Printing-HOWTO

Grant Taylor <gtaylor+pht@picante.com>, traduction Jean-Michel VANSTEENE <J.M.Vansteene@frcl.bull.fr>, <vanstee@worldnet.fr>

12 Novembre 1997

Contents

1 Introduction

Le Printing-HOWTO devrait contenir tout ce dont vous avez besoin pour mettre en place les services d'impression sur votre système Linux. Comme vous vous en doutez peut-être, ceci est un tantinet plus compliqué que le "simple" *clic* du monde *Microsoft* et *Apple*, mais en contre-partie, c'est beaucoup plus souple et certainement plus facile à administrer pour les grands réseaux.

Ce document est organisé de façon à ce que la lecture de la première moitié suffise à la mise en place d'un système qui "tourne". Les choses plus obscures voire particulières sont concentrées dans la deuxième moitié. Vous pouvez vous référer à la table des matière pour les détails.

Depuis la version 3.x, ce document a été complètement réécrit et un certain nombre d'informations ont été retirées. La principale raison est que l'ancien document faisait une soixantaine de pages et contenait pas mal de parties redondantes voire inutiles. Ainsi fait, la sanction est tombée: vous trouverez ici un document sensiblement plus concis. S'il manque une information qui vous semble importante, nous vous encourageons soit à parcourir l'ancien HOWTO, soit à nous envoyer un courrier nous proposant d'ajouter ladite information. L'adresse du site de G. Taylor est `http://www.picante.com/~gtaylor/pht/`.

En vous souhaitant comme à l'accoutumée, moulte satisfaction dans vos recherches et pleine réussite dans la configuration de votre imprimante (Note : Désolé : ça vouloir dire nous espérer toi réussir configurer imprimante :-)).

Les HOWTOs français peuvent être trouvés en France, notamment sur les sites

```
ftp.lip6.fr            (/pub/linux/french/docs/HOWTO)
ftp.univ-angers.fr     (/pub/linux/french/docs/HOWTO)
```

1.1 Historique

Ce document est le troisième du nom, c'est à dire la troisième version complètement revue (en espérant que sa structure se stabilise un peu). L'histoire du PHT peut etre brièvement décrite ainsi :

J'ai commencé à écrire le premier HOWTO en réponse à un nombre assez conséquent de questions qui avaient été posées dans comp.os.linux. C'était en fait une FAQ que j'avais nommée HOWTO. Cette version était disponible uniquement en ASCII.

Le PHT a ensuite été regroupé avec la FAQ de LPD écrite par Brian McCauley <B.A.McCauley@bham.ac.uk>, nous avons continué ce document de concert pendant à peu près deux ans. Nous avons alors ajouté à notre document les travaux de Karl Auer <Karl.Auer@anu.edu.au>. Cette version du PHT était disponible en TeXInfo, PostScript, HTML, ASCII et Info.

Après avoir un peu laissé le PHT vieillir pendant une bonne année, sans avoir pu trouver quelqu'un souhaitant le maintenir, j'ai décidé de le réécrire. Cette version est conforme au "standard" Linuxdoc-SGML.

1.2 Copyright

Le coordinateur des HOWTOs est actuellement Greg Hankins <gregh@sunsite.unc.edu>. Veuillez quand meme ne pas encombrer sa boite aux lettres de questions sauf de la plus haute importance. En France, le coordinateur est Eric Dumas <dumas@freenix.fr, dumas@Linux.EU.Org>.

2 Qu'est-ce que l'impression sous UNIX ?

Je me permets de consacrer un petit chapitre au B.A.BA de l'impression sous UNIX et donc sous Linux. Les gourous trouveront ce chapitre inutile et je les invite à passer au suivant. Les débutants trouveront l'information suffisante, je l'espère, pour comprendre comment les données sont véhiculées vers leur chère imprimante.

La façon la plus simple d'imprimer sous Unix (et donc Linux) est d'envoyer les données directement au pilote de l'imprimante.

```
ls >/dev/lp0
```

Cette méthode a néanmoins un gros inconvénient : elle ne tire pas parti des caractéristiques multi-tâches du système d'exploitation. En effet vous devrez attendre que le tampon de l'imprimante ait absorbé toutes les données pour pouvoir continuer à travailler.

Une meilleure méthode consiste à utiliser un spool d'impression, dont le rôle est de collecter les données temporairement dans des fichiers afin de les envoyer, en tâche de fond, à l'imprimante. Lorsque plusieurs fichiers sont soumis, ils seront imprimés dans l'ordre de soumission (premier entré, premier sorti). La zone de spool est donc bien une **file**. On dit que les travaux sont dans la **file d'impression**. Cette méthode met en jeu deux acteurs : le client (*lpr*) permet à tout utilisateur de soumettre des travaux d'impression. Le serveur quant à lui (*lpd*) les prend en charge en tâche de fond. Il récupère les données dans le spool ainsi que des informations qui lui sont nécessaires pour l'impression proprement dite.

Nous verrons dans un chapître un peu plus loin comment tout cela interagit. Arrêtons-nous là pour le moment et penchons-nous déjà sur le plus important: les périphériques du système.

3 Comment Imprimer

Si le démon d'impression de votre machine est correctement configuré, vous pouvez lire le *Printing-Usage HOWTO*

<`http://www.loria.fr/services/linux/HOWTOFRENCH/Printing-Usage-HOWTO/Printing-Usage-HOWTO.html`

qui traite plus particulièrement de la commande `lpr` et de la manipulation des files d'impression.

Si, par contre, vous venez d'installez un nouveau système (un qui marche, quoi...) vous souhaitez certainement configurer les services d'impression. Restez sur cette chaîne :-) ne zappez pas et allons-y.

3.1 Les périphériques d'imprimantes dans le système

Les imprimantes sont souvent rattachées (surtout les imprimantes personnelles) au port parallèle. Elles sont gérées par les périphériques de type `/dev/lp?`.

Si vous disposez d'une imprimante série, vous utiliserez bien sûr `/dev/ttyS?` (ou `/dev/ttys?`), et non pas `/dev/lp?` ni `/dev/cua?`. Le majeur de `/dev/ttyS?` est 4, celui de `/dev/cua?` est 5.

3.2 Le périphérique lp

3.2.1 Noyaux antérieurs à 2.1.32

Si vous utilisez une imprimante parallèle, vous devez disposer d'un noyau Linux compilé avec les gestionnaires de périphériques d'impression `lp`. Verifiez-le à l'aide de la commande

```
cat /proc/devices
```

qui doit vous montrer un ou plusieurs périphériques lp. De plus le répertoire `/dev` doit comporter la liste des périphériques `/dev/lp0`, `/dev/lp1`, ...

Sur un système de type XT, `LPT1:` correspond à `/dev/lp0` (majeur 6, mineur 0), alors que sur un AT, LPT1: correspond à `/dev/lp1` (majeur 6, mineur 1).

```
Nom     Majeur  Mineur  Port
lp0     6       0       0x3bc
lp1     6       1       0x378
lp2     6       2       0x278
```

Le port parallèle peut être géré par le pilote du noyau de deux manières. D'une part par polling (scrutation), d'autre part par interruption. La méthode de gestion par interruption est en théorie plus efficace, puisque le gestionnaire n'est sollicité par une interruption qu'au moment où un travail est à effectuer. En pratique, il semble que ce soit dépendant de la machine. Dans beaucoup de cas, il n'y a pas tant de différence que cela.

Un certain nombre d'utilisateurs se sont plaints que leur port parallèle bidirectionnel n'était pas détecté lorsqu'ils utilisent un vieux cable unidirectionnel. Vérifiez que le vôtre est récent.

Vous ne pourrez pas utiliser les pilotes `plip` et `lp` en même temps. Vous pouvez cependant charger l'un ou l'autre soit manuellement, soit par `kerneld` version 2.x (et noyaux au moins 1.3.x). En configurant les interruptions avec attention, vous pourrez théoriquement utilisez plip sur un port et lp sur l'autre. Quelqu'un l'a fait en modifiant les pilotes... J'attends avec impatience que quelqu'un me dise comment faire à partir d'une simple et ingénieuse commande.

Un petit utilitaire appelé `tunelp` vous permet, sous le compte super-utilisateur, de régler les interruptions d'un périphérique `lp`, le mode de fonctionnement du pilote, la fréquence de scrutation, et plein d'autres options...

Avec LILO et LOADLIN, vous pouvez configurer les adresses et interruptions utilisés par le pilote.

```
Syntaxe:      lp=port0[,irq0[,port1[,irq1[,port2[,irq2]]]]]

Par exemple:   lp=0x378,0    ou    lp=0x278,5,0x378,7 **
```

(**) le port `lp0` en `0x3bc` n'est plus utilisé sur les PC récents.

Si cette caractéristique est utilisée, vous devez spécifier tous les ports que vous souhaitez utiliser, il n'y a pas de valeurs par défaut. Vous pouvez invalider un pilote en indiquant `lp=0`.

Lorsque le pilote est chargé en tant que module (version noyau 2 et 1.3.x), il est également possible de spécifier les lignes d'interruption utilisées soit dans le fichier `/etc/conf.modules`, soit sur la ligne de commande de insmod en utilisant la même syntaxe. Les paramètres sont `io=port0,port1,port2` et `irq=irq0,irq1,irq2`. Lisez les mages de manuel de `insmod` pour plus d'information.

Le code source du pilote de port parallèle se trouve dans `/usr/src/linux/drivers/char/lp.c`.

3.2.2 Noyaux postérieurs à 2.1.32: le périphérique parport.

Depuis le noyau 2.1.33 (il existe un patch pour la version 2.0.30), le périphérique lp est simplement un client du nouveau périphérique `parport`. L'ajout de ce `parport` corrige un certain nombre de problèmes dont était affublé `lp` - il peut partager le port avec d'autres pilotes, il met dynamiquement en relation les ports parallèles disponibles avec les numéros de périphériques plutôt que de forcer une correspondance statique entre addresse d'entré/sortie et numéro de port, ...

Une prochaine version de ce document devrait couvrir le `parport` lorsque j'en aurai utilisé un, mais en attendant, vous pouvez lire le fichier `Documentation/parport.txt` dans les sources du noyau, ou aller sur le site *parport* à `http://www.cyberelk.demon.co.uk/parport.html`

3.3 Les périphériques série

Les périphériques série utilisables sont appelés `/dev/ttyS?` (les périphériques `/dev/cua?` sont aussi série mais pas utilisables dans ce domaine). L'utilitaire `stty` permet de visualiser ou de modifier les caractértiques d'un port donné. `setserial` permet de controler et configurer les IRQ et adresses d'entrée/sortie si besoin est. Référez-vous au *Serial-HOWTO* pour de plus amples renseignements.

Si vous utilisez une imprimante série lente avec le contrôle de flux, vous constaterez peut-être que certaines impressions sont tronquées. Ceci est certainement dû au port série dont le comportement par défaut est de purger son tampon des caractères non transmis dans les 30 secondes après sa fermeture. Une imprimante trop lente n'aura pas le temps de le vider. Le tampon peut contenir 4096 caractères.

Si la commande `cat file > /dev/ttyS2` produit une sortie correcte pour les fichiers courts et tronquée pour les longs fichiers, vous êtes peut-être dans ce cas.

Le délai de 30 secondes peut être ajusté à l'aide du paramètre "closing_wait" (attente fermeture) de la commande `setserial` (version 2.12 et ultérieure).

Notez que les ports série sont généralement configurés au démarrage de la machine grace à un appel à `setserial` dans le fichier `/etc/rc.d/rc.serial`. Vous pouvez rajouter toute option nécessaire dans ce fichier.

4 Les imprimantes supportées

Le noyau Linux supporte quasiment toutes les imprimantes que vous pouvez physiquement connecter au port série ou parallèle. Il y a néanmoins des petites choses à savoir, notamment certaines imprimantes à éviter bien qu'elles puissent (électriquement parlant) communiquer avec Linux. En premier lieu, il existe toute une génération d'imprimantes incompatibles s'appuyant sur le "Windows Printing System" et qui répondent au label "pour Windows". Ces imprimantes ne fonctionnent pas avec Linux. Elles font travailler l'unité centrale pour des tâches normalement laissées au microprocesseur de l'imprimante. Malheureusement ces tâches ne peuvent etre effectuées que par le pilote du constructeur qui ne tourne que sous Windows. Conclusion n'achetez pas ce genre d'imprimante pour Linux.

Evidemment, sous Linux, comme d'ailleurs sous tout autre système, le meilleur choix est d'avoir une imprimante PostScript. Presquer tous les logiciels Unix produisent du *PostScript* et bien évidemment le mieux est d'avoir une imprimante qui le comprend. Cela simplifie. Malheureusement *PostScript* n'est pratiquement pas disponible en dehors du domaine des imprimantes laser. Et c'est cher. Rassurez-vous, vous pouvez utiliser toute autre imprimante.

Si vous ne comptez pas avoir de PostScript à imprimer, vous pouvez connecter une simple imprimante matricielle retrouvée dans votre grenier. Sinon, comme nous le verrons, il faudra passer par un logiciel d'interprétation du *PostScript* (le plus célèbre est le logiciel gratuit *GhostScript* de *Alladin software*). Utilisez alors une imprimante reconnue par ce logiciel.

Voir `http://www.cs.wisc.edu/~ghost/printer.html` pour des informations mises à jour selon les versions disponibles et les pilotes en test.

Les imprimantes supportées sont:

```
Canon BubbleJet BJ10e
Canon BubbleJet BJ200
Canon BubbleJet BJC-210 (4.01) N/B seulement
Canon BubbleJet BJC-240 (3.33, 4.03) N/B seulement
Canon BubbleJet BJC-600
Canon BubbleJet BJC-610 (3.53) 360dpi seulement, N/B & couleur. Voir correctif bjc610.
```

```
Canon BubbleJet BJC-4000
Canon BubbleJet BJC-4100 (4.01) pas couleur.
Canon MultiPASS C2500
Canon BJC-240 (5.01)
Canon BJC-70 (5.01)
Canon BubbleJet BJC-800

HP DeskJet (3.33)
HP DeskJet Plus (3.33)
HP DeskJet 500 (3.53)
HP DeskJet Portable (4.01)
HP DeskJet 400 (3.33, 4.03) N/B teste seulement.
HP DeskJet 500C (3.53)
HP DeskJet 540C (3.53)
HP DeskJet 690C (4.03) 1bit/pixel et 32bit/pixel
HP DeskJet 693C (4.03)
HP DeskJet 550C (3.53)
HP DeskJet 560C (3.53)
HP DeskJet 600 (3.53) N/B teste seulement (1bit/pixel or 32bit/pixel)
HP DeskJet 660C (3.53)
HP DeskJet 682C (4.01) Utiliser gamma=0.3
HP DeskJet 683C (3.33, 4.03)
HP DeskJet 693C (4.03) 24bit/pixel
HP DeskJet 850 (3.53) 300dpi
HP DeskJet 870Cse (4.03) (16 ou 32 bits/pixel)
HP DeskJet 850
HP DeskJet 870Cse (4.03)
HP DeskJet 870Cxi (4.03)
HP DeskJet 680 (5.01)
HP DeskJet 500C (3.53)
HP DeskJet 500C (3.53)
HP DeskJet 510 (3.53)
HP DeskJet 520 (3.53)
HP DeskJet 540C (3.53)
HP DeskJet 693C (4.03)
HP DeskJet 600 (3.53)
HP DeskJet 600 (3.53) marges incorrectes
HP DeskJet 870Cse (4.03)
HP LaserJet 5 (4.01) 300dpi ou 600dpi
HP LaserJet 5L (4.03) 300dpi ou 600dpi Marges incorrectes dans GS 3.33. Correct dans GS 4.03.
Oki OL410ex LED printer (4.03) 300dpi ou 600dpi
HP PaintJet XL300
HP DeskJet 600 (3.53) 300dpi OK, 600 dpi tres lent
HP DeskJet 1200C (3.53)
HP DeskJet 1600C (4.03) 24bit/pixel. -dShingling=2 -dDepletion=1 -dPrintQuality=1
Ricoh 4081 laser printer (3.53)
Ricoh 6000 laser printer (3.53)
Epson Stylus Color (3.53)
Epson Stylus Color II (3.53) -r360 -dMicroweave voir devices.txt et GS stcolor FAQ
Epson Stylus 500 (4.03) -r360 or -r720, -dMicroweave voir les notes
                        d’Alan Williams sur GS. (Ne marche pas avec 3.33.)
Epson Stylus 800 (3.53) -sModel=st800
```

5 Quel démon d'impression ?

Jusque récemment, le choix sous Linux était simple puisque tout le monde disposait du seul démon lpd sorti tout droit et presque tel quel du code BSD Net-2. Aujourd'hui la plupart des fournisseurs proposent ce logiciel. Mais les choses sont en train de changer. Les systèmes SVR4 tels que Sun Solaris sont fournis avec un paquetage logiciel d'impression centré sur `lpsched`. D'autre part, sous Linux, quelques fournisseurs proposent `LPRng`, une implémentation beaucoup plus récente et disponible gratuitement. `LPRng` est très facilement administrable notamment sur des plate-formes importantes et contient du code plus robuste (moins farfelu??) que `lpd`.

A l'heure actuelle malgré toutes ces nouveautés, `lpd` est certainement ce qui convient le mieux à la plupart des utilisateurs. Même si ce n'est pas le fin du fin, il tourne bien une fois configuré et, chose importante, il est très bien documenté dans les livres sur Unix.

Pour plus d'information sur `LPRng`, allez voir sur `http://ltpwww.gsfc.nasa.gov/ltpcf/about/unix/Depotdoc/LPRng/`

6 L'impression, comment ça marche

Nous considérons ici que vous utilisez la suite logicielle `lpd` que nous décrirons d'ailleurs. Elle est la plus répandue et fonctionne très bien.

6.1 Impression locale et impression distante

L'impression locale permet aux utilisateurs d'envoyer des travaux d'impression à l'imprimante directement rattachée à leur machine.

L'impression distante, par contre, permet de soumettre des travaux d'impression depuis une machine, à une autre machine sur le réseau, sur laquelle est connectée une imprimante.

6.2 De quoi avez vous besoin

Nous supposons que vous savez éditer un fichier texte sous Linux et que vous avez une bonne compréhension des notions de droits d'accès et de propriété (`chmod`, `chown`).

Nous supposons également que votre système Linux fonctionne correctement. En particulier, si vous souhaitez faire de l'impression distante, que votre réseau fonctionne déjà.

Consultez à ce propos les nombreuses documentations disponibles sur le sujet (comme on dit: RTFM, ce qui en bon francais veut dire LLBD `:-)`).

6.3 Les programmes importants

Le système d'impression Unix comprend (au moins) 5 programmes. Ils doivent se trouver à l'endroit décrit (c'est le mieux) ou dans un répertoire accessible (avec la variable PATH), appartenir à root (groupe lp), et avoir les permissions suivantes :

```
-r-sr-sr-x     root    lp     /usr/bin/lpr
-r-sr-sr-x     root    lp     /usr/bin/lpq
-r-sr-sr-x     root    lp     /usr/bin/lprm
-r-xr-sr-x     root    lp     /usr/sbin/lpc
-rwxr--r--     root    lp     /usr/sbin/lpd
```

Les quatre premiers sont utilisés pour soumettre, visualiser, annuler, contrôler les travaux d'impression. Le dernier est le démon.

Il existe bien entendu des pages de manuel en ligne pour ces commandes que vous pourrez consulter pour plus d'information. Le point important à noter est que les commandes `lpr`, `lpq`, `lpc` et `lprm` opèrent sur une imprimante par défaut nommée `lp`. La variable d'environnement `PRINTER` peut contenir le nom de l'imprimante que vous avez choisie. La spécification du nom d'une imprimante sur la ligne de commande surchargera ces définitions (les imprimantes de l'exemple sont hors-ligne):

```
# echo $PRINTER
                           (vide)
#
# lpq
waiting for lp to become ready (offline ?)
...
# export PRINTER=mon_imprimante
# lpq
waiting for mon_imprimante to become ready (offline ?)
...

# lpq -Plpr0
waiting for lpr0 to become ready (offline ?)
...
```

6.3.1 Le client et le serveur

Voici ci-dessous les interactions client - démon

```
                                              ---------
                                            +/BlaBla  /+
                                           //_______//|
                                          /          / +
+----------------+                       +----------+ /
|      LPR       |                       |=      oo |/
+----------------+                       +----------+
        |                                 Imprimante
        |                                     ^
        V                                     |
+----------------+                  +------------------+
|      LPD       |------>------>----|        LPD       |
+----------------+                  +------------------+

            Soumission d'un requete d'impression


+----------------+
|      LPQ       |
+----------------+
                  \
                   \_______>______
                                  \
+----------------+                 \+------------------+
|      LPD       |                  |        LPD       |
+----------------+                  +------------------+
```

```
Soumission d'une demande d'information
```

Lorsque le système démarre, `lpd` est chargé. Il lit le fichier `/etc/printcap` (dont vous trouverez une explication plus loin) qui décrit les imprimantes connues.

6.3.2 Les fichiers soumis par le client

Lorsqu'un programme client soumet un travail d'impression, il génère deux fichiers qu'il écrit dans le spool:

- Un fichier de données qui contient une copie des données que vous souhaitez soumettre à l'imprimante. Il s'agit bien d'une copie, ce qui signifie que toute modification ultérieure de votre fichier n'altérera pas l'impression.
- Un fichier de description du travail à effectuer.

6.3.3 La commande lpr

La commande `lpr` soumet un travail d'impression. Elle se charge de mettre les données à imprimer dans un fichier dans le spool d'impression. Ces données peuvent provenir soit d'un fichier (les données sont dupliquées et toute modification ultérieure du fichier d'origine n'affectera pas l'impression), soit de l'entrée standard (stdin). Le démon est averti de l'existence d'un nouveau fichier et envoie, dès que possible, les données vers l'imprimante physique (ou la machine distante).

La taille du spool est bien entendu limitée à la place disponible sur votre disque dans `/usr/spool/` ou à la taille limite spécifiée dans le fichier de configuration `printcap`. Vous pouvez néanmoins imprimer un gros fichier en demandant à `lpr` de ne pas dupliquer le fichier. Le démon d'impression ira alors chercher le fichier que vous indiquez dans la ligne de commande et non plus dans le spool.

6.3.4 La commande lpq

La commande `lpq` affiche le contenu du spool, pour une imprimante donnée. Une des informations importantes fournies est le numéro du travail (job). C'est lui qui pourra servir à annuler un des travaux soumis, y compris celui en cours d'impression. Parmi tous les travaux soumis, l'indication "active" indique le travail en cours d'impression (ou que `lpd` essaie d'envoyer à l'impression).

6.3.5 La commande lprm

La commande `lprm` enlève un travail de la file (et donc le fichier du spool). Vous pouvez soit spécifier un numéro de job, soit un tiret permettant de supprimer tous les travaux vous appartenant. Si vous êtes root, tous les travaux sont supprimés. Pour supprimer les travaux d'un utilisateur, spécifiez son nom.

```
# lprm 1
dfA001Aa00484 dequeued
cfA001Aa00484 dequeued
#
```

Le premier fichier contient les données à imprimer. Il a été créé par `lpr`. Le deuxième contient des informations que le démon utilise pour savoir que faire des données (impression locale, distante, ...) Consultez le manuel en ligne : `lpd(8)`.

6.3.6 La commande lpc

La commande `lpc` permet de contrôler les travaux en cours ainsi que l'imprimante, et certains aspects de son utilisation. En particulier, vous pouvez démarrer ou stopper la sortie des travaux du spool pour l'impression, valider ou invalider une imprimante, et même modifier l'ordre d'impression des fichiers. Les commandes suivantes permettent d'invalider l'impression sur `mon_imprimante`, de valider le spool sur `ton_imprimante`, et de faire passer le job 37 en début de file:

```
lpc down mon_imprimante
lpc enable ton_imprimante
lpc topq 37
```

`lpc` peut fonctionner en interactif si aucun paramètre ne lui est passé. Vous pouvez lire les pages du manuel en ligne pour obtenir des instructions complètes. A noter que certaines actions de `lpc` sont réservées au super-utilisateur (root).

6.4 Les répertoires importants

Le répertoire le plus important est le répertoire de spool, dans lequel les données vont être stockées avant d'être imprimées. Typiquement, un système sera configuré pour avoir un répertoire de spool par imprimante. Cela rend la gestion plus facile. Sur mon système, par exemple, le répertoire `/usr/spool/lp` est le répertoire principal. Sous ce répertoire, on y trouve le sous-répertoire `lpr0`, correspondant à la déclaration que j'ai faite dans `/etc/printcap` pour le répertoire de spool de mon imprimante.

NDT: Ce qui va suivre décrit une façon de faire pour donner les bons droits d'accès aux répertoires de spool. Différentes méthodes sont possibles, sachant que, comme sous Un*x, beaucoup de choses sont possibles dans ce domaine, il convient de faire attention de ne pas offrir de failles à la sécurité de l'ensemble.

Le répertoire de spool doit appartenir à `root`, et au groupe `lp`, avec les droits de lecture/d'écriture pour utilisateur et groupe, et lecture seule pour le reste du monde.

```
chmod ug=rwx,o=rx lpr0
chgrp lp lpr0
```

```
drwxrwxr-x   2 root     lp          1024 Feb 11 10:51 lpr0/
```

Un autre répertoire doit également être présent : `/usr/spool/lpd` avec les mêmes droits. Vous aurez plus d'informations plus avant dans ce document.

6.5 Les fichiers importants

En dehors des programmes que nous avons déjà évoqués précédemment, quatre fichiers doivent se trouver dans chaque répertoire de spool. Avec les versions récentes des gestionnaires d'impression, vous n'avez pas à vous soucier de ces fichiers. Ils sont créés automatiquement s'ils n'existent pas : `.seq`, `errs`, `lock` et `status`. Ces fichiers doivent avoir les droits `-rw-rw-r-`. Le fichier `.seq` contient un compteur pour l'affectation des numéros de jobs. Le fichier `status` contient le message devant être émis par la commande `lpc stat`. Le fichier `lock` est utilisé par le démon pour qu'il n'imprime qu'un fichier à la fois. Le fichier `errs` contient les erreurs survenues sur l'imprimante.

Le fichier `errs` n'est pas obligatoire. De plus, il peut s'appeler comme vous le souhaitez, pourvu que son nom soit déclaré dans le fichier `/etc/printcap` que nous décrirons dans la suite.

7 Configurer les services d'impression

La configuration minimale pour un système d'impression permet de mettre des fichiers en file d'impression puis de les imprimer. Il ne prête aucune attention au fait que votre imprimante puisse les imprimer (les comprendre même) et ne vous permettra pas d'imprimer des choses extraordinaires. Néanmoins, c'est un premier pas.

7.1 Le fichier `/etc/printcap`

Le fichier `/etc/printcap` décrit toutes les imprimantes que votre système doit connaître. Il peut être modifié avec votre éditeur préféré, doit appartenir à root et avoir les droits suivants:

```
-rw-r--r--   1 root     system        164 Oct 25 21:23 /etc/printcap
```

Le contenu du fichier semble assez incompréhensible à première vue. Il respecte effectivement une syntaxe particulière et malgré les apparences, assez simple lorsque l'on connaît ! Il n'y a pas toujours de manuels concernant ce fichier, et cela complique un peu les choses. Un petit conseil en passant : essayez, dans la mesure du possible de rendre votre fichier le plus lisible possible, avec des commentaires. Vous pouvez consulter les pages du manuel en ligne concernant `printcap(5)` (ou empressez-vous de les récupérer si vous ne les avez pas). Plus loin sont décrits les paramètres importants.

Une entrée de `printcap` décrit une imprimante, c'est-à-dire une correspondance nom logique - imprimante physique, puis décrit la façon de transmettre les données. Par exemple, une entrée va décrire le périphérique physique à utiliser, le répertoire de spool, les traitements à effectuer sur les données avant impression, ou encore le répertoire dans lequel seront notifiées les erreurs. Vous pouvez aussi limiter la quantité de données pour un job, ou même limiter l'accès d'une imprimante à une classe d'utilisateurs. Vous trouverez dans la partie suivante la description des champs.

Il est tout à fait possible d'avoir plusieurs entrées décrivant différentes façons d'envoyer des données à une même imprimante physique. Par exemple, une imprimante physique peut supporter les formats HP LaserJet et PostScript, en fonction de la séquence de caractères envoyée au début d'un travail. Vous définirez donc deux entrées, l'une permettant de traiter le format HP, l'autre le format *PostScript*. Les programmes générant des données "HP" les enverront à l'imprimante HP, ceux générant des données *PostScript* les enverront à l'imprimante *PostScript*. Toutes les deux représentent la même imprimante physique.

Les programmes qui modifient les données avant de les envoyer à l'imprimante physique sont des filtres.

Exemple d'entrée d'un fichier `/etc/printcap`:

```
# LOCAL djet500
lp|dj|deskjet:\
        :sd=/var/spool/lpd/dj:\
        :mx#0:\
        :lp=/dev/lp0:\
        :sh:
```

Ceci définit une imprimante dont les noms sont `lp` (par défaut), `dj` et `deskjet`. Les deux derniers sont des alias de la même imprimante. La file (on dit très souvent le spool, même en français...) de cette imprimante de trouve dans le répertoire `/var/spool/lpd/dj` (`sd` signifie spool directory). Le périphérique utilisé est `/dev/lp0`. La page d'en-tête est supprimée (`sh`) et aucune limite de taille de fichier n'est fixée (`mx`)

Notez que la même entrée pourrait s'écrire:

```
lp|dj|deskjet:sd=/var/spool/lpd/dj:mx#0:lp=/dev/lp0:sh:
```

mais c'est moins beau.

Vous pouvez consultez la page de manuel de `printcap` sur

`http://www.picante.com/~gtaylor/pht/man/printcap.html`. Toutes les options y sont décrites.

Tous les champs exceptés les noms d'imprimantes sont entourés de deux-points et repérés par un symbole de deux lettres suivi du signe égal. Ensuite est indiquée la valeur qui peut être de type numérique, booléenne ou chaîne de caractères:

```
champ          type           signification

lp             string         designe le peripherique d'impression
sd             string         designe le repertoire de spool
lf             string         designe le fichier de rapport d'erreurs
if             string         specifie le nom du filtre d'entree
rm             string         designe le nom d'un site d'impression distant
rp             string         designe le nom d'une imprimante distante
sh             booleen        indique s'il faut supprimer les en-tetes
sf             booleen        indique s'il faut supprimer les sauts de pages
                              de fin de travaux
mx             numerique      indique la taille maximum d'un job
                              (en blocs = 1Ko sous linux)
```

7.1.1 Détails sur le champ lp

Si vous spécifiez `/dev/null` comme périphérique, tous les traitements se feront, mais tout partira à la poubelle. Ca semble ridicule, mais cela vous permet par exemple de tester une configuration. Lisez le chapitre "Imprimantes qui ne sont pas de simples périphériques". Si vous désignez une imprimante distante avec `rp` et `rm`, `lp` doit contenir `:lp=:`.

Ne laissez pas ce champ vide en cas d'impression locale, le démon signalerait une erreur.

7.1.2 Détails sur le champ lf

Tout fichier spécifié ici doit exister, sinon le rapport d'erreurs ne se ferait pas.

7.1.3 Détails sur le champ if

Les filtres d'entrée sont des utilitaires transformant les données qu'il reçoivent sur leur entrée standard en un format particulier qu'il sortent sur leur sortie standard. Typiquement, la conversion texte - *PostScript* déjà mentionnée.

Si vous spécifiez un filtre d'entrée, le démon n'envoie pas directement les données au périphérique. Il exécute le filtre en dirigeant les donnés sur son entrée standard et en désignant le périphérique de sortie comme sortie standard.

7.1.4 Détails sur les champs rm et rp

Envoyer des données à une imprimante rattachée à une machine distante est très simple: il suffit de spécifier le nom de la machine avec `rm` et le nom de l'imprimante avec `rp`. S'assurer que l'entrée `lp` est vide. A noter que les données seront d'abord mises dans le spool local avant d'être transférées. Même si votre imprimante est distante, il faudra également un spool local.

7.1.5 Détails sur les champs sh et sf

Les bannières concernent éventuellement les utilisations à plusieurs personnes. Elles identifient les jobs.

La suppression de ces bannières vous permet d'économiser du papier. Par contre la gestion des sauts de page sera plus intéressante, surtout si vous utilisez des traitements de textes qui formatent toujours des pages pleines. Dans ce cas, pas besoin de saut de page supplémentaire. Vous auriez sinon une page blanche en fin de chaque travail. Si vous utilisez des listings ou autres documents, ajouter un saut de page garantit que chaque travail commancera bien en début de page.

7.1.6 Détail sur le champ mx

Ce champ permet de limiter la taille des données pour chaque job. Le nombre à spécifier est en blocs de `BUFSIZE` (pardon, de 1 Ko) sous Linux. La valeur 0 rend la taille illimitée, permettant la soumission de travaux limitée uniquement à la taille du disque. Notez que la limite concerne la taille des données mises en spool, et non pas les données envoyées à l'imprimante physique. Si la limite est dépassée, le fichier est tronqué avec l'émission d'un message disant: `lpr: <fichier>: copy file is too large`.

Cela peut être intéressant pour des imprimantes physiques en mode texte, notamment si des utilisateurs ou des programmes créent accidentellement des données trop volumineuses.

Si vous manquez de mémoire de masse, pourquoi n'inventeriez-vous pas un filtre qui décompresse ce qu'il a à envoyer à l'imprimante ? Vous soumettriez alors des données compressées.

8 Les filtres

Si avec les explications précédentes, tout marche c'est formidable, mais en règle général, cela ne suffit pas. Regardez ci-dessous ce que l'on obtient sur la DeskJet 500, lorsque j'envoie un fichier texte:

```
Ceci est la premiere ligne.
                           Celle-ci est la deuxieme.
                                                    Voici la troisieme.
```

Et alors l'impression d'un fichier *PostScript*... Vous obtenez le listing complet du code *PostScript* avec les même effets d'escalier. Une horreur.

Il faut donc quelque chose de plus et c'est le rôle des filtres. Les plus observateurs d'entre vous auront peut-être remarqué l'existence des paramètres `if` (input filter) et `of` (output filter) dans le fichier printcap. Pour l'instant nous avons besoin de `if`.

Un filtre est un simple programme exécutable qui lit les données sur son entrée standard et sort le résultat sur sa sortie standard.

Commençons par écrire un script que vous appellerez `filtre` et qui ajoute des retours chariot avant chaque caractère fin de ligne. Ceci élimine l'effet d'escalier.

```
#!/usr/local/bin/perl
# La ligne ci-dessous doit contenir le chemin complet vers perl
# Ce script doit etre executable: chmod 755 filtre
while(<STDIN>){chop $_; print "$_\r\n";};
# Vous pouvez aussi vouloir terminer avec une fin de page: print "\f";
```

Dans `/etc/printcap`, l'entrée est modifiée en conséquence:

```
lp|dj|deskjet:\
             :sd=/var/spool/lpd/dj:\
             :mx#0:\
             :lp=/dev/lp0:\
             :if=/var/spool/lpd/dj/filtre:\
             :sh:
```

Essayez d'écrire le filtre en shell qui sera plus efficace que de charger `perl`. Bon allez, je vous aide un peu:

```
#!/bin/sh
if [ "$1" = -c ]; then
  cat
else
  sed -e s/$/^M/
fi
# echo -ne suppose que /bin/sh correspond a bash
echo -ne \\f
```

Notez que '^M' symbolise le caractère retour-chariot et non pas un '^' suivi d'un 'M'. Dans emacs, pour saisir ce caractère, entrez la séquence `C-q C-m`, alors que sous vi, entrez `C-v C-m`. Le test de $1 permet d'invalider l'insertion du retour-chariot par la commande `lpr -l` A savoir que `lpr` génère des paramètres qui sont passés au filtre. Par défaut il passe `-w0`. Si l'option `-l` est donnée, il passe `-c`. Ce script est traditionnellement nommé `/usr/lib/lpf`. Si vous avez plusieurs scripts de la sorte, une bonne idée consiste à les mettre tous dans un sous-répertoire, par exemple `/usr/lib/lpd`.

Il se peut aussi que votre imprimante puisse passer dans un mode permettant l'ajout de retour-chariots grâce à une séquence d'échappement. Voici un exemple de filtre utilisant la commande `echo -ne` pour envoyer cette séquence:

```
#!/bin/sh
# Filtre pour imprimantes HP, permettant de traiter LF comme CRLF
# La commande echo -ne suppose que /bin/sh correspond a bash
echo -ne \\033&k2G
cat
echo -ne \\f
```

Vous pouvez compliquer les filtres comme bon vous semble. Le mieux est d'avoir un filtre qui reconnaît le fichier d'entrée et le convertit au bon format pour votre imprimante. Un tel filtre est appelé filtre magique. Ne vous embêtez pas à les écrire vous-même, il en existe sûrement déjà un qui vous convient. Allez voir sur `tsx-11.mit.edu:/pub/linux/sources/usr.bin/magic-filter-x.y.tar.gz`.

9 Les fichiers, leur emplacement et les droits d'accès

Les différences qui existent entre les nombreuses distributions font que l'on ne peut ici être exhaustif. Je pense que beaucoup de gens utilisent maintenant les distributions type `slackware` et on peut raisonnablement s'appuyer sur cet exemple.

Pensez à inclure `lpd` dans le fichier rc.local après le démarrage éventuel de `syslogd`. Voici les fichiers tels que l'on peut les trouver:

```
-r-sr-xr-x   1 root     lp          9308 Aug 23 21:45 /usr/bin/lpq*
-r-sr-xr-x   1 root     lp         10056 Aug 23 21:45 /usr/bin/lpr*
```

```
-r-sr-xr-x   1 root     lp          8900 Aug 23 21:45 /usr/bin/lprm*
-r-x------   1 root     lp          1596 Aug 23 21:45 /usr/bin/lptest*

-r-xr-sr-x   1 root     lp         17160 Aug 23 21:45 /usr/sbin/lpc*
-rwxr--r--   1 root     lp         34072 Aug 23 21:45 /usr/sbin/lpd*
```

et pour chaque répertoire de spool:

```
/usr/spool/lp/lpr0/
total 5
drwxr-xr-x   2 root     lp          1024 Feb 12 15:15 ./
drwxr-xr-x   3 root     lp          1024 Sep  2  1993 ../
-rw-r----x   1 root     lp             4 Feb 12 15:15 .seq
-rw-r--r--   1 root     lp             3 Feb 13 20:46 lock
-rw-rw-r--   1 root     root          27 Feb 12 15:15 status
```

Ces trois fichiers sont créés par `lpr` et `lpd`. Ils peuvent être absents si vous ne les avez encore jamais lancés. Avec d'anciennes versions il fallait exécuter `touch` sur ces fichiers ou bien modifier leurs droits. Les bugs concernant ces fichiers ont maintenant été corrigés dans les versions récentes.

Il est à noter également que le groupe d'appartenance était `daemon` avec d'anciennes versions, et est maintenant `lp`.

Ne soyez pas surpris de trouver des choses légèrement différentes sur votre système. D'un autre côté, si quelque chose ne fonctionne pas, pensez à soupçonner ces droits avant d'affoler nos boîtes aux lettres (Si vous saviez le nombre de courriers électroniques reçus et dont la solution se trouve là !).

On peut trouver le programme `lpr` avec ou sans le bit setuid(root). En fait ce n'est pas si évident que cela. Tout dépend des droits et permissions des répertoires de spool. Autant que je sache, il y a une totale sécurité avec `lpr`, même si il est setuid(root). Donc, à la limite, positionnez le bit pour ne pas vous soucier des droits d'accès au répertoire de spool.

Vous êtes libre de mettre les binaires dans les répertoires que vous voulez, bien qu'ils se trouvent couramment dans `/usr/bin` ou `/usr/sbin`. (`lpc` et `lpd` peuvent se trouver par exemple dans `/etc`). Certaines commandes étant intéressantes pour tout utilisateur, il est bon de les laisser aux endroits habituels.

Attention toutefois, car les gens qui conçoivent les distributions sont également libres de choisir. Pensez à supprimer les anciennes versions, si vous changez de distribution.

L'emplacement du fichier de verrouillage principal du démon `lpd` (`lpd.lock`), est fixé en dur dans le code. Il se trouve dans `/var/spool/lpd/lpd.lock`. Donc, vous devrez prévoir un répertoire `/var/spool/lpd` même si votre répertoire de spool est différent. Les binaires anciens mettaient ce fichier dans `/var/spool/lpd.lock`/

Typiquement, chez moi, on trouve

```
/var/spool/lpd/
drwxr-xr-x   4 root     lp          1024 Aug 18  1994 ./
drwxr-xr-x  18 root     root        1024 Aug 17  1994 ../
-rw-r--r--   1 root     root           3 Feb 14 20:12 lpd.lock

/var/spool/lp/lpr0
drwxr-xr-x   2 root     lp          1024 Feb 12 15:15 ./
drwxr-xr-x   3 root     lp          1024 Sep  2  1993 ../
-rw-r----x   1 root     lp             4 Feb 12 15:15 .seq*
-rw-r--r--   1 root     root           3 Feb 14 20:12 lock
-rw-rw-r--   1 root     root          27 Feb 12 15:15 status
```

Etant donné que l'on jongle en permanence entre `/usr` et `/var`, il est clair qu'un lien doit exister entre les deux. Soit vous définissez vos répertoires dans `/usr/spool/`... et définissez le lien `/var` vers `/usr`, soit vous mettez tout sous `/var/spool/lpd`... et définissez le lien `/usr/spool` vers `/var/spool`.

Si vous avez, comme moi, une partition root (/) et une partition /usr, les deux cas ne sont pas identiques. Dans le premier, vos fichiers seront stockés dans la partition de root, `/var` etant créé sous `/`, dans l'autre, ce sera dans la partition `/usr`, puisque `/usr` est monté. Vous pouvez aussi avoir un système de fichiers `/var` réservé.

Le fichier de configuration principal est `/etc/printcap`. Il existe aussi, pour l'impression distante, les fichiers `/etc/hosts.allow` et `/etc/hosts.lpd`.

Désormais, le répertoire `/etc` est le répertoire où sont situés les fichiers de configuration. Vous pouvez choisir de les mettre ailleurs, mais définissez toujours un lien symbolique de `/etc` vers vos fichiers. Si votre système comporte des binaires qui vont toujours chercher leur configuration dans `/usr/etc` ou `/etc/inet`, ils sont sûrement très anciens et vous gagneriez à mettre votre système à jour.

10 Ou trouver des filtres d'impression ?

Pas mal de filtres sont déjà rédigés et disponibles sur *Sunsite* ou *lip6* dans /pub/sunsite/linux/system/printing.

10.1 Les filtres magiques

```
Titre:          magicfilter
Version:        1.1b
Date-entree:    04APR95
Description:    Un filtre d'impression automatique, extensible, parametrable.
                Detecte tout type de fichier pour lequel existe un utilitaire
                de conversion. Ce filtre est ecrit en C et completement controle'
                par un fichier de configuration externe. Cette version
                apporte la creation "automagique" de ce fichier
                d'apres les logiciels installes sur votre systeme
                grace a 'GNU Autoconf'.
                Cette version corrige les bogues de la version 1.1/1.1a;
                En plus: filtres pour imprimantes PostScript non-ASCII
Auteur:         H. Peter Anvin <hpa@zytor.com>
Site-initial:   sunsite.unc.edu
                53000 /pub/Linux/system/printing/magicfilter-1.1b.tar.gz
licence-copie:  GPL
```

10.2 Les filtres APS

```
Titre:          apsfilter
Version:        4.9.1
Date-entree:    Lundi, 10. Juillet 1995, 21:22:35  MET DST
Description:    magicfilter for lpd with auto filetype detection
Mots-cles:      lpd magicfilter aps apsfilter
Site-initial:   sunsite.unc.edu
                /pub/Linux/system/printing/
                211KB aps-491.tgz
Platformes:     C-Compiler, gs Postscript emulator, pbmutils
Licence-copie:  GPL
```

Les filtres APS se configurent dans l'entrée *if* du fichier `/etc/printcap` et convertissent la plupart des types de fichiers connus (texte, *PostScript*, dvi, gif, ...) en commandes compréhensibles par votre imprimante.

10.3 Les filtres EZ-magic

```
Titre:          ez-magic printer filter
Version:        1.0.5
Date-entree:    26 Janvier 1997
Description:    ez-magic est un filtre d'impression supportant 8 formats
                de fichiers (txt,ps,gif,bmp,pcx,png,jpg,tif).
                Il permet l'impression via un reseau (SMB), ou vers une
                imprimante locale. Lit depuis un fichier, STDIN ou lpd.
                Simple a utiliser et a configurer. Seul un fichier 'script'
                est necessaire, pas de multitudes de manuels et pilotes.
                Necessite des programmes de conversion (tels que netpbm et
                ghostscript). Ecrit en 'bash'. Ajout de nouveaux formats
                facile. Preconfigure pour HP DeskJet 870Cse en reseau.
                Comparable a apsfilter et autres.
Mots-cles:      magic filter, print, graphics, samba, network, smb,
                ghostscript, postscript, gif, jpg, simple
Auteur:         toby@eskimo.com (Toby Reed)
Maintenu-par:   toby@eskimo.com (Toby Reed)
Site-initial:   http://www.eskimo.com/~toby/ez-magic-1.0.5.tar.gz
                38 kb ez-magic-1.0.5.tar.gz
Site-secondaire:sunsite.unc.edu /pub/Linux/system/printing
                38 kb ez-magic-1.0.5.tar.gz
Licence-copie:  Copyright, droits complets de manipulation sauf 1 ou 2
                restrictions.
```

11 Les logiciels d'impression

Les logiciels d'impression sont disponibles en France par exemple sur `ftp://ftp.lip6.fr/pub/linux/sunsite/system/Printing`.

11.1 GhostScript

GhostScript est un logiciel majeur pour l'impression sous Linux. En effet la plupart des logiciels génèrent du *PostScript*. Ce logiciel, gratuit, est capable de convertir le *PostScript* en langage compréhensible par votre imprimante (si le pilote est disponible). Il joue le rôle de filtre afin que vous puissiez considérer votre imprimante comme *PostScript*. Ceci vous simplifie grandement la vie.

GhostScript est disponible sous deux formes. Une version commerciale, appelée *Alladin GhostScript*, peut être utilisée librement pour des besoins privés mais ne doit pas être distribuée par les distributions payantes de Linux. Elle est généralement en avance d'une année sur la version gratuite.

La version gratuite est sous licence GNU et n'est rien d'autre qu'une version plus ancienne de *Alladin GhostScript*.

11.1.1 Utiliser GhostScript

`gs` est le nom de l'exécutable. `gs -help` vous donne une aide rapide sur les paramètres disponibles. (La liste des pilotes est la liste des pilotes compilés avec la version et non la liste complète.)

Quoi que vous fassiez avec `gs`, il est conseillé d'invalider l'accès aux fichiers (par `-dSAFER`). En effet *PostScript* est un langage pleinement opérationnel et un fichier *PostScript* peu scrupuleux peut endommager vos fichiers et vous donner un terrible mal de crâne..

Exemple de ligne de commande pour un imprimante Stylus 800

```
gs -dNOPAUSE -sDEVICE=escp2 -sPAPERSIZE=a4 -sOutputFile=/dev/lp1 fichier.ps
```

11.1.2 Réglages

La taille, l'aspect et la situation d'une image sur une page sont réglés par le pilote de l'imprimante dans *GhostScript*. Si vous trouvez que vos impressions sont tronquées ou mal cadrées il faudra soit retoucher le code du pilote (déconseillé), soit modifier les fichiers de configuration (`gs_init.ps`, `gamma.ps`)

11.1.3 Gamma, tailles de points, ...

Il se peut que vous trouviez vos impressions trop sombres. Ceci peut arriver si votre imprimante n'a pas une définition suffisante. Dans ce cas vous devez créer votre propre fonction de transfert. Pour ceci, créez le fichier `gamma.ps` dans le répertoire des librairies de *GhostScript* et appelez le fichier sur la ligne de commande de gs avant le fichier à imprimer. Pour éclaricir le résultat, vous devez diminuer les valeurs indiquées. Notamment si votre pilote utilise l'algorithme de *Floyd-Steinberg* pour rastériser les couleurs, des valeurs comprises entre 0.15 et 0.2 sont mieux adaptées.

```
---8<---- gamma.ps ----8<---
%!
%transfer functions for cyan magenta yellow black
{0.3 exp} {0.3 exp} {0.3 exp} {0.3 exp} setcolortransfer
---8<------------------8<---
```

Vous pouvez aussi modifier les dominantes de couleurs (voir le répertoire `/examples` de *GhostScript* qui contient une page de test des couleurs).

12 L'impression à distance

Une des caractéristiques de `lpd` est qu'il supporte l'impression sur des imprimantes rattachées à d'autres machines que la vôtre. Avec, en plus, une combinaison de filtres soignée, vous aurez un système d'impression transparent, réparti et performant.

12.1 Vers un hôte Unix/lpd

Pour que des machines distantes puissent utiliser l'imprimante attachée à votre machine, le nom de ces machines doit être référencé soit dans le fichier `/etc/hosts.lpd`, soit dans le fichier `/etc/hosts.equiv`. Ce sont des fichiers textes normaux, dans lesquels on indique un nom de machine par ligne.

Il est préférable de déclarer les machines dans `/etc/hosts.lpd` qui est spécialement réservé à l'impression, le fichier `/etc/hosts.equiv` donnant des droits plus étendus.

Vous pouvez restreindre les droits d'accès distants par groupe ou par utilisateur. Les groupes autorisés sont indiqués grâce au paramètre `:rg=:` du fichier `printcap`: `rg=admin` restreint l'utilisation aux utilisateurs du groupe `admin`. Le paramètre booléen `:rs=:` du même fichier restreint l'accès aux utilisateurs ayant un compte sur votre machine.

12.1.1 Avec lpd

Pour imprimer vers une autre machine, vous devez créer une entrée `printcap` telle que:

```
# REMOTE djet500
lp|dj|deskjet:\
        :sd=/var/spool/lpd/dj:\
        :rm=machine.out.there.com:\
        :rp=printername:\
        :lp=/dev/null:\
        :sh:
```

Vous noterez qu'il doit bien exister un répertoire de spool local géré par votre `lpd` local. Les fichiers soumis y seront copiés puis envoyés vers la machine distante.

12.1.2 Avec rlpr

Vous pouvez utiliser `rlpr` pour soumettre une impression directement à la machine distante sans passer par un démon local et toute sa configuration. C'est particulièrement intéressant lorsque vous imprimer rarement et vers différentes imprimantes.

`rlpr` s'appuie sur TCP/IP. Il n'est pas nécessaires que les imprimantes soient connues explicitement. N'ayant pas besoin de fichier printcap, il est plus facile à gérer. Il est compatible avec `lpr`.

12.2 Imprimer vers une imprimante attachée à Windows 95, NT, LanManager ou Samba.

Il existe un mini-document (`Printing to Windows HOWTO`) qui décrit ceci très bien.

Il est possible de rediriger une file `lpd` vers un service d'impression SMB grâce au programme `smbclient` (`http://www.picante.com/~gtaylor/pht/man/smbclient.html`). Samba contient un script (`smbprint`) qui fait cela. Vous devez mettre un fichier de configuration pour l'imprimante en question dans le répertoire de spool et décrire le programme `smbprint` en tant que filtre dans `/etc/printcap`:

```
lp|remote-smbprinter:\
    :lp=/dev/null:sh:\
    :sd=/var/spool/lpd/lp:\
    :if=/usr/local/sbin/smbprint:
```

Vous pouvez également utiliser le programme `smbclient` pour soumettre un fichier directement à un service d'impression SMB sans impliquer `lpd`. Lisez la documentation de tous ces programmes pour plus d'information.

12.3 Vers une imprimante NetWare

La suite logicielle `ncpfs` contient un utilitaire appelé `nprint` qui fournit les mêmes fonctionnalités que smbprint pour NetWare. Vous pouvez l'obtenir sur `ftp://sunsite.unc.edu/pub/Linux/system/filesystems/ncpfs/`. Avec `ncpfs` vous pouvez monter des volumes du serveur NetWare sous Linux. Vous pouvez également soumettre des travaux d'impression vers NetWare ou mettre des travaux d'impression de NetWare en file sur votre système Linux. Vous devez disposer d'un noyaux 1.2.x ou 1.3.54 et ultérieur. `ncpfs` **NE** fonctionne **PAS** avec un noyau 1.3.x ($x < 54$).

Pour que `nprint` fonctionne via `lpd`, vous devez écrire un shell-script pour diriger stdin sur l'imprimante NetWare. Vous l'installerez comme un filtre (`if`) d'une file d'impression `lpd`. Vous obtiendrez :

```
sub2|remote-NWprinter:\
            :lp=/dev/null:sh:\
            :sd=/var/spool/lpd/sub2:\
            :if=/var/spool/lpd/nprint-script:
```

le script `nprint-script` ressemble à quelque chose comme :

```
#! /bin/sh
# Essayez en premier le compte invite (guest)sans mot de passe!
/usr/local/bin/nprint -S net -U name -P passwd -q printq-name -
```

12.4 Vers une imprimante EtherTalk

Le paquetage logiciel `netatalk` contient l'équivalent de `nprint` et `smbclient`. Référez-vous au `Netatalk-HOWTO` dans lequel est bien décrite la procédure d'impression vers et depuis un réseau *Apple*.

12.5 Vers une imprimante HP ou autre imprimante Ethernet

Certaines imprimantes (HP et autres) sont fournies avec une interface Ethernet que vous pouvez directement adresser pour soumettre vos travaux d'impression. Conformez-vous au manuel du constructeur. En général, ces imprimantes font “tourner” un `lpd` et fournissent une ou plusieurs files vers lesquelles vous pouvez imprimer. Une imprimante HP, par exemple, pourra fonctionner avec une entrée `printcap` telle que:

```
lj-5|remote-hplj:\
            :lp=/dev/null:sh:\
            :sd=/var/spool/lpd/lj-5:\
            :rm=printer.name.com:rp=raw:
```

Les imprimantes LaserJet HP avec une interface JetDirect ont en général deux files incorporées; l'une “raw” accepte le PCL (et peut-être le PostScript), l'autre “text” accepte l'ascii pur (et s'arrange pour résoudre d'elle-même les problèmes d'effets d'escalier).

Dans un environnement dans lequel plusieurs imprimantes ne supportent pas le *PostScript*, il peut être bon de configurer un serveur d'impression dédié vers lequel toutes les machines enverront leurs travaux et sur lequel GhostScript tournera.

12.6 Vers d'anciennes HP

Certaines anciennes imprimantes HP ne supportent qu'un protocole mal foutu s'appuyant sur des connexions TCP, notamment les premièrs modèles à base de cartes JetDirect (et quelques JetDirectEx). Pour imprimer vers de telles imprimantes, vous devez ouvrir une connexion TCP vers un port dédié (9100) et envoyer votre impression vers cette connexion. Voici le script Perl correspondant :

```
#!/usr/bin/perl
# Thanks to Dan McLaughlin for writing the original version of this
# script (And to Jim W. Jones for sitting next to Dan when writing me
# for help ;)
```

```
$fileName = @ARGV[0];

open(IN,"$fileName") || die "Can't open file $fileName";

$dpi300     = "\x1B*t300R";
$dosCr      = "\x1B&k3G";
$ends = "\x0A";

$port =  9100 unless $port;
$them = "bach.sr.hp.com" unless $them;

$AF_INET = 2;
$SOCK_STREAM = 1;
$SIG{'INT'} = 'dokill';
$sockaddr = 'S n a4 x8';

chop($hostname = `hostname`);
($name,$aliases,$proto) = getprotobyname('tcp');
($name,$aliases,$port) = getservbyname($port,'tcp')
    unless $port =~ /^\d+$/;;

($name,$aliases,$type,$len,$thisaddr) =
        gethostbyname($hostname);
($name,$aliases,$type,$len,$thataddr) = gethostbyname($them);
$this = pack($sockaddr, $AF_INET, 0, $thisaddr);
$that = pack($sockaddr, $AF_INET, $port, $thataddr);

if (socket(S, $AF_INET, $SOCK_STREAM, $proto)) {
#    print "socket ok\n";
}
else {
    die $!;
}
# Give the socket an address.
if (bind(S, $this)) {
#    print "bind ok\n";
}
else {
    die $!;
}

# Call up the server.

if (connect(S,$that)) {
#    print "connect ok\n";
}
else {
    die $!;
}

# Set socket to be command buffered.

select(S); $| = 1; select(STDOUT);

#    print S "@PJL ECHO Hi $hostname! $ends";
```

```
#    print S "@PJL OPMSG DISPLAY=\"Job $whoami\" $ends";
#    print S $dpi300;

# Avoid deadlock by forking.

if($child = fork) {
    print S $dosCr;
    print S $TimesNewR;

    while (<IN>) {
        print S;
    }
    sleep 3;
    do dokill();
} else {
    while(<S>) {
        print;
    }
}

sub dokill {
    kill 9,$child if $child;
}
```

12.7 Les filtres d'entrée pour des imprimantes distantes

Une des bizarreries de `lpd` est que le filtre d'entrée (`if`) n'est pas exécuté pour des imprimantes distantes. Si vous devez absolument passer par un filtre, il vous faudra utiliser deux files, la première redirigeant vers la deuxième. Par exemple :

```
lj-5:remote-hplj:\
        :lp=/dev/null:sh:\
        :sd=/var/spool/lpd/lj-5:\
        :if=/usr/lib/lpd/filter-lj-5:
lj-5-remote:lp=/dev/null:sh:rm=printer.name.com:\
        :rp=raw:sd=/var/spool/lpd/lj-5-raw:
```

et le filtre `filter-lj-5` (exemple):

```
#!/bin/sh
gs <options> -q -dSAFER -sOutputFile=- - | \
        lpr -Plj-5-remote -U$5
```

L'option `-U` ne fonctionne que si `lpr` est lancé en tant que démon. Elle positionne correctement le nom du soumissionnaire du travail dans la deuxième queue. Il serait d'ailleurs mieux d'utiliser une méthode plus sûre pour récupérer ce nom car ce n'est pas toujours le 5ème paramètre.

12.8 Imprimer depuis Windows

L'impression depuis un client Windows vers un serveur Unix est directement supporté par `SMB` en utilisant le paquetage `SAMBA` (qui supporte également le partage de fichiers du système de fichiers Linux vers les clients Windows).

Samba est fourni avec une documentation complète. Vous pouvez soit installer un filtre magique sur Linux et imprimer du *PostScript* soit installer un pilote d'imprimante spécifique sous Windows et décrire une file sans filtre. En s'appuyant sur les pilotes Windows, vous pourrez obtenir de meilleurs résultats, mais c'est un peu plus compliqué à administrer si vous avez plusieurs stations sous Windows. Donc essayez d'abord la première solution.

12.9 Depuis Netware

Il y a des services NetWare disponibles pour Linux, mais je n'ai aucune idée si vous pouvez offrir des services d'impression depuis des client Netware. Des informations sont les bienvenues.

12.10 Depuis un Apple

Netatalk permet d'imprimer depuis une station Apple sur EtherTalk. Voir le `Netatalk HOWTO` (`http://thehamptons.com/anders/netatalk`) pour de plus amples renseignements.

12.11 Imprimer vers un fax

12.11.1 Utiliser un modem/fax

Si vous disposez d'un modem/fax, vous pouvez configurer votre système pour envoyer ou recevoir des fax aux formats *PostScript*, dvi, ascii, etc... Vous pourrez même faire en sorte que votre courrier électronique soit faxé!

Les modems/Fax supportent les commandes de classe 1 ou 2. Les modems de classe 1 ont un sous-ensemble de fonctionnalités d'un fax disons traditionnel (donc, le logiciel doit faire le reste... et le coût de traitement est parfois critique!)

La classe 1 correspond au standard EIA 578. Les modems de classe 2 répondent au sandard EIA 592. Vous trouverez dans la documentation de votre modem à quelle classe il appartient. Ne confondez pas classe et groupe. Le groupe est généralement le groupe III.

Les logiciels de fax tournant sous Linux doivent savoir convertir les données reçues en un format compatible avec le groupe III pour la transmission. Comme d'habitude *Ghostscript* sait faire ! Le pilote `tiffg3` génère des messages fax encodés au format `g3/tiff`. Vous devrez compiler et intégrer le pilote si ce n'est déjà fait.

Un des logiciels les plus complets sur le sujet, `HylaFax` est disponible sur :

```
ftp.sgi.com:/sgi/fax/?????.src.tar.Z
```

Il supporte toutes sortes de choses comme les multiples modems et la diffusion.

`mgetty + sendfax` est un couple de logiciels, contenant un `getty` pour Linux et les modems fax ainsi qu'un logiciel d'envoi de fax assez simple. Ce paquetage se trouve à:

```
sunsite.unc.edu:/pub/Linux/system/serial/getty/mgetty+sendfax-1.0.0.tar.gz
```

Enfin, `efax` mérite d'être mentionné. C'est un excellent choix pour Linux. Il supporte les classes 1 et 2.

```
sunsite.unc.edu:/pub/Linux/apps/serialcomm/fax/efax08a.tar.gz
```

(Répertoire dans lequel on trouvera également `vfax10.tar.z`, `qfax1.3.tar.gz`, `xfax.v1.07s.tar.gz`)

12.11.2 Utiliser le service d'impression distant

C'est un service expérimental vous permettant d'envoyer un courrier électronique que vous souhaitez imprimer sur un fax distant. Des formats tels que PostScript sont supportés. Bien que la couverture géographique de ce service soit très faible, il est très prometteur. Pour plus d'information, référez-vous au site Web "Remote Printing WWW Site" (`http://www.tpc.int/`).

13 Les logiciels qui permettent de réaliser de belles impressions

Linux sait faire tourner un grand nombre de binaires avec plus ou moins de réussite: Linux/x86, Linux/Alpha, Linux/Sparc, iBCS, Windows (Un jour, avec Wine), Mac/68k (avec Executor) et Java. WordPerfect, traitement de texte commercial tourne bien avec l'émulation iBCS. La suite Corel Office en Java est également prometteuse...)

En ce qui concerne Linux, les choix sont limités aux logiciels Unix classiques:

13.1 Les langages à balises

La plupart des langages balisés sont bien adaptés aux projets conséquents ou répétitifs pour lesquels vous souhaitez que l'ordinateur contrôle la mise en forme pour un résultat homogène. Vouloir ajouter un bel effet dans un tel langage choquera certainement.

13.1.1 nroff

C'est l'un des plus anciens langages balisés sous Unix. Les pages de manuel en ligne sont l'exemple le plus connu de pages formattées en macros nroff; beaucoup de gens ne jurent que par lui... mais sa syntaxe est quand même plus compliquée que nécessaire; ce n'est certainement pas le bon choix pour les nouveaux projets. Il est intéressant de savoir que vous pouvez imprimer une page de manuel directement en PostScript avec `groff`. La syntaxe est la suivante: `man -t truc | lpr`. Le résultat en vaut vraiment la peine.

13.1.2 TeX

TeX (et le paquetage de macros LaTeX) est l'un des langages balisés le plus répandu sous Unix. Les travaux techniques sont souvent rédigés en LaTeX parce qu'il simplifie grandement la mise en page et il est l'un des rares logiciels à traiter correctement et puissamment les fomules mathématiques. Le format de sortie de TeX est dvi, et peut être converti en *PostScript* ou PCL (HP) à l'aide du programme `dvips` ou `dvilj`.

13.1.3 SGML

Il y a au moins un interpréteur de SGML gratuit sous Unix et Linux; il est la base du système de documentation LinuxDoc-SGML. Il supporte bien entendu d'autres types de documents.

13.1.4 HTML

Bien connu, il permet d'écrire des documents simples.

13.2 Traitements de textes WYSIWYG

Ca y est! Linux et Unix ne manquent plus de tels traitement de texte. Il existe plusieurs suites logicielles dont une disponible gratuitement pour une utilisation personnelle: *StarOffice*.

13.2.1 StarOffice

Une compagnie allemande distribue `StarOffice` pour Linux. On peut trouver une version libre de droits sur les serveurs ftp classiques. Cette version est limitée à un usage personnel. Cette suite logicielle est très complète; vous trouverez tout ce dont vous rêviez. Il existe d'ailleurs un mini-HOWTO décrivant où la trouver et comment l'Installer. La plupart des imprimantes sont reconnues.

13.2.2 LyX

`LyX` est une interface pour LaTeX assez prometteuse. Rendez-vous sur la page Web de Lyx à `http://www-pu.informatik.uni-tuebingen.de/users/ettrich/` pour de plus amples renseignements.

13.2.3 L'interface utilisateur 'Andrew'

Cette interface comprend un éditeur WYSIWYG appelé `ez` comportant la plupart des fonctionnalités de base d'un traitement de texte: HTML, et MIME pour courrier et forums de discussions (niouzes, si vous préférez).

13.2.4 Offres commerciales

Caldera et Red Hat vendent des paquetages logiciels comprenant les applications principales, à savoir un traitement de texte et un tableur. Caldera vend également (à vérifier) WABI, l'émulation Windows de Sun qui permet de faire tourner les applications de Microsoft bien connues.

Pour plus d'information, consultez les sites de Caldera (`http://www.caldera.com`) et de Red Hat (`http://www.redhat.com`).

D'autres vendeurs peuvent m'envoyer un courrier descriptif de leur offre.

14 Logiciels de pré-visualisation

Tout ce que vous imprimez peut être également visualisé à l'écran. Ceci permet dans bien des cas d'économiser du papier.

14.1 PostScript

GhostScript possède un pilote X11 utilisé par Ghostview (`http://www.picante.com/~gtaylor/pht/man/ghostview.htm`
La dernière version de ce logiciel devrait permettre également la visualisation de fichiers au format PDF.

14.2 TeX dvi

les fichiers TeX dvi (DeVice Independent - indépendant du périphérique) peuvent être visualisés sous X11 à l'aide de `xdvi` (`http://www.picante.com/~gtaylor/pht/man/xdvi.html`). Les versions récentes de `xdvi` appellent `ghostscript` pour le rendu de spécificités *PostScript*.

Un pilote VT100 (`dgvt`) existe également. `Tmview` fonctionne sous Linux avec `svgalib`, si cela vous suffit.

14.3 Adobe PDF

Acrobat Reader d'Adobe est disponible pour Linux. Vous pouvez le charger depuis `http://www.adobe.com`. Vous pouvez également utiliser `xpdf`, un logiciel gratuit fournit avec les sources. *GhostScript*, devrait, comme je l'ai dit plus haut, supporter le format pdf.

15 Les imprimantes série

Le démon `lpd` fournit cinq attributs que vous pouvez positionner dans `/etc/printcap` afin de contrôler le port série sur lequel se trouve votre imprimante.

```
br
     (numerique) definit le taux de transfert en bauds (appel a ioctl(2))
fc
     (num) efface des indicateurs (sgtty.h)
fs
     (num) positionne des indicateurs (inverse de 'fc')
xc

xs
```

Pour définir la vitesse du port, la syntaxe est évidente. Exemple: br \#9600.

Les autres paramètres à positionner correspondent à un ensemble de bits, que l'on pourra soit positionner, soit mettre à 0. Pour effacer des bits, on utilisera les paramètres fc et xc, pour les positionner, fs et xs.

Faites bien attention aux bits que vous sélectionnez. Mais au fait que sont-ils ? Souvenez-vous..., la commande `stty`. Elle indique de nombreux paramètres caractérisant un tty. La commande `stty -a` affiche en clair les paramètres du tty, certains d'entre-eux étant précédés d'un tiret s'ils sont invalidés et sans tiret s'ils sont validés. La commande `stty` peut être appliqués au port série (voir exemple). Ce sont certains de ces paramètres (des drapeaux, des flags, donc des bits) que l'on va manipuler.

```
# stty -a < /dev/ttyS2
     speed 9600 baud; rows 0; columns 0; line = 0;
     intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
     eol2 = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R; werase = ^W;
     lnext = ^V; min = 1; time = 0;
     -parenb -parodd cs8 hupcl -cstopb cread -clocal -crtscts
     -ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr
     -igncr -icrnl ixon -ixoff -iuclc -ixany -imaxbel
     -opost -olcuc -ocrnl -onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0
     bs0 vt0 ff0
     -isig -icanon -iexten -echo -echoe -echok -echonl -noflsh -xcase -tostop
     -echoprt -echoctl -echoke
```

Note: utilisez toujours `stty` de cette façon (`stty < /dev/ttyS?`); cette command utilise en effet l'entrée standard).

Vous pouvez utiliser cette commande pour configurer le port de façon à obtenir une impression correcte. Par exemple, les différences que l'on peut noter entre le stty ci-dessus et l'initialisation du port au démarrage de ma machine réside dans les informations `-clocal`, `-crtscts` et `ixon`. (La configuration de votre port pourra très bien être différente selon la manière dont votre imprimante gère le contrôle de flux).

Votre port étant bien configuré, faites : `cat fichier > /dev/ttyS?` (? est le numéro de votre port) pour imprimer un fichier.

Imprimez par exemple le fichier `/usr/src/linux/include/asm-i386/termbits.h`. Vous y découvrirez un tas de définitions de constantes et de structures. Nous allons voir quelles valeurs définies dans ce fichier vont nous servir pour configurer le port, non plus avec stty, mais avec les paramètres `fc`, `xc`, `fs` et `xs` du fichier printcap. Regardez la section commençant par :

```
/* c_cflag bit meaning */
#define CBAUD   0010017
```

Elle décrit justement les bits manipulables à l'aide de fc et fs dont on parlait. On y voit les constantes des vitesses de modulation en baud, puis des lignes qui nous intéressent particulièrement : ce sont les mêmes paramètres que dans la commande `stty`. Je sens que vous voyez où on veut en venir. `stty` n'est qu'un interface nécessaire au positionnement (ou effacement) de bits.

Vous savez maintenant que chaque paramètre affiché par `stty` correspond à un bit, et qui ont la valeur 0 lorsqu'il y a un tiret devant. Notez alors les bits à effacer (ce sera fait avec la paramètre `fc`) et ceux à positionner (paramètre `fs`). Exemple: `'fc\#0177777'` (Attention le paramètre `fc` semble surcharger le paramètre `br`, donc prenez garde à les positionner correctement).

Ensuite occupez-vous des bits à positionner. Par exemple s'il faut positionner les bits `cs8`, `hupcl` et `cread`, regardez les constantes `CS8` (0000060), `HUPCL` (0002000) et `CREAD` (0000200). Pensez à la vitesse de modulation qu'il faut aussi définir, dans mon cas, ce sera `B9600` (0000015). Tous ces bits ensemble font '0002275'. Indiquez cette valeur au paramètre `fs`.

Effectuez les même réglages avec la section suivante intitulée

```
/* c_lflag bits */
```

Dans mon cas je n'ai rien à positionner, j'ai donc simplement à fournir la valeur `xc\#0157777`, puis `xs\#0`. Une fois votre fichier printcap correctement défini, essayez d'imprimer. Si quelque chose ne va pas, continuez à lire les paragraphes suivants.

Souvenez-vous de toujours commencer par les bits que vous souhaitez voir à 0 (`fc` et `xc`), puis de définir seulement après des bits à positionner (`fs` et `xs`).

La commande `cat` fonctionne pour le port série, mais pas `lpd`

La mise en place de `lpd` n'est pas traitée ici, mais sachez que si vous avez des problèmes avec la configuration du port série, vous pouvez empêcher `lpd` de le configurer en considérant votre imprimante comme ne présentant pas une interface normale. Lisez également à ce propos le chapitre suivant.

Donnez à votre imprimante le périphérique `/dev/null1` (`mknod /dev/null1 c 1 3`). N'utilisez pas `/dev/null`, pour ne pas qu'il soit ouvert de manière exclusive. Enlevez les paramètres de vitesse et de positionnement des bits du fichier `printcap`.

Créez un shell-script comme ci-dessous :

```
#!/bin/sh
echo if: $* >> /var/spool/lpd/results
# /dev/lp est un lien vers /dev/ttyS2 auquel est reliee l'imprimante
exec votre_vieux_filtre $* > /dev/lp
```

...ou si vous n'avez pas de paramètre 'if' configuré...

```
#!/bin/sh
echo if: $* >> /var/spool/lpd/results
cat > /dev/lp
# la commande ‘‘echo -ne’’ suppose que /bin/sh correspond a bash
echo -en \\f > /dev/lp
```

Donnez-lui les droits de lecture/écriture pour tout le monde. Essayez-le: `/usr/lib/lpd/if <FICHIER`.

Définissez un filtre d'entrée dans votre fichier `printcap` pour appeler ce script. `:if=/usr/lib/lpd/if:`.

Utilisez la commande `stty` pour configurer le port correctement. Essayez d'imprimer. Vous devriez pouvoir déterminer si le(s) fichier(s) sont bien mis dans le répertoire de spool. Cela devrait imprimer, si votre essai manuel du script précédent a fonctionné. Bien entendu, le mieux serait de pouvoir se passer du script d'entrée et donc du paramètre `if`.

Supposons donc que la méthode précédente a fonctionné, et que vous pensez avoir correctement configuré votre fichier printcap. Exécutez la commande `stty -a </dev/ttyS?`. Si certains paramètres ne sont pas corrects, vérifiez les constantes du fichier `termbits.h`. Si la configuration est incorrecte, malgré tous vos efforts de vérification, n'hésitez pas à installer un démon récent.

