/rpm

/usr/include/rpm/dbindex.h

/usr/include/rpm/header.h

/usr/include/rpm/messages.h

/usr/include/rpm/rpmerr.h

/usr/include/rpm/rpmlib.h

/usr/lib/librpm.a

Ces options peuvent être utilisées sur un fichier rpm que vous souhaitez installer pour savoir ce qu'il contient : il suffit d'ajouter l'option -p :

```
gandalf # rpm -qlp giftrans-1.11.1-4.i386.rpm
```

/usr/bin/giftrans

/usr/man/man1/giftrans.1

Si vous ajoutez l'option -i, vous obtenez les renseignements sur le paquetage.

## 4.6 - Vérification

La vérification permet de comparer des informations relatives aux fichiers installés sur le système avec celles du package d'origine.

Entre autres, la vérification se fait sur la taille, sur la somme MD5, sur les permissions, sur le type, sur le propriétaire et sur le groupe de chaque fichier.

rpm -V vérifie un package. Vous pouvez aussi combiner d'autres options décrites plus haut (Package Selection Options). La commande rpm -V foo vérifie que tous les fichiers du package foo sont tels qu'ils étaient lors de l'installation d'origine. Exemples :

\* Pour vérifier qu'un package contient un fichier particulier :

```
rpm -Vf /bin/vi
```

\* Pour vérifier tous les packages installés :

```
rpm -Va
```

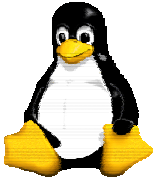
\* Pour vérifier un package installé par rapport au package RPM :

```
rpm -Vp foo-1.0-1.i386.rpm
```

\* Ceci peut être utile si vous pensez que votre base de données est corrompue.

Si la vérification s'est correctement déroulée, vous ne verrez rien apparaître. Si la vérification met à jour des différences, une chaîne de huit caractères apparaît. Le caractère c suivi du nom du fichier indique qu'il s'agit d'un fichier de configuration. Chacun des huit caractères représente une comparaison, si un point (.) apparaît cela signifie que la comparaison n'a pas échoué. Voici les caractères utilisés lorsque la comparaison a échoué :

5	Somme MD5
S	Taille du fichier



# Le système GNU/Linux

Version 9  
30/09/2003

---

L	Lien symbolique
T	Date de modification du fichier
D	Périphérique
U	Utilisateur
G	Groupe
M	Mode (permissions et type de fichier)

Suivant le résultat, c'est à vous de déterminer la marche à suivre (supprimer le package, le réinstaller, corriger le problème, ...).

## 5 - La gestion des disques et des partitions

### 5.1 - Montage manuel de partitions : mount

Comme nous l'avons vu dans le paragraphe précédent, les partitions sont montées lors de l'amorçage grâce au fichier `/etc/fstab`. Vous pouvez toutefois les monter à la main. Il faut alors utiliser la commande `mount`. Par exemple, pour monter la disquette A:, faire : `mount -t msdos /dev/fd0 /mnt`. Le `-t` indique le type du support. Parmi les plus courants, on trouve

- `ext2` : filesystem Linux ;
- `msdos` : disque(tte)s MS-DOS ;
- `vfat` : disque(tte)s MS-Windows 95 ;
- `iso9660` : Cd-ROM ;
- `nfs` : montage de partitions à travers le réseau.

Bien évidemment, il faut que ces types de systèmes de fichiers aient été intégrés au noyau...

Le `/dev/fd0` est le nom du fichier spécial correspondant au périphérique. Le répertoire est le point de montage.

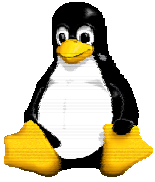
Pour démonter, tapez `umount /mnt`.

### 5.2 - Montage automatique de partitions : /etc/fstab

Le fichier `/etc/fstab` contient toutes les informations concernant le montage de vos partitions.

Voici un exemple :

```
#/etc/fstab
# Device      Repertoire  type      options    frequence  passe
# Linux
/dev/hda2     /           ext2      defaults   5          1
/dev/hdb2     /usr2      ext2      defaults   5          2
/dev/sda2     /usr3      ext2      defaults   10         2
# MS-DOS
/dev/hda1     /dos       msdos     defaults   0          0
/dev/hdb1     /dos2     msdos     defaults   0          0
# Processus
none         /proc      proc      defaults   0          0
# Partition de swap
/dev/hda3     none       swap      defaults   0          0
# Fichier de swap
/usr2/swap_file /usr2     swap      defaults   0          0
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

# NFS

```
zeppelin:/users      /users nfs          rw,rsize=8192,wsiz
```

Voici une description des six colonnes :

1. device (périphérique) de la partition. Dans le cas d'un fichier de swap, c'est le nom du fichier.
2. point de montage de la partition ;
3. type de la partition ;
4. options (vous pouvez spécifier une partition en lecture seule, etc).
5. fréquence correspond au nombre de jours entre deux traitements du fichier par la commande dump. Cette commande n'existe que pour ext2fs (c'est un portage de la version 4.4BSD) mais n'est pas encore incluse dans toutes les distributions.
6. Ordre de tests des partitions (fsck). Si vous mettez 0, aucune vérification automatique n'est effectuée lors du démarrage. Les partitions situées sur un même disque seront testées d'une manière séquentielle mais si elles sont situées sur deux disques différents, c'est fait en parallèle.

Ajouter ici les partitions dont vous avez besoin.

Les volumes NFS profiteront des paramètres suivants : rsize=8192,wsiz=8192, surtout si vous possédez une carte réseau rapide.

Pour autoriser un utilisateur à monter un volume il faut créer une ligne contenant l'option user.

Exemple (cas d'un CD-ROM SCSI) :

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

N'importe quel utilisateur pourra dès lors monter et démonter un CD (en utilisant mount /mnt/cd, umount /mnt/cd) La page de man de mount, section 8 (man 8 mount) explique la signification des options possibles.

la RedHat fournit également l'utilitaire fstool sous X qui permet de rajouter et supprimer les différents périphériques montés à l'aide de quelques clics de souris et autres boîtes de dialogues.

## 5.3 - Opérations sur disques et file systems (mke2fs, e2fsck, fdformat)

Pour créer un file system de type ext2 sur une disquette

```
mke2fs -c /dev/fd0
```

Pour contrôler l'état d'un file system

```
e2fsck /dev/fd0
```

Formater une disquette de densité 1.4

```
fdformat /dev/fd0H1440
```

## 5.4 - Comment générer une disquette de boot

Il peut être très utile d'avoir sur disquette un noyau pour booter dessus. Pour faire cela, prenez une disquette formatée et allez à la racine de votre disque.

Identifiez votre noyau. Il doit s'appeler zImage ou vmlinuz.

Vous le copiez d'abord sur disquette :

```
cat /zImage > /dev/fd0
```

puis faites (exemple) :

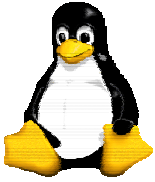
```
rdev /dev/fd0 /dev/hda2
```

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

Le deuxième paramètre de la première commande doit être le nom de la partition racine Linux de votre système.

Cette disquette vous permet alors de booter (tester !).

Si vous utilisez la Slackware, vous pouvez aussi utiliser la commande /sbin/makebootdisk qui fera tout cela à votre place.



# Le système GNU/Linux

---

## 5.5 - Partition de swap

### 5.5.1 - Swap : création de la partition

Pour rajouter une partition de swap de 8 Mo (8192 Ko), il faut d'abord créer la partition à l'aide du programme `fdisk`, puis soit vous passez par le programme `setup` de la Slackware, soit vous le faites à la main (ATTENTION : commande DANGEREUSE, destruction du contenu de la partition concernée ASSURÉE) : `mkswap /dev/hda3 8192`

Rajoutez la ligne `/dev/hda3 swap swap defaults 0 0` dans `/etc/fstab`.

Enfin, pour que ça fonctionne, réamorçez la machine ou lancez `swapon -a`.

### 5.5.2 - Swap : fichier

Pour le fichier, c'est plus compliqué. Invoquer :

```
dd if=/dev/zero of=/usr2/swap_file bs=1024 count=8192
```

```
mkswap /usr2/swap_file 8192
```

Ajoutez dans `/etc/fstab` la ligne : `/usr2/swap_file /usr2 swap defaults 0 0`.

Faites bien attention à mettre cette ligne après le montage de la partition `/usr2`. Sinon, ce n'est pas près de fonctionner.

Désormais, vous êtes obligé pour quitter Linux de faire un `swapon -a`

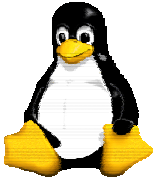
```
reboot      #Ou autre
```

En effet, si vous ne désactivez pas le fichier d'échange, Linux ne va pas pouvoir démonter la partition, et donc, il fera un `fsck` dessus à chaque fois que vous relancerez votre machine. Il est aussi possible, avec certains systèmes, de placer la commande `swapon -a` dans `/etc/rc.d/rc.K` ou bien `/etc/rc.d/init.d/halt` afin d'automatiser cela.

## 5.6 - Optimisation des disques

Sur certaines distributions l'accès aux disques durs IDE est réalisé en 16 bits alors qu'il pourrait l'être en 32 bits, ceci améliore grandement les performances du système. Pour pallier ce problème on utilise l'utilitaire `hdparm` comme le montre l'exemple ci-dessous :

```
[root@tux /root]# hdparm -c /dev/hda
/dev/hda:
I/O support = 0 (default 16-bit)
[root@tux /root]# hdparm -t /dev/hda
/dev/hda:
Timing buffered disk reads: 64 MB in 17.58 seconds = 3.64 MB/sec
[root@tux /root]# hdparm -c 1 -d 1 /dev/hda
/dev/hda:
setting 32-bit I/O support flag to 1
setting using_dma to 1 (on)
I/O support = 1 (32-bit)
using_dma = 1 (on)
[root@tux /root]# hdparm -t /dev/hda
/dev/hda:
Timing buffered disk reads: 64 MB in 5.24 seconds =12.21 MB/sec
[root@tux /root]# hdparm -k 1 /dev/hda
/dev/hda:
```



# Le système GNU/Linux

```
setting keep_settings to 1 (on)
keepsettings = 1 (on)
```

L'option `-t` de `hdparm` permet de tester les paramètres du disque en lecture, `-c` permet de passer en mode 32 bits alors que `-d` active l'utilisation du DMA. Enfin, l'option `-k` enregistre ces paramètres jusqu'à l'extinction de la machine. Il faudra donc bien veiller à insérer ces lignes au démarrage du système dans le `rc.local` par exemple.

## 6 - L'administration du système

### 6.1 - Ajouter un utilisateur, un groupe

Si vous possédez la Red Hat, vous pouvez utiliser le panneau de configuration (control-panel ou bien directement `usercfg`, utilisables par root sous X Window) : c'est vraiment très bien fait mais CELA N'EMPECHE PAS DE CONNAITRE LES VRAIS COMMANDES.

Ajouter un utilisateur se fait en deux temps et trois mouvements, la première solution est d'utiliser la commande `adduser` dont la syntaxe est la suivante :

```
useradd [-c comment] [-d home_dir]
        [-e expire_date] [-f inactive_time]
        [-g initial_group] [-G group[,...]]
        [-m [-k skeleton_dir] | -M] [-p passwd]
        [-s shell] [-u uid [-o]] [-n] [-r] login
```

Pour plus d'information, consulter l'aide en ligne.

La deuxième solution est manuelle, deux fichiers sont importants : `/etc/passwd` et `/etc/group`.

Une ligne du fichier `passwd` est composée de cette manière :

```
gandalf::501:100:Eric Dumas:/home/gandalf:/bin/bash
sgandalf::0:0:Super Eric Dumas:/:/bin/sh
```

Une ligne est composée de 7 colonnes :

1. Nom de l'utilisateur : 8 caractères max ;
2. Mot de passe chiffré (périmé avec les shadow password);
3. Numéro d'identificateur unique de l'utilisateur (uid). Rq : 0 = root ;
4. Numéro du groupe (gid) ;
5. Nom "officiel" de l'utilisateur (dit GECOS) ;
6. Répertoire racine de l'utilisateur ;
7. Interpréteur de commandes.

Si le champ mot de passe contient le caractère `x`, c'est que vous utilisez les shadow. Le mot de passe chiffré se trouve dans le fichier `/etc/shadow`. Cette configuration permet en plus de définir la durée de validité des mots de passe.

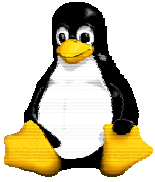
Le fichier `/etc/group` contient la liste des groupes de la machine. Par exemple, le groupe 100 est :

```
users::100:games,gandalf
```

Pour ajouter un utilisateur *toto*, faire

1. ajoutez-le dans `/etc/passwd` ;
2. si besoin est, créez le groupe dans lequel il va se trouver, en modifiant `/etc/group` ;
3. créez son home directory :

```
mkdir /home/toto
cp /etc/skel/* /home/toto
chown -R toto.le_groupe_de_toto /home/toto
```



4. Enfin, donnez-lui un mot de passe en tapant `passwd toto` .

Vous n'avez plus qu'à essayer.

Je vous conseille très fortement de travailler sous un utilisateur commun. En effet, il est très dangereux de travailler en root. Une fausse manipulation et vous devez tout réinstaller, voire perdre de précieux fichiers personnels.

## 6.2 - Enchaînement des scripts de démarrage

Le processus `init` lit ses paramètres dans `/etc/inittab`. Ce dernier donne les scripts à lancer en fonction d'un niveau de démarrage. Le niveau 3 est le niveau par défaut. « `init` » lance ensuite le scripts `/etc/rc n` qui va par la suite lancer les scripts présents dans le répertoire `/etc/rc.d/rcn.d` ou " `n` " est le niveau de démarrage choisi... exemple : `/etc/rc 3 -> /etc/rc.d/rc3.d`

Dans ce répertoire on trouve un ensemble de scripts commençant par `S` ou `K` et qui sont des liens vers `/etc/rc.d/init.d`. Pour un niveau donné, afin de lancer ou ne pas lancer des daemons au démarrage il suffit de mettre ou d'enlever les scripts correspondant dans le répertoire `rc3.d`.

On trouve des fichiers commençant par `S` s'il s'agit de lancer le script ou `K` s'il s'agit de tuer un process... Par exemple, pour ne pas démarrer en niveau 3 le serveur NIS , il suffit d'invalider le script `/etc/rc3.d/S65ypserv` , simplement en renommant `S65ypserv` en `K65ypserv`.

Le numéro se trouvant après le `S` indique l'ordre dans lequel vont être exécutés les scripts, plus le numéro est petit, plus il est prioritaire et sera donc exécuté avant les autres. Ceci s'explique par le fait que par exemple, la couche réseau doit être démarrée avant de démarrer le daemon apache (serveur web).

L'ensemble de ces scripts est utilisé aussi bien lors du démarrage, ils reçoivent donc l'argument `start`, qu'à l'extinction du système avec l'argument `stop`. De plus, un troisième argument « `status` » permet de connaître l'état du service lancé par ce script.

Enfin, un dernier fichier est exécuté : `/etc/rc.d/rc.local`. Ainsi, lorsque l'on veut paramétrer un nouveau service, deux solutions sont envisageables, soit créer un nouveau script `Syydaemon` (basé sur un script existant), soit ajouter des lignes dans le fichier `rc.local`. Il est bien évident que la première solution est la meilleure.

## 6.3 - Ordonnancement de travaux : la *crontab*

La *crontab* est une table qui contient des commandes qui doivent être lancées à intervalle régulier. Vous pouvez y mettre la sauvegarde journalière, etc.

Le format est assez simple :

```
# Faire une sauvegarde tous les Lundi a 2 heures du matin
```

```
0 2 * * 1 /home/gandalf/backup
```

```
# Rebooter la machine tous 1ers et 15 du mois a 4h15.
```

```
15 4 1,15 * * /sbin/shutdown -r +3
```

Les 5 premières colonnes sont :

1. Minute (0 à 59) ;

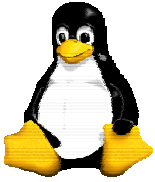
2. Heure (0 à 23) ;

3. Jour du mois (1 à 31) ;

4. Mois (1 à 12) ;

5. Jour de la semaine (0 à 6 : 0 = Dimanche, 1 = Lundi).





# Le système GNU/Linux

Version 9  
30/09/2003

Ensuite vient la commande. Cette crontab est très utile si vous utilisez UUCP (pour envoyer le courrier, récupérer les news, etc).

Tout utilisateur peut se créer une crontab grâce à la commande crontab.

## 6.4 - Le fichier `/etc/syslog.conf`

Le démon syslogd permet de récupérer différents messages du noyau, qui concernent des messages de lancement de certains programmes, des messages de login, et également des messages d'erreur. Il est parfois intéressant de pouvoir consulter ces messages pour détecter une éventuelle erreur ou le dysfonctionnement d'un programme ou d'un service.

Pour cela le fichier `syslog.conf` permet avec des règles de choisir où ces messages doivent parvenir. Il peut donc être utile de rajouter en première ligne du fichier `/etc/syslog.conf` :

```
*.* /dev/tty8
```

Elle va donc rediriger tout ces messages dans la console `tty8`.

Il faut rappeler que pour changer de console, il suffit de faire `Alt+Fx`, `Fx` correspondant à la touche de fonction comportant le numéro de la console que vous voulez atteindre. Sous X, il faut faire `Ctrl+Alt+Fn`.

## 6.5 - La rotation des fichiers de logs : `logrotate`

Le programme `/usr/sbin/logrotate` permet une rotation des fichiers de log, on lui passe en argument un fichier de config (par exemple `/etc/logrotate`). Dans la distribution redhat, ce programme est appelé par un script placé dans le répertoire `/etc/cron.daily`, ce qui veut dire qu'il est appelé tous les jours car tous les fichiers placés dans ce répertoire sont exécutés tous les jours par le démon `cron`.

Le fichier `/etc/resolv.conf`

## 6.6 - Mise en place des quota

### 6.6.1 - Logiciels nécessaires

Le support des quota disque a été intégré dans le noyau Linux depuis la version 1.3.46. Il vous faut donc utiliser un noyau postérieur pour pouvoir bénéficier des quota.

Vous avez également besoin des outils de gestion des quota. On peut les trouver sur :

<ftp://ftp.funet.fi/pub/Linux/PEOPLE/Linus/subsystems/quota>

ou, plus près, sur :

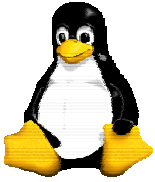
<ftp://ftp.lip6.fr/pub/linux/kernel/sources/subsystems/quota>

### 6.6.2 - Compilation des utilitaires

Après avoir extrait le contenu de l'archive, il vous faut compiler ces utilitaires s'il s'agit d'un fichier `.tar.gz`. Cela se fait en tapant tout simplement `make`. Dans certains cas (selon la version de bibliothèque C utilisée), il peut être nécessaire d'ajouter `-I.` à la variable `CFLAGS` définie dans le `Makefile`.

Si vous utilisez le système de fichiers `ext2` et que vous avez installé les bibliothèques contenues dans la distribution `e2fsprogs` (utilitaires de gestion du système de fichiers `ext2`), vous pouvez inclure le support `ext2fs` dans les utilitaires des quota, afin d'accélérer leur exécution. Pour cela, éditez le fichier `Makefile` afin que la variable `CFLAGS` contienne `-DEXT2_DIRECT` et que la variable `EXT2LIBS` contienne `-lext2fs -lcom_err` avant de compiler les programmes.





# Le système GNU/Linux

Version 9  
30/09/2003

Après la compilation, installez les utilitaires par la commande `make install`. Attention, la commande `quota` est installée dans le répertoire `/usr/ucb` qui n'existe pas sur la plupart des installations Linux. Il vous faut donc modifier le `Makefile` pour installer cette commande dans un autre répertoire (`/usr/bin` est un bon choix). Il peut également être nécessaire de définir la variable `LN` à la valeur `ln -sf`. Si il s'agit d'un package `rpm`, la commande est décrite précédemment.

## 6.6.3 - Activation de la gestion des quota dans le noyau

Pour utiliser les quota, le noyau doit avoir été compilé avec le support des quota. Pour cela, vous devez avoir un noyau dans lequel vous ayez répondu "Y" à la question "Quota support". Attention : le support des quota n'est pour l'instant effectif que sur les systèmes de fichiers `ext2`.

## 6.6.4 - Activation des quota sur les systèmes de fichiers

Il existe deux types de quota : les quota liés aux utilisateurs et les quota liés aux groupes d'utilisateur. Les premiers définissent le nombre maximal de fichiers et de blocs disque associés à chaque utilisateur, les seconds définissent le nombre maximal de fichiers et de blocs disque associés à chaque groupe d'utilisateurs. Les deux types de quota peuvent être activés séparément.

Pour activer les quota pour les utilisateurs, il faut indiquer l'option `usrquota` pour les systèmes de fichiers concernés dans `/etc/fstab`. Les quota concernant les groupes sont régis par l'option `grpquota`. Les fichiers de définition des quota s'appellent respectivement `quota.user` et `quota.group` et sont situés dans la racine de chaque système de fichiers concerné.

Il est possible de modifier les noms de fichiers de gestion des quota en utilisant les syntaxes suivantes :

```
usrquota=nom_de_fichier
```

```
grpquota=nom_de_fichier
```

Voici un exemple de fichier `/etc/fstab` :

```
/dev/hda2 / ext2 defaults,rw 0 1
```

```
/dev/hdb2 /home ext2 defaults,rw,usrquota,grpquota 0 1
```

```
/dev/sda1 /usr/src ext2 defaults,rw,usrquota 0 1
```

L'activation des quota est ensuite lancée par la commande `quotaon`. Pour les activer automatiquement lors du démarrage du système, il faut ajouter dans un fichier d'initialisation (situé généralement dans `/etc/rc.d`) les lignes :

```
# Enable quota
```

```
/usr/sbin/quotaon -avug
```

Il peut également être nécessaire de vérifier la cohérence des informations de gestion des quota après des arrêts intempestifs. Pour cela, il faut utiliser la commande `quotacheck` :

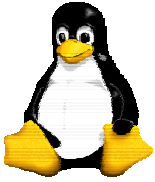
```
/usr/sbin/quotacheck -avug
```

Cette commande doit également être exécutée la première fois pour créer les fichiers de gestion des quotas : `quota.user` et `quota.group`. En effet, ils ne sont pas créés automatiquement.

Si les utilitaires ont été compilés avec `-DEXT2_DIRECT`, la commande `quotacheck` doit être relativement rapide. Dans le cas contraire, elle peut être très lente car elle doit explorer tous les répertoires du système de fichiers de manière hiérarchique. Il est, de toutes façons, conseillé de la lancer automatiquement lors du démarrage avant l'activation des quota par `quotaon` :

```
# Check quota
```

```
/usr/sbin/quotacheck -avug
```



# Le système GNU/Linux

## 6.6.5 - Attribution d'un quota à un utilisateur

La commande `edquota` est utilisée pour affecter un quota à un utilisateur ou à un groupe d'utilisateurs. Sa syntaxe est `edquota -u utilisateur` ou `edquota -g groupe`. Cette commande lance un éditeur de texte contenant la définition des quota attribués à l'utilisateur ou au groupe et prend en compte leur nouvelle valeur lorsque le fichier est ré-écrit.

Pour chaque utilisateur ou groupe, il existe deux limitations : le nombre de fichiers et le nombre de blocs disque (exprimés en blocs de 1024 octets). Pour chacune, il existe deux limites :

1. la limite "douce" : lorsque cette limite est atteinte ou dépassée, un message d'avertissement est affiché lors de chaque nouvelle allocation de fichier ou de bloc ;
2. la limite "dure" : lorsque cette limite est atteinte, il est impossible à l'utilisateur de créer de nouveau fichier ou d'allouer de nouveau bloc.

La limite "douce" se transforme en limite "dure" quand elle a été atteinte ou dépassée depuis un certain temps (sept jours par défaut).

## 6.6.6 - Statistiques sur les quota

Tout utilisateur peut obtenir l'état des quota qui lui sont attribués (les limites sur le nombre de fichiers et de blocs qui lui sont alloués) grâce à la commande `quota`.

Le super-utilisateur peut obtenir les mêmes informations sur tout utilisateur ou groupe avec la même commande : `quota -u utilisateur` ou `quota -g groupe`. Il peut de plus utiliser la commande `repquota` pour obtenir une liste des quota associés à un ou plusieurs systèmes de fichiers.

## 6.6.7 - Autres documentations sur les quota

L'implémentation des quota dans Linux est compatible avec l'implémentation originale dans 4.2BSD Unix. Toutes les documentations concernant les quota dans BSD, que ce soit au niveau des commandes ou au niveau de l'interface de programmation, sont donc utilisables.

Les pages de manuel des utilitaires sont assez claires et très instructives. Enfin, les documentations diffusées en format HTML forment une bonne introduction à la gestion des quota.

## 6.7 - Lancer des " R "(emote ) commandes

Pour autoriser un utilisateur sur une machine distante à lancer des commandes sur la machine locale, il faut l'autoriser en le répertoriant dans le fichier `~/rhosts`

Ceci n'est peut être pas suffisant...il arrive que " root " ne puisse pas se rloguer ou faire des remote commande (comme rsh) depuis une autre machine. Ceci est dû au système de sécurité de `/etc/securetty` visant à n'autoriser des connexions que depuis un terminal " sûr ". On sécurise les accès depuis une console authentifiée.

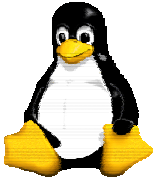
Pour invalider cette sécurité il faut commenter la ligne suivante dans le fichier `/etc/pam.conf`.

```
#rlogin auth required
```

et laisser

```
rlogin auth sufficient
```

La commande `/sbin/pamconfig` permet de paramétrer ce fichier. Ce fichier `/etc/pam.conf` régit la possibilité d'utiliser ou pas la plupart des rcommandes à distance...depuis certains comptes ou certaines consoles tty.



# Le système GNU/Linux

---

## 6.8 - Les fichiers temporaires et la commande *tmpwatch*

Le programme `/usr/sbin/tmpwatch` permet d'effacer des fichiers qui n'ont pas été utilisés depuis un certain temps. Par exemple la ligne : `/usr/sbin/tmpwatch 240 /tmp /var/tmp /var/catman/cat?` peut être ajoutée dans un fichier dans le répertoire `/etc/cron.daily`, répertoire dont les fichiers sont exécutés tous les jours par le programme `cron` dans la distribution `redhat` (voir `crontab`).

## 6.9 - La commande *strace*

On peut tracer un processus à l'aide de la commande `strace`. Par exemple `strace -p <pid du process>`, ceci permet d'avoir les signaux et les appels systèmes. C'est utile pour savoir si un processus est planté ou pas.

## 6.10 - La commande *lsdf*

Avec ce programme, vous pouvez apprendre quels fichiers sont ouverts par quels processus. Il convient de noter que `lsdf` ne se limite pas aux fichiers 'normaux'. Vous pouvez aussi demander des informations sur les périphériques blocs et caractères, les bibliothèques, les 'streams' ou les fichiers réseau (fichiers NFS ou sockets de domaine UNIX). Vous pouvez aussi demander des informations sur des systèmes de fichiers complets...

## 6.11 - La commande *stat*

Affiche le contenu de l'inode, à savoir essentiellement la date de dernier accès et la date de dernière modification.

## 6.12 - *locate*

Lorsque vous installez le package GNU "find", en plus du programme de recherche se trouve un programme `locate` qui permet de retrouver rapidement l'emplacement d'un fichier. Ce programme se trouve dans certaines distributions dans `/usr/lib/locate`.

Lancez, en tant que `root`, `/usr/lib/locate/updatedb`. Cela invoquera un `find` / sur les disques montés et placera les noms de tous les fichiers (sous forme d'une table de hachage) dans `/usr/lib/locate/find.codes`. `locate` permettra ensuite de localiser l'un d'eux.

Remarque : lorsque vous lancez ce programme, pensez à démonter les partitions MS-DOS et votre CD-ROM !

Une autre solution assez agréable consiste à insérer dans la *crontab* la ligne :

```
updatedb --prunepaths="/tmp /usr/tmp /var/tmp /proc /users /root /dos  
/mnt /var/adm /var/spool /var/catman /home"
```

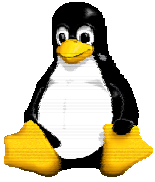
Cette commande réalise la mise à jour de l'arborescence sans les fichiers temporaires, l'arborescence utilisateur, une partition MS-DOS, etc.

Il est souhaitable d'exclure les répertoires utilisateurs des utilisateurs. En effet, ces partitions contiennent normalement de nombreux fichiers. Mais cela pose un grave problème de sécurité, car tout utilisateur peut sinon connaître le nom des fichiers du voisin.

## 6.13 - Les fichiers d'initialisation des interpréteurs de commandes

Voici la description des fichiers qui sont chargés lors de la connexion d'un utilisateur. Le nom de ou des fichiers est spécifique à l'interpréteur de commandes utilisé.

Dans le cas de `bash`, c'est un peu compliqué. D'après la page de manuel de `bash`, lors d'une connexion interactive, il exécute :



# Le système GNU/Linux

- /etc/profile s'il existe ;
- ~/.bash\_profile s'il existe, sinon ~/.bash\_login s'il existe, sinon ~/.profile s'il existe ;
- ~/.bashrc s'il existe.

Dans le cas où bash est invoqué en tant que sh, il n'exécute que les fichiers /etc/profile et ~/.profile s'ils existent.

Dans le cas de ksh, la page de manuel est moins précise mais une étude rapide du code source montre que :

- /etc/profile est exécuté s'il existe ;
- ~/.profile est exécuté s'il existe, si le numéro d'utilisateur effectif est identique au numéro d'utilisateur réel et si le numéro de groupe effectif est égal au numéro de groupe réel ;
- /etc/suid\_profile est exécuté s'il existe, si les numéros d'utilisateur effectif et réel diffèrent ou si les numéros de groupe effectif et réel diffèrent ;
- enfin, si la variable d'environnement ENV est positionnée, le fichier dont le nom est contenu dans \$ENV est exécuté s'il existe.

Enfin, pour être complet sur les interpréteurs de commandes compatibles avec le Bourne Shell, citons le cas de zsh qui exécute :

- /etc/zshenv s'il existe ;
- ~/.zshenv (ou \$ZDOTDIR/.zshenv si ZDOTDIR est positionné) s'il existe ;
- /etc/zprofile s'il existe ;
- ~/.zprofile (ou \$ZDOTDIR/.zprofile) s'il existe ;
- /etc/zshrc s'il existe ;
- ~/.zshrc (ou \$ZDOTDIR/.zshrc) s'il existe.

Dans le cas de csh (enfin plutôt tcsh) :

- /etc/csh.cshrc est exécuté s'il existe ;
- /etc/csh.login est exécuté s'il existe ;
- ~/.tcshrc est exécuté s'il existe, sinon ~/.cshrc est exécuté s'il existe ;
- ~/.history (ou le contenu de la variable histfile) est chargé en mémoire s'il existe ;
- ~/.login est exécuté s'il existe ;
- ~/.cshdirs (ou le contenu de la variable dirsfile) est chargé en mémoire s'il existe.

## 6.14 - Les sauvegardes sous Linux

Comme tout système Unix, Linux fournit des outils standards pour réaliser les sauvegardes des disques. Principalement deux types d'outils sont actuellement utilisés.

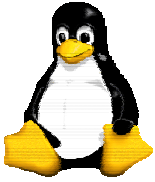
### 6.14.1 - dump et restore

Les outils dump et restore ont été adaptés pour Linux par Rémy Card. Ils sont livrés dans les distributions.

Une fois les sources compilés et installés, l'utilisation de dump et restore est relativement simple. Pour effectuer la sauvegarde d'une partition /dev/sda1 sur /dev/rmt0, il suffit de faire par exemple :

```
gandalf# dump 0sfu 3600 /dev/rmt0 /dev/sda1  
gandalf# dump 0sfu ondee:/dev/rmt0 /dev/sda1
```

La deuxième commande permet de sauvegarder un disque sur un périphérique distant (par exemple situé ici sur la machine ondee). Les options de dump peuvent sembler complexes. En voici une courte description :



# Le système GNU/Linux

Version 9  
30/09/2003

– 0 à 9 : niveau de sauvegarde. 0 correspond à une sauvegarde complète, alors que les autres niveaux n correspondent à la sauvegarde des fichiers qui ont été modifiés depuis la nième sauvegarde ;

– s : taille de la bande en pieds ;

– f : fichier - peut être composé de machine:fichier ;

– u : écriture de la date et du niveau de sauvegarde dans le fichier /etc/dumpdates.

Il existe d'autres options. Pour plus renseignements, consultez les pages de manuel.

Il existe deux modes pour effectuer une restauration : en ligne de commande ou en mode dit "interactif". Le deuxième mode est plus simple pour des restaurations partielles. Le premier est surtout utilisé pour des restaurations complètes.

Pour restaurer la bande en mode interactif il suffit de faire

```
gandalf# restore -if /dev/rmt0
```

```
gandalf# restore -if ondee:/dev/rmt0
```

Dans ce cas, un mini-interpréteur de commandes est lancé. Utiliser la commande help pour plus de détails.

Pour restaurer une bande complètement, lancez :

```
gandalf# restore rf /dev/rmt0
```

Note importante : pour l'utilisation de dump et restore à travers un réseau (sauvegarde sur des périphériques distants), vous devez utiliser des fichiers .rhosts. Dans l'exemple de sauvegarde ci-dessus, la machine ondee doit avoir

```
#cat ~root/.rhosts
```

```
gandalf
```

```
#
```

Attention toutefois aux failles de sécurité engendrées par les fichiers .rhosts.

L'utilisation de périphériques distants nécessite également la présence du programme rmt sur la machine gérant les périphériques de sauvegarde. Ce programme est inclus dans la distribution source de dump pour Linux.

## 6.14.2 - tar

A la différence de dump ou restore, tar permet de sauvegarder les fichiers désirés, d'exclure certains répertoires, etc. Il est à noter que le tar utilisé sous Linux est le tar GNU. Il possède certaines options particulières.

Pour connaître toutes les options possibles, je vous conseille de faire tar --help . Une utilisation simple de tar peut être illustrée avec la sauvegarde d'une partition d'utilisateurs :

```
# tar cvf /dev/rmt0 /users | mail backup-user
```

La liste des fichiers sera ainsi envoyée à l'utilisateur *backup-user*.

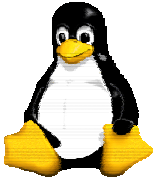
Certains sites utilisent exclusivement tar pour effectuer leurs sauvegardes, c'est un choix.

On l'utilise également pour faire des archives « .tar », on l'obtient par exemple avec la commande suivante :

```
tar -cvf monfichier.tar *
```

qui va mettre tous les fichiers du repertoire courant dans l'archive monfichier.tar, on pourra extraire cette archive avec la commande suivante :

```
tar -xvf monfichier.tar
```



# Le système GNU/Linux

---

## 6.15 - Les fichiers core

Certains développeurs aiment exploiter le contenu des fichiers core. Or, avec certaines distributions ils sont désactivés par défaut. Pour pouvoir les générer, il faut introduire : `ulimit -c unlimited` sous bash ou `limit coredumpsize unlimited`

## 6.16 - Consoles virtuelles

### 6.16.1 - Aspects pratiques

Linux vous permet de travailler en même temps sur plusieurs fenêtres textes. Pour passer de l'une à l'autre, fais `Alt + Fn` ou `n` est le numéro de la fenêtre virtuelle et `Fn` une touche de fonction (par exemple `F1`, `F2` ...). Vous pouvez configurer le nombre de fenêtres dans le fichier `/etc/inittab`.

Quand vous êtes sous X, la combinaison devient `Ctrl + Alt + Fn`. Il est alors possible de retourner sous X en accédant ainsi à la console qu'il emploie (la première "libre", c'est-à-dire non gérée par un `getty`).

Utiliser `maj + pages` (touches `SHIFT` et page précédente ou page suivante) pour "remonter". Ceci fonctionne aussi sous `xterm`.

### 6.16.2 - Aspects théoriques

#### 6.16.2.1 - Principe

Les consoles virtuelles sont des périphériques. Elles sont au nombre de 63 (ou plus, ou moins, si l'on change ça lors de la compilation du noyau). Elles correspondent aux fichiers "spéciaux" `tty1` ... `tty63` du répertoire `/dev`.

Maintenant, pour pouvoir les utiliser, il faut pouvoir attacher un processus à ce terminal. C'est le rôle des utilitaires tels que `agetty`. Il est d'ailleurs intéressant de suivre (à l'aide de `top`) l'évolution des processus lors du login :

Tout d'abord, `agetty` (de propriétaire `root`) vous demande votre login. Dès que vous appuyez sur entrée, `agetty` exécute le programme `login` (toujours de propriétaire `root`) par recouvrement.

`login` (attaché à la VC puisqu'héritant de `agetty`) vous demande votre mot de passe. Il lance alors votre shell de login (de propriétaire "vous").

Vous pouvez noter que quand vous terminez votre session (i.e. votre shell de login), un `agetty` est relancé.

#### 6.16.2.2 - Configuration

C'est là qu'intervient le fichier `inittab`. `init` (de PID 1), est le processus ancêtre de tous les processus. Comme c'est le premier processus, c'est lui qui se charge de lancer les `agetty`. Il se sert pour cela du fichier de configuration `/etc/inittab`.

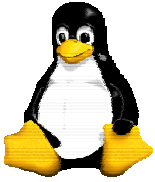
Entre autres choses, ce fichier contient des lignes du type :

```
c1:1235:respawn:/sbin/agetty 38400 tty1 linux
```

Pour une explication, se référer à la page de manuel `d'inittab(5)`. Ca veut dire en gros que `init` lance un `agetty` sur `/dev/tty1` qu'il relance chaque fois qu'il meurt (`respawn`) avec un type de terminal Linux pour les cas où on est dans les "runlevels" 1,2,3 ou 5.

Donc, il faut mettre une ligne de ce type pour chaque console virtuelle.





# Le système GNU/Linux

Version 9  
30/09/2003

On voit ici qu'il y a autant de agetty lancés que de consoles virtuelles utilisables, et qu'on est limité par ce nombre des lignes dans inittab (sauf si l'on lance 63 agetty...).

Il existe une autre méthode, c'est l'allocation de VC dynamiquement. Cela se fait grâce à un petit démon (cspawnd) (<ftp://ftp.lip6.fr/pub/linux/sunsite/utis/console/dynamic-vc-1.1.tar.gz>) :

« cspawnd is normally started in rc.local as a daemon that is signaled by the kernel when the console user presses the Spawn\_Console key (see below). Upon receipt of a signal from the kernel, cspawnd activates the newly allocated VC and spawns a getty process to enable the user to login. In its default configuration, it also attempts to periodically deallocate unused VCs ».

Il convient alors de laisser une VC gérée par la première méthode, les autres étant alouées dynamiquement.

## 6.16.2.3 - Les touches

Maintenant, il faut pouvoir passer d'une VC à l'autre. C'est le noyau qui s'en charge. Quand une touche "ConsoleN" (N=1...63) est frappée, Le noyau fait correspondre la VC numéro N à la console (la vrai).

Il y a aussi "Decr\_Console", "Incr\_Console" qui respectivement, incrémente et décrémente le numéro de la console ; ainsi que Last\_Console qui fait passer à la dernière console utilisée. cspawnd rajoute la touche "Spawn\_Console" (voir plus haut).

Il ne reste plus qu'à faire correspondre ceci à des touches (ou des combinaisons de touches) du clavier. Cela est fait grâce à l'utilitaire loadkeys (lancé par init au boot [dans un fichier /etc/rcXXX]). Cet utilitaire prend en paramètre un fichier .map (sûrement /usr/lib/kbd/keytables/fr-lat1.map pour vous) qui contient des lignes du style :

```
keycode 87 = F11 F11 Console_23
control    keycode 87 = F11
alt        keycode 87 = Console_11
control alt keycode 87 = Console_11
```

Bien sûr, vous pouvez retrouver tout ceci dans les pages de man et toutes les aides en ligne mises à votre disposition. Citons loadkeys(8), init(8), inittab(5), cspawnd(8) ainsi que les fichiers du répertoire /usr/src/linux/Documentation.

## 7 - X Window

Maintenant que tout fonctionne bien, on passe à un autre gros morceau : l'installation de X Window. Pour pouvoir utiliser X Window, il faut posséder une configuration raisonnable (au moins 8Mo de mémoire). Dans le cas contraire, la machine sera très lente : elle va passer son temps à swapper.

Dans un premier temps, installez à l'aide de votre distribution le serveur correspondant à la carte graphique.

### 7.1 - XF86Config

Le problème avec X Window, c'est de configurer le fichier XF86Config. Il s'agit ici du serveur XFree86 3.2, basé sur X11R6. Avant de commencer à construire votre fichier, lancez le programme SuperProbe. Ceci donne par exemple les renseignements suivants :

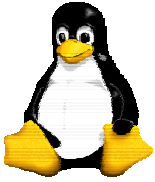
First video: Super-VGA

Chipset: Cirrus CL-GD5428

Memory: 1024 Kbytes

RAMDAC: Cirrus Logic Built-in 15/16/24-bit DAC

(with 6-bit wide lookup tables (or in 6-bit mode))



# Le système GNU/Linux

Version 9  
30/09/2003

Ensuite, le plus dur reste à faire : construire le fichier de configuration. Attention : dans certaines distributions, le fichier XF86config est placé dans le répertoire /etc/X11, ce qui est une bonne idée dans le cas d'un réseau de stations Linux partageant /usr par NFS. Si vous utilisez une ou plusieurs machines indépendantes, vous pouvez supprimer /etc/X11/XF86config et utiliser /usr/X11R6/lib/X11/XF86Config.

Les nouvelles versions de distributions sont livrées avec certains outils permettant d'automatiser l'installation et la configuration du serveur. Essayez dans un premier temps avec cet outil : il est fort probable que cela fonctionne directement. (Xconfigurator pour redhat.)

Ce fichier se présente sous la forme de sections. En voici une description :

– Fichiers

Section "Files"

```
RgbPath "/usr/X11R6/lib/X11/rgb"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/misc/"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/Type1/"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/Speedo/"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/75dpi/"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/100dpi/"
```

```
FontPath "/usr/X11R6/lib/X11/fonts/xtel/"
```

EndSection

Dans cette section, il n'y a pas grand chose à modifier. Ajoutez ou supprimez les chemins des polices de caractères, et la liste des couleurs RGB.

– Options pour le serveur :

Section "ServerFlags"

EndSection

– Clavier

Section "Keyboard"

```
Protocol "Standard"
```

```
# AutoRepeat 500 5
```

```
ServerNumLock
```

```
# Pour cela, voir plus bas
```

```
LeftAlt Meta
```

```
RightAlt ModeShift
```

```
# RightCtl Compose
```

```
ScrollLock ModeLock
```

– La souris :

Section "Pointer"

```
# pour le protocole, utilise MouseMan, MouseSystems, Logitech ou Microsoft
```

```
Protocol "MouseMan"
```

```
# Peripherique a utiliser
```

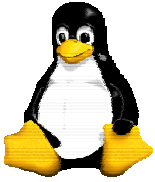
```
Device "/dev/mouse"
```

```
BaudRate 1200
```

```
# Emulation 3 boutons: le click sur les 2 boutons
```

```
# simulent le bouton du milieu... pour les souris a deux boutons.
```





# Le système GNU/Linux

Version 9  
30/09/2003

---

Emulate3Buttons

# ChordMiddle is an option for some 3-button Logitech mice

# ChordMiddle

EndSection

– Le moniteur. Il s'agit de détailler les caractéristiques du moniteur.. Les fichiers situés dans le répertoire doc contiennent les caractéristiques de tout un tas d'écrans. Il est fort possible que vous y trouviez votre bonheur.

Section "Monitor"

Identifier "Sync3N"

VendorName "Samsung"

ModelName "3N"

# Frequences:

Bandwidth 25.2

HorizSync 31.5-80

VertRefresh 60-90

# Les modes dans lesquels l'ecran peut fonctionner

Modeline "800x600" 40 800 840 968 1056 600 601 605 628

Modeline "640x480" 25.175 640 664 760 800 480 491 493 525

Modeline "1024x768i" 45 1024 1040 1216 1264 768 768 778 819 interlace

Modeline "1024x768" 75 1024 1048 1184 1328 768 771 777 806

EndSection

– la carte graphique.

Section "Device"

# Baratin

Identifier "CartePerso"

VendorName "Cirrus"

# Nom de la carte

BoardName "clgd5428"

#memoire video

VideoRam 1024

# nom du chip

Chipset "clgd5428"

# clocks d'horloges.

Clocks 25.23 28.32 41.16 36.08 31.50 39.99 45.08 49.87

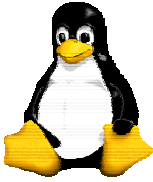
Clocks 64.98 72.16 75.00 80.01 85.23

EndSection

– Enfin, on réunit tout : la section écran :

Section "Screen"

Driver "svga"



# Le système GNU/Linux

---

```
# References aux autres sections.
```

```
Device "CartePerso"
```

```
Monitor "Sync3N"
```

```
# Le display utilise
```

```
Subsection "Display"
```

```
Depth 8 # 256 couleurs max
```

```
# Modes par priorite : de gauche a droite
```

```
Modes "1024x768" "800x600" "640x480"
```

```
ViewPort 0 0
```

```
# Resolution virtuelle
```

```
Virtual 1024 768
```

```
option "linear"
```

```
EndSubsection
```

```
EndSection
```

Si votre carte le permet, vous pouvez ajouter d'autres sous-sections pour le display :

```
Subsection "Display"
```

```
Depth 32
```

```
Modes "640x480"
```

```
ViewPort 0 0
```

```
Virtual 800 600
```

```
EndSubsection
```

Dans ce cas, le serveur fonctionnera en 16 millions de couleurs.

La configuration du serveur X est soit très rapide (marche tout de suite), soit très longue. Dans le deuxième cas, voici quelques conseils :

1. lire la documentation se trouvant dans `./doc`. Un bon nombre d'écrans et de cartes sont décrites avec les horloges à utiliser.
2. commencer avec une résolution faible (640x480) : c'est plus simple ;
3. si vous avez besoin des clocks de la carte, lancez `X -probeonly 2>/tmp/err` (si vous utilisez comme interpréteur de commandes `csh` ou `tsh`, remplacez `2>` par `>&`). Dans le fichier `/tmp/err` vont se trouver tous les messages du serveur, dont le chip de la carte ainsi que ses clocks. La seule chose qu'il va rester à configurer, c'est l'écran.
4. lire le "Video HOWTO", écrit par J.M. Vansteene et disponible sur `ftp.lip6.fr` dans le répertoire `/pub/linux/french/docs/HOWTO` ;
5. utiliser le programme `xvidtune` : il permet "d'affiner" la mise au point de l'image (centrage, etc). Par exemple, lors d'un tremblement de l'image, on peut résoudre le problème avec ce programme. Si jamais l'anglais vous rebute, vous pouvez lire une version française de toute la documentation XFree86 3.2, au format HTML. Pour cela, allez sur le serveur `http://www.kheops.com`.

## 7.2 - Clavier français sous X... utiliser `xkb`

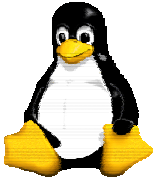
Il est nécessaire d'utiliser un serveur XFree86 3.2 minimum pour mettre en place cela. Dans la section "Keyboard" du fichier de configuration, ajouter :

```
XkbKeycodes "xfree86"
```

```
XkbTypes "default"
```

```
XkbCompat "default"
```

```
XkbSymbols "us(pc101)"
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
XkbGeometry "pc"  
XkbRules "xfree86"  
XkbModel "pc102"  
XkbLayout "fr"
```

Si le clavier n'offre pas 101 touches il faudra remplacer le "102" par "101", "pc102", "104" ou "105" afin de pouvoir d'utiliser les touches spéciales (MS-Windows).

Pour plus d'informations, regardez le contenu du répertoire /usr/X11R6/lib/X11/xkb.

## 7.3 - Clavier français sous X, ancienne méthode : Xmodmap

Comme XFree86 a été conçu par des gens se servant de claviers américains (qwertyiop), la configuration du clavier pouvait sembler assez affolante sur les serveurs Xfree86 inférieures à la 3.2. Il fallait pour cela utiliser xmodmap.

## 7.4 - xdm

Pour démarrer X Window directement au boot (xdm), vous pouvez modifier le fichier /etc/inittab en changeant la ligne contenant initdefault. Par exemple, il suffit de remplacer :

```
id:3:initdefault:      par      id:5:initdefault:
```

Mais attention ! Certaines distributions n'utilisent pas les mêmes niveaux. Le plus simple est de regarder dans le fichier /etc/inittab : celui-ci contient généralement une description des niveaux d'exécution et de ce à quoi ils correspondent.

Pour lancer xdm en 16 ou 32 bits, il faut modifier le fichier /usr/X11/lib/X11/xdm et mettre :

```
:0 local /usr/X11/bin/X -bpp 16 (ou 32)
```

## 7.5 - Sécurité : le fichier .Xauthority

Le fichier .Xauthority est généré par le programme xauth. Il s'agit d'un système d'authentification pour les applications graphiques. Cela permet d'éviter que d'autres personnes envoient des images, des fenêtres sur votre écran - mais également que des personnes puissent "voir" ce qu'il y a sur votre écran.

Le principe est de donner une clef d'identification, en hexadécimal avec un nombre pair de caractères.

Lancer xauth et faire :

```
add MaMachine:0 MIT-MAGIC-COOKIE-1 MonCode  
add MaMachine/unix:0 MIT-MAGIC-COOKIE-1 MonCode
```

Pour la machine locale, c'est en fait "hostname:NoDisplay".

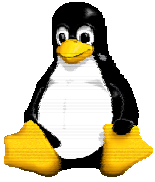
Un fois lancé, le serveur X interdit toute connexion, sauf si l'application :

- est exécutée sur une machine référencée ;
- possède le code.

Vous pouvez désactiver le système pour certaines machines avec un xhost +toto par exemple.

Rq: certaines versions obligent à lancer le serveur X avec la commande

```
xinit -- -auth $HOME/.Xauthority
```



## 8 - Recompiler le noyau

### 8.1 - Cyrix

Il existe des patches non officiels pour le noyau 2.0.xx. Toutefois, le mieux est d'utiliser l'utilitaire `set6x86` qui positionne certains registres du cyrix pour optimiser son fonctionnement. Le site incontournable pour tout ce qui concerne linux/cyrix : est <http://wauug.erols.com/~balsa/linux/cyrix/index.html>.

### 8.2 - Pentium

Linux gère sans aucun problème les machines basées sur Pentium. Pour optimiser un peu la compilation, il suffit de spécifier Pentium lors de la recompilation du noyau. Il existe toutefois un gcc spécialisé pour Pentium, il suffit de consulter la page <http://www.goof.com/pcg> pour plus de renseignements.

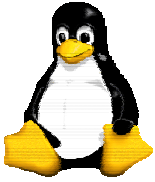
### 8.3 - Compilation

Les nouveaux sources du noyau Linux peuvent être récupérés sur le site <ftp.lip6.fr> dans le répertoire `/pub/linux/kernel/sources`. Il s'agit d'un miroir du site <ftp.kernel.org>.

Installez donc les sources (normalement dans le répertoire `/usr/src/linux`). La compilation est très simple à réaliser mais vous devez suivre quelques points très scrupuleusement :

1. conserver à portée de la main une disquette permettant de démarrer le système et de monter la partition root, afin de pouvoir reprendre le contrôle en cas d'erreur et tester qu'elle permet bien de démarrer.
  2. `make config` va demander la configuration que vous souhaitez. Certains points peuvent vous sembler assez obscurs. Dans ce cas, ne les changez pas. Vous pouvez lancer `make menuconfig` en mode texte pour avoir le programme en couleur ou `make xconfig` sous X à condition d'avoir installé Tcl/Tk. Remarque importante : il s'agit du noyau 2.0.26, donc certaines options ou certaines indications peuvent ne pas exister sur d'autres versions.
  3. `make dep` se charge de créer les dépendances.
  4. `make clean` va nettoyer toute l'arborescence.
  5. Lancez `make zImage` pour compiler votre noyau. Vous pouvez aller boire un bon café ou plusieurs suivant la machine ( avec un 486 DX2/66 et 16 Mo, la compilation exige un peu plus de 40 minutes). Cela dépend surtout de la quantité de mémoire vive disponible.
  6. si vous avez des modules, lancer leur compilation par : `make modules` puis les installer par `make modules_install`. Ils seront copiés dans le répertoire `/lib/modules/2.0.26`.
  7. Quand c'est terminé, aller dans `/usr/src/linux/arch/i386/boot`. Le nouveau noyau flambant neuf est le fichier `zImage`, qui est déjà compacté. Le copier dans la racine : `/` en lui attribuant un nouveau nom (par exemple "vmlinuz2"). Modifier la configuration de LILO mais ne rien retirer : copier/coller en tête du fichier une section existante puis la modifier afin qu'elle concerne le nouveau noyau. Puis relancer lilo pour que le nouveau noyau soit pris en compte et reboote.
- Remarque : si vous ne parvenez pas à modifier la configuration de lilo il faudra installer le nouveau fichier du noyau en lieu et place de l'ancien. Je vous conseille fortement de laisser l'ancien noyau dans un coin et surtout de pouvoir booter dessus, par exemple grâce à une disquette, afin de ne pas tout perdre.

Réamorcer et cela doit fonctionner.



# Le système GNU/Linux

---

## 8.4 - Modules Chargeables

### 8.4.1 - Structure du noyau

Le noyau d'un système Unix peut être représenté sous la forme d'un objet monolithique. Toutefois, un tel objet possède l'inconvénient d'être gros et statique. A chaque fois que l'on désire rajouter un nouveau périphérique, il est nécessaire de recompiler le noyau. De plus, si l'on n'utilise certains gestionnaires particuliers que rarement, on est obligé de l'avoir dans le noyau, ce qui a tendance à consommer de la mémoire.

### 8.4.2 - Qu'est-ce qu'un module chargeable ?

Les modules chargeables permettent de joindre l'utile à l'agréable en ayant un noyau le plus petit que possible, chargeant à la demande ce dont il a besoin, soit d'une manière manuelle par le super utilisateur de la machine, soit d'une manière automatique. De cette manière, le gain de ressources est non négligeable.

La première question que l'on peut se poser, c'est : " Pourquoi deux techniques de chargement ?"

La première technique est manuelle : il faut charger ou décharger les modules à la main. La deuxième est automatique, grâce à l'utilisation d'un démon spécialisé qui est l'esclave du noyau et qui charge et décharge les modules pour lui. En fait, la version 1.2 de Linux n'offrait que la possibilité d'un chargement manuel qui est limité au super-utilisateur de la machine et qui est assez lourd à manipuler. Au fil du développement de la version 2.0, un nouveau système implémenté par Bjorn Ekwall permit d'effectuer un chargement dynamique et automatique des modules.

### 8.4.3 - Compilation du noyau

Lors de la compilation du noyau, il est nécessaire de spécifier des options particulières pour activer l'utilisation des modules chargeables :

```
gandalf# make config
```

```
*
```

```
* Loadable module support
```

```
*
```

```
Enable loadable module support (CONFIG_MODULES) [Y/n/?]
```

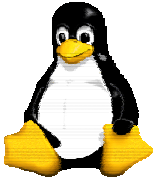
```
Set version information on all symbols for modules (CONFIG_MODVERSIONS) [N/y/?]
```

```
Kernel daemon support (e.g. autoload of modules) (CONFIG_KERNELD) [Y/n/?]
```

Voici le détail de ces trois options :

– CONFIG\_MODULES : active le mécanisme de modules chargeables. Sans cette option, ils ne fonctionneront pas sur votre système.

– CONFIG\_MODVERSIONS : si vous laissez cette option à N (par défaut), vous devrez recompiler les modules à chaque mise à jour du noyau de la machine. Cela signifie que des modules compilés pour un noyau 2.0.n ne fonctionneront pas avec un module compilé pour 2.0.n+1. Si vous activez cette option, les modules pourront être lancés... Cette option est particulièrement intéressante dans le cas de modules externes au noyau livrés sous forme binaire (ils sont accompagnés d'un patch à appliquer sur les sources du noyau). Il est fortement déconseillé de l'utiliser dans le cadre de modules intégrés au noyau (il est plus simple de les compiler et de les installer que d'aller regarder si les anciens modules sont compatibles avec les nouveaux).



# Le système GNU/Linux

Version 9  
30/09/2003

– CONFIG\_KERNELD : cette option active le chargement automatique des modules. Vous devez également activer les IPC Système V car le noyau et le démon communiquent via des files de messages.

Une fois configuré, il vous suffit de lancer la compilation ainsi que l'installation :

```
gandalf# make dep ; make clean
gandalf# make zImage
gandalf# make modules ; make modules_install
```

Une fois ces opérations effectuées, les modules se trouvent alors dans le répertoire /lib/modules/x.y.z où x.y.z correspond au numéro de version du noyau. Il ne nous reste plus qu'à voir le chargement.

Note importante : tous les outils de manipulation des modules se trouvent dans modules-2.0.0.tar.gz situé dans le répertoire v2.0. Ils sont installés par défaut avec les distributions standards, mais il est important d'utiliser la bonne version.

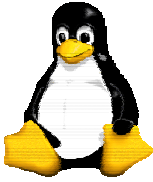
## 8.4.4 - Chargement : méthode manuelle

Le chargement manuel est basé sur trois commandes :

- insmod : insère un module dans le noyau ;
- rmmod : décharge un module, si plus aucun processus ne l'utilise ;
- lsmod : affiche la liste des modules chargés.

Leur utilisation oblige d'être en super-utilisateur. Voici un exemple d'utilisation :

```
gandalf# insmod nfs.o
gandalf# lsmod
Module:      #pages: Used by:
nfs          12      4
gandalf# mount -t nfs /truc /mnt
gandalf# lsmod
Module:      #pages: Used by:
nfs          12      5
gandalf# cd /mnt
...
gandalf# cd /
gandalf# umount /mnt
Module:      #pages: Used by:
nfs          12      4
gandalf# ps axu | grep nfs
root  5535 0.0 0.0  0  0 q2 SW 17:15  0:00 (nfsiod)
root  5536 0.0 0.0  0  0 q2 SW 17:15  0:00 (nfsiod)
root  5537 0.0 0.0  0  0 q2 SW 17:15  0:00 (nfsiod)
root  5538 0.0 0.0  0  0 q2 SW 17:15  0:00 (nfsiod)
root  5557 0.0 0.4 864 300 q2 S 17:16  0:00 grep nfs
gandalf# kill -9 5535 5536 5537 5538
gandalf# lsmod
gandalf# rmmod nfs.o
```



# Le système GNU/Linux

Version 9  
30/09/2003

Il est nécessaire de "tuer" les 4 démons nfsiod car ils sont lancés dès que NFS est activé. Comme vous pouvez le voir, ces opérations deviennent relativement pénibles. C'est pour cette raison que le système de chargement automatique a été créé.

## 8.4.5 - Chargement automatique : kerneld

### 8.4.5.1 - Introduction

Le système de chargement automatique de modules permet de réduire au minimum la taille de son noyau. Le principe de fonctionnement est particulièrement simple : un démon en mode utilisateur est à l'écoute des ordres du noyau (via une file de messages de type IPC Système V). Lorsque un processus essaye d'accéder à une ressource système (via un appel système open, etc...), le noyau envoie l'ordre de chargement du module à kerneld. Une fois le message reçu, kerneld exécute modprobe pour charger les modules nécessaires.

### 8.4.5.2 - Conseils pour la compilation du noyau

Par contre, lors de la compilation du noyau, il est nécessaire d'y mettre au moins le support pour permettre l'amorçage de la machine et le montage de la racine de votre système de fichiers (par exemple, support IDE + ext2fs). Vous pouvez y mettre tout le reste en module (carte son, systèmes de fichiers, carte SCSI, etc).

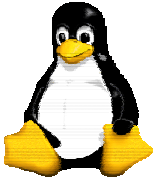
### 8.4.5.3 - Mise en place

Cette partie de modification n'est valable que si la machine n'était pas pourvue de kerneld. Les nouvelles distributions effectuent une installation tout à fait correcte.

Pour réaliser la mise en place du système de chargement de modules, il est nécessaire d'effectuer certaines modifications au niveau de votre configuration. En effet, il est nécessaire lors de l'amorçage de la machine de lancer le démon kerneld et de réaliser une espèce de liste des dépendances des modules : certains modules ne peuvent être lancés avant que d'autres ne le soient. Dans un premier temps, il faut créer le fichier /etc/rc.d/rc.modules dans lequel, il faut mettre :

```
# Modules

#
# Creation d'un lien logique pour le noyau courant
#
#
/bin/rm -f /lib/modules/current
ln -sf /lib/modules/`uname -r` /lib/modules/current
#
# Creation des dependances
if [ ! -r /lib/modules/current/modules.dep ]
then
    echo "Creating module dependencies"
    /sbin/depmod -a
fi
#
# Chargement des modules d'amorçage...
#
```



# Le système GNU/Linux

```
if [ -x /sbin/kerneld ]
then
    if find /lib/modules/boot -type f -o type l > /dev/null 2>&1
    then
        echo "Loading boot-time modules"
        /sbin/modprobe -a -t boot \*
    fi
else
    echo "Loading modules"
    /sbin/modprobe -a \*
fi
#
# Si vous possédez d'autres types de démons kerneld à lancer...
#
if [ -x /sbin/kdsound ]
then
    echo "Starting sound daemon"
    /sbin/kdsound &
fi
```

Cela permet de générer la dépendance de vos modules à chaque fois que vous amorcez votre machine. Ensuite, dans le fichier `/etc/rc.d/rc.S` (peut dépendre de la distribution...), il convient d'ajouter :

```
# Start update.
/sbin/update &

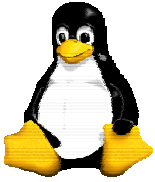
# *** A AJOUTER ***
# Lancement de kerneld le plus tôt possible, de telle manière
# que les modules de systèmes de fichiers puissent être chargés
if [ -x /sbin/kerneld ]
then
    echo "kerneld running"
    /sbin/kerneld
fi
...

# Un peu de ménage
cat /dev/null > /var/adm/utmp

# Lancement du script des modules
if [ -f /etc/rc.d/rc.modules ]; then
    /etc/rc.d/rc.modules
fi
```

Une fois ces modifications effectuées et la machine réamorcée, tout doit être en place. Si `kerneld` permet de charger automatiquement les modules, il permet également de les décharger au bout d'un





# Le système GNU/Linux

Version 9  
30/09/2003

certain temps de non utilisation. Par défaut, si aucun processus n'accède au module pendant plus de 60 secondes, il est automatiquement déchargé. Il est possible de modifier cette valeur en rajoutant le paramètre `delay=Nb_Secondes` à `kerneld` où `Nb_Secondes` est le délai en nombre de secondes.

## 8.4.5.4 - Le fichier `/etc/conf.modules`

Il peut arriver qu'il soit nécessaire de configurer un dernier fichier : le fichier `/etc/conf.modules`. Ce fichier contient les chemins où se trouvent les modules devant être chargés et ensuite des alias pour les modules. Si vous n'avez pas ce fichier, vous pouvez le créer avec :

```
gandalf# /sbin/modprobe -c | grep -v '^path' >/etc/conf.modules
```

Il peut arriver que lors du premier amorçage vous obteniez ce message :

```
Cannot locate module for net-pf-3
```

```
Cannot locate module for net-pf-4
```

```
Cannot locate module for net-pf-5
```

Pas de panique ! Ce message n'est pas méchant et pour ne plus l'avoir, ajouter les lignes suivantes dans le fichier `/etc/conf.modules` :

```
alias net-pf-3 off
```

```
alias net-pf-4 off
```

```
alias net-pf-5 off
```

Il peut arriver que certains périphériques aient besoin de certains paramètres particulier. Consultez le document `Kernel HowTo`.

Au boot, des messages d'erreurs concernant les modules créés peuvent apparaître, par exemple :

```
Unresolved symbol in module /lib/module/2.0.30/ppa.o
```

```
insmod : <symbole> wrong verion or undefined
```

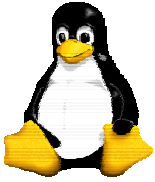
```
insmod : loading failed the modules symbol (from your 2.0.30à don't match your linux 2.0.30
```

C'est parceque les modules compilés présents dans `/lib/modules/2.0.*` sont ceux correspondant à l'ancien noyau et pas au nouveau que l'on vient de recompiler (si tant est qu'on ait modifié les modules à compiler lors de la compilation du nouveau noyau, par "`make modules ; make modules_install`")!

Après avoir déplacé `/usr/src/linux/System.map` vers `/boot/System.map`, la liste des fonctions définies pour les modules est donc adaptée au nouveau noyau mais les modules trouvables dans `/lib/modules/2.0.30/` sont les anciens correspondant au noyau précédent. Ils risquent d'être trop nombreux par rapport à la version précédente, si lors de la recompilation du nouveau noyau, on a demandé moins de modules que le nombre pre-installé).

On peut alors déplacer `/lib/modules/2.0.30` en `/lib/modules/2.0.30.old` (par exemple) et refaire un '`make modules; make modules_install`'. On aura un nouveau `/lib/modules/2.0.30/` avec juste les modules déclarés pour le nouveau noyau, donc dans le `System.map`, et lors du test des dépendances de modules au boot, le noyau ne trouvera plus de fichiers `.o` sous `modules/2.0.30/` avec des fonctions non déclarées.

On peut également déclarer des options pour un module particulier dans ce fichier sous la forme :



# Le système GNU/Linux

---

options le\_module irq=5 dma=0 dma2=1 mpu\_io=0x388 io=0x370

## 8.5 - SMP, autres architectures

SMP, abréviation de *Symmetric Multi-Processors*, désigne la capacité des noyaux Linux 2.0 et versions suivantes de fonctionner sur des machines à plusieurs processeurs. Sur architecture ix86, le support SMP de Linux suit les spécifications Intel MP (v1.1, v1.4) qui vont théoriquement jusqu'à 16 processeurs. En fait, le SMP n'est pas spécifique aux plates-formes Intel : il existe également sur les plates-formes Alpha, Sparc, PowerPC. La version 2.0 ne gère le SMP que pour les plates-formes de type Intel.

Pour compiler un noyau avec le support SMP, il faut *décommenter* la ligne `SMP = 1` dans le fichier Makefile. Il est en outre recommandé de compiler le support pour l'*Enhanced Real Time Clock* (`CONFIG_RTC`).

Lors du démarrage du noyau, Linux boot d'abord sur un processeur, puis active le(s) autre(s). Extraits des informations données par la commande `dmesg` :

```
Intel MultiProcessor Specification v1.1
```

```
Virtual Wire compatibility mode.
```

```
OEM ID: INTEL Product ID: Nightshade APIC at: 0xFEE00000
```

```
Processor #1 Pentium(tm) Pro APIC version 17
```

```
Processor #0 Pentium(tm) Pro APIC version 17
```

```
I/O APIC #2 Version 17 at 0xFEC00000.
```

```
Processors: 2
```

```
...
```

```
Calibrating delay loop.. ok - 396.49 BogoMIPS
```

```
...
```

```
Booting processor 0 stack 00002000: Calibrating delay loop.. ok - 396.49 BogoMIPS
```

```
Total of 2 processors activated (792.99 BogoMIPS).
```

Ce qui précède suffit pour avoir un système multiprocesseur qui fonctionne. Ce qui suit va le rendre plus agréable à utiliser et plus efficace.

Le package `procps` contient entre autre les commandes `top` et `ps` qui peuvent être modifiées pour afficher le numéro du processeur sur lequel une commande tourne. La marche à suivre ainsi que les patches sont disponibles (en anglais) à :

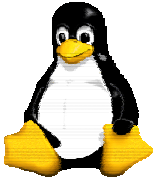
<http://www.cs.inf.ethz.ch/~rauch/procps.html>.

GNU make accepte l'option `-j` qui spécifie un nombre maximal de tâches pouvant être lancées simultanément. Il peut donc être intéressant d'utiliser cette option (`make -j 5` par exemple) sur un système multi-processeurs.

Dans la majorité des cas, un système SMP ne se distingue pas parce qu'une même tâche se déroule plus vite, mais bien parce que l'on peut faire tourner plusieurs grosse tâches simultanément (par exemple compiler un programme C et un fichier LaTeX en même temps) ou bien parce que le système répond instantanément malgré la présence d'une grosse tâche (p.ex. lire son mail tout en compilant le noyau).

Pour vraiment faire en sorte qu'un même programme utilise toute la puissance CPU de la machine, il faut investir dans le parallélisme. Lire à ce sujet l'excellentissime *Linux Parallel Processing HOWTO* (que l'on trouve aux endroits habituels).

Pour en savoir plus, vous pouvez consulter la Linux SMP FAQ de David Mentré, <http://www.irisa.fr/prive/mentre/smp-faq/>.



## 9 - Configuration du réseau.

### 9.1 - Installation ou changement de carte ethernet (3c509,3C590)

- Voir si la carte ethernet en question possède un driver sous Linux en lisant soit le Ethernet-Howto, soit en allant chercher dans `/lib/modules/2.0.30/net/` si il y a un module objet.o correspondant à la carte à installer.... exemple 3c59x.o

Les sources peuvent être trouvés dans `/usr/src/linux/drivers/net` ...on peut recompiler le module à partir de là aussi...

- Vérifier les modules réseau installés par le noyau en faisant :

```
$ /sbin/lsmmod
```

```
Module:      #pages:  Used by:
nfs          12         10 (autoclean)
3c59x        3           1 (autoclean)
```

- Si le module chargé n'est pas le bon (ou pour mettre à jour une dernière version de pilote):

booter en single-user ,

enlever le module existant : `rmmod <nom du module>` (ex. `rmmod 3c509`)

tester et réinstaller dynamiquement le module, sans rebooter, : `$ insmod 3c59x.o`

si tout va bien , placer le module nouveau au bon endroit dans `/lib/modules/2.0.30/net/`

modifier le fichier `/etc/conf.modules` :

```
$ more /etc/conf.modules
```

```
alias eth0 3c59x
```

```
alias cdrom isp16
```

```
$ vi /etc/conf.modules
```

et modifier l'alias pour l'interface eth0 par le bon module driver

- Enfin, la bonne page pour ce thème est là : <http://cesdis.gsfc.nasa.gov/linux/misc/>

### 9.2 - Support des interfaces réseau

Si vous avez spécifié des paramètres corrects lors de l'installation de la machine, le support des interfaces réseau doit déjà fonctionner.

Si ce n'est pas le cas, il n'est pas trop tard pour le faire. Tout d'abord le noyau doit supporter les interfaces réseau. Ensuite vérifier que les interfaces sont bien supportées avec la commande `ifconfig`, si vous avez les interfaces loopback et Ethernet c'est ok.

Si vous avez des problèmes à faire reconnaître votre carte Ethernet essayez d'utiliser la fonction `append` dans le fichier `/etc/lilo.conf`

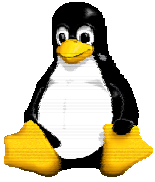
Les interfaces activées automatiquement au boot sont définies dans les fichiers du répertoire `/etc/sysconfig/network-scripts/`

Si vous ne voulez pas configurer les interfaces de façon automatique, vous pouvez bien sûr utiliser la commande `ifconfig` en faisant par exemple (cas d'une machine d'adresse IP 192.168.1.202) :

```
$ifconfig eth0 192.168.1.202
```

### 9.3 - Ajout des routes

Une fois vos interfaces configurées correctement, il faut maintenant ajouter les routes, c'est à dire les adresses IP des ordinateurs avec lesquels vous voulez vous connecter. La commande `netstat -r` affiche les routes déjà configurées, il faut au moins avoir la route vers loopback.



# Le système GNU/Linux

Pour les ajouter de nouvelles routes faire :

```
$route add 127.0.0.0  
$route add 192.168.1.202
```

De la même manière, ajouter les routes vers les ordinateurs vers lesquels vous voulez avoir accès en spécifiant leurs adresses IP. Cependant, vous n'êtes pas obligé de configurer une route pour chaque ordinateur. En ajoutant une route vers le sous-réseau (par exemple 192.168.1.0) tous les ordinateurs connectés au brin local qui ont une adresse commençant par 192.168.1 seront accessibles.

Pour vérifier que le réseau marche bien, il faut maintenant tester grâce à un petit ping en spécifiant l'adresse d'un ordinateur de votre réseau.

## 9.4 - Spécifier le nom de la machine, puis des hôtes de votre réseau

Si vous voulez modifier le nom de votre ordinateur, il faudra modifier :

- la ligne HOSTNAME du fichier `/etc/sysconfig/network`
- le contenu du fichier `/etc/HOSTNAME`
- la ligne correspondante de votre fichier `/etc/hosts`
- et tous les autres endroits où vous l'avez spécifié dans un fichier de configuration...

Spécifier les adresses IP de chaque machine n'est évidemment pas très humain, on peut donc associer le nom des machines aux adresses IP, cela se fait dans le fichier `/etc/hosts`

```
127.0.0.1    localhost  
192.168.1.200  zeppelin  
192.168.1.201  pantera  
192.168.1.202  fresne  
192.168.1.203  trixie  
192.168.1.209  flupke
```

Ensuite toutes les manipulations qui concernent l'adresse IP pourront se faire en spécifiant le nom de la machine.

## 9.5 - Regrouper des machines par sous-réseau

À partir d'un certain nombre de machine, le fichier `hosts` peut prendre une taille plutôt conséquente, c'est pourquoi il est possible de regrouper les machines par leurs adresses IP dans un sous-réseau auquel on donnera alors un nom.

Les sous-réseaux sont définis dans le fichier `/etc/networks` comme suit :

```
zep-net  192.168.1.0  
autre-net 198.163.8.0
```

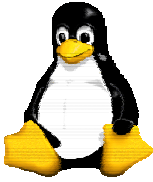
Le sous-réseau `zep-net` regroupe en fait toutes les machines commençant par 192.168.1, c'est-à-dire les machines définies plus haut dans le fichier `/etc/hosts`.

Cela simplifie ainsi beaucoup l'ajout des routes ou les autres manipulations que l'on fera alors pour le sous-réseau entier plutôt que machine par machine.

## 9.6 - Le fichier `/etc/resolv.conf`

Le fichier `/etc/resolv.conf` est le principal fichier de configuration pour le code de résolution de nom. Son format est très simple.

C'est un fichier texte avec un mot-clé par ligne. Il y a trois mots-clés typiquement utilisés, qui sont :



# Le système GNU/Linux

Version 9  
30/09/2003

domain

ce mot-clé indique le nom de domaine local.

search

ce mot-clé spécifie une liste d'autres noms de domaine pour rechercher un nom d'hôte.

nameserver

ce mot-clé, qui peut être utilisé plusieurs fois, spécifie l'adresse IP d'un serveur de nom de domaine pour la résolution de noms.

Un exemple de `/etc/resolv.conf` pourrait ressembler à ceci :

```
domain maths.wu.edu.au
search maths.wu.edu.au wu.edu.au
nameserver 192.168.10.1
nameserver 192.168.12.1
```

Cet exemple spécifie que le nom de domaine par défaut à ajouter aux noms non qualifiés (c'est-à-dire sans domaine) est `maths.wu.edu.au`, et que si l'hôte n'est pas trouvé dans ce domaine on peut aussi essayer le domaine `wu.edu.au` directement. Deux entrées de serveurs de noms sont fournies, chacune d'elles pouvant être appelée par le solveur de noms.

## 10 - De Windows à Linux

Il existe un bon nombre d'outils permettant l'interopérabilité entre le monde Windows et Linux, certains de ces programmes sont encore au stade de développement et ne peuvent être utilisés en production alors que d'autres sont parfaitement utilisables et opérationnels.

### 10.1 - Montage de partitions dos/windows

Nous avons vu avec la commande `mount` qu'il était possible de monter des partitions de types tout à fait variés. En effet, la commande `mount` accepte en paramètre l'option `-t` qui permet de préciser le type de partition. En ce qui nous concerne ici, il faut utiliser le type `msdos` pour des partitions dos (il ne doit plus y en avoir encore beaucoup à notre époque) et `fat` ou `vfat` pour des partitions Windows formatés en fat (comme Win95, 98 et NT si le formatage a bien été précisé comme cela). Le support de NTFS (Windows NT) est encore au stade expérimental.

### 10.2 - mtools

Plutôt que de monter vos partitions MS-DOS, je vous conseille d'utiliser les `mtools`. Il s'agit de l'équivalent des commandes MS-DOS, mais avec un `m` avant !

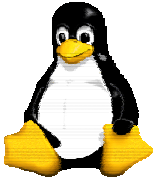
Exemple : `mcopy toto.c a: .` Si vous souhaitez copier un fichier texte pour qu'il soit exploitable sous MS-DOS, ou l'inverse, n'oubliez pas d'ajouter l'option `-t` : cela convertit le caractère de fin de ligne qui est différent entre les deux systèmes.

En principe, ils sont déjà à peu près configurés. Au cas où, vous devez modifier le fichier `/etc/mtools.conf`.

```
drive a: file="/dev/fd0" exclusive
drive b: file="/dev/fd1" exclusive
```

```
# 1er disque Dur
drive c: file="/dev/hda1"
```

```
# 2nd disque Dur
drive d: file="/dev/sda1"
```



# Le système GNU/Linux

mtools\_lower\_case=1

Attention : le fichier a été modifié sur les versions les plus récentes. C'est ici le dernier format qui est présenté car beaucoup plus clair et simple à utiliser. De plus, cette nouvelle version permet d'utiliser les noms longs (vfat de MS-Windows 95).

## 10.3 - Executer des applications Dos/Windows

### 10.3.1 - Dosemu et xdos

Dosemu est un programme qui crée une machine virtuelle x86 permettant de faire fonctionner la quasi totalité des programmes Dos, son fichier de configuration est en général dosemu.conf dans lequel on peut définir des lecteurs, configurer les différents ports, ... Pour executer dosemu dans une fenetre X il suffit d'executer xdos. Pour plus d'information sur dosemu on pourra se referer à la homepage : <http://www.dosemu.org/>.

### 10.3.2 - Wine

WINE signifie au choix WINdows Emulator ou Wine Is Not an Emulator. Wine est composé de plusieurs éléments l'application Wine et les librairies. En effet, ces dernières permettent de recompiler des applications windows pour être executable directement sous Linux. Wine permet quand a lui d'executer des applications win16 et win32 dans une fenetre X. Malheureusement, les spécifications de Microsoft Windows n'étant pas diffusées, le travail est plus difficile et beaucoup plus long. C'est pourquoi Wine est encore au stade de développement mais permet tout de même l'execution d'un grand nombre d'applications Windows. On trouvera plus d'informations à ce sujet sur la page : <http://www.winehq.org>.

## 10.4 - Samba et les réseaux MS-Windows

Samba est une collection d'outils disponibles de façon libre sur les systèmes Unix qui permettent d'accéder à un réseau MS-Windows. Samba permet ainsi à une machine Linux de devenir serveur de fichier pour tout un réseau sous MS-Windows, ou encore d'accéder aux différentes ressources qu'un serveur MS-Windows aura partagé.

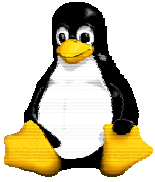
### 10.4.1 - Configuration de Samba sur la machine Linux

Si vous avez installé correctement samba, les 2 démons smbd et nmbd doivent être lancés automatiquement au boot de votre ordinateur. Si ça n'est pas le cas installez les packages.

Les fichiers de configurations par défaut de samba se sont grandement améliorés récemment, et permettent d'accéder à des répertoires homes assez facilement. Du moins, ils sont suffisamment commentés pour permettre de servir de modèle à des modifications personnelles.

Exemple de fichier /etc/smb.conf :

```
[global]
workgroup = mongroup
comment = Mon serveur samba
volume = RedHat
netbios name = Samba1
printing = bsd
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

```
printcap name = /etc/printcap
load printers = yes

log file = /var/log/samba-log.%m
max log size = 50
lock directory = /var/lock/samba
locking = yes
strict locking = no
share modes = yes

security = SHARE
null passwords = yes
socket options = TCP_NODELAY

; Permet au serveur Samba de devenir serveur du domaine
os level = 33
domain master = yes

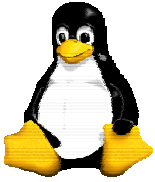
; Facilite la gestion des noms longs
preserve case = yes
short preserve case = yes

; Chaque client accède à son répertoire
[homes]
comment = Home Directories
browseable = yes
read only = no
create mode = 0750

; Service d'impression pour partager les imprimantes
[printers]
comment = All Printers
path = /var/spool/samba
browseable = no
printable = yes
public = no
writable = no
create mode = 0700

; Répertoire public /home/samba, accessible en lecture/écriture pour
; tout le monde
[public]
comment = Répertoire Public
path = /home/samba
public = yes
writable = yes
browsable = yes
```





# Le système GNU/Linux

---

printable = no

Une fois le fichier de configuration modifié, il faut relancer Samba avec :

```
/etc/rc.d/init.d/smb stop
```

```
/etc/rc.d/init.d/smb start
```

Enfin, si vous modifiez le fichier `smb.conf`, utilisez l'utilitaire `testparm` qui permet de vérifier la cohérence des informations de votre fichier. Il permet également d'afficher la totalité de la configuration de votre serveur. Ce qui permet de vérifier la validité de vos modifications.

## 10.4.2 - Configuration de la machine MS-Windows

Pour le moment, Samba ne permet d'accéder à la machine MS-Windows que si elle utilise une pile TCP/IP. En attendant la version qui devrait offrir un service direct avec NetBIOS. Il est donc nécessaire de configurer sous MS-Windows le support TCP/IP pour votre carte Ethernet, et d'affecter une adresse IP à votre machine. Ceci étant fait après quelques reboots, devrait apparaître dans votre Voisinage Réseau votre nouveau serveur Samba sous Linux.

Vous allez normalement alors pouvoir accéder au compte de l'utilisateur avec lequel vous êtes connectés sous MS-Windows, ainsi qu'au répertoire public `/home/samba`

## 10.4.3 - Les outils `smbclient` et `smbmount`

`Smbclient` et `smbmount` permettent d'accéder aux différentes ressources offertes par le serveur MS-Windows.

**`smbclient`** : le ftp pour MS-Windows. Il permet de tester, répertorier et accéder aux ressources offertes par le serveur MS-Windows.

Tout d'abord, il est possible de vérifier si le serveur MS-Windows est bien disponible, et de lister les différentes ressources qu'il partage grâce à la commande :

```
$smbclient -L nom_serveur_windows
```

`nom_serveur_windows` représentant le nom que vous avez donné à votre machine sous MS-Windows.

Ensuite, il est possible d'accéder au différents fichiers de la ressource partagée en s'y connectant avec `smbclient`, puis de la parcourir et de récupérer les fichiers exactement comme avec un ftp :

```
$smbclient \\\nom_serveur_windows\nom_ressource
```

Avec la majorité des Shell, il est nécessaire de doubler les backslash (`\`) pour qu'ils soient pris en compte. `nom_ressource` représente le nom de la ressource partagé que l'on peut obtenir avec un `smbclient -L`

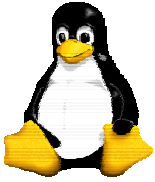
Après cela, il devrait vous être demandé un mot de passe (qui correspond à celui spécifié pour la ressource sous MS-Windows). Une fois le mot de passe entré, il n'y a plus qu'à parcourir les fichiers exactement comme avec ftp.

**`Smbmount`** : monter sous Linux des répertoires MS-Windows. Il permet de monter (comme par NFS) des répertoires MS-Windows sur l'arborescence Linux et de les manipuler le plus naturellement du monde.

```
$smbmount //nom_serveur_windows/nom_ressource /mnt -f 777
```

`smbmount` utilise les `'/'` à la différence de `smbclient`.

Il est nécessaire de rajouter une entrée dans le fichier `/etc/hosts` pour `nom_serveur_windows` avec son adresse IP, pour que `smbmount` puisse marcher correctement.



# Le système GNU/Linux

Version 9  
30/09/2003

Enfin, il est nécessaire de passer quelques paramètres pour assurer la compatibilité des droits de MS-Windows avec les droits Unix. La clause `-f 777` met tous les fichiers en lecture-écriture pour tout le monde.

On peut bien sûr la combiner avec l'option `-u uid` et `-g gid` qui permettrons de définir les droits précis des fichiers de la ressource pour pallier les carences de MS-Windows dans ce domaine.

## 11 - Les services reseaux

### 11.1 - NFS

NFS (abréviation de Network File System) est un produit développé par SUN (les premières versions datent de 1985) et il est disponible en standard sur de nombreux systèmes UNIX. Il existe des versions NFS sur d'autres systèmes. C'est un service TCP/IP basé sur le concept de RPC (Remote Procedure Call).

Le service NFS est un service serveur. On dit que le serveur "exporte" des disques ou une arborescence pour les clients NFS. Un client NFS, à besoin d'un produit client pour accéder aux serveurs. On dit qu'un client "monte" des disques exportés par une serveur. Pour le client, le fait de "monter" une arborescence exportée par une machine B, donne l'illusion aux utilisateurs de A que l'arborescence est un disque local. Un client peut également être serveur si le système d'exploitation le permet.

#### 11.1.1 - Coté client

Sur la machine cliente (pc linux) il faut dire quels sont les partitions à monter

De façon automatique :

```
vi /etc/fstab (et faire les modifications nécessaires)
mount -av
```

Ou manuelle :

```
mount -t nfs nom_du_serv:/rep_exporte /rep_local
```

Il faut tout de même que le noyau supporte le système de fichier nfs, vérifier en faisant un `cat /proc/filesystems`, une ligne `nodev nfs` doit apparaître, si ça n'est pas le cas, vous devez recompiler votre noyau.

#### 11.1.2 - Coté serveur

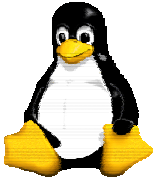
Sur les machines hôtes (celles qui ont le disque à monter) , il faut autoriser les machines extérieures à monter une partition. Pour cela, il faut éditer le fichier `/etc/exports` et rajouter le nom de la machine Linux autorisée à faire un montage nfs puis lancer `exportfs -va` pour mettre à jour.

Exemple :

```
/users  fresne(rw) pantera(rw) flupke(rw)
/       fresne(ro)
/usr/local
```

Évidement, il est souhaitable pour des raisons de sécurité de ne pas permettre à tout le monde de monter des fichiers en lecture-écriture(rw).

Vous pouvez maintenant lancer sur le serveur les démons NFS qui sont `rpc.mountd` et `rpc.nfsd`.



# Le système GNU/Linux

---

## 11.2 - NIS

Le service NYS permet de distribuer sur un certain nombre de machines des cartes. Ces cartes sont en fait les fichiers de configuration correspondant aux comptes utilisateurs, aux adresses des différentes machines...

Pratiquement, NYS permet en fait d'utiliser son compte utilisateur avec le même mot de passe et les mêmes paramètres sur n'importe quelle machine appartenant au réseau NYS. Pour cela, on exportera les comptes utilisateur par NFS, et on utilisera NYS pour partager les données et les paramètres sur tous les clients du réseau.

Une fois installé, tous les fichiers de configuration sont centralisés sur le serveur. Cela simplifie grandement les mises à jour, la maintenance, et les ordinateurs clients ont ainsi en local des fichiers minimum.

Les cartes sont créées à partir des fichiers du serveur :

```
/etc/passwd -> carte des mots de passe  
/etc/group -> groupe des utilisateurs  
/etc/hosts -> les machines du réseau  
/etc/networks -> liste des sous-réseaux
```

### 11.2.1 - Coté client

#### 11.2.1.1 - Avec les distributions récentes

Il y a encore quelque temps, il était nécessaire de lancer un démon sur le client NIS. Ce client s'appelait ypbind, mais il est maintenant devenu inutile avec la distribution RedHat. En effet les fonctions de NIS ont été intégrées directement dans les bibliothèques, qui ont maintenant le nom de NYS.

La documentation n'est malheureusement pas très claire à ce sujet, et il est parfois difficile de trouver la bonne façon de configurer son client NYS car les différents docs se mélangeaient allègrement les pieds entre les 2 implémentations.

Ce sont les fonctions NYS qui sont maintenant utilisés et dont nous allons parler.

Il faut tout d'abord configurer comme pour le serveur le nom de domaine. Vous pouvez mettre dans le fichier `/etc/sysconfig/network` la ligne :

```
NIS_DOMAIN=mon_domaine
```

mais comme le script `ypserv` n'est évidemment pas lancé, il faudra définir `domainname` dans un des scripts de lancement. Il faut donc mettre dans `/etc/rc.d/rc.sysinit` à la suite du `hostname` la ligne `domainname ${NIS_DOMAIN}`

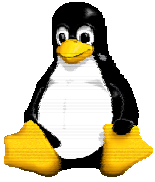
Il faut ensuite créer le fichier `/etc/yp.conf`

```
domainname mon_domaine_nis  
ypserver mon_serveur_yp  
ypsever mon_deuxieme_serveur
```

Il permet d'indiquer le nom du serveur yp que l'on utilisera. Il est tout à fait possible de spécifier plusieurs serveurs avec la ligne `ypserver`. Celui qui sera utilisé est celui qui répondra en premier. Cela sert essentiellement pour des machines qui sont souvent déplacées entre des réseaux différents.

Éditer également le fichier `/etc/nsswitch.conf`

```
passwd: files nis  
group: files nis  
hosts: files nis dns  
networks: files nis
```



# Le système GNU/Linux

Version 9  
30/09/2003

Ce fichier permet de déterminer par où commencera la recherche d'une information dans les différentes cartes.

À partir de là, les différentes fonctions de NYS doivent fonctionner. Pour le tester, il suffit d'essayer un :

```
$ypcat passwd
```

## 11.2.1.2 - Avec ypbind

Les étapes sont sensiblement les mêmes :

1. Dire à quel domaine NIS le PC client veut accéder par la commande `domainname`, si la commande ne retourne rien, on peut et doit assigner le nom du domaine NIS par la commande :

```
domainname <nomdudomaine_NIS>
```

2. Lancer `/usr/sbin/rpc.portmap`

3. Créer le répertoire `/var/yp` (il est en général déjà créé avec la distribution redhat)

4. Lancer le daemon `ypbind` (après avoir lu la man page de `ypbind`) et après avoir paramétré le fichier `/etc/yp.conf` de la manière suivante :

```
$ cat /etc/yp.conf
ypserver <adresse_IP_du_serveur_NIS_pour_le_domaine_considere>
(dans ce cas on force l'adresse du serveur)
```

ou

```
$ cat /etc/yp.conf
domain <nom_du_domaine_NIS> server <adresse_IP_du_serveur_NIS >
```

ou

```
$ cat /etc/yp.conf
domain <nom_du_domaine_NIS> broadcast
```

5. On peut vérifier le handshake entre `ypbind` (coté client) et `ypserv` (coté serveur) en lançant `ypbind` en mode debug

```
ypbind -debug
```

6. vérifier le fonctionnement de `ypbind` par

```
rpcinfo -p localhost
rpcinfo -u localhost ypbind
```

7. Modification du fichier `/etc/passwd`

Rajouter la ligne `+ :* :0 :0 : : :` à la fin du fichier `passwd` ... pour indiquer que les noms de login doivent être obtenus depuis la map "passwd" du serveur NIS. Pour vérifier l'accès et la récupération de la map `passwd` du serveur NIS, lancer

```
ypcat passwd
```

## 11.2.2 - Coté serveur

Avant tout, il faut spécifier à quel domaine NIS appartient la machine. Cela se fait par la commande `domainname`.

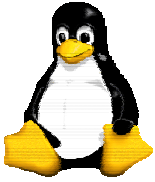
```
$domainname mon_domaine
```

Ce domaine est totalement différent du domaine DNS. Vous pouvez donc donner un nom de domaine différent ou identique à votre domaine DNS. Pour éviter les confusions, il est parfois préférable de distinguer ces 2 noms.

Pour ne pas avoir à le faire à chaque fois que vous redémarrez votre machine il suffit d'ajouter dans le fichier `/etc/sysconfig/network` la ligne :

```
NIS_DOMAIN=mon_domaine
```

Le nom de domaine sera initialisé automatiquement lors du lancement de `ypserv`.



# Le système GNU/Linux

Version 9  
30/09/2003

Aller ensuite dans le répertoire `/var/yp` et éditer le `Makefile`. Aller alors à la ligne `all:`, elle contient la liste des cartes possibles. Recopiez-là et commentez-là pour en garder une trace.

Laissez une ligne contenant simplement : `all: passwd hosts group`

Ce sont les trois cartes généralement suffisantes pour la majorité des utilisations, en tout cas elles suffisent largement pour tester. Il sera ensuite toujours temps de rajouter les autres cartes dont vous avez besoin, comme par exemple la carte de `/etc/networks`.

Ensuite il n'y a plus qu'à faire un `make`, les cartes vont alors être créés dans le répertoire `/var/yp/mon_domaine`.

Il faut également éditer le fichier `ypserv.conf`

```
sunos_kludge: no
tryresolve: no
dns: no
# Host      : Map      : Security :Passwd_mangle
#
192.168.1.  :passwd.byname  : port    : yes
192.168.1.  :passwd.byuid   : port    : yes
# Not everybody should see shadow password, not secure, since
# under MSDOG everybody is root and can access ports < 1024 !!! *
      : shadow.byname : port    : yes
```

Enfin, il faut autoriser les clients à se connecter, pour cela il faut éditer le fichier `hosts.allow` si ce n'est pas encore fait pour qu'il ressemble à quelque chose comme cela :

```
ALL: 192.168.1.      ou bien
ALL: fresne, pantera
```

Il ne reste plus qu'à lancer le démon `ypserv`.

Le serveur NYS à proprement parler est prêt. Pour que les utilisateurs puissent se connecter de façon transparente sur les clients, il faut maintenant exporter le répertoire de leurs comptes par NFS. Vous pouvez alors exporter les répertoires `/home/un_utilisateur` de chaque utilisateur par NFS et les monter sur le client, de préférence en lecture-écriture.

Toutefois, si vous avez un grand nombre d'utilisateurs avec NYS, il paraît judicieux de les regrouper dans un seul et même répertoire, `/nisusers` par exemple que vous n'aurez plus qu'à exporter. Dans ce cas, il faudra également modifier le champ `home_directory` du fichier `/etc/passwd`, et le script `adduser` pour qu'il soit compatible.

## 11.2.3 - Le démon `yppasswd`

Lorsque l'on utilise NYS et les mots de passe distribués, la commande `passwd` sur un client risque de ne pas avoir le comportement attendu puisque qu'elle va éditer le fichier local `/etc/passwd`.

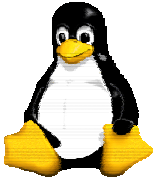
C'est donc le démon `yppasswd` du serveur qui doit se charger de cela. En fait lorsqu'un utilisateur voudra changer son mot de passe, il utilisera la commande `yppasswd`, qui ira modifier le fichier `/etc/passwd` du serveur NYS, et qui également mettra à jour les cartes, en faisant appel aux fonctions de notre bon démon.

Pour que l'utilisation de `yppasswd` soit transparente pour les utilisateurs, vous pouvez renommer le fichier `/usr/bin/passwd` en `lpasswd` par exemple, et ensuite faire un lien `passwd` vers `yppasswd` avec la commande :

```
ln -sf yppasswd passwd
```

Les utilisateurs pourront ainsi changer leur mot de passe sans se rendre compte qu'ils utilisent un compte NYS.

[ne/linux/sendmail.rtc.html](http://ne/linux/sendmail.rtc.html)



# Le système GNU/Linux

---

## 11.3 - APACHE / httpd : un serveur Web

Sa configuration est des plus simple. Tout d'abord, il faut installer apache sur votre machine si ça n'est pas encore fait. À vous de choisir votre solution préférée : package ou recompilation qui offre plus de liberté d'adaptation.

Apache peut fonctionner un peu comme le kernel de linux, c'est à dire qu'il supporte le chargement automatique des modules lorsqu'il en a besoin. Par exemple le module perl, ... Ce mécanisme porte le nom de DSO (Dynamic Shared Object). Par contre il est désactive par défaut.

Un module intéressant que vous pouvez installer est le module rewrite, votre ligne de compilation d'apache ressemblera donc a ceci :

```
./configure --enable-module=rewrite --enable-shared=max
```

Vous regarderez attentivement le fichier INSTALL pour comprendre exactement a quoi servent ces paramètres, puis :

```
make
```

```
make install
```

Pour démarrez apache vous devrez tapez la ligne suivante :

```
/usr/local/apache/bin/apachectl start
```

### 11.3.1 - Configuration d'Apache

Tous les fichiers de configuration de httpd se trouvent dans le répertoire `/etc/httpd/conf/`. Le fichier principal est le fichier `httpd.conf`. Il suffit de réactiver la ligne `ServerName` qui est normalement commentée, et de spécifier le nom auquel le serveur répondra, par exemple :

```
ServerName nom_de_ma_machine
```

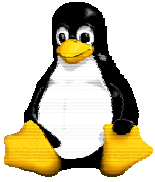
Bien évidemment, ce nom doit être un nom valide de la machine, c'est-à-dire un nom auquel elle répondra soit directement, soit après une résolution par NYS ou DNS. Pour le moment, mettre tout simplement le nom habituel de la machine.

Il ne reste plus qu'à tester le bon fonctionnement du serveur Web, en lançant un browser quelconque à l'adresse `http://nom_de_ma_machine/` ou `http://localhost/` si vous êtes directement sur le serveur. Cela devrait afficher la page de présentation de apache. Celle-ci est installée dans le répertoire `/home/httpd/` où vous pourrez bien entendu placer vos pages Web.

Les utilisateurs peuvent maintenant insérer leurs pages Web dans un répertoire `public_html` sur leur compte qui sera accessible par `http://nom_de_ma_machine/~user_name/`. Le nom de ce répertoire ainsi que les pages chargées par défaut (style `index.html`) sont paramétrés et modifiables dans le fichier `srm.conf`.

Voici un exemple de fichier `httpd.conf` :

```
##  
## httpd.conf -- Apache HTTP server configuration file  
##  
  
### Section 1: Global Environment  
# ServerType : inetd ou standalone  
ServerType standalone
```



# Le système GNU/Linux

```
# ServerRoot: Le répertoire d'apache
ServerRoot "/usr/local/apache"

# PidFile: Le fichier dans lequel apache doit écrire son pid
PidFile /usr/local/apache/logs/httpd.pid

# ScoreBoardFile: Fichier utilisé pour stocker les informations du serveur.
ScoreBoardFile /usr/local/apache/logs/httpd.scoreboard

# Timeout: nombre de secondes du délai imparti entre la réception et l'envoi.
Timeout 300

# KeepAlive: autorise ou non plus d'un requête par connexion
KeepAlive On

# MaxKeepAliveRequests: Nombre de requêtes maximums par connexion
MaxKeepAliveRequests 100

# KeepAliveTimeout: Nombre de seconde max d'attente entre deux requêtes
KeepAliveTimeout 15

# Nombre de daemons apaches lancés simultanément en attente de connexion
# apache essayera toujours d'avoir un nombre de daemons situés entre
# le min et le max
MinSpareServers 5
MaxSpareServers 10

# Nombre de daemons a lancer au démarrage
StartServers 5

# Nombre max de daemons simultanés
MaxClients 150

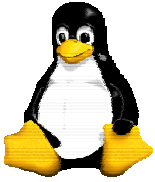
# Nombre de requêtes traitées par chaque fils avant de mourir 0 illimité
MaxRequestsPerChild 0

# Listen: Adresse et port d'écoute supplémentaire du serveur apache
#Listen 3000
#Listen 12.34.56.78:80

# BindAddress: Pour écouter sur plusieurs adresses IPs
#BindAddress *

# Dynamic Shared Object (DSO) Support
LoadModule env_module      libexec/mod_env.so
LoadModule config_log_module libexec/mod_log_config.so
```





# Le système GNU/Linux

Version 9  
30/09/2003

---

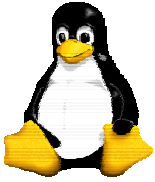
```
LoadModule mime_module      libexec/mod_mime.so
LoadModule negotiation_module libexec/mod_negotiation.so
LoadModule status_module    libexec/mod_status.so
LoadModule includes_module  libexec/mod_include.so
LoadModule autoindex_module libexec/mod_autoindex.so
LoadModule dir_module       libexec/mod_dir.so
LoadModule cgi_module       libexec/mod_cgi.so
LoadModule asis_module      libexec/mod_asis.so
LoadModule imap_module      libexec/mod_imap.so
LoadModule action_module    libexec/mod_actions.so
LoadModule userdir_module   libexec/mod_userdir.so
LoadModule alias_module     libexec/mod_alias.so
LoadModule rewrite_module   libexec/mod_rewrite.so
LoadModule access_module    libexec/mod_access.so
LoadModule auth_module      libexec/mod_auth.so
LoadModule setenvif_module  libexec/mod_setenvif.so
LoadModule php4_module      libexec/libphp4.so
```

```
# Reconstruction of the complete module list from all available modules
# (static and shared ones) to achieve correct module execution order.
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE UPDATE THIS,
# TOO]
```

```
ClearModuleList
AddModule mod_env.c
AddModule mod_log_config.c
AddModule mod_mime.c
AddModule mod_negotiation.c
AddModule mod_status.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_imap.c
AddModule mod_actions.c
AddModule mod_userdir.c
AddModule mod_alias.c
AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
AddModule mod_so.c
AddModule mod_setenvif.c
AddModule mod_php4.c
```

```
### Section 2: 'Main' server configuration
```

```
# Port: Port d'ecoute
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

Port 80

```
# Utilisateur et groupe sous lequel apache doit s'executer
User nobody
Group nobody

# ServerAdmin: adresse mail de l'administrateur
ServerAdmin root@tp4-pcXX.iut.fr

# Nom du serveur (si différent du nom de la machine)
#ServerName tp4-pc14.iut.fr

# DocumentRoot: Le répertoire contenant la racine du site
DocumentRoot "/usr/local/apache/htdocs"

# Configuration par défaut des permissions sur les répertoires
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

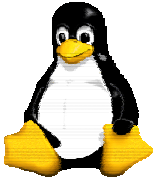
# Configuration spécifique de ce repertoire
<Directory "/usr/local/apache/htdocs">
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

# UserDir: Le nom du répertoire sous lequel un utilisateur met ses pages
<IfModule mod_userdir.c>
    UserDir public_html
</IfModule>

# DirectoryIndex: Nom du fichier lu par défaut dans chaque répertoire
<IfModule mod_dir.c>
    DirectoryIndex index.html index.htm
</IfModule>

# AccessFileName: Le nom du fichier de contrôle d'accès pouvant se
# trouver dans chaque répertoire
AccessFileName .htaccess

# Config de securite
<Files ~ "\.ht">
    Order allow,deny
    Deny from all
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

</Files>

# Configuration des types mimes

<IfModule mod\_mime.c>

TypesConfig /usr/local/apache/conf/mime.types

</IfModule>

DefaultType text/plain

<IfModule mod\_mime\_magic.c>

MIMEMagicFile /usr/local/apache/conf/magic

</IfModule>

#

# HostnameLookups: Log le nom des clients ou juste leurs IPs

HostnameLookups Off

#

# ErrorLog: L'emplacement du fichier de log des erreurs

ErrorLog /usr/local/apache/logs/error\_log

# LogLevel: Niveau de log

LogLevel warn

# Definitions des alias sur les repertoires

<IfModule mod\_alias.c>

Alias /icons/ "/usr/local/apache/icons/"

<Directory "/usr/local/apache/icons">

Options Indexes MultiViews

AllowOverride None

Order allow,deny

Allow from all

</Directory>

ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"

<Directory "/usr/local/apache/cgi-bin">

AllowOverride None

Options None

Order allow,deny

Allow from all

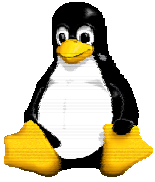
</Directory>

</IfModule>

### Section 3: Virtual Hosts

# VirtualHost : La machine peut avoir plusieurs noms différents

# et fournir une page html suivant le nom utilisé



# Le système GNU/Linux

Version 9  
30/09/2003

## 11.3.2 - Accès aux fichiers et gestion des droits d'accès

Avant tout, pour que le démon httpd puisse accéder aux fichiers, ceux-ci doivent avoir le droit de lecture pour tous. Pour l'appliquer à tous les fichiers d'un répertoire, nous lancerons ainsi la commande `chmod a+r *`. Par ailleurs, Apache fournit un système permettant de définir les droits d'accès des différents répertoires. Les droits d'accès par défaut sont définis dans le fichier `/etc/httpd/conf/access.conf`.

Il est toutefois possible de préciser un droit d'accès pour chaque répertoire, ce qui peut être utile si l'on installe des pages qui doivent être accessibles uniquement de façon interne à votre entreprise/association/réseau, et non à tous les internautes.

Cela se fait dans un fichier appelé par défaut `.htaccess` qui contient par exemple :

```
order deny,allow
deny from all
allow from .votre_domaine.fr
```

Ce qui permet d'autoriser uniquement les utilisateurs de `.votre_domaine.fr` à accéder aux fichiers du répertoire où est placé ce fichier.

## 11.3.3 - Configuration et utilisation des scripts CGI

L'un des principaux intérêts d'un serveur Web est de permettre d'exécuter des programmes sur le serveur. Ces programmes sont généralement dénommés des scripts cgi.

Apache permet bien évidemment d'exécuter de tels scripts. Par défaut, ces scripts sont autorisés à être exécuté uniquement si ils sont stockés dans le répertoire : `/home/httpd/cgi-bin`.

Mettons donc dans ce répertoire un petit script perl `test.pl` :

```
#!/usr/bin/perl
print "Content-type: text/html\n\n";
print "Tout baigne\n";
```

Et à l'affichage par votre browser préféré de `http://nom_de_ma_machine/cgi-bin/test.pl` vous lirez le message "Tout baigne".

Vous pouvez également autoriser le lancement de scripts CGI dans d'autres répertoires ajoutant l'option `AddHandler cgi-script.cgi` dans votre fichier `srn.conf`. Cependant, il est préférable de limiter au cas par cas cette possibilité pour des raisons de sécurité grâce au fichier `access.conf`.

La configuration fournie ici est suffisante pour la majorité des utilisations, mais les fichiers de configuration sont suffisamment bien commentés pour ne pas mériter de plus amples détails.

## 11.3.4 - PHP et Mysql

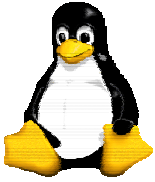
### Mysql

Pour installer Mysql le plus simple est de le récupérer sous forme de package ce qui ne nécessite aucune recompilation, vous devrez par contre faire attention à choisir un mot de passe pour l'administrateur de la base après l'installation :

```
/usr/bin/mysqladmin -u root -p password 'new-password'
/usr/bin/mysqladmin -u root -h tp4-pc14.iut.fr -p password 'new-password'
```

Un script est normalement automatiquement crée : `/etc/rc.d/init.d/mysql`

### PHP



# Le système GNU/Linux

Version 9  
30/09/2003

Vous pouvez recompiler php pour qu'il soit utilisé comme module pour apache, ce mécanisme est plus simple car les modules peuvent être rajoutés au fur et à mesure sans recompiler apache entièrement.

Comme indiqué dans le fichier INSTALL, la commande configure doit être exécutée avec les arguments suivants :

```
./configure --with-mysql --with-apxs=/usr/local/apache/bin/apxs
```

puis

```
make  
make install
```

## 12 - inetd, tcpd et xinetd

Tout système BSD ou System V dispose d'un processus à l'écoute des demandes de connexions TCP ou UDP via le réseau. Ce processus se nomme généralement inetd. C'est lui qui détecte une demande de connexion à la machine, qui détermine le protocole utilisé et le port demandé, afin de lancer l'action appropriée par lecture de sa configuration.

Par opposition au fonctionnement des démons, inetd permet de lancer un processus sur le système à la demande, et non en permanence. Ce mode favorise les opérations à faible fréquence ou occasionnelle (connexion finger, talk, etc.) au détriment des performances (ne jamais mettre un serveur HTTP en lancement via inetd, le temps nécessaire au démon inetd pour « aiguiller » sur le service HTTP deviendrait un goulot d'étranglement et nuirait gravement aux performances du serveur web.

### 12.1 - Le super démon inetd

Son fonctionnement est le suivant :

- Il est lancé au démarrage du système et va gérer un certain nombre de services.
- Lorsqu'une connexion est demandée pour un service dont il s'occupe, il lance le programme adéquat.
- Il se remet éventuellement à l'écoute juste après ou attend la fin de la communication.

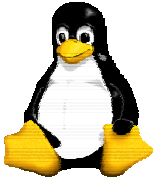
Le fichier /etc/inetd.conf est le fichier de configuration du serveur démon inetd. Il sert à dire à inetd ce qu'il doit faire lorsqu'il reçoit une demande de connexion pour un service particulier. Pour les services où vous acceptez une connexion vous devez dire à inetd quel démon serveur de réseau doit tourner, et comment. Son format est aussi très simple. C'est un fichier texte dont chaque ligne décrit un service que vous voulez fournir. Tout texte suivant un '#' est ignoré et considéré comme commentaire. Chaque ligne contient sept champs séparés par un nombre quelconque d'espaces (espace ou tabulation). Le format général est comme suit :

```
service  type_de_socket  protocole  drapeaux  utilisateur  chemin  arguments
```

*service* : Est le nom de service applicable à cette configuration, pris dans le fichier /etc/services.

*type\_de\_socket* : Ce champ décrit le type de socket que cette entrée considère comme pertinent, les valeurs permises sont : stream, dgram, raw, rdm, ou seqpacket. C'est un peu technique par nature, mais par expérience, presque tous les services basés sur tcp utilisent stream et presque tous les services basés sur udp utilisent dgram. Il n'y a que quelques types de serveurs démons spéciaux utilisant d'autres valeurs.

*protocole* : Le protocole considéré comme valide pour cette entrée. Il doit correspondre à l'entrée



# Le système GNU/Linux

Version 9  
30/09/2003

appropriée dans le fichier `/etc/services` et sera donc soit `tcp` soit `udp`. Les serveurs basés sur Sun RPC (Remote Procedure Call) utilisent `rpc/tcp` ou `rpc/udp`.

*drapeaux* : Il n'y a en fait que deux valeurs pour ce champ. Celles-ci disent à `inetd` si le programme serveur réseau libère le socket après démarrage, et donc si `inetd` peut prendre en compte une des prochaines demandes de connexion, ou bien si `inetd` doit attendre qu'un autre démon serveur tournant déjà prenne en charge la nouvelle demande de connexion. C'est encore compliqué, mais en pratique tous les serveurs `tcp` doivent avoir cette entrée positionnée sur `nowait` et la plupart des serveurs `udp` ont cette entrée positionnée sur `wait`. Attention il y a quelques exceptions notables.

*utilisateur* : Ce champ décrit quel compte utilisateur extrait de `/etc/passwd` sera considéré comme propriétaire du démon réseau lorsqu'il est lancé. C'est très utile lorsque vous voulez vous protéger contre les trous de sécurité. Vous pouvez mettre `nobody` comme utilisateur pour une entrée si bien que dans le cas où le réseau comporte une brèche, les dommages éventuels seront minimisés. Cependant habituellement ce champ est réglé sur `root`, car beaucoup de serveurs ont besoin des privilèges de `root` pour tourner correctement.

*chemin\_de\_serveur* : Ce champ est le chemin réel du programme à exécuter pour cette entrée.

*arguments* : Ce champ correspond au reste de la ligne et est optionnel. Il sert à indiquer les arguments de commande que vous voulez passer au programme serveur au lancement.

Exemple de fichier `/etc/inetd.conf` :

```
#echo          stream  tcp     nowait  root    internal
#echo          dgram  udp     wait    root    internal
daytime        stream  tcp     nowait  root    internal
daytime        dgram  udp     wait    root    internal
...
telnet         stream  tcp     nowait  root    /usr/sbin/in.telnetd
ftp            stream  tcp     nowait  root    /usr/sbin/in.ftpd
talk           dgram  udp     wait    root    /usr/sbin/in.talkd
ntalk          dgram  udp     wait    root    /usr/sbin/in.ntalkd
smtp           stream  tcp     nowait  root    /usr/sbin/in.smtpd
pop-3          stream  tcp     nowait  root    /usr/sbin/gnu-pop3d
```

Ce fichier de configuration est lu par `inetd` à son démarrage, et peut être relue sur la réception d'un signal `SIGHUP` :

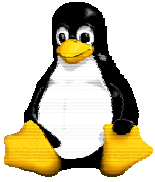
```
kill -HUP `cat /var/run/inetd.pid`
```

On peut noter que le fichier `inetd.conf` utilise des noms de service comme `telnet`, la définitions de ces services sont définis dans un fichier de configuration particulier : `/etc/services`.

Le fichier `/etc/services` est une simple base de données qui associe des noms compréhensibles par l'homme à des ports service compréhensibles par la machine. Son format est tout à fait simple. Le fichier est un fichier texte dont chaque ligne représente une entrée de la base de données. Chaque entrée comprend trois champs séparés par des caractères espace ou tabulation. Ces champs sont :

```
nom           port/protocole      alias           # commentaire
```

*nom* : Un simple mot qui représente le service décrit.



# Le système GNU/Linux

Version 9  
30/09/2003

*port/protocole* : ce champ est divisé en deux.

- *port* : Un nombre qui spécifie le numéro de port où le service désigné sera disponible. La plupart des services ont des numéros assignés. Ils sont décrits dans la RFC-1340.

- *protocole* : C'est soit tcp soit udp.

Il est important de noter qu'une entrée comme 18/tcp est très différente de 18/udp et qu'il n'y a pas de raisons techniques que le même service existe sur les deux. Normalement le bon sens prévaut et c'est vraiment pour un service particulier disponible à la fois sur tcp et udp que vous verrez une entrée pour les deux.

*alias* : Autre nom qui peut être utilisé pour désigner ce service.

Tout texte apparaissant après le caractère '#' est ignoré et traité comme commentaire.

Exemple : un extrait d'un fichier */etc/services* :

```

Daytime      13/tcp
Daytime      13/udp
ftp          21/tcp
telnet       23/tcp
smtp         25/tcp      mail
time         37/tcp      timserver
time         37/udp      timserver
pop-3        110/tcp     # POP version 3
pop-3        110/udp

```

## 12.2 - Le wrapper *tcpd*

Le wrapper *tcpd* est un programme conçu pour améliorer la sécurité de *inetd*, il est donc placé entre *inetd* et programme associé au service demandé. Ainsi lorsque *inetd* reçoit une demande de connexion pour un service dont il s'occupe, il n'exécute pas directement le programme associé, mais *tcpd* en lui passant en argument le nom du service. *Tcpd* va ainsi effectuer une vérification sur la validité de cette connexion.

Peu de modifications sont à effectuer dans le fichier de configuration *inetd.conf*, si l'on reprend l'exemple précédent, ce fichier sera le suivant :

```

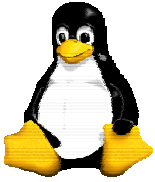
#echo        stream  tcp    nowait  root    internal
#echo        dgram  udp    wait    root    internal
daytime      stream  tcp    nowait  root    internal
daytime      dgram  udp    wait    root    internal
...
telnet       stream  tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/in.telnetd
ftp          stream  tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/in.ftpd
talk         dgram  udp    wait    root    /usr/sbin/tcpd  /usr/sbin/in.talkd
ntalk        dgram  udp    wait    root    /usr/sbin/tcpd  /usr/sbin/in.ntalkd
smtp         stream  tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/in.smtpd
pop-3        stream  tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/gnu-pop3d

```

Il reste ensuite à configurer *tcpd*, ceci est réalisable au travers de deux fichiers : */etc/hosts.allow* pour les connexions autorisées et */etc/hosts.deny* pour celles refusées

Lors de la prochaine demande de connexion au service *pop3* par exemple, *inetd* va donc exécuter la commande */usr/sbin/tcpd /usr/sbin/gnu-pop3d*. Le binaire *tcpd* (le wrapper) sait donc qu'à l'issue de la vérification qu'on lui demande d'opérer, si l'accès n'est pas refusé, alors la commande à





# Le système GNU/Linux

Version 9  
30/09/2003

lancer est in.telnetd.

Le segment TCP ou UDP possède dans son entête l'adresse IP source (de la machine émettrice). Le wrapper va donc parcourir les fichiers /etc/hosts.deny et /etc/hosts.allow pour y trouver quel comportement adopter face à cette demande.

Une bonne sécurité amène généralement les administrateurs à refuser toute connexion, sauf celle qui seront explicitement autorisées.

```
# /etc/hosts.deny
# service: adresse(s) source(s)
ALL :      ALL

# /etc/hosts.allow
# service: adresse(s) source(s)
gnu-pop3d : 10.0.0.1, 10.0.0.2
smtpd :     ALL
```

Historiquement, le fichier hosts.deny décrivait les interdictions, alors que le fichier hosts.allow décrivait les autorisations. Les différentes évolutions du wrapper font qu'il est possible aujourd'hui de se contenter d'un unique fichier au choix (généralement hosts.allow).

```
# /etc/hosts.allow
# service: adresse(s) source(s)           : accès
gnu-pop3d : 10.0.0.0/255.255.255.0       : allow
gnu-pop3d : ALL                           : deny
smtpd : ALL                                : allow
ALL : ALL                                  : deny
```

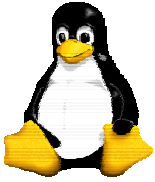
Ces deux fichiers (/etc/hosts.allow et /etc/hosts.deny) renseignent le wrapper sur les machines autorisées ou refusées. Si l'appelant n'a pas d'autorisation d'accès au service, la socket est alors fermée. Dans le cas contraire, le programme spécifié est lancé, et prend en charge le flux de données (TCP ou UDP) envoyé par l'appelant. Voici le comportement normal lorsque la connexion est acceptée :

```
% telnet 0 110
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
+OK POP3 Welcome to GNU POP3 Server Version 0.9.8
quit
```

Voici maintenant un exemple qui montre un accès au service pop-3 refusé par le wrapper. Ce dernier a donc fermé la socket avant d'avoir appelé le service pop-3 :

```
% telnet 0 110
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
Connection closed by foreign host.
```

Il est possible d'effectuer des actions particulières lors d'une tentative de connexion comme



# Le système GNU/Linux

Version 9  
30/09/2003

envoyer un message sur la console, ... pour plus de renseignements sur les options possibles des fichiers `hosts.allow` et `hosts.deny` on consultera la page de manuel `hosts_options`.

## 12.3 - *xinetd*

Ce daemon est destiné à remplacer `inetd`, il est de plus en plus présent sur les distributions. Il est compatible avec `inetd`. Le fichier de configuration s'appelle `/etc/xinetd.conf`, il est possible de se passer des fichiers `/etc/hosts.allow` et `deny`. Sa syntaxe est :

```
Service nom_du_service
{
  attribut operateur valeur
}
```

Voici un exemple pour le service `ftp` :

```
service ftp
{
  socket_type = stream
  protocol = tcp
  wait = no
  user = root
  server = /usr/sbin/in.ftpd
  server_args = -l -a
  only_from = 192.168.0.0/16
}
```

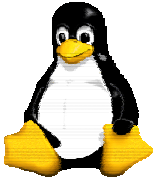
La configuration générale passe par le service « `defaults` » qui fixe les règles générales. Les autres services peuvent modifier ces valeurs localement.

Voici la liste des attributs utilisés par `xinetd` :

- `Flags` (voir `man`)
- `Log_type` : permet de choisir son mode de log
- `No_access` : liste les clients indésirables
- `Only_from` : clients autorisés
- `Port` : nnn
- `Protocol`
- `Server` : chemin du serveur
- `Socket_type` : `tcp`, `udp`, `raw`
- `Wait` : `yes/no` ó monatache ou forcké
- `Cps` : nombre de connexions entrantes
- `Max_load` : charge maximale
- `Bind` : permet de lier un serveur à une adresse ip : `bind = 160.1.3.180`
- `Redirect` : permet de rediriger un service `tcp` vers une autre machine (`proxy`)
- `Redirect = 160.1.3.180 23` (dans le service `telnet` redirige la connexion)

En général, seul le service `default` est défini dans le fichier `xinetd.conf`, ensuite un répertoire `/etc/xinetd.d/` est inclus et contient la configuration de tous les services, un fichier par service.

```
# Fichier xinetd.conf
defaults
{
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
instances = 60
log_type = SYSLOG authpriv
log_on_success = HOST PID
log_on_failure = HOST
cps = 25 30
}
includedir /etc/xinetd.d

[rahan@tux /]$ ls -al /etc/xinetd.d/
total 80
drwxr-xr-x  2 root  root  4096 avr  9 15:48 .
drwxr-xr-x 56 root  root  4096 jui 16 18:19 ..
-rw-r--r--  1 root  root   297 aoû 29 2001 chargen
-rw-r--r--  1 root  root   317 aoû 29 2001 chargen-udp
-rw-r--r--  1 root  root   297 aoû 29 2001 daytime
-rw-r--r--  1 root  root   317 aoû 29 2001 daytime-udp
-rw-r--r--  1 root  root   289 aoû 29 2001 echo
-rw-r--r--  1 root  root   308 aoû 29 2001 echo-udp
-rw-r--r--  1 root  root   318 jun 25 2001 finger
-rw-r--r--  1 root  root   259 jui 24 2001 ntalk
-rw-r----- 1 root  root   346 déc 18 2001 proftpd
-rw-r--r--  1 root  root   361 jui 24 2001 rexec
-rw-r--r--  1 root  root   378 jui 24 2001 rlogin
-rw-r--r--  1 root  root   431 jui 24 2001 rsh
-rw-r--r--  1 root  root   317 aoû 17 2001 rsync
-rw-r--r--  1 root  root   392 aoû 23 2001 sgi_fam
-rw-r--r--  1 root  root   247 jui 24 2001 talk
-rw-r--r--  1 root  root   304 avr  9 15:47 telnet
-rw-r--r--  1 root  root   321 aoû 29 2001 time
-rw-r--r--  1 root  root   317 aoû 29 2001 time-udp
[rahan@tux /]$
```

## 13 - Le DNS

Le service DNS est utilisé pour la « résolution de noms ». Cette opération consiste à fournir aux clients DNS qui en font la demande une association adresse IP, un nom symbolique, et vice-versa. C'est à dire qu'on pourra demander à un DNS : « Quelle est ou quelles sont les adresses IP de `www.yahoo.com` ». Le serveur DNS réalisera alors diverses opérations afin de vous répondre, souvent en contactant différentes entités sur l'Internet pour trouver la réponse.

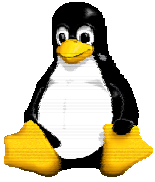
### 13.1 - Le protocole DNS

Lorsque les machines communiquent sur un réseau informatique, c'est toujours par l'utilisation d'une adresse (IP ou autre) source ou destination. Mais ces adresses bien que nécessaires, sont difficiles à mémoriser et ne permettent pas de souplesse dans les configurations des stations.

Pour quelqu'un de normalement constitué, il est difficile de se souvenir de `155.124.198.56`, alors que `www.victim.com` sera assez aisé à mémoriser. C'est le but du protocole DNS : fournir une association (adresse IP, nom FQDN) et inversement.

Lorsqu'un client DNS (votre station Windows qui possède une entrée DNS dans sa configuration IP par exemple, un programme quelconque d'un système Unix configuré pour utiliser le DNS, etc.) émet une requête à son serveur DNS, les opérations réalisées sont les suivantes :

t=1 Le client (`ST1.FOO.COM`) demande au serveur de résoudre un nom de station



# Le système GNU/Linux

Version 9  
30/09/2003

---

WWW.AV.COM

- t=2 Le serveur DNS détermine (configuration) si ce domaine est géré localement ou par un tiers
- t=3 Le serveur DNS ne connaissant la réponse, demande à un serveur (contenu dans la configuration) particulier qui sont les DNS de la zone AV.COM. Ces serveurs connaissant tous les serveurs de noms de chaque domaine sont appelés les ROOT SERVERS
- t=4 Le ROOT SERVER répond à notre serveur DNS que les serveurs de noms de la zone AV.COM sont situés sur les adresses IP X.X.X.X, Y.Y.Y.Y, etc...
- t=5 Notre serveur DNS contacte alors le premier serveur de noms renvoyé par le ROOT SERVER, et lui pose l'ultime question: « Quelle est l'adresse IP de WWW.AV.COM » ?
- t=6 Le serveur DNS de la zone AV.COM répond alors X.X.X.X
- t=7 Notre serveur DNS ayant obtenue la réponse escomptée répond alors à la demande initiale du client, en lui fournissant la réponse à sa question. La réponse est également placée dans un cache DNS (côté serveur) pour une utilisation ultérieure (réponse plus rapide)

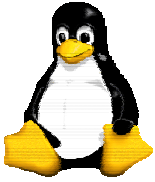
A la question simple de savoir quelle adresse IP est associée à quel nom (symbolique), on s'aperçoit que les échanges de données nécessaires sont multiples.

Mais les DNS sont également capables de se redistribuer les informations qu'ils contiennent, par le jeu de serveur primaire et secondaires. Pour chaque zone (par exemple domain.com), un serveur DNS est désigné comme primaire (et sera en priorité interrogé par les clients DNS du monde entier), et des serveurs DNS secondaires seront également présents, dans le cas d'une défaillance du primaire. Les serveurs secondaires sont généralement configurés pour prendre une copie de la zone en question sur un autre serveur DNS (par exemple, les secondaires vont chercher une copie de la zone toutes les 8 heures).

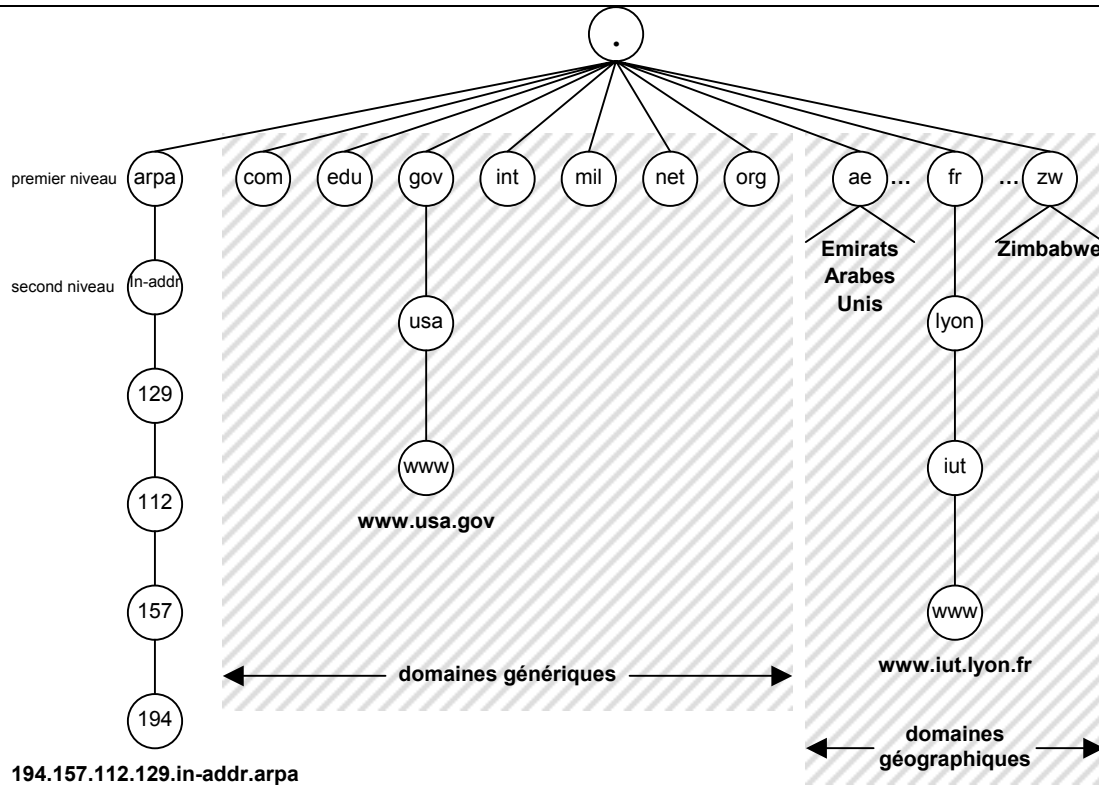
Les requêtes DNS des clients (demande de résolution) se font par dessus UDP, alors que les transferts de zone se font via TCP. Dans les deux cas, c'est le port 53 qui est utilisé. On trouvera donc dans les définitions normalisées des ports (/etc/services) les entrées 53/tcp et 53/udp associés tous les deux au service DNS.

## 13.2 - Hiérarchie des DNS

Les zones (généralement des noms de domaine, mais également des plages d'adresses IP) sont hiérarchisées pour permettre une recherche rapide d'une information.



# Le système GNU/Linux



Lorsqu'une requête est soumise par le resolver à un serveur DNS, ce dernier contacte un ROOT SERVER présent dans sa configuration (puis un autre si celui-ci ne répond pas jusqu'à épuisement de la liste) et demande quelle est l'autorité sur la zone demandée. Il contactera ensuite ce serveur pour poser la question à laquelle se dernier doit savoir répondre, et retournera cette information au resolver.

La zone « in-addr.arpa » est une zone quelque peu spéciale, puisqu'elle permet de résoudre les adresses IP en noms symboliques (requêtes inverses). Les zones sont déclarées dans le sens opposé à leur utilisation (reverse), on aura donc une déclaration DNS de la classe IP « 192.168.16.0/24 » qui ressemblera à « 16.168.192.in-addr.arpa. ».

Il est important de noter que le caractère « . » est le délimiteur de niveau hiérarchique. La racine la plus haute de la hiérarchie étant ce même caractère « . ». C'est la racine non nommée. Les extensions « .com. » sont donc une sous-hiérarchie de la zone « . ».

## 13.3 - Bases de données « whois »

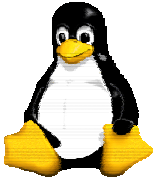
Les informations sur les noms de domaines sont déclarées par les prestataires de service lors de l'achat du domaine auprès des organismes de nommage (NIC France, InterNIC, etc.). Ces informations sont généralement les noms des serveurs de noms qui seront autoritaires sur la zone, leurs adresses IP, ainsi que le NIC HANDLE technique du domaine, le NIC HANDLE du responsable légal, et le NIC HANDLE du payeur.

L'accès à cette base de données est libre, et permet donc à chacun de savoir si un nom de domaine est libre, ou par qui il est utilisé. La syntaxe de la commande whois est la suivante :

```
whois <zone>@<server_whois>
```

On peut donc connaître les informations fournies lors de l'enregistrement de la zone « av.com » comme suit :

```
% whois av.com@whois.internic.net
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

```
[whois.internic.net]
```

```
Whois Server Version 1.1
```

```
Domain names in the .com, .net, and .org domains can now be
registered
with many different competing registrars. Go to
http://www.internic.net
for detailed information.
```

```
Domain Name: AV.COM
Registrar: NETWORK SOLUTIONS, INC.
Whois Server: whois.networksolutions.com
Referral URL: www.networksolutions.com
Name Server: NS2.ALTAVISTA.COM
Name Server: NS1.ALTAVISTA.COM
Name Server: NS3.ALTAVISTA.COM
Updated Date: 28-jan-2000
```

```
>>> Last update of whois database: Thu, 31 Aug 00 04:46:50 EDT <<<
```

```
The Registry database contains ONLY .COM, .NET, .ORG, .EDU domains
and Registrars.
```

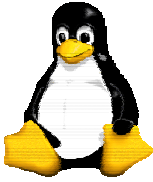
Historiquement, l'InterNIC était le seul prestataire habilité à gérer les zones de premier niveau, mais depuis le début de l'année 2000, il existe plusieurs prestataires mondiaux habilités à gérer les domaines. On peut donc se connecter à nouveau sur le serveur whois de l'organisme d'enregistrement (le registrar), ici NETWORK SOLUTIONS, INC.

```
% whois av.com@whois.networksolutions.com
[whois.networksolutions.com]
The Data in Network Solutions' WHOIS database is provided by Network
Solutions for information purposes, and to assist persons in obtaining
information about or related to a domain name registration record.
Network Solutions does not guarantee its accuracy. By submitting a
WHOIS query, you agree that you will use this Data only for lawful
purposes and that, under no circumstances will you use this Data to:
(1) allow, enable, or otherwise support the transmission of mass
unsolicited, commercial advertising or solicitations via e-mail
(spam); or (2) enable high volume, automated, electronic processes
that apply to Network Solutions (or its systems). Network Solutions
reserves the right to modify these terms at any time. By submitting
this query, you agree to abide by this policy.
```

```
Registrant:
Digital Equipment Corporation (AV22-DOM)
529 Bryant St.
Palo Alto, CA 94301
US
```

```
Domain Name: AV.COM
```

```
Administrative Contact, Technical Contact, Zone Contact, Billing
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

**Contact:**

AltaVista Search Operations (AO111-ORG) dns-technical@AV.COM  
AltaVista Company  
529 Bryant St.  
Palo Alto, CA 94301  
US  
650-295-2508  
Fax- 650-617-3200

Record last updated on 28-Jan-2000.  
Record expires on 21-Oct-2000.  
Record created on 21-Oct-1998.  
Database last updated on 31-Aug-2000 08:34:56 EDT.

Domain servers in listed order:

NS1.ALTAVISTA.COM	204.152.190.79
NS2.ALTAVISTA.COM	204.152.190.6
NS3.ALTAVISTA.COM	204.152.190.1

Dans la réponse à nos requêtes whois, on voit que les informations principales (notamment sur le serveur de l'InterNIC) sont les adresses des serveurs de noms de cette zone. C'est à eux qu'il faut s'adresser pour toute résolution dans la zone av.com. On contactera par exemple l'un de ces serveurs pour la résolution de www.av.com.

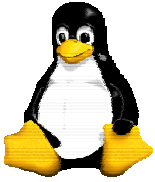
```
% whois altavista.fr@whois.nic.fr  
[winter.nic.fr]
```

```
Rights restricted by copyright.  
See http://www.ripe.net/db/dbcopyright.html  
Tous droits reserves par copyright.  
Voir http://www.nic.fr/info/whois/dbcopyright.html
```

```
domain:      altavista.fr  
descr:      ALTAVISTA  
descr:      54-56 Avenue Hoche  
descr:      75008 Paris  
admin-c:    HR4629-RIPE  
tech-c:     IN195-RIPE  
zone-c:     NFC1-RIPE  
nserver:    ukuk3d01.aveurope.co.uk  
nserver:    ukuk3d02.aveurope.co.uk  
mnt-by:     FR-NIC-MNT  
mnt-lower:  FR-NIC-MNT  
changed:    ripe-dbm-updates@nic.fr 20000403  
source:     RIPE
```

```
role:       ISDnet NOC  
address:    Paris  
e-mail:     noc@isdnet.net  
admin-c:    CB58-RIPE  
tech-c:     DP22-RIPE  
nic-hdl:    IN195-RIPE  
mnt-by:     ISDNET-NOC
```





# Le système GNU/Linux

Version 9  
30/09/2003

---

```
changed:      david@isdnet.net 19981110
source:       RIPE

role:         NIC France Contact
address:      AFNIC (NIC France)
address:      Domaine de Voluceau B.P. 105
address:      F-78153 Le Chesnay CEDEX, France
phone:        +33 1 39 63 56 16
fax-no:       +33 1 39 63 55 34
e-mail:       tech@nic.fr
trouble:      Information: http://www.nic.fr/
trouble:      Questions:  mailto:nic@nic.fr
trouble:      Spam:       mailto:abuse@nic.fr
trouble:      Test:       mailto:ping@nic.fr
admin-c:      AR41
tech-c:       AR41
tech-c:       PL12-RIPE
tech-c:       JP1110-RIPE
tech-c:       EM634-RIPE
tech-c:       MS1887-RIPE
tech-c:       VL229-RIPE
tech-c:       PR1249-RIPE
tech-c:       PV827-RIPE
tech-c:       GO661-RIPE
tech-c:       FT1632-RIPE
tech-c:       PB9432-RIPE
nic-hdl:      NFC1-RIPE
mnt-by:       FR-NIC-MNT
changed:      ripe-dbm-updates@nic.fr 20000803
source:       RIPE

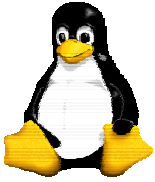
person:       Howard Rosen
address:      ALTAVISTA
address:      54-56 Avenue Hoche
address:      75008 Paris
phone:        +33 1 56 60 51 95
fax-no:       +33 1 56 60 55 07
nic-hdl:      HR4629-RIPE
changed:      ripe-dbm-updates@nic.fr 20000208
source:       RIPE
```

## 13.4 - Tester un DNS

Les serveurs DNS sont généralement packagés avec un outil d'interrogation interactif nommé nslookup. Un autre outil en ligne de commande nommé host est également disponible sur les distributions Unix récentes.

La commande nslookup peut être utilisée de la manière suivante :

```
% nslookup
Default Server:  ns.dns.net
Address:  192.168.16.1
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
> server a.root-servers.net.  
Default Server:  a.root-servers.net  
Address:  198.41.0.4
```

```
> set query=ns  
> altavista.com.  
Server:  a.root-servers.net  
Address:  198.41.0.4
```

Authoritative answers can be found from:

```
com      nameserver = A.GTLD-SERVERS.NET  
com      nameserver = G.GTLD-SERVERS.NET  
com      nameserver = H.GTLD-SERVERS.NET  
com      nameserver = C.GTLD-SERVERS.NET  
com      nameserver = I.GTLD-SERVERS.NET  
com      nameserver = B.GTLD-SERVERS.NET  
com      nameserver = D.GTLD-SERVERS.NET  
com      nameserver = L.GTLD-SERVERS.NET  
com      nameserver = F.GTLD-SERVERS.NET  
com      nameserver = J.GTLD-SERVERS.NET  
com      nameserver = K.GTLD-SERVERS.NET  
com      nameserver = E.GTLD-SERVERS.NET  
com      nameserver = M.GTLD-SERVERS.NET  
A.GTLD-SERVERS.NET      internet address = 192.5.6.30  
G.GTLD-SERVERS.NET      internet address = 192.42.93.30  
H.GTLD-SERVERS.NET      internet address = 192.54.112.30  
C.GTLD-SERVERS.NET      internet address = 192.26.92.30  
I.GTLD-SERVERS.NET      internet address = 192.36.144.133  
B.GTLD-SERVERS.NET      internet address = 192.33.14.30  
D.GTLD-SERVERS.NET      internet address = 192.31.80.30  
L.GTLD-SERVERS.NET      internet address = 192.41.162.30  
F.GTLD-SERVERS.NET      internet address = 192.35.51.30  
J.GTLD-SERVERS.NET      internet address = 210.132.100.101  
K.GTLD-SERVERS.NET      internet address = 213.177.194.5  
E.GTLD-SERVERS.NET      internet address = 192.12.94.30  
M.GTLD-SERVERS.NET      internet address = 202.153.114.101
```

```
> server A.GTLD-SERVERS.NET  
Default Server:  A.GTLD-SERVERS.NET  
Address:  192.5.6.30
```

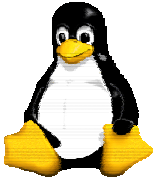
```
> altavista.com.  
Server:  A.GTLD-SERVERS.NET  
Address:  192.5.6.30
```

Non-authoritative answer:

```
altavista.com  nameserver = NS1.altavista.com  
altavista.com  nameserver = NS3.altavista.com  
altavista.com  nameserver = NS2.altavista.com
```

Authoritative answers can be found from:

```
NS1.altavista.com      internet address = 209.73.164.76  
NS3.altavista.com      internet address = 209.73.176.204
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

```
NS2.altavista.com      internet address = 209.73.164.7
```

```
server 209.73.164.76
```

```
Authoritative answers can be found from:
```

```
(root)  nameserver = A.ROOT-SERVERS.NET
(root)  nameserver = H.ROOT-SERVERS.NET
(root)  nameserver = C.ROOT-SERVERS.NET
(root)  nameserver = G.ROOT-SERVERS.NET
(root)  nameserver = F.ROOT-SERVERS.NET
(root)  nameserver = B.ROOT-SERVERS.NET
(root)  nameserver = J.ROOT-SERVERS.NET
(root)  nameserver = K.ROOT-SERVERS.NET
(root)  nameserver = L.ROOT-SERVERS.NET
(root)  nameserver = M.ROOT-SERVERS.NET
(root)  nameserver = I.ROOT-SERVERS.NET
(root)  nameserver = E.ROOT-SERVERS.NET
(root)  nameserver = D.ROOT-SERVERS.NET
```

```
A.ROOT-SERVERS.NET      internet address = 198.41.0.4
H.ROOT-SERVERS.NET      internet address = 128.63.2.53
C.ROOT-SERVERS.NET      internet address = 192.33.4.12
G.ROOT-SERVERS.NET      internet address = 192.112.36.4
F.ROOT-SERVERS.NET      internet address = 192.5.5.241
B.ROOT-SERVERS.NET      internet address = 128.9.0.107
J.ROOT-SERVERS.NET      internet address = 198.41.0.10
K.ROOT-SERVERS.NET      internet address = 193.0.14.129
L.ROOT-SERVERS.NET      internet address = 198.32.64.12
M.ROOT-SERVERS.NET      internet address = 202.12.27.33
I.ROOT-SERVERS.NET      internet address = 192.36.148.17
E.ROOT-SERVERS.NET      internet address = 192.203.230.10
D.ROOT-SERVERS.NET      internet address = 128.8.10.90
```

```
Default Server:  [209.73.164.76]
```

```
Address:  209.73.164.76
```

```
> www.altavista.com
```

```
Server:  [209.73.164.76]
```

```
Address:  209.73.164.76
```

```
www.altavista.com      canonical name = altavista.com
altavista.com          nameserver = ns1.altavista.com
altavista.com          nameserver = ns2.altavista.com
altavista.com          nameserver = ns3.altavista.com
ns1.altavista.com      internet address = 209.73.164.76
ns2.altavista.com      internet address = 209.73.164.7
ns3.altavista.com      internet address = 209.73.176.204
```

```
> set q=a
```

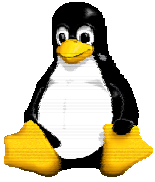
```
> www.altavista.com
```

```
Server:  [209.73.164.76]
```

```
Address:  209.73.164.76
```

```
Name:      altavista.com
```

```
Addresses:  209.73.180.9, 209.73.180.10, 209.73.164.90,
209.73.164.91, 209.73.164.92, 209.73.164.93, 209.73.164.94,
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

```
209.73.164.95, 209.73.164.96, 209.73.180.8  
Aliases: www.altavista.com
```

&gt;

La commande `host` peut également être utile pour rapidement vérifier le bon fonctionnement du `resolver` et du `DNS`.

```
% host www.microsoft.com  
www.microsoft.com is a nickname for www.microsoft.akadns.net  
www.microsoft.akadns.net has address 207.46.197.113  
www.microsoft.akadns.net has address 207.46.230.219  
www.microsoft.akadns.net has address 207.46.230.229  
www.microsoft.akadns.net has address 207.46.197.101  
% host -t ns altavista.com  
altavista.com name server ns2.alta-vista.net  
altavista.com name server ns3.alta-vista.net  
altavista.com name server ns1.alta-vista.net  
% host -t mx altavista.fr  
altavista.fr mail is handled (pri=10) by  
inbound.altavista.fr.criticalpath.net
```

## 13.5 - Les objets DNS

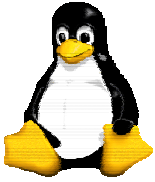
Chaque zone dispose donc (en théorie) de deux DNS (minimum requis pour l'achat d'un nom de domaine auprès des organismes de nommage). Les administrateurs de ces zones vont déclarer différents objets dans cette zone, comme par exemple le numéro de la version actuelle, l'adresse de courrier électronique de l'administrateur de cette zone, les serveurs de courriers à utiliser, etc...

<b>A</b>	Un enregistrement A définit une adresse IP
<b>CNAME</b>	CNAME permet d'associer un nom à un autre nom. C'est un alias
<b>HINFO</b>	Chaîne d'informations sur la station, contenant (par convention d'utilisation) le CPU de la station suivi du nom du système d'exploitation.
<b>MX</b>	Définit le ou les serveurs de courrier pour la zone. Lors d'un envoi d'email à un utilisateur de cette zone, le serveur SMTP émetteur contactera le serveur SMTP contenu ici, en se connectant sur le port standardisé 25/tcp
<b>NS</b>	Définit quels sont les serveurs de noms pour cette zone
<b>PTR</b>	Enregistrement de pointeur (PTR). Cet enregistrement permet de définir un nom de machine associé à une adresse IP.

## 13.6 - Contrôleurs primaire et secondaires

On a vu que chaque zone dispose d'un minimum de deux serveurs de noms (ceux-là même qui sont enregistrés dans les bases whois). Il existe donc une hiérarchie entre ces serveurs de noms, l'un étant primaire, l'autre secondaire. Ce schéma s'applique avec la règle suivante : « chaque zone dispose d'un serveur DNS primaire, et de serveurs DNS secondaires (entre 1 et 3) ». Chacun de ces serveurs DNS possèdera donc l'ensemble des informations sur la zone concernée.

Puisqu'il est laborieux pour un administrateur DNS (`hostmaster`) de modifier plusieurs fichiers de configuration de zone sur différents serveurs pour le simple ajout d'un enregistrement DNS, le protocole DNS prévoit une possibilité de transfert des configurations entre DNS autonome : les



# Le système GNU/Linux

Version 9  
30/09/2003

transferts de zone.

Un serveur DNS secondaire est configuré avec l'adresse du ou des serveurs auxquels il peut demander une copie de la zone en question. Ces connexions des serveurs secondaires se font sur le port 53/TCP du serveur DNS.

Si une configuration est modifiée sur le DNS primaire, son numéro de version (un numéro par zone) est donc incrémenté, et le serveur recharge la configuration de la zone modifiée, et notifie les DNS secondaires de cette zone (déclarés dans la zone) de la modification.

Si une zone est modifiée, sans que le numéro de version ne soit incrémentée, les données conservées en mémoire du serveur DNS seront obsolètes, mais considérées comme actuelles par le serveur DNS disposant (de son point de vue) de la dernière version de la zone.

## 13.7 - Syntaxe d'un fichier de zone DNS

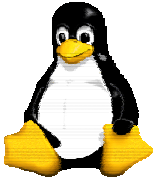
Une définition de zone doit contenir un minimum d'informations, parmi lesquelles :

- Le nom FQDN de la zone
- Le nom FQDN du serveur DNS considéré comme serveur primaire (SOA )
- L'adresse email de l'administrateur de la zone, le caractère '@' sera remplacé par un caractère '.'
- Le numéro de version de la zone
- L'intervalle de temps (en secondes) au bout duquel un serveur secondaire devra se connecter pour obtenir une nouvelle (éventuelle) version de cette zone
- Le délai d'attente avant de retenter un transfert de zone en cas d'échec précédent (serveur primaire non disponible ou non joignable)
- Le temps maximum où une copie de zone sera stockée par le secondaire sans mise à jour du serveur primaire
- Le temps maximum de présence (par défaut) des enregistrements de cette zone dans les caches de DNS extérieurs

```
internet.com. IN SOA mars.internet.com. hostmaster.mars.internet.com. (
1999070301          ; Numéro de série
10800              ; délai de rafraîchissement des secondaires
3600               ; délai avant une nouvelle tentative si échec
604800            ; délai maximum de validité des données
86400             ) ; TTL par défaut des enregistrements
```

Les enregistrements ajoutés à chaque entête de zone DNS sont les serveurs de noms de cette zone, les serveurs de courrier, et toutes les associations et alias entre les noms symboliques et les adresses IP.

```
internet.com. IN SOA mars.internet.com. hostmaster.mars.internet.com. (
1999070301          ; Numéro de série
10800              ; délai de rafraîchissement des secondaires
3600               ; délai avant une nouvelle tentative si échec
604800            ; délai maximum de validité des données
86400             ) ; TTL par défaut des enregistrements
; déclaration des serveurs de noms de cette zone
    IN  NS  lune.internet.com.
    IN  NS  remote.backup.com.
    IN  NS  another.network.net.
; déclaration des serveurs de courrier de cette zone
internet.com.  IN   MX   10   mx-01.internet.com.
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
internet.com.    IN    MX    10    mx-02.internet.com.
internet.com.    IN    MX    100   bk1.mail.net.
internet.com.    IN    MX    200   bk2.mail.net.
internet.com.    IN    MX    200   fail.safe.net.
; déclaration des enregistrements concernant les noms
w1.internet.com. IN    A            10.0.0.1
www              IN    CNAME         w1
www              IN    HINFO        "Linux/1.2.13 AlphaPC 64"
lune             IN    A            10.0.0.10
mail.in          IN    CNAME        mx-01
mx-01            IN    A            10.0.0.2
mx-02.internet.com. IN  A            10.0.0.2
```

## 13.8 - Déploiement de Bind (ISC Software)

Une implémentation libre du protocole DNS existe sous Linux, et présente d'excellente performance tant en qualité de service que d'administration. Bind est un serveur DNS accompagné de tous les outils nécessaires à la gestion d'un tel service (nslookup, dig, host, etc.).

Les sources sont disponibles sur le site ftp d'ISC : <ftp://ftp.isc.org/isc/bind/>

### 13.8.1 - Compilation

```
% mkdir /opt/src/bind-version
% cd /opt/src/bind-version
% tar xzf /path/to/bind-src.tar.gz
% tar xzf /pah/to/bind-doc.tar.gz
% cd src
% make clean
% make depend
% make all
% rm .settings
% make install
% make clean
```

### 13.8.2 - intégration système

Vérifier que le nom FQDN de votre station est bien présent sous la forme ci-dessous dans votre fichier `/etc/hosts`.

```
10.0.0.1 mars.internet.com mars
```

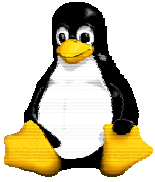
Modifier votre fichier `/etc/resolv.conf` pour fournir les informations de base à votre resolver comme ci-dessous.

```
search internet.com
nameserver 10.0.0.1
nameserver 10.0.0.2
nameserver 10.0.0.3
```

### 13.8.3 - Configuration

La configuration d'un serveur DNS demande la création des fichiers suivants :

- Un fichier de configuration du service DNS comportant une définition exhaustive des zones utilisées (`/etc/named.conf`)
- Un fichier de définition par zone configurée (y compris les zones inverses)
- Un fichier de définition de zone pour l'interface loopback



# Le système GNU/Linux

Version 9  
30/09/2003

- Un fichier contenant une liste de ROOT SERVERS à contacter pour les résolutions récursives

## Configuration du DNS : /etc/named.conf

Ce fichier définit les caractéristiques d'un serveur DNS. Il est généralement nommé /etc/named/named.conf et lié symboliquement pour une utilisation de /etc/named.conf.

```
# /etc/named.conf - configuration de BIND
```

```
options {
    directory "/etc/named";
};

zone "internet.com" in {
    type master;
    file "internet.com";
};

zone "0.0.10.in-addr.arpa" in {
    type master;
    file "rev.10.0.0";
};

zone "backup.net" in {
    type slave;
    masters {
        10.0.0.2;
        10.0.0.3;
    };
    file "db.backup.net";
};

zone "0.0.127.in-addr.arpa" in {
    type master;
    file "db.127.0.0";
};

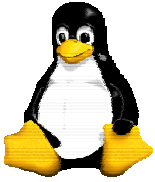
zone "." in {
    type hint;
    file "named.cache";
};
```

Le fichier de configuration suivant montre donc qu'il va être nécessaire de configurer les fichiers /etc/named/internet.com, /etc/named/rev.10.0.0, /etc/named/rev.127.0.0 et /etc/named/named.cache. Le fichier /etc/named/db.backup.net sera quant à lui créé automatiquement par le serveur, et contiendra les informations sur la zone backup.net issue d'un des serveurs primaires déclarés dans la configuration.

## Liste des ROOT SERVERS : named.cache

Ce fichier contenant exhaustivement la liste des ROOT SERVERS de l'Internet, il est évident que ce fichier est disponible librement sur les serveurs ftp des organismes de gestion des zones. Il appartient donc à chaque administrateur de mettre régulièrement à jour cette liste.

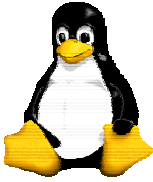




# Le système GNU/Linux

Version 9  
30/09/2003

```
; This file holds the information on root name servers needed
to
; initialize cache of Internet domain name servers
; (e.g. reference this file in the "cache . <file>"
; configuration file of BIND domain name servers).
;
; This file is made available by InterNIC registration
services
; under anonymous FTP as
;   file /domain/named.root
;   on server FTP.RS.INTERNIC.NET
; -OR- under Gopher at RS.INTERNIC.NET
;   under menu InterNIC Registration Services
(NSI)
;   submenu InterNIC Registration Archives
;   file named.root
;
; last update: Aug 22, 1997
; related version of root zone: 1997082200
;
;
; formerly NS.INTERNIC.NET
;
. 3600000 IN NS A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000 A 198.41.0.4
;
; formerly NS1.ISI.EDU
;
. 3600000 NS B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000 A 128.9.0.107
;
; formerly C.PSI.NET
;
. 3600000 NS C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000 A 192.33.4.12
;
; formerly TERP.UMD.EDU
;
. 3600000 NS D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET. 3600000 A 128.8.10.90
;
; formerly NS.NASA.GOV
;
. 3600000 NS E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET. 3600000 A 192.203.230.10
;
; formerly NS.ISC.ORG
;
. 3600000 NS F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET. 3600000 A 192.5.5.241
;
; formerly NS.NIC.DDN.MIL
;
```



# Le système GNU/Linux

Version 9  
30/09/2003

---

```
.                3600000      NS      G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET. 3600000      A       192.112.36.4
;
; formerly AOS.ARL.ARMY.MIL
;
.                3600000      NS      H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET. 3600000      A       128.63.2.53
;
; formerly NIC.NORDU.NET
;
.                3600000      NS      I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET. 3600000      A       192.36.148.17
;
; temporarily housed at NSI (InterNIC)
;
.                3600000      NS      J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET. 3600000      A       198.41.0.10
;
; housed in LINX, operated by RIPE NCC
;
.                3600000      NS      K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET. 3600000      A       193.0.14.129
;
; temporarily housed at ISI (IANA)
;
.                3600000      NS      L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET. 3600000      A       198.32.64.12
;
; housed in Japan, operated by WIDE
;
.                3600000      NS      M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET. 3600000      A       202.12.27.33
; End of File
```

## 13.9 - Exploitation du serveur DNS

Les opérations sur le serveur DNS (named) se font via un contrôleur, `ndc`. C'est lui qui reçoit les ordres, et informe le processus `named` des changements d'états à effectuer.

Démarrage du serveur DNS

```
% ndc start
```

Arrêt du serveur DNS

```
% ndc stop
```

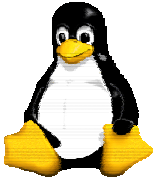
Prise en compte une modification de configuration

Si la configuration globale du serveur DNS a été modifiée, alors il est nécessaire de recharger l'ensemble de la configuration.

```
% ndc reload
```

Si seule une zone a été modifiée, il est possible d'économiser du temps machine en ne rechargeant que cette zone.

```
% ndc reload internet.com
```



## 14 - Le courrier électronique

### 14.1 - Introduction

Parmi toutes les fonctionnalités proposées par Internet, on compte 3 services de base. Tout d'abord des outils de connexion à distance comme par exemple Telnet. Ensuite des utilitaires de transfert de fichiers dont le logiciel le plus populaire et le plus utilisé est « Ftp ». Et enfin, et c'est ce qui va concentrer notre attention dans la suite de ce document, un protocole de transport de courrier, comme SMTP ( Simple Mail Transfer Protocol ).

Le courrier électronique (ou e-mail) est le plus populaire des services d'Internet. En effet, au niveau de l'utilisateur, il présente toutes les caractéristiques du courrier classique, avec la simplicité de ce dernier. Ceci permet donc l'envoi de courriers électroniques, aussi appelés e-mails. Mais ce qui fait sa popularité, c'est qu'il touche le plus grand nombre, des entreprises pour l'envoi d'informations professionnelles, aux particuliers pour se substituer au courrier classique, en passant par le monde étudiant pour un contact privilégié avec les entreprises.

Petit à petit le courrier électronique s'est imposé comme un moyen de communication rapide et fiable. En effet, la possibilité d'envoyer instantanément des messages est un atout considérable par rapport au courrier classique. De plus, ses capacités à faire des envois en nombre en toute simplicité, ajouté à des options intéressantes comme les envois de fax, en fait l'allié indispensable des entreprises. Et les possibilités d'envois de documents multimédia (textes, images, sons, etc.) ont apporté des atouts indéniables venus consolider sa position de leader dans le monde des outils de communication du monde moderne.

### 14.2 - Les principes du courrier électronique

#### 14.2.1 - Le principe de la messagerie

Pour envoyer un courrier, il faut :

- un texte
- une enveloppe

Mais ce n'est pas suffisant !

En effet, il faut aussi :

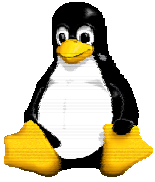
- l'adresse précise de l'expéditeur
- l'adresse précise du destinataire
- un timbre

Et il reste encore à poster la lettre...

Ensuite, il y a la levée du courrier, qui part dans un centre de tri (routage), puis est expédié par train, avion, etc. Arrivé au bureau de poste de la ville du destinataire, le courrier est à nouveau trié, puis il est déposé dans la boîte aux lettres du destinataire par le facteur.

Un courrier électronique opère exactement de la même façon !

Des logiciels permettent d'écrire et de lire son courrier, une adresse électronique identifie tout expéditeur/destinataire, et tout le cheminement d'un courrier classique par le biais de la Poste est effectué par des modules logiciels.



# Le système GNU/Linux

Version 9  
30/09/2003

---

## 14.2.2 - Les composantes du système du courrier électronique

Le système de courrier électronique s'appuie sur plusieurs composantes formant ainsi des modules, qui coopèrent les uns avec les autres. Cette architecture permet une grande souplesse, par exemple, le remplacement d'un seul module par un élément plus récent suffit à faire évoluer le système de courrier électronique.

Les composantes sont au nombre de quatre :

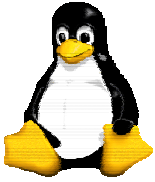
Mail Transfer Agent (MTA) : c'est un module logiciel effectuant un travail analogue à celui d'un centre de tri postal. En effet, tous les mails émanant du réseau ou d'une machine locale transitent par ce Mail Transfer Agent. Ensuite, en fonction de l'adresse de destination, ce module décide soit de renvoyer le mail vers un autre centre de tri (MTA) si l'adresse n'est pas directement accessible, ou alors de le donner à un facteur (MDA) s'il s'agit d'une adresse locale.

Mail Delivery Agent (MDA) : c'est un module logiciel effectuant un travail analogue à celui d'un facteur. Il est chargé de distribuer les courriers dans les boîtes aux lettres en copiant les mails reçus par le centre de tri (MTA) dans les boîtes aux lettres des utilisateurs locaux.

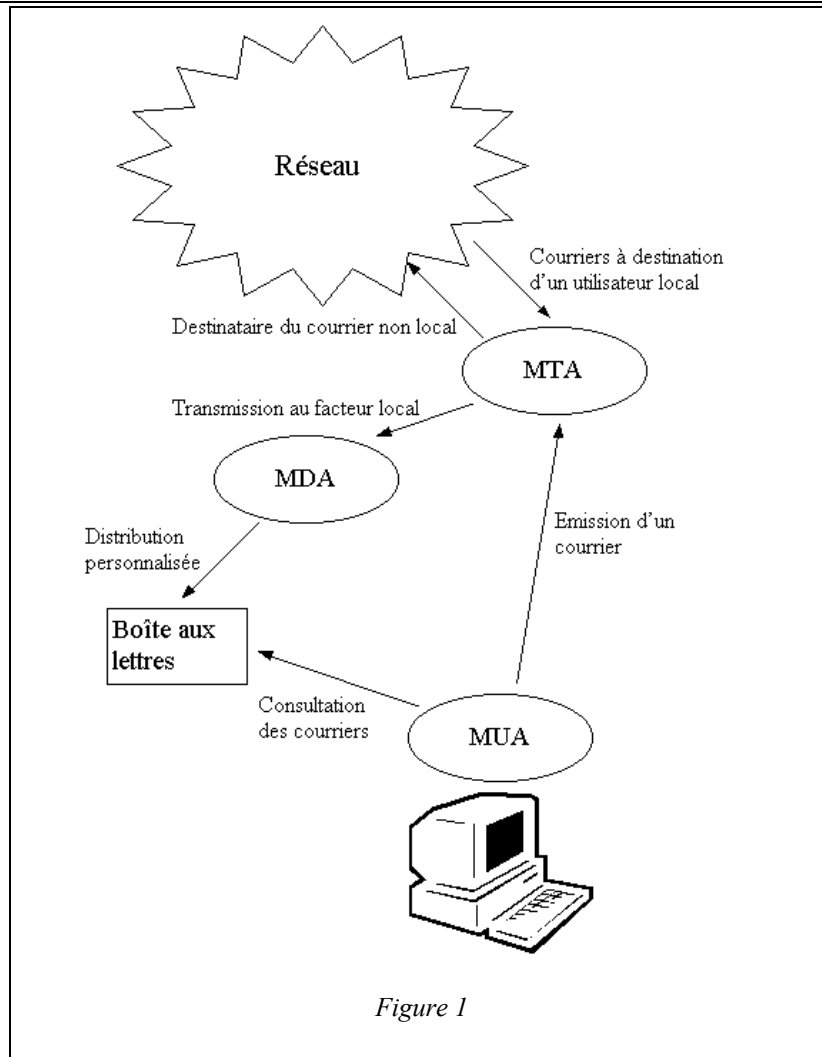
Boîte aux lettres : sa forme dépend du système utilisé. Sur Unix, il s'agit juste d'un fichier dans lequel le MDA copie les messages qu'il reçoit du MTA.

Mail User Agent (MUA) : il s'agit de la partie logicielle dont se sert l'utilisateur pour lire les messages contenus dans sa boîte aux lettres, ainsi que pour écrire de nouveaux mails, qui seront envoyés au MTA afin d'être traités.

Le schéma suivant (cf. figure 1) présente les échanges effectués entre les différentes composantes du courrier électronique.



# Le système GNU/Linux

Version 9  
30/09/2003

Voici maintenant comment est réalisée la coopération entre les composants du système de courrier électronique. Chaque tâche appelle une tâche partenaire de la façon suivante :

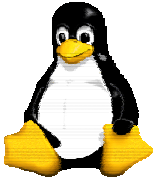
○ Réseau  $\Rightarrow$  MTA et MTA  $\Rightarrow$  Réseau : comme la plupart des machines connectées utilisent le protocole IP, les communications entre les MTAs se font simplement en écoutant sur un *well known port*, c'est-à-dire sur un numéro de port dédié et connu de tout le monde. Généralement, le port SMTP (lorsque le MTA est précisément le Sendmail) est décrit dans `/etc/services`, et indique de la façon suivante qu'il s'agit du port 25, utilisant le protocole TCP :

```
smtp          25/tcp
```

Une fois connectés à ce port, les MTAs s'échangent des données selon les caractéristiques du protocole SMTP, décrit à l'origine dans la RFC 821.

○ MTA  $\Rightarrow$  MDA : le MTA appelle le MDA suivant une méthode qui est généralement précisée dans le fichier de configuration.

○ MUA  $\Rightarrow$  MTA : il existe beaucoup de MUAs différents, mais pour qu'ils puissent facilement communiquer avec le MTA, tous les MUAs suivent la même méthode, équivalente à :



# Le système GNU/Linux

---

cat message | MTA

où « message » est le courrier à envoyer que l'on aura composé grâce au MUA.

Le MTA peut être de plusieurs types. Le choix du MTA est alors précisé dans le fichier de configuration du MUA.

## 14.2.3 - Le Mail Transfer Agent (MTA)

Ceci constitue la partie logicielle la plus complexe du système de courrier électronique. En effet, celui-ci doit interagir avec tous les autres composants du courrier électronique. Il doit pouvoir communiquer avec le milieu extérieur, c'est-à-dire le réseau et les autres MTAs, il doit pouvoir donner des ordres aux MDAs, et doit pouvoir répondre aux requêtes des MUAs.

Il existe beaucoup de MTAs différents, avec chacun leurs spécificités et leurs fonctions plus ou moins complexes. Généralement, chaque MTA est lié à un système d'exploitation.

On trouve ainsi smail, mmdf, mais surtout **Sendmail** qui nous intéresse ici puisque c'est celui que l'on retrouve de base sur les systèmes Linux, \*BSD, SunOS, Solaris, HP-UX, IRIX, etc. C'est le MTA qui sera étudié en détails dans la suite de ce document, car c'est sans doute le plus répandu des agents de transfert du courrier. La partie la plus délicate concerne l'écriture du fichier de configuration du Sendmail. En effet, la difficulté de celui-ci croît très rapidement avec la complexité du réseau sur lequel il est implanté. Les secrets de la configuration seront percés un peu plus loin.

## 14.2.4 - Le Mail Deliver Agent (MDA)

Le MDA est appelé par le MTA afin de copier les messages reçus dans la boîte aux lettres de l'utilisateur concerné.

En général, chaque constructeur fournit sa version du module MDA, qui, souvent, fonctionne assez mal en milieu homogène, et est inutilisable en milieu hétérogène.

Cependant, quelques programmes de substitutions existent dans le domaine public, qui sont en général plus performants. Par exemple :

mail	MDA de base
sendmail 8.7.5	Cette version de Sendmail contient un utilitaire MDA
procmil	Un des utilitaires les plus robustes. Il prend en compte les problèmes liés au fonctionnement en milieu NFS, hétérogène.

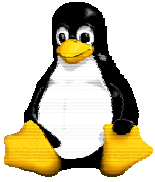
Tableau 1

## 14.2.5 - Le Mail User Agent (MUA)

Le MUA est le logiciel utilisé au niveau de l'utilisateur. Le marché offre un choix très important, comme par exemple mail, pine, mutt, elm, mailx, Mail, mailtool, netscape, etc.

Une partie importante des logiciels disponibles ne sont pas très fiables, et dans tous les cas ne sont pas adaptés à une utilisation en milieu hétérogène.

Le choix du MUA sera donc fait en fonction de la configuration de la consultation du courrier électronique sur le site considéré.



# Le système GNU/Linux

Voici les principaux *Mail User Agent* que l'on trouve sur le marché :

elm	
mail	MUA d'Unix
Mail	Alternative de <i>mail</i>
mailtool	
mailx	Alternative de <i>mail</i>
netscape	
Pine	... de l'Université de Washington
rmail	Utilisé avec <i>mail</i> pour l'UUCP
Metamail	Permet le support MIME
deliver	Classement/Gestion du courrier
procmail	Classement/Gestion du courrier
Majordomo	Gère les listes e-mail (listes de distribution)
Mserv	Permet les requêtes de fichiers par mail

Tableau 2

## 14.2.6 - Format des adresses électroniques

L'évolution des protocoles des réseaux connectés à Internet, a obligé les MTAs à s'adapter, afin de pouvoir communiquer quelque soit le protocole. On se trouve donc aujourd'hui face à plusieurs types d'adresses de courrier électronique possibles. Et on retrouve ces différents formats dans les fichiers de configuration.

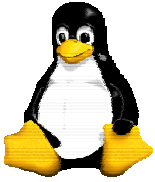
Voici les types d'adresses de courrier électronique qu'il est possible de rencontrer.

### 14.2.6.1 - Adresses Internet globales

Ces adresses sont valides en tout point du réseau Internet, car elles ne spécifient pas le chemin pour arriver au site qu'elles représentent.

- toto@domaine.fr : c'est l'adresse la plus simple. Les espaces entre les composants sont acceptés.
- toto (toto le vrai) @ domaine.fr : il s'agit de la même adresse que la précédente, mais avec un commentaire entre parenthèses, qui peut être inséré n'importe où dans l'adresse, puisqu'il est ignoré par le MTA lors du traitement.
- "toto le vrai" @ domaine.fr : la partie entre guillemets est une partie locale, considérée comme un seul mot. Le courrier est adressé à l'utilisateur "toto le vrai" dans le domaine domaine.fr.
- <toto@domaine.fr> : la signification de ces adresses est analogue aux adresses globales. De manière générale, la présence des caractères < et > indique que c'est une adresse facilement exploitable par le MTA, en quelque sorte "prédigérée".
- toto le vrai <toto@domaine.fr> : un commentaire (sans parenthèses) est ajouté à l'adresse. Le MTA traite en priorité ce qui est entre les caractères < et >, et ignore tout le reste.





# Le système GNU/Linux

Version 9  
30/09/2003

○ "toto le vrai" <toto@domaine.fr> : idem à l'adresse précédente, mais ici, le commentaire est considéré comme un mot unique puisqu'il est entre guillemets.

## 14.2.6.2 - Adresses Internet avec routage explicite

Par exemple : @etape1.fr,@etape2.fr :toto@domaine.fr

Cette adresse spécifie que le courrier doit être envoyé à etape1.fr, qui transmettra à etape2.fr, qui enverra le courrier à toto@domaine.fr.

Leur utilisation est à proscrire, parce que beaucoup de sites ne les reconnaissent pas correctement, et cela pose donc quelques problèmes. De plus, dans le cas où etape2.fr ne fonctionnerait pas, le courrier ne pourrait plus être acheminé. Le RFC 1123 déconseille d'ailleurs formellement leur utilisation.

## 14.2.6.3 - Adresses Internet littérales

Par exemple : toto@[172.20.0.22]

Les crochets indiquent qu'il s'agit d'une adresse numérique qui doit être traitée telle quelle. Le courrier doit donc être envoyé directement à la machine indiquée par l'adresse entre crochets. Il est déconseillé d'utiliser ce type d'adresses.

## 14.2.6.4 - Adresses Internet avec routage explicite à la RFC 1123

Par exemple : toto%domaine.fr%etape2.fr@etape1.fr

Cette adresse s'interprète de la même façon que @etape1.fr,@etape2.fr :toto@domaine.fr. Cependant, le RFC 1123 spécifie qu'il est préférable d'utiliser ce type d'adresse plutôt que celui avec routage explicite.

## 14.2.6.5 - Adresse Internet particulière de la RFC 1123

L'adresse particulière <> spécifie une adresse à laquelle on ne peut pas répondre. En effet, lorsqu'un message d'erreur est généré, l'adresse de l'expéditeur doit être spécifiée, et spécifier Postmaster, ou Mailer-Demon peut poser quelques problèmes, en particulier des boucles de courrier. <> évite ce genre de problèmes.

## 14.2.6.6 - Adresse UUCP ( Unix to Unix Copy Program )

Les adresses UUCP sont de la forme suivante :

```
site1!site2!...!siten!user
```

Cela signifie que le courrier UUCP doit d'abord être envoyé à « site1 » qui doit faire suivre à « site2 », et ainsi de suite jusqu'à « siteN », qui le délivrera à l'utilisateur spécifié dans « user ».

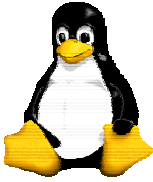
Une adresse UUCP peut également être mélangée avec une adresse Internet. On obtient alors quelque chose de la forme :

```
site2!site3!...!siten!user@site1
```

Dans ce cas, le message doit être envoyé à « site1 », qui le fera suivre par UUCP aux sites suivants, jusqu'à l'utilisateur.

## 14.2.7 - Structure d'un courrier

Un courrier électronique a une structure similaire à celle d'un courrier classique. Autrement dit, un courrier électronique est composé d'une enveloppe (aussi appelé « **en-tête du message** »),



# Le système GNU/Linux

Version 9  
30/09/2003

comportant les données relatives aux adresses des émetteurs et récepteurs, ainsi que le sujet du message, la date, etc. A cela s'ajoute évidemment le contenu du courrier (aussi appelé « **corps du message** »). Ces deux éléments doivent être séparés par au moins une ligne vide.

Les caractéristiques du courrier électronique sont définies dans la RFC 822, qui a été corrigée par la RFC 1123.

## 14.2.7.1 - En-tête du message

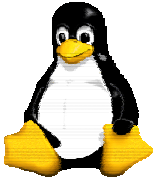
Cette partie contient toutes les informations qui vont permettre au courrier électronique d'être acheminé jusqu'à son destinataire. Chaque ligne de l'en-tête est composée de la même façon, c'est-à-dire : *mot-clé* : *arguments*

Voici un exemple d'en-tête de courrier électronique :

```
Return-Path: <rahan@iut-amiens.fr>
X-Sieve: cmu-sieve 2.0
Received: from morglum.iut-amiens.fr (www.iut.u-picardie.fr [194.57.109.129])
        by gip.u-picardie.fr (Postfix) with ESMTTP id C67BB7DCA
        for <Cyril.Drocourt@u-picardie.fr>; Thu, 3 Jan 2002 13:47:36 +0100
(CET)
Received: from etud.iut.fr (etud.info.iut.u-picardie.fr [194.57.109.130])
        by morglum.iut-amiens.fr (8.11.2/8.11.2) with ESMTTP id g03Clal06505
        for <rahan@iut-amiens.fr>; Thu, 3 Jan 2002 13:47:36 +0100
Received: from morglum.iut-amiens.fr (morglum.iut-amiens.fr [194.57.109.129])
        by etud.iut.fr (8.11.2/8.11.2) with ESMTTP id g03Clah28249
        for <drocourt@info.iut-amiens.fr>; Thu, 3 Jan 2002 13:47:36 +0100
Received: (from rahan@localhost)
        by morglum.iut-amiens.fr (8.11.2/8.11.2) id g03Clal06500
        for drocourt@info.iut-amiens.fr; Thu, 3 Jan 2002 13:47:36 +0100
Date: Thu, 3 Jan 2002 13:47:36 +0100
From: rahan@iut-amiens.fr
To: drocourt@info.iut-amiens.fr
Subject: Petit bonjour
Message-ID: <20020103134736.A6467@morglum.iut-amiens.fr>
Mime-Version: 1.0
Content-Type: text/plain; charset=us-ascii
X-Mailer: Mutt 1.0pre3us
```

Voici les différents mots-clés utilisés et leur signification :

<b>Return-Path</b>	Champ utilisé pour indiquer l'adresse à utiliser pour une réponse éventuelle.
<b>Received</b>	Chaque machine par laquelle passe le message rajoute les informations suivantes : <i>From son nom, By le nom de celle qui l'a expédié, With le protocole utilisé.</i>
<b>From</b>	Identité de l'expéditeur (non conforme à la RFC 822, champ spécifique à UNIX).
<b>Sender</b>	Adresse de l'expéditeur réel (s'il est différent du précédent).
<b>Reply-to</b>	Adresse pour une réponse éventuelle spécifiée par l'expéditeur (contrairement au champ Return-Path qui est rempli automatiquement).
<b>Date</b>	Date d'émission du message.
<b>To</b>	Adresse du ou des destinataire(s).
<b>Cc</b>	Pour envoyer une copie d'un message à une autre adresse (« Carbon Copy »).
<b>Bcc</b>	Pour envoyer une copie d'un message à une autre adresse sans que les autres destinataires ne soient au courant (« Blind Carbon Copy »).
<b>Message-Id</b>	Numéro d'identification du message.
<b>In-Reply-To</b>	Utiliser pour spécifier que le présent message se rapporte à un courrier précédent.



# Le système GNU/Linux

Version 9  
30/09/2003

<b>References</b>	Références aux messages précédents.
<b>Keywords</b>	Mots-clefs
<b>Subject</b>	Sujet du message.
<b>Comments</b>	Commentaires.
<b>Encrypted</b>	Message crypté, et méthode de cryptage utilisée.
<b>X- ???</b>	Messages d'extension

Tableau 3

## 14.2.7.2 - Corps du message

Cette partie contient les données du courrier électronique. Il s'agit de texte, codé soit sur 7 bits (protocole SMTP : *Simple Mail Transfer Protocol*) ce qui interdit l'utilisation des caractères accentués, soit sur 8 bits (protocole ESMTP : *Extended Simple Mail Protocol*) et dans ce dernier cas, le problème précédent ne se pose plus.

## 14.2.8 - L'enveloppe mail

Une des notions importantes du mail, que les utilisateurs ont beaucoup de mal à assimiler, est la notion « d'enveloppe ».

Comme nous l'avons vu précédemment, un message est composé de 2 parties :

- Une entête qui est une suite de champs spécifiés par les **RFC 822, 1341 à 1345**.
- Un corps qui est le contenu du mail proprement dit.

Pour permettre la remise de courrier à divers destinataires, le programme MTA utilise la notion d'enveloppe.

L'entête du mail supporte plusieurs champs permettant de spécifier un ou plusieurs destinataires ( via les champs **To:**, **Cc:**, **Bcc:** ) et le ou les expéditeurs ( via les champs **From:**, **Reply-to:**, **Return-Path:**, **Resent-From:** ). Or ces champs ne permettent pas , par eux-mêmes, de délivrer du courrier.

En effet, si le champ **To:**, était utilisé pour acheminer du courrier jusqu'à sa destination, cela poserait de gros problèmes. Par exemple :

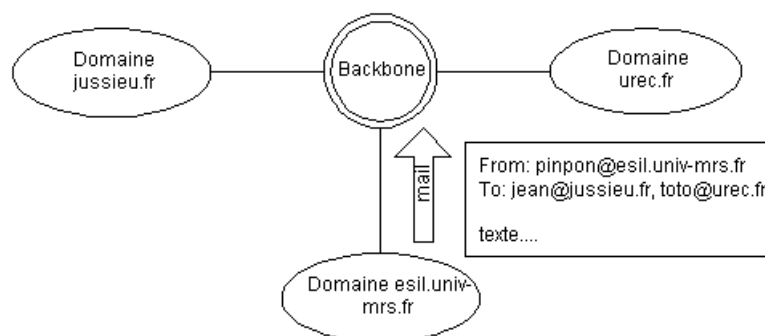
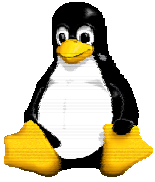


Figure 2

Le message serait envoyé aux sites **urec.fr** et **jussieu.fr**. Lors de son arrivée à **jussieu.fr**, comme le champ **To:** est utilisé, le message est remis à jean, mais est également transmis à **urec.fr**, puisque le



# Le système GNU/Linux

Version 9  
30/09/2003

champ **To:** contient deux destinataires. Le site **urec.fr** reçoit donc maintenant deux exemplaires de ce courrier. S'il utilise lui aussi le champ **To:** il va renvoyer le message à **jussieu.fr** et on voit que l'on a créé une boucle de courrier. « *Les en-têtes ne permettent donc pas à eux seules une remise du courrier* ».

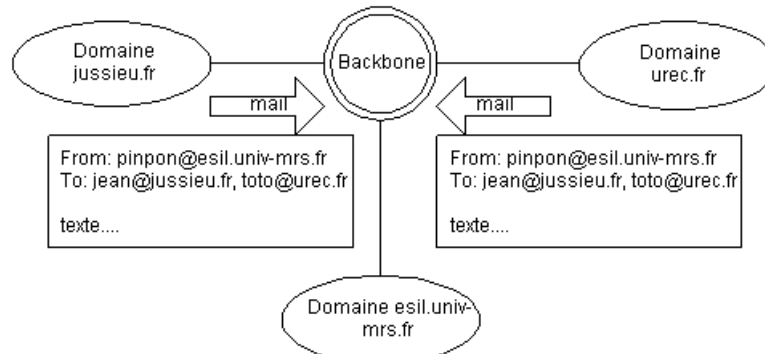


Figure 3

Il convient donc de véhiculer une nouvelle information avec un message. Cette information par analogie avec la poste, s'appelle l'**enveloppe**. Elle sert à router les messages.

Au début de l'envoi du courrier, l'enveloppe contient les mêmes adresses destination que les adresses mentionnées dans le champ **To:**, mais elles évoluent au fur et à mesure du traitement du courrier. Si l'on reprends l'exemple précédent, lors de la remise du courrier à **jussieu.fr**, l'enveloppe ne contiendra que **jean@jussieu.fr** et lors de la remise à **urec.fr**, l'enveloppe ne contiendra que **toto@urec.fr**. En revanche, le champ **To:** de l'entête du courrier contiendra toujours les deux destinataires **To: jean@jussieu.fr, toto@urec.fr**.

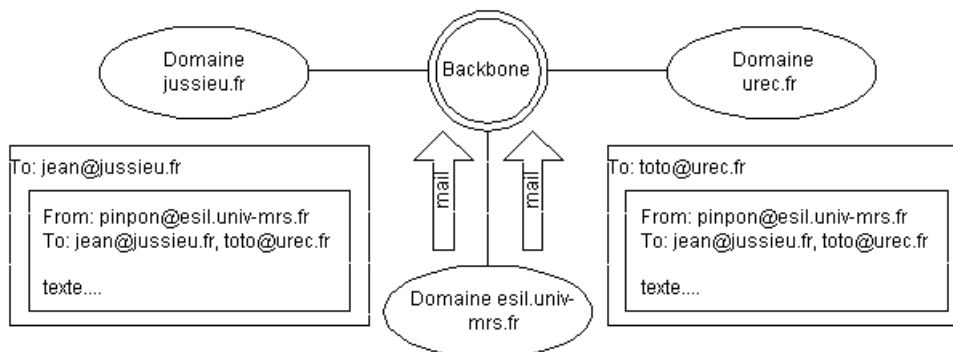
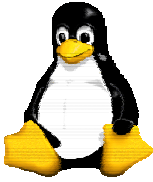


Figure 4

L'enveloppe sert seulement pendant le processus de remise du courrier, alors que les en-têtes servent aux destinataires du courrier à titre d'information.

## 14.3 - Le protocole SMTP

### 14.3.1 - Caractéristiques du protocole SMTP



# Le système GNU/Linux

Version 9  
30/09/2003

SMTP signifie *Simple Mail Transfer Protocol*. Il s'agit donc du protocole de transfert de courrier électronique le plus basique, aussi bien dans ses spécificités que dans son utilisation.

Tout envoi de courrier électronique commence par l'établissement d'un canal de communication entre les deux correspondants que sont l'expéditeur et le destinataire. SMTP est un protocole application, au dessus de la couche Transport TCP.

Application	SMTP/ESMTP
Présentation	
Session	
Transport	
Réseau	
Ligne	
Physique	
	TCP
	IP
	Dépendant du réseau

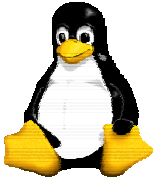
Après l'ouverture du canal, des commandes spécifiques permettent de traiter le courrier à envoyer, mais il existe aussi un certain nombre de commandes qui effectuent des actions propres au suivi de la communication entre les deux entités.

Chaque commande invoquée est suivie d'une réponse, composée d'un code numérique, et d'un message. Par exemple, le code numérique « 250 » est utilisé pour signifier « OK », tandis que « 500 » signifiera « commande inconnue ».

Voici les principales commandes et leur rôle :

<b>DATA</b>	Ce qui suit cette commande est le texte du courrier électronique.
<b>EXPN</b> <i>argument</i>	Donne des informations sur un utilisateur, ou un groupe d'utilisateurs. « <i>argument</i> » est le nom de l'utilisateur.
<b>HELO</b>	Première commande à effectuer qui permet de vérifier la connexion avec la machine distante. Si cette dernière comprend qu'il s'agit d'un message SMTP, elle renverra le code 250, signifiant OK.
<b>HELP</b>	Aide, pouvant prendre en argument une commande spécifique.
<b>MAIL</b> <b>From:</b> < <i>adresse</i> >	Juste après la commande HELO, « <i>adresse</i> » est l'adresse de l'émetteur.
<b>NOOP</b>	Commande sans action (« No Opération »). Elle peut être utile pour vérifier que le canal de communication n'est pas rompu.
<b>QUIT</b>	L'émetteur demande à quitter la session, mais cela ne deviendra effectif qu'après un acquittement positif du destinataire.
<b>RCPT</b> <b>To:</b> < <i>adresse</i> >	Spécifie le ou les destinataire(s) du courrier électronique. « <i>adresse</i> » est l'adresse du destinataire du courrier.
<b>RSET</b>	Permet d'interrompre la session.
<b>TURN</b>	Permet d'invertir les rôles entre l'émetteur et le récepteur
<b>VERFY</b> <i>argument</i>	Vérifie si le destinataire connaît un utilisateur spécifique. « <i>argument</i> » est le nom de l'utilisateur.

Tableau 4



# Le système GNU/Linux

Voici un exemple de session SMTP :

```
[rahan]$ telnet morglum 25
Trying 194.57.109.129...
Connected to morglum.iut-amiens.fr.
Escape character is '^]'.
220 morglum.iut-amiens.fr ESMTP Sendmail 8.11.2/8.11.2; Thu, 3 Jan 2002 13:52:55
+0100
HELO morglum
250 morglum.iut-amiens.fr Hello rech2 [194.57.109.131], pleased to meet you
HELP
214-2.0.0 This is sendmail version 8.11.2
214-2.0.0 Topics:
214-2.0.0         HELO      EHLO      MAIL      RCPT      DATA
214-2.0.0         RSET      NOOP      QUIT      HELP      VRFY
214-2.0.0         EXPN      VERB      ETRN      DSN       AUTH
214-2.0.0         STARTTLS
214-2.0.0 For more info use "HELP <topic>".
214-2.0.0 To report bugs in the implementation send email to
214-2.0.0         sendmail-bugs@sendmail.org.
214-2.0.0 For local information send email to Postmaster at your site.
214 2.0.0 End of HELP info
MAIL FROM: drocourt@iut-amiens.fr
250 2.1.0 drocourt@iut-amiens.fr... Sender ok
RCPT TO: <rahan>
250 2.1.5 <rahan>... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
Subject: Essai de mail
Ceci est le corps du message ...
.
250 2.0.0 g03CsN107973 Message accepted for delivery
NOOP
250 2.0.0 OK
QUIT
221 2.0.0 morglum.iut-amiens.fr closing connection
Connection closed by foreign host.
[rahan]$
```

Les limitations du protocole SMTP sont présentées dans la RFC 821. Elles sont les suivantes :

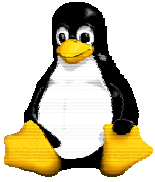
Codage du message	sur 7 bits
Nom de l'utilisateur	< 64 caractères
Nom de domaine	< 64 caractères
Nombre de destinataires	< 100
Lignes	< 1000 caractères

Tableau 5

## 14.3.2 - Caractéristiques du protocole ESMTP

Ce protocole n'est qu'une extension du protocole SMTP. Il est donc naturellement compatible avec ce dernier, mais il présente des nouveautés intéressantes.

La commande d'ouverture (« HELO ») est remplacée par « EHLO » (*Extended HELO*). Le destinataire ne répond plus seulement « OK », mais donne aussi la liste des mots-clefs qu'il connaît et qu'il est capable de traiter. Le destinataire peut renvoyer un message d'erreur signifiant qu'il ne supporte pas le protocole ESMTP et la communication continue alors en mode SMTP.



# Le système GNU/Linux

Version 9  
30/09/2003

Le contenu des messages est maintenant codé sur 8 bits ce qui autorise l'utilisation des caractères accentués. Ceci a abouti aux spécifications MIME (*Multipurpose Internet Mail Extensions*), dont le but est de permettre le transfert de données multimédias via le courrier électronique.

Voici un exemple de session ESMTP :

```
[rahan]$ telnet morglum 25
Trying 194.57.109.129...
Connected to morglum.iut-amiens.fr.
Escape character is '^]'.
220 morglum.iut-amiens.fr ESMTP Sendmail 8.11.2/8.11.2; Thu, 3 Jan 2002 14:00:03
+0100
ehlo morglum
250-morglum.iut-amiens.fr Hello rech2 [194.57.109.131], pleased to meet you
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-SIZE 5000000
250-DSN
250-ONEX
250-ETRN
250-XUSR
250 HELP
MAIL FROM: rahan@iut-amiens.fr
250 2.1.0 rahan@iut-amiens.fr... Sender ok
RCPT TO: Cyril.Drocourt@u-picardie.fr
250 2.1.5 Cyril.Drocourt@u-picardie.fr... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
message ...
.
250 2.0.0 g03D1t110892 Message accepted for delivery
QUIT
221 2.0.0 morglum.iut-amiens.fr closing connection
Connection closed by foreign host.
[rahan]$
```

## 14.4 - La réception du courrier

### 14.4.1 - Le mail local

Le plus simple et le plus basique pour récupérer ses mails est d'utiliser la boîte aux lettres locale. Sur les systèmes Unix par exemple, les mails reçus sont déposés dans un répertoire spécifique sur la machine destinatrice, en attendant d'être lu par l'utilisateur. Toutefois, chaque machine ne peut gérer ses propres mails (problème d'administration), de plus il ne doit pas être nécessaire d'avoir un compte Unix sur une machine pour récupérer ses mails. C'est pourquoi d'autres solutions existent.

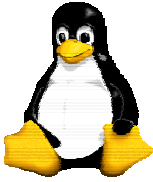
### 14.4.2 - Utiliser une interface web

Certains sites proposent une interface web pour consulter et envoyer des mails, dans ce cas c'est le serveur sur lequel est hébergé la page que utilise le protocole SMTP pour recevoir et envoyer des messages sur internet. Le dialogue entre ce serveur et le poste client (sur lequel fonctionne le navigateur) est uniquement réalisé en HTTP. Tous les messages sont alors stockés sur le serveur.

### 14.4.3 - Utiliser une connexion POP ou IMAP

Il existe pour gérer le courrier électronique à distance deux protocoles de gestion de boîtes aux lettres:





# Le système GNU/Linux

Version 9  
30/09/2003

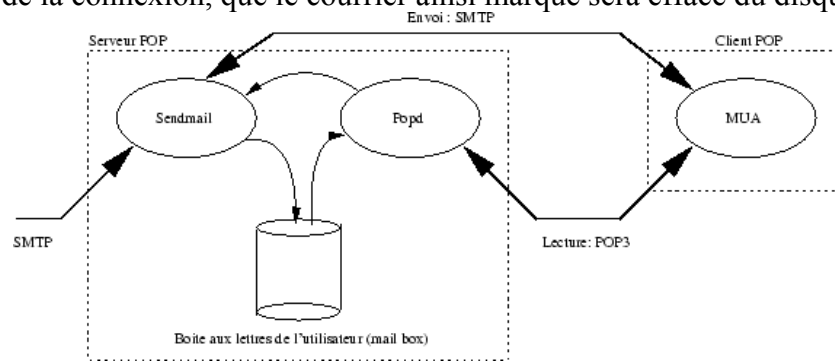
- ✓ "Post Office Protocol" ou POP
- ✓ "Internet Message Acces Protocol" ou IMAP

Le protocole POP est le plus ancien et le plus largement répandu. Très simple, il possède environ une dizaine de commandes alors que le protocole IMAP est plus récent et propose plus de fonctionnalités que POP.

Si on utilise une connexion POP pour lire ses messages, on récupère les courriers directement sur un serveur POP qui les stocke pour l'utilisateur (serveur du fournisseur d'accès par exemple). On peut également laisser une copie des courriers sur le serveur POP, pour pouvoir être ensuite récupérés d'un autre endroit. Si on utilise une connexion IMAP, on consulte directement la boîte aux lettres à distance, ce qui permet d'en avoir une seule et unique copie, où que l'on soit dans le monde.

## 14.4.3.1 - Le protocole POP3

Ce protocole est défini par la RFC1939. Généralement, ce protocole est utilisé en mode off line, c'est à dire que le client ramène dans sa boîte aux lettres locale son courrier stocké sur le serveur depuis sa dernière relève. Le courrier est alors lu après l'avoir rapatrié, hors connexion, en local. Lorsque le client souhaite effacer son courrier, il doit rendre un "flag" ou drapeau actif. C'est seulement à la fin de la connexion, que le courrier ainsi marqué sera effacé du disque du serveur.



Par contre, il n'y a qu'une seule boîte aux lettres sur le serveur. En général, le serveur POP écoute le port 110 de la machine.

Nous allons aborder maintenant la dizaines de commandes qui permettent de communiquer avec un serveur POP. La syntaxe des commandes du protocole POP est la suivante:

*[ commande ] [ argument ] [ argument ] ( retour chariot )*

Les commandes peuvent être:

**USER** : prend comme argument le nom de l'utilisateur qui veut se connecter à sa boîte aux lettres.

**PASS** : prend comme argument le mot de passe du client qui veut se connecter.

**STAT** : pas d'argument, affiche le nombre de messages de la boîte aux lettres et sa taille.

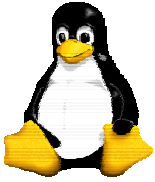
**LIST** : si il n'y a pas d'argument, affiche pour chaque message son numéro et sa taille, sinon ne le fait que pour le message passé en argument.

**RETR** : prend en argument le numéro du message que l'on veut rapatrier.

**DELE** : prend en argument le numéro du message que l'on souhaite effacer.

**NOOP** : rien faire.....

**RSET** : permet de retirer tous les drapeau "effacés".



# Le système GNU/Linux

Version 9  
30/09/2003

---

**QUIT** : pas d'argument, permet de quitter la connexion en cours.

Les réponses retournent +OK en début de ligne si tout ce passe bien sinon retournent - ERR en cas de problème. Voici un exemple de session POP :

```
[rahan]$ telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.
+OK POP3 Welcome to GNU POP3 Server Version 0.9.8 <11965.1010143243@morglum.iut-amiens.fr>
USER rahan
+OK
PASS motdepasse
+OK opened mailbox for rahan
STAT
+OK 1 369
LIST 1
+OK 1 369
RETR 1
+OK
Return-Path: <rahan@iut-amiens.fr>
Received: (from rahan@localhost)
    by morglum.iut-amiens.fr (8.11.2/8.11.2) id g04BKZJ10755
    for rahan; Fri, 4 Jan 2002 12:20:35 +0100
Date: Fri, 4 Jan 2002 12:20:35 +0100
From: rahan@iut-amiens.fr
Message-Id: <200201041120.g04BKZJ10755@morglum.iut-amiens.fr>
To: rahan@iut-amiens.fr
Subject: Essai de mail

Bonjour rahan

.
DELE 1
+OK Message 1 marked
QUIT
+OK
Connection closed by foreign host.
[rahan]$
```

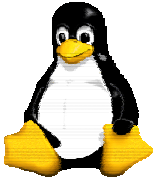
## 14.4.3.2 - Le protocole IMAP

Ce protocole est définie par la RFC2060. Il permet la gestion de plusieurs boites aux lettres sur plusieurs serveurs à partir de plusieurs clients. Il permet également la consultation de messages MIME. Le mode on line supporté veut dire que le client et le serveur sont connecté pendant toute la durée de la manipulation du courrier électronique. Le client peut donc consulter son courrier en restant connecté.

De ce fait, on peut créer toute une arborescence de boites aux lettres à partir de la principale (inbox). Par exemple, l'utilisateur toto peut avoir des sous boites aux lettres, une pour les messages personnels, une pour les message du travail, une pour la cuisine, ....

La syntaxe des commandes du protocole IMAP est la suivante:

[ tag ] [ commande ] [ argument ] [ argument ] ( retour chariot )



# Le système GNU/Linux

Version 9  
30/09/2003

Le tag est un code alphanumérique que vous pouvez choisir à votre guise. En générale, on utilise un code croissant pour identifier temporellement les commandes. Les commandes principales sont:

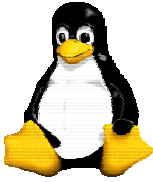
**LOGIN** : prend comme argument le nom et le mot de passe du client.  
**SELECT** : prend comme argument le nom de la boîte que l'on veut sélectionner.  
**CREATE** : le nom de la nouvelle boîte est passée en argument.  
**DELETE** : efface la boîte aux lettres qui est passée en argument.  
**RENAME** : prend comme argument le nom de l'ancienne boîte et le nouveau.  
**LIST** : prend comme argument la référence (ex : etc/mail/....) et le nom de la boîte.  
**STATUS** : donne les informations sur une boîte.  
**SEARCH** : commande qui permet de rechercher des messages selon des critères spécifiques.  
**CLOSE** : pas d'argument, ferme la boîte mais attend de nouveau un login.  
**FETCH** : prend plusieurs arguments, permet de consulter un message

Le serveur peut retourner plusieurs types de réponses:

**tag \* OK** : la commande c'est bien déroulée.  
**tag \* NO** : échec de la commande.  
**BAD** : erreur de protocole.  
**PREAUTH**: message d'accueil, indique des fois qu'il n'est pas nécessaire de se logger.  
**BYE** : le serveur va fermer sa session.

Voici un exemple de session IMAP :

```
[rahan]$ telnet mailx.u-picardie.fr 143
Trying 193.49.184.17...
Connected to gip.u-picardie.fr.
Escape character is '^]'.
* OK gip Cyrus IMAP4 v2.0.16 server ready
a001 LOGIN roger motdepasse
a001 OK User logged in
a002 select inbox
* FLAGS (\Answered \Flagged \Draft \Deleted \Seen)
* OK [PERMANENTFLAGS (\Answered \Flagged \Draft \Deleted \Seen \*)]
* 7 EXISTS
* 1 RECENT
* OK [UNSEEN 1]
* OK [UIDVALIDITY 1004181443]
* OK [UIDNEXT 3652]
a002 OK [READ-WRITE] Completed
a003 FETCH 7 FULL
* 7 FETCH (FLAGS (\Recent) INTERNALDATE " 4-Jan-2002 12:43:40 +0100" RFC822.SIZE
636 ENVELOPE ("Fri, 4 Jan 2002 12:43:40 +0100" "essai de mail IMAP" ((NIL NIL
"rahan" "iut-amiens.fr")) ((NIL NIL "rahan" "iut-amiens.fr")) ((NIL NIL "rahan"
"iut-amiens.fr")) ((NIL NIL "cyril.drocourt" "u-picardie.fr")) NIL NIL NIL
"<200201041143.g04Bhe018099@morglum.iut-amiens.fr>") BODY ("TEXT" "PLAIN"
("CHARSET" "us-ascii") NIL NIL "7BIT" 11 1))
a003 OK Completed
a004 FETCH 7 body
* 7 FETCH (BODY ("TEXT" "PLAIN" ("CHARSET" "us-ascii") NIL NIL "7BIT" 11 1))
a004 OK Completed
a005 FETCH 7 BODY[TEXT]
* 7 FETCH (FLAGS (\Recent \Seen) BODY[TEXT] {11}
Voila ...
)
```



# Le système GNU/Linux

```
a005 OK Completed
a006 logout
* BYE LOGOUT received
a006 OK Completed
Connection closed by foreign host.
[rahan@morglum raahan]$
```

## 14.5 - Interaction du mail avec le DNS

L'envoi de courrier à **user@site** peut se faire de deux façons:

- la première méthode consiste à contacter **site** et de lui remettre le mail,
- la seconde consiste à délivrer le message à une machine qui assurerait un service de poste centrale pour un groupe de machine.

Si la première méthode est simple à l'esprit à mettre en oeuvre, il n'en est pas de même dans la réalité, car elle nécessite la maintenance d'un MTA sur chaque machine du réseau. C'est pour cela que dans la pratique, on utilise généralement la deuxième méthode.

C'est à ce niveau que le DNS intervient, car il permet de connaître ces machines postales. Il est en effet capable de stocker dans sa base de données ce type d'informations; il s'agit en l'occurrence des RRs<sup>1</sup> de type MX<sup>2</sup>.

Au lieu de contacter le DNS pour obtenir l'adresse IP de la machine destinatrice, on va le contacter pour lui demander le MX de la station destinatrice, puis l'adresse IP de cette dernière et c'est à lui que **sendmail** se connectera pour transmettre le message.

Dans la pratique, l'utilisation d'un MX présente plusieurs avantages :

- Il permet de joindre des machines en bordure de l'Internet en spécifiant la passerelle vers ces machines, par exemple une passerelle capable de communiquer en UUCP.
- Mettre le même MX pour toutes les stations d'un même domaine permet de centraliser la distribution du courrier. On peut ainsi fiabiliser la distribution en passant par une machine maintenue, administrée et à jour qui distribue les courriers à l'intérieur du campus. Le DNS, pour résumer grossièrement, ne contient que des enregistrements de données de type (*clé, valeur*). C'est ce qui permet ainsi d'envoyer du courrier à un domaine en mettant simplement dans la base le RR suivant :

```
domaine.fr          IN      MX      10      machine1
```

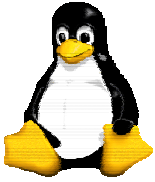
Le domaine *domaine.fr* a donc un MX et on peut donc utiliser l'adresse *(username)@domaine.fr* qui dirigera les courriers vers *machine.domaine.fr*.

- On peut indiquer plusieurs MXs dans le DNS, avec des priorités différentes, ce qui permet d'assurer la continuité du service du courrier électronique en cas de panne ( envoi des mails vers les MXs secondaires ) et aussi d'éviter l'inondation du MX principal lors de sa remise en marche puisque les mails en attente ne se présenteront pas tous successivement à lui mais au

---

<sup>1</sup> Enregistrement ( Resource Record ).

<sup>2</sup> Enregistrement précisant le relais pour le courrier électronique adressé à une machine ou à un domaine.



# Le système GNU/Linux

Version 9  
30/09/2003

contraire feront l'objet d'une seule connexion SMTP de la part des MXs de secours qui, selon le principe des MXs, retransmettent les mails au MX le plus prioritaire.

```
domaine.fr      IN    MX    10    machine1
domaine.fr      IN    MX    20    machine2
```

Même si un MX présente énormément d'intérêts, on peut néanmoins trouver quelques limitations. En effet, comme l'envoi de courrier passe automatiquement par le MX, cela peut être coûteux en termes de performances ( passages inutiles par des routeurs internes ) et le surcharger inutilement. On pourrait par exemple avoir la situation suivante où le MX se trouve derrière quelques routeurs :

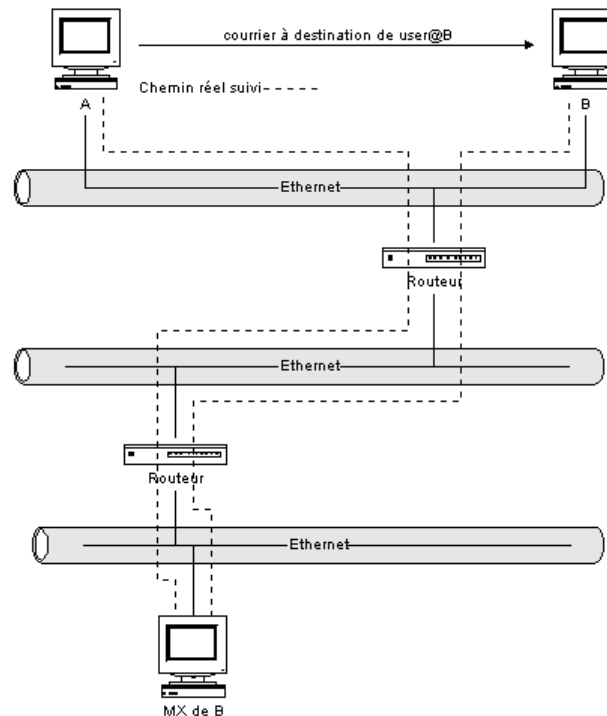


Figure 6

## 14.6 - Sendmail

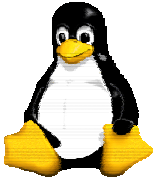
### 14.6.1 - Les versions

Le sendmail est le plus connu des agents de transfert, il a été conçu en 1982 par Éric ALLMAN. La version actuelle de sendmail est la version 8, dite V8 qui est la dernière version majeure après sendmail V5. Cette dernière outre le fait qu'elle présentait des trous de sécurité importants ne connaissait que le protocole SMTP. En revanche la version V8 supporte les protocoles SMTP et ESMTP ( Extended Simple Mail Transfert Protocol ) qui permet l'envoi de messages multimédia.

### 14.6.2 - Sendmail et la sécurité

Il y a de nombreuses raisons qui sont à l'origine des problèmes de sécurité avec le sendmail, notamment qu'il :

- s'exécute avec le SUID de root, donc si un utilisateur arrive à le "planter", il se retrouve avec les droits du root,
- soit écrit en C, qui est un langage très peu sécurisé,



# Le système GNU/Linux

- possède de nombreuses options.

Il est fortement conseillé d'utiliser la dernière version de sendmail, qui corrige tous les bugs et les trous de sécurité trouvés à ce jour.

## 14.6.3 - Appel de sendmail

Il existe différentes manières d'invoquer sendmail :

- par la commande mail ou tout autre MUA ( Mail User Agent ) comme mailx, elm...
- directement par un utilisateur en utilisant sendmail comme une commande :

```
> /usr/sbin/sendmail rahan@iut-amiens.fr
From : casimir
To : rahan@iut-amiens.fr
Subject : essai
```

```
Ceci est un essai
:
>
```

- par un agent de transport de messages : SMTP, UUCP, X400...
- par le démon sendmail lui-même qui va périodiquement vérifier si la file d'attente est vide et donc que tous les messages ont bien été envoyés.

## 14.6.4 - Les fichiers et répertoires de sendmail

Comme on l'a vu précédemment, les fichiers et les répertoires de sendmail dépendent de la version de sendmail utilisé, mais aussi du système. Nous nous restreindrons donc seulement à la version de sendmail  $\geq 8.9.X$  sur un système Linux.

### 14.6.4.1 - Les fichiers

- fichier sendmail                    fichier binaire exécutable
- fichier sendmail.cf                fichier de configuration
- fichier sendmail.hf                fichier d'aide lors du dialogue SMTP
- fichier sendmail.st                fichier de statistiques
- fichier aliases                    fichier contenant les alias en clair
- fichier aliases.db                version "compilée" des alias
- fichier .forward                  fichier permettant de renvoyer les courriers d'un utilisateur.

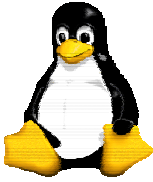
### 14.6.4.2 - Les répertoires

- répertoire mqueue                la file d'attente.
- répertoire mail                    ce répertoire contient les boites aux lettres.

### 14.6.4.3 - Le fichier « aliases »

Ce fichier permet de définir des listes de diffusion. Il est possible de diriger des courriers destinés à une adresse vers une autre adresse. Les alias permettent aussi de rediriger des mails sur des programmes. Ce fichier doit respecter la syntaxe suivante :

Format 1 : alias : adresse\_locale => L'utilisateur est sur la machine.  
sysadmin : root



# Le système GNU/Linux

Version 9  
30/09/2003

Format 2 : alias : adresse\_distante => L'utilisateur est sur une autre machine  
mdup : casimir@ileauxenfants.fr

Format 3 : alias : fichier => Il faut indiquer un nom absolu de fichier. Sendmail rajoute les messages dans le fichier. Le fichier doit exister, être accessible en lecture pour tout le monde. Il ne doit pas être exécutable.

phill : /var/spool/mail/phill

Format 4 : alias : "| ligne de comande" => Sendmail passe le message comme entrée standard de la commande.

Format 5 : alias::include:fichier => Le message est transmis à tous les utilisateurs indiqués dans le fichier. Ces listes de distribution peuvent être mises à jour par les utilisateurs. Le fichier d'alias permet aussi de faire des listes de diffusion en mettant plusieurs utilisateurs séparés par de virgules pour un alias :

superliste : rahan, drocourt@iut-amiens.fr

Un alias owner-superliste : rahan (facultatif) indique rahan comme étant responsable de la liste de diffusion superliste ; si un membre de cette liste ne peut pas être joint, rahan recevra un message d'erreur.

## Ligne trop longue

Si vous avez une ligne trop longue (~ 2000 caractères) dans le fichier /etc/aliases : (vous le savez en tapant "newaliases" qui vous répond "alias line too long")

### ✓ 1ere possibilité :

Décomposer la la ligne en plusieurs lignes :

fichier aliases :

grand-alias: petit-alias1, petit-alias2

petit-alias1: membre1,membre2...

petit-alias2: membre100,membre101...

### ✓ 2eme possibilité :

Découper le fichier aliases en plusieurs fichiers :

fichier aliases :

grand-alias: :include:/etc/mail/long-alias

fichier /etc/mail/long-alias :

membre1

membre2

...

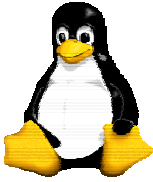
membre101

### ✓ 3eme possibilité :

Créer un script qui construit dynamiquement la liste des membres depuis le fichier group ou le fichier passwd => fichier aliases : mon\_group: "|un\_script mon\_group"

où un\_script prend le mail, créé la liste des destinataires et invoke sendmail.





# Le système GNU/Linux

Version 9  
30/09/2003

Ce script doit avoir une gestion des erreurs, et il doit aussi définir le champ "from" avec "-f"

## 14.6.4.4 - Le fichier « .forward »

Ce fichier permet lui aussi de rediriger les mails d'un utilisateur sur une autre boîte aux lettres, ou bien un programme. Celui-ci est propre à chaque utilisateur et se trouve dans le home directory de l'utilisateur. L'utilisation de ce fichier peut être la cause de problèmes, notamment des boucles de courrier :

L'utilisateur A renvoie son courrier à l'utilisateur B et l'utilisateur B renvoie son courrier à l'utilisateur A.

## 14.6.4.5 - La file d'attente

Comme tous les mails ne peuvent pas être délivrés immédiatement, sendmail les enverra plus tard, il doit donc les sauver dans un répertoire : la file d'attente. Ce répertoire contient les messages tant qu'ils n'ont pas été délivrés.

Un message est mis dans la file d'attente :

- Lorsque la machine destination n'est pas joignable ou en panne. Le courrier sera délivré quand la machine destination sera de nouveau en service.

- Quand un message est destiné à plusieurs destinataires, certains messages peuvent être délivrés d'autres non. Ceux qui ont échoués seront sauvés dans la file d'attente pour une remise différée.

- Lorsqu'un message est coûteux en terme de distance, le message sera sauvé dans la file d'attente, en attendant que le réseau soit moins chargé.

Le répertoire de file d'attente contient deux type de fichiers, les fichiers commençant par df ( data file ) qui contiennent les messages et les fichiers commençants par qf ( queue file ) qui contiennent les entêtes, mais aussi des informations concernant le traitement du message.

Il est possible de connaître les messages en cours d'attente grâce à la commande mailq, qui donne par exemple :

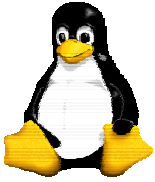
```
[root]# mailq
                               /var/spool/mqueue (1 request)
----Q-ID----- --Size--  -----Q-Time-----  -----Sender/Recipient-----
g01776110337      4136 Tue Jan  1 08:07 MAILER-DAEMON
                  (<newsletter@mail.megahardcoresex.com>... reply: read error)
                  <newsletter@mail.megahardcoresex.com>

[root]#
```

## 14.6.4.6 - Emplacement des fichiers et répertoires sur le système

Fichier ou répertoire	emplacement
Fichier sendmail	/usr/sbin/sendmail
Fichier de configuration	/etc/(mail)/sendmail.cf
Fichier de statistiques	/var/log/sendmail.st
Fichier des alias	/etc/(mail)/aliases
Fichier d'aide de SMTP	/usr/lib/sendmail.hf
Répertoire des fichiers de configuration	/etc/mail
Répertoire de la file d'attente	/var/spool/mqueue
Répertoire contenant les boîtes aux lettres	/var/spool/mail

Tableau 6



# Le système GNU/Linux

Version 9  
30/09/2003

La Figure 7 illustre l'intégration et la position de sendmail dans l'architecture de courrier d'une machine Unix.

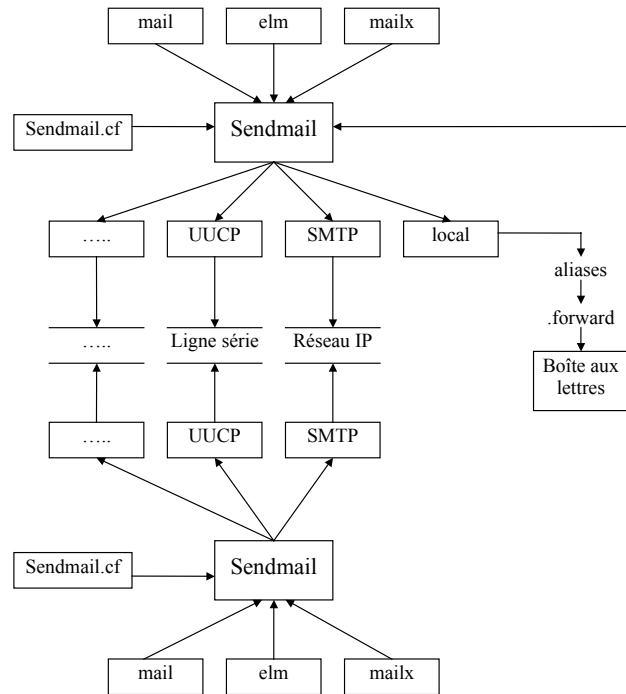


Figure 7

## 14.6.5 - Le fichier de configuration sendmail.cf

Ce fichier est un fichier texte qui permet de configurer le sendmail. Il se trouve généralement dans le répertoire `/etc` ou `/etc/mail`, mais cela peut varier selon le système. Ce fichier se décompose en plusieurs sections :

- Définition de macros,
- Définition de mailers,
- Règles de réécriture,
- Diverses options.

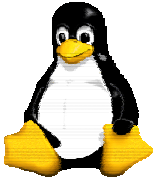
Le premier caractère de chaque ligne détermine la syntaxe de la ligne. Le caractère `#` introduit une ligne de commentaire. Les lignes vides sont autorisées. Le problème de ce fichier réside dans sa complexité, pour cette raison, on ne le modifie quasiment jamais mais on préfère passer par un fichier `m4` qui est un fichier traité par le pré-processeur `m4`, à partir duquel sera généré le fichier "sendmail.cf" final.

## 14.6.6 - Le fichier de configuration m4

Le but ici n'est pas d'expliquer le fonctionnement du pré-processeur `m4`, mais uniquement le contenu du fichier de configuration de sendmail dans ce format. Pour plus d'information on consultera la documentation de `m4` à l'adresse <http://www.gnu.org/software/m4/m4.html>.

Etant donné qu'un bon exemple vaut toujours mieux qu'un long discours, nous allons donc étudier un fichier de configuration `m4 type`.

```
include(`/usr/local/src/sendmail/cf/m4/cf.m4')
```



# Le système GNU/Linux

---

```
OSTYPE(`linux')
```

```
DOMAIN(generic)
GENERICS_DOMAIN(mondomaine.net)
```

```
dnl sendmail.cw contient les noms des domaines locaux
dnl par exemple :
dnl mondomaine.net
dnl mondomaine.fr
dnl autredomaine.com
dnl localhost
```

```
FEATURE(use_cw_file)
define(`confCW_FILE', `/etc/mail/sendmail.cw')
```

```
dnl G E N E R I C S T A B L E
dnl Mettre en place les tables
dnl
dnl Permet de changer un login local en login correspondant
dnl pour internet comme changer le from d'un mail
dnl userlocal          logininternet@domain.com
dnl exemple :
dnl rahan             drocourt@iut-amiens.fr
dnl
dnl Pour le mettre en place il faut le "compiler"
dnl makemap hash /etc/mail/genericstable.db </etc/mail/genericstable
```

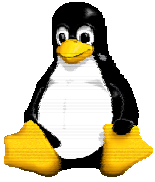
```
FEATURE(genericstable,`hash -o /etc/mail/genericstable')
```

```
dnl V I R T U S E R T A B L E
dnl
dnl Si la machine gere plusieurs domaines en local et les memes
dnl noms de login. Permet de repartir aux bonnes personnes
dnl en fonction du domaine
dnl help@mondomaine.net      help-societe1
dnl help@mondomaine.fr      help-societe2
dnl
dnl Pour le mettre en place il faut le "compiler"
dnl makemap hash /etc/mail/virtusertable.db </etc/mail/virtusertable
```

```
FEATURE(virtusertable,`hash -o /etc/mail/virtusertable')
```

```
dnl M A I L E R T A B L E
dnl
dnl Permet de repartir le courrier.
dnl
dnl Par exemple, si une passerelle recoit tous les mails d'une
dnl societe, et qu'elle doit le distribuer aux bons serveurs
dnl ensuite, on pourra mettre par exemple sur cette passerelle
dnl dans la mailertable :
dnl mondomaine.net          smtp:mail-net.domaineinterne.fr
dnl mondomaine.fr          smtp:mail-fr.domaineinterne.fr
dnl
dnl -----
dnl Pour le mettre en place il faut le "compiler"
dnl makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
```

```
FEATURE(mailertable, `hash -o /etc/mail/mailertable')
```



# Le système GNU/Linux

```
dnl Ajouter le domaine

FEATURE(nocanonify)dnl
FEATURE(always_add_domain)

dnl Me permet d'etre certain que l'enveloppe est bonne et affiche bien
dnl le bon nom de réseau

FEATURE(masquerade_entire_domain)dnl masque meme les sous domaines
FEATURE(allmasquerade)
FEATURE(masquerade_envelope)
FEATURE(limited_masquerade)
MASQUERADE_AS(mondomaine.net)
MASQUERADE_DOMAIN(mondomaine.net)
MASQUERADE_DOMAIN(mondomaine.fr)

dnl Utiliser les capacités de redirect et indiquer la localisation
dnl de procmail
dnl Redirect me permet de renvoyer un message en cas de changement de
dnl nom de boîte du destinataire
dnl Procmail permet aux users de trier leurs courriers

FEATURE(redirect)
FEATURE(local_procmail,"/usr/bin/procmail")
define(`PROCMail_MAILER_PATH',`/usr/bin/procmail')
define(`LOCAL_MAILER_PATH',`/usr/bin/procmail')

dnl Mettre les statistiques d'utilisation du courrier

define(`STATUS_FILE',`/var/log/sendmail.st')

dnl Utiliser l'id mail:mail pour sendmail (user:group).

define(`confDEF_USER_ID',`mail:mail')

dnl Definition du fichier d'alias, ici on en definit 2, aliases et aliases2

define(`ALIAS_FILE',`/etc/mail/aliases,/etc/mail/aliases2')

dnl Interdire les verifications (mesure anti-spam)

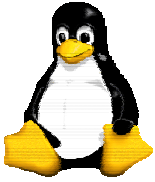
define(`confPRIVACY_FLAGS',`noreceipts,novrfy')

dnl SMART HOST : C'est sur cette machine que seront rediriges tous
dnl les mails sortant, a part ceux specifiés dans la mailertable
dnl ceci permet par exemple de centraliser l'envoi de mail
dnl a partir d'une unique machine.
dnl Utile aussi quand vous avez un fournisseur d'accès et vous
dnl voulez donc le laisser travailler a votre place.
dnl define(`SMART_HOST',`smtp:[smtp.dial.oleane.com]')dnl

define(`SMART_HOST',`smtp:[mailx.iut.fr]')

dnl taille maxi des messages

define(`confMAX_MESSAGE_SIZE',`5000000')
```



# Le système GNU/Linux

Version 9  
30/09/2003

```
dnl Parametre divers

define(`confCF_VERSION',` iut - 01/01/02')dnl version du fichier
define(`confSMTP_MAILER',`smtp8')dnl utiliser smtp en mode 8 bits
define(`confNO_RCPT_ACTION',`add-apparently-to')dnl Pour les pbs de spam

dnl normalement pas besoin
dnl define(`QUEUE_DIR',`/var/spool/mqueue')

MAILER(local)
MAILER(smtp)
MAILER(procmail)

dnl Activer le mode uucp ou non. INACTIF PAR DEFAULT

dnl MAILER(`uucp')

dnl FORCER LA PRESENCE D'UN CHAMP D'INFO POUR LE CAS OU UN USER FERAIT LE
dnl CON AVEC LE MAIL. (BIEN sur cela n'arrive jamais, en theorie ).

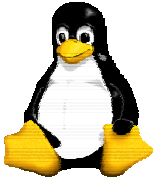
HX-ABUSE: write to abuse@mondomaine.fr

dnl Retirer les definitions des "choses" qui ne nous servent pas
undefine(`UUCP_RELAY')
undefine(`BITNET_RELAY')

dnl Reconstruit automatiquement les fichiers alias si besoin
define(`confAUTO_REBUILD')

dnl Utilisation du fichier /etc/mail/access pour filtrer

FEATURE(`access_db')
FEATURE(`blacklist_recipients')
```



# Le système GNU/Linux

## 15 - Index

/etc/services.....	56	MUA.....	77
adduser.....	22	Mysql.....	<i>Voir Apache</i>
Apache.....	53	named.cache.....	71
Bind.....	70	named.conf.....	71
CGI.....	<i>Voir Apache</i>	NFS.....	49
Clavier Français.....	10	NIS.....	50
conf.modules.....	41	Noyau Linux.....	36
Consoles virtuelles.....	30	nslookup.....	65
Courrier électronique.....	74	NYS.....	<i>Voir NIS</i>
crontab.....	23	Partitionnement du disque.....	8
Disquette de boot.....	20	PHP.....	<i>Voir Apache</i>
DNS.....	60	POP3.....	86
Contrôleurs.....	68	printcap.....	13
Hiérarchie.....	61	quota.....	24
zone.....	69	Recompiler le noyau.....	<i>Voir Noyau Linux</i>
Dosemu.....	46	resolv.conf.....	45
DSO.....	<i>Voir Apache</i>	restore.....	<i>Voir sauvegardes</i>
dump.....	<i>Voir sauvegardes</i>	rmmod.....	38
e2fsck.....	20	route.....	44
edquota.....	26	RPM.....	14
ESMTP.....	85	Samba.....	46
fdformat.....	20	sauvegardes.....	28
fstab.....	19	sendmail.....	91
hdparm.....	21	smb.conf.....	<i>Voir Samba</i>
hosts.allow.....	57	smbclient.....	48
hosts.deny.....	58	smbmount.....	48
httpd.conf.....	<i>Voir Apache</i>	SMTP.....	83
IMAP.....	88	stat.....	27
inetd.....	55	strace.....	27
inetd.conf.....	55	Swap.....	21
inittab.....	23	syslog.....	24
insmod.....	38	tar.....	<i>Voir sauvegardes</i>
kerneld.....	39	tcpd.....	57
LILO.....	11	tmpwatch.....	27
locate.....	27	whois.....	62
logrotate.....	24	Wine.....	46
lsmod.....	38	X Window.....	31
lsof.....	27	xdm.....	35
MDA.....	77	xdos.....	46
mke2fs.....	20	XF86Config.....	<i>Voir X Window</i>
Montage de partitions.....	19	xinetd.....	59
mount.....	19	xinetd.conf.....	59
MTA.....	77	yppbind.....	<i>Voir NIS</i>
mtools.....	45	yppasswd.....	<i>Voir NIS</i>