The Linux Printing HOWTO

Grant Taylor, <gtaylor+pht@picante.com>.
Traduzione di Luca Ronchi, <ronchi@xtreme.it>

Version 3.9, 1996/07/25 18:55:33

Questo documento contiene informazioni su come generare, vedere e stampare qualsiasi tipo di documento sotto Linux e su sistemi Unix in generale.

Contents

1 Introduzione

Questa e' una completa riscrittura e molte delle informazioni precedentemente incluse sono assenti. Questa e' stata una scelta voluta, visto che le ultime versioni superavano le 60 pagine. Pertanto, se non trovate qui le risposte che cercate, potete
a) controllare le versioni precedenti (in lingua inglese) presso la *The PHT Home Page* `http://www.picante.com/~gtaylor/pht/` , oppure
b) scrivermi cosa vi aspettavate di trovare.

La *Printing HOWTO Home Page* `http://www.picante.com/~gtaylor/pht/` e' il posto migliore dove trovare l'ultima versione di questo documento (NdT:in lingua inglese); ovviamente viene distribuito da SunSite, e dal vostro mirror LDP.

1.1 Storia delle versioni

Questa e' la terza generazione del PHT, cioe' la terza riscrittura completa. La storia del documento puo' essere riassunta cosi':

1. La prima versione e' stata scritta in risposta alle numerose domande inerenti la stampa, poste su comp.os.linux. Cio' anticipo' il progetto degli HOWTO di alcuni mesi, e fu pertanto il primo documento ad essere definito un 'howto'. Questa edizione era in Ascii piano.

2. Dopo essere entrato a far parte del progetto HOWTO, il PHT e' stato unito ad un FAQ su Lpd di Brian McCauley <`B.A.McCauley@bham.ac.uk`>; per circa due anni il PHT e' stato sviluppato in collaborazione. Ad un certo punto e' stato introdotto anche il lavoro di Karl Auer <`Karl.Auer@anu.edu.au`>. Questa generazione del PHT era in TexInfo e disponibile in PS, HTML, Ascii ed Info.

3. Dopo aver abbandonato il PHT per oltre un anno, ed aver cercato inutilmente di convincere qualcun altro ad occuparsene, e' stata scritta questa versione, in Linuxdoc-SGML.

1.2 Copyright originale

Il Copyright (c) 1996 appartiene a Grant Taylor. Copiate e distribuite liberamente questo documento, ma non modificatene il testo e non omettete il mio nome.

2 Come stampare

Se siete riusciti a configurare lpd per stampare sulla vostra stampante, o il vostro amministratore di sistema, o il vostro rivenditore, lo hanno fatto per voi, dovete solo imparare ad usare il comando lpr. Il *Printing Usage HOWTO* `http://sunsite.unc.edu/mdw/HOWTO/Printing-Usage-HOWTO.html`

tratta di questo e di qualche altro comando di manipolazione della coda di stampa.

3 I dispositivi di stampa del Kernel.

3.1 Il dispositivo lp

Il Kernel di Linux, dando per scontato che vi abbiate compilato, o caricato, il dispositivo lp (l'uscita di cat /proc/devices deve includere lp), fornisce i dispositivi /dev/lp0, /dev/lp1 e /dev/lp2. Questi non sono assegnati dinamicamente, ma corrispondono ad uno specifico indirizzo di I/O hardware. Cio' significa che la vostra stampante potrebbe essere lp0 o lp1, a seconda della configurazione hardware del vostro sistema.

Non e' possibile eseguire contemporaneamente i driver plip e lp su una data porta. E' possibile comunque caricare l'uno o l'altro driver, sia manualmente che automaticamente dal kerneld dei kernel versione 2 (o le ultime 1.3.x). Configurando adeguatamente gli interrupt e gli indirizzi di I/O dovrebbe essere possibile eseguire plip su una porta ed lp su un'altra. Qualcuno e' riuscito a farlo modificando il codice dei driver; si spera che qualcun altro riesca a farlo direttamente dalla linea di comando.