15.1 Imprimantes anciennes et caractères perdus

Certaines anciennes imprimantes séries ont des petits tampons mémoire et gèrent mal le contrôle de flux. Supprimer le FIFO du port série (16550) avec `setserial` (faites croire à setserial que c'est un port 8250 et ça marchera).

16 Compléments et réglages

Si votre imprimante est *PostScript*, elle peut ne pas être capable de traiter du texte pur. Si tel est le cas, vous devrez mettre en place un filtre pour transformer le texte en *PostScript*. Un excellent freeware (logiciel libre de tout droit de distribution et d'utilisation) appelé `nenscript` réalise cela très bien. Si vous ne mettez pas en place un tel filtre, vous devez vous assurer par d'autres moyens que l'imprimante ne reçoit bien que du *PostScript*.

Vous pouvez également définir dans vos fichiers de login (`.profile`, par exemple) ou celui par défaut, une variable d'environnement `PRINTER` définissant l'imprimante à utiliser. Exemple:

```
export PRINTER=lpr0
```

Ceci évite d'avoir à spécifier `-Plpr0` à chaque fois.

Il est possible de "réutiliser" une entrée printcap. Si vous déclarez votre propre machine comme machine hôte distante, et une autre imprimante comme imprimante distante, vous pouvez rediriger les données à imprimer de l'une vers l'autre. Souvenez-vous que si vous utilisez cette technique, les données passeront par chaque filtre de la chaîne et seront mises successivement dans chaque spool.

Bien que vous puissiez spécifier pour une imprimante autant d'alias que vous le souhaitez, il semble que pour la meilleure utilisation, les deux premiers doivent être identiques et doivent correspondre au nom réel. Certains programmes n'utiliseront que ces deux entrées. La commande lpc indiquera seulement le premier alias, alors que les commandes `lpr`, `lprm` et `lpq` comprennent tous les alias.

Plutôt que de spécifier une taille maximum de fichier pour l'impression, vous préfereriez sans doute que les fichiers du spool ne puissent remplir votre disque, même temporairement. Pour ce faire, créez un fichier appelé `minfree` dans chaque répertoire de spool, contenant, sous forme d'un nombre de blocs (1 Ko pour

Linux), la quantité minimum d'espace disque devant rester pour que les données puissent être acceptées dans le spool. Vous créerez un fichier réel dans le répertoire principal de spool et, dans chaque sous-répertoire, un lien symbolique vers ce fichier.

17 Résumé

Voici un guide de configuration étape par étape pour une imprimante nommée `/dev/lp0`. Vous pouvez l'étendre à votre guise. Pour faire ce qui suit, vous devez être root. (NDT: L'auteur a tout installé dans `/usr/spool/lpd`. J'ai préféré modifier légèrement cette configuration en définissant `/usr/spool/lp/lpr0` et `/usr/spool/lpd`, comme dans la plupart des distributions)

- Vérifiez les droits d'accès et l'emplacement de `lpr`, `lprm`, `lpq`, `lpc` et `lpd`. Voir à ce propos: Les programmes importants.
- Créez le répertoire de spool pour votre imprimante, appelée dans notre exemple `lpr0`:

```
mkdir /usr/spool/lp /usr/spool/lp/lpr0
chowm root.lp /usr/spool/lp /usr/spool/lp/lpr0
chmod ug=rwx,o=rx /usr/spool/lp /usr/spool/lp/lpr0
```

- Créez un répertoire permettant au démon d'y mettre son propre fichier de verrouillage `lpd.lock`:

```
mkdir /usr/spool/lpd
chowm root.lp /usr/spool/lpd
chmod ug=rwx,o=rx /usr/spool/lpd
```

- Dans les répertoires `/usr/spool/lp/lpr0` et `/usr/spool/lpd`, créez les fichier nécessaires, avec les bons droits (Cette manipulation n'est pas nécessaire avec les versions récentes du gestionnaire d'impression):

```
cd /usr/spool/lp/lpr0
touch .seq errs status lock
chown root.lp .seq errs status lock
chmod ug=rw,o=r errs status
chmod u=rw,go=r lock
chmod u=rw,g=r,o=x .seq

cd /usr/spool/lpd
touch .seq errs status lock
chown root.lp .seq errs status lock
chmod ug=rw,o=r errs status
chmod u=rw,go=r lock
chmod u=rw,g=r,o=x .seq
```

- Créez le shell-script `filtre_entree` dans le répertoire `/usr/spool/lp/lpr0`. Utilisez le filtre décrit précédemment. Donnez-lui les bons droits.

```
cd /usr/spool/lp/lpr0
chmod ug=rwx,o=rx filtre_entree
```

- Créez le fichier `/etc/printcap`, s'il n'existe pas. Enlevez les entrées qu'il contient et ajoutez une entrée de test décrite précédemment. Donnez-lui les droits `-rw-r-r-`.
- Editez le fichier `/etc/rc.d/rc.local` ou `rc.multi`. Ajoutez la ligne `/usr/sbin/lpd` à la fin. Cela lancera le démon au boot. Vous pouvez aussi le lancer à la main :

```
root# /usr/sbin/lpd
```

- Effectuez un test d'impression :

```
ls -l | lpr -Plpr0
```

- Regardez dans `/tmp` et vérifiez la présence du fichier `testlp.out`. Il devrait contenir le listing du répertoire dans lequel vous étiez.
- Editez `/etc/printcap`. Dupliquez l'entrée `lpr0`. Vous avez alors 2 entrées identiques. Dans la première entrée, changez, sur la première ligne uniquement, les occurrences de `lpr0` par `testlp`. Dans la seconde entrée, changez `/dev/null` par le périphérique réel, par exemple `/dev/lp0`. Dans la seconde entrée, enlevez le champ `if` complètement.
- Rebootez le système ou tuez le démon d'impression et relancez-le, afin qu'il reprenne en compte le fichier `/etc/printcap` modifié.
- Refaites un test d'impression - Allumez votre imprimante. Cela devrait imprimer!

```
ls -l | lpr -Plpr0
```

18 Problèmes avec la LaserJet 5M

18.1 PostScript

Cette imprimante peut poser des problèmes d'impression en `PostScript`. Si vous constatez, après avoir imprimé un document `PostScript`, qu'elle ne sort plus les documents suivants et qu'elle affiche “DATA RECEIVED” en permanence, suivez les conseils qui suivent. Merci à <Patrick.Begou@hmg.inpg.fr>.

- l'imprimante est en configuration d'usine avec le flag d'impression des erreurs postscript validé.
- Le fichier `printcap` est très simple:

```
# /etc/printcap
#
# This file can be edited with the printtool in the control-panel.
laser5M|lp:\
    :lp=/dev/lp2:\
    :sh:\
    :sd=/var/spool/lpd/laser5M:\
    :lf=/var/spool/lpd/laser5M/erreurs:\
    :if=/var/spool/lpd/laser5M/filter_court:
```

- Le filtre utilisé (`filter_court`) est donné ci-dessous. Il n'accepte que les fichiers *ascii*, *PostScript* et *PostScript compressé*. Ce qui est intéressant, ce sont les chaines d'initialisation en PCL. En gros:

```
\033%-12345X      passage en mode PCL (d'apres l'ingenieur de MDS)
\033E             reinitialisation de l'imprimante les \n
                  semblent indispensables dans le filtre.
```

  Avec `echo -ne`, Esc c'est `033` en octal (`27` en décimal)

```
#!/bin/sh
#
# Filtre d'impression gerant l'ascii, le postscript et le postscript
# compresse pour une laserjet  5M. A declarer comme filtre "if" dans
# /etc/printcap.
# Reinitialise l'imprimante entre chaque impression postscript pour
# eviter les erreurs sur timeout.
#
# Version 0.1   P. BEGOU    8/97  (Patrick.Begou@hmg.inpg.fr)
#
# recuperation des donnees dans un fichier temporaire
#
FILE=/tmp/laser5M.$$
cat - > ${FILE}
#
# Scrutation du type de fichier
#
filetype=`file $FILE| cut -d":" -f2`
case $filetype in
  *PostScript*)
       (echo -ne "\033%-12345X\n"; cat ${FILE}; echo -ne "\033%-12345X\033E\033%-12345X\n")
        echo " $FILE postscript" >&2
        ;;
  *text*|*script*)
       (echo -ne "\033%-12345X\n"; /usr/bin/nenscript -s -2rG -p- ${FILE};
        echo -ne "\033%-12345X\033E\033%-12345X\n")
        echo " $FILE texte" >&2
        ;;
  *compress*)
       # verifions qu'il s'agit bien de postscript.
       resu=`zcat ${FILE} |head -1|grep "^%!"|wc -l`
       if [ ${resu} -eq 1 ]
       then
          (echo -ne "\033%-12345X\n"; zcat ${FILE}; echo -ne "\033%-12345X\033E\033%-12345X\n")
          echo " $FILE compresse" >&2
       else
          echo " $FILE compresse non reconnu" >&2
       fi
       ;;
  *)
        echo " $FILE non reconnu" >&2
        ;;
esac

#
# On repond Ok, tout s'est bien passe.
#
rm ${FILE}
exit 0
```

18.2 Recto-verso

Voici comment configurer, par logiciel, le passage en mode recto-seulement ou recto-verso aussi bien pour de l'ascii que du PostScript.

Il suffit d'insérer après la première ligne...

```
%!Ado...
```

...les lignes suivantes pour valider le recto-verso:

```
%%BeginFeature: *Duplex DuplexNoTumble <<Duplex true /Tumble false>> setpagedevice
%%EndFeature
```

ou, pour valider le recto seul:

```
%%BeginFeature: *Duplex None <<Duplex false>> setpagedevice %%EndFeature
```

19 Informations concernant ce document

Sauf changements, les documents HOWTO sont placés sous copyright par leurs auteurs respectifs. Les documents Linux HOWTO peuvent être reproduits et distribués en tout ou partie, par quelque moyen physique que ce soit, sans l'autorisation de l'auteur. Les traductions et travaux dérivés sont également permis sans autorisation expresse. La distribution à titre commercial est permise et même encouragée; cependant, l'auteur souhaiterait en être averti.

En bref, nous souhaitons que l'information contenue dans ces document soit répandue le plus largement possible. Cependant, nous souhaitons maintenir le copyright sur ce document, et souhaiterions être avertis de toute re-distribution. Si vous avez des questions à ce propos, contactez Greg Hankins, le nouveau coordinateur des documents Linux HOWTO à l'adresse <gregh@cc.gatech.edu>. Son représentant français est Eric Dumas <dumas@freenix.fr,dumas@Linux.EU.Org>.