Esiste una utility chiamata ***tunelp*** con la quale e' possibile, come root, mettere a punto alcuni parametri di funzionamento del dispositivo lp, quali l'uso degli interrupt, la frequenza del polling e simili.

I kernel v2 ed alcuni v1.3.x accettano una opzione lp= che permette di settare gli interrupt e gli indirizzi di I/O:

Quando il driver lp e' compilato nel kernel, si puo' usare la linea di comando del LILO/LOADLIN per passare i parametri di funzionamento al driver.

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

Ad esempio: lp=0x378,0   oppure   lp=0x278,5,0x378,7 **
```

Si noti che se si usa questa tecnica, si devono specificare tutte le porte che devono essere considerate, non esistono parametri predefiniti. Si puo' disabilitare un driver con lp=0.

Quando il driver viene caricato come modulo in kernel v2 o v1.3.x, e' possibile specificare gli indirizzi di I/O e gli interrupt nella linea di comando di insmod (oppure in /etc/conf.modules) usando la solita sintassi. I parametri sono `io=port0,port1,port2` e `irq=irq0,irq1,irq2`. Consultate la pagina di man relativa a insmod per ulteriori informazioni.

**Per tutti quelli che non si ricordano gli indirizzi standard quando servono, sono quelli del secondo esempio. L'altra porta (*lp0*) e' a 0x3bc.

3.2 Dispositivi seriali

I dispositivi seriali vengono solitamente identificati come */dev/ttyS1* e simili sotto Linux. L'utility *stty* permette di vedere e modificare le impostazioni per una porta seriale; *setserial* permette di controllare alcuni parametri particolari e configurare gli interrupt e gli indirizzi di I/O per porte non standard. Ulteriori informazioni sulle porte seriali sotto Linux possono essere trovate nel *Serial-HOWTO* `http://sunsite.unc.edu/mdw/HOWTO/Serial-HOWTO.html` .

4 Principi di funzionamento.

Per far funzionare correttamente il sistema di stampa occorre prima capire come funzioni.

Lpd sta per Line Printer Daemon, e si riferisce, a seconda del contesto, sia al demone che all'intero insieme di programmi responsabili dello spooling e della stampa; questi sono:

lpd

Il demone di spooling. Una sua istanza controlla l'intero sistema, ed una ulteriore istanza e' attiva per ciascuna stampante in fase di stampa.

lpr

Il comando utente per la stampa. Lpr contatta lpd ed inserisce un nuovo elemento nella coda di stampa.

lpq

Fornisce la lista degli elementi della coda di stampa.

lpc

Il comando di controllo di lpd. Con lpc si possono attivare, disattivare, riordinare e, in generale, manipolare, le code di stampa.

lprm

`lprm` permette di rimuovere un elemento dalla coda di stampa.

Vediamo adesso come tutto cio' funziona. Quando il sistema viene avviato viene eseguito lpd. Lpd consulta il file */etc/printcap* per determinare per quali stampanti dovra' gestire le code di stampa. Ogni volta che un utente esegue lpr, lpr contatta lpd attraverso il socket denominato /dev/printer, e fornisce a lpd sia il file da stampare, sia alcune informazioni su chi sta stampando e come tale stampa deve essere eseguita. Lpd si occupa di stampare il file sulla stampante appropriata.

Il sistema lp e' stato progettato quando le stampanti erano perlopiu' stampanti di linea, cioe' adatte a stampare solo Ascii piano. Come sara' chiaro in seguito, e' pero' sufficiente qualche script in piu' per stampare PostScript, text, dvi e quant'altro.

5 Come configurare il tutto, fondamenti.

5.1 La configurazione tradizionale di lpd

La configurazione minima di lpd permette di ottenere un sistema in grado di accodare i file in una coda di stampa ed inviarli alla stampante. In questo caso non si pone alcuna attenzione al fatto che la stampante sia in grado di interpretare i file, ed in generale non si potranno ottenere stampe di qualita'. Comunque e' il primo passo per comprendere il funzionamento del sistema di stampa, quindi continuate a leggere.

Fondamentalmente, per aggiungere un a coda di stampa ad lpd, si deve aggiungere un elemento in */etc/printcap*, e creare una nuova directory di spooling in */var/spool/lpd*.

Un elemento in */etc/printcap* e' qualcosa del genere:

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

Questo definisce uno spool chiamato *lp*, *dj* o *deskjet*, il cui spooling viene eseguito nella directory */var/spool/lpd/dj*, senza nessun limite sulla dimensione massima dei file stampabili, che stampa sul dispositivo */dev/lp0*, e che non usa una pagina di intestazione (con il nome di chi ha eseguito la stampa e altre informazioni).

Adesso leggetivi la man page di printcap .

Quello sopra e' un esempio molto semplice, ma se non si manda in stampa un file che la Deskjet puo' capire si ottengono strani risultati. Stampando un normale file Unix si ottengono dei newline interpretati letteralmente, cioe' la seguente cosa:

```
Questa e' la prima riga.
                                        Questa e' la seconda riga.
                                                                          Questa e' la terza.
```

e cosi' via. Mandando in stampa un file PostScript si ottiene una bella lista di comandi PostScript con il solito effetto a gradinata, ma niente di piu' utile.

Ovviamente e' necessario qualcosa in piu': e' necessario un filtro. I lettori piu' attenti della man page del printcap avranno notato gli attributi *if* e *of* dello spool. Bene, *if*, cioe' il filtro di input e' quello che serve in questo caso.

Se scriviamo un piccolo script chiamato *filtro*, che aggiunge il carriage-return prima dei newline, l'effetto a gradinata puo' essere eliminato. A tal scopo dobbiamo aggiungere una riga nell'elemento del printcap illustrato prima:

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

Un semplice filtro potrebbe essere:

```
#!perl
while(<STDIN>){chop $_; print "$_\r\n";};
```

In questo modo otteniamo uno spool su cui possiamo stampare normali file di testo Unix ed ottenere delle pagine leggibili. Ci sono ovviamente molti altri modi di ottenere lo stesso risultato, ma questo e' uno dei piu' illustrativi. Siete comunque invitati a provarne altri.

L'unico problema e' che stampare solo testo piano non e' esattamente il massimo; certamente sarebbe meglio poter stampare anche PostScript o altre forme di testo formattato e grafica. Naturalmente e' possibile farlo, ed e' semplice. Il metodo da usare e' una semplice estensione di quello presentato sopra per risolvere il problema della gradinata. Se scriviamo un filtro che puo' accettare un qualsiasi file di input e trasformarlo in qualcosa di comprensibile per una DeskJet, il gioco e' fatto.

Questo tipo di filtro viene detto un *filtro magico*. Non preoccupatevi di scriverne uno da soli a meno che stampiate cose molto strane, ci sono molti ottimi filtri a disposizione in rete.

5.2 Permessi sui file

A grande richiesta includero' la lista dei permessi di accesso di alcuni file interessanti. Ci sono modi migliori di ottenere lo stesso risultato, idealmente usando solo eseguibili SGID e non rendendo tutto SUID root, ma questo e' il modo in cui il tutto era configurato dopo l'installazione e funziona. Sinceramente, se il vostro rivenditore non e' in grado di fornirvi un lpd che funziona, aspettatevi grossi problemi.

```
-r-sr-sr-x   1 root     lp    /usr/bin/lpr*
-r-sr-sr-x   1 root     lp    /usr/bin/lprm*
-rwxr--r--   1 root     root  /usr/sbin/lpd*
-r-xr-sr-x   1 root     lp    /usr/sbin/lpc*
drwxrwxr-x   4 root     lp    /var/spool/lpd/
drwxr-xr-x   2 root     lp    /var/spool/lpd/lp/
```

Lpd deve essere eseguito da root, in modo che possa connettersi alle porte riservate di lp. Dovrebbe probabilmente diventare UID lp.lp o qualcosa del genere dopo la connessione, ma non credo che lo faccia.

6 Come ottenere un filtro magico adeguato

6.1 Il filtro APS

Uno dei migliori filtri magici e' aps, di Andreas Klemm. La Linux Software Map fornisce le seguenti informazioni:

```
Begin3
Title:          apsfilter
Version:        4.9.1
Entered-date:   Montag, 10. Juli 1995, 21:22:35 Uhr MET DST
Description:    magicfilter for lpd with auto filetype detection
Keywords:       lpd magicfilter aps apsfilter
Author:         andreas@knobel.GUN.de (Andreas Klemm)
Maintained-by:  sledge@hammer.oche.de (Thomas Bueschgens)
                andreas@knobel.GUN.de (Andreas Klemm)
Primary-site:   sunsite.unc.edu
                /pub/Linux/system/Printing/
                211KB aps-491.tgz
Original-site:  ftp-i2.informatik.rwth-aachen.de
                /pub/Linux/pak/APSfilter/aps-491.tgz
Platforms:      C-Compiler, gs Postscript emulator, pbmutils
Copying-policy: GPL
End
```

Il filtro APS viene installato come if (filtro di input), e d e' in grado di tradurre molti tipi di file comuni in comandi appropriati per la vostra stampante. Tra gli altri aps comprende text, PostScript, dvi, e gif.

6.2 Filtri magici

Il filtro magico scritto da B.A.McCawley lavora all'incirca nello stesso modo di aps, ma essendo scritto in bash e' probabilmente piu' facile da modificare. La LSM fornisce:

```
Version: 0.4
Description: Bash scripts to simplify the task of configuring lpd to
             auto-detect file types using /etc/magic. (As advertised in the
             printing-HOWTO).
             This package uses existing filters so it only prints file types for
             which one already has filters (or compressed versions
             thereof). Alterative magic filter approches are described in the
             printing-HOWTO.
Keywords: lpd, /etc/magic, printing, filter
Author: B.A.McCauley@bham.ac.uk (Brian McCauley)
Maintained-by: B.A.McCauley@bham.ac.uk
Primary-site: tsx-11.mit.edu pub/linux/sources/usr.bin
              magic-filter-0.4.tar.gz  5709
Copying-policy: GNU GPL
```

6.3 Gli altri filtri magici

Per la massima confusione possibile, sembra che esistano due filtri magici chiamati semplicemente magicfilter (o forse e' lo stesso?)

```
Title:          magicfilter
Version:        1.1b
Entered-date:   04APR95
Description:    A customizable, extensible automatic printer filter.
                Lets you automatically detect and print just about any
                data type you can find a conversion utility for.  This
                filter is written in C and is controlled completely
```

```
                 from an external printer configuration file.
                 This version adds automagic creation of configuration
                 files based on the installed software on your system,
                 courtesy of GNU Autoconf.
                 This version is a bug fix from 1.1/1.1a; filters for
                 non-Ascii capable PostScript printers have been added.
Author:          H. Peter Anvin <Peter.Anvin@linux.org>
Primary-site:    sunsite.unc.edu
                 53000 /pub/Linux/system/Printing/magicfilter-1.1b.tar.gz
Copying-policy: GPL
```

7 Le stampanti seriali e lpd

7.1 Configurazione di printcap.

Lpd prevede cinque attributi da configurare in */etc/printcap/* per controllare il funzionamento della porta seriale a cui e' collegata la stampante. Leggete la man page di printcap e notate il significato degli attributi *br#*, *fc#*, *xc#*, *fs#* e *xs#*. Gli ultimi quattro sono mappe di bit che indicano i settaggi della porta. L'attributo *br#* e' semplicemente il baud rate, ad esempio '`br#9600`'.

E' molto semplice tradurre i settaggi da *stty* in attributi per printcap. Se ne avete bisogno date un'occhiata alla man page di stty adesso.

Usate stty per configurare la porta della stampante, in modo che possiate fare il cat di un file sulla porta ed ottenere una stampa corretta. Nel mio caso '*stty -a*' fornisce le seguenti informazioni:

```
dina:/usr/users/andy/work/lpd/lpd# 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
```

L'unica differenza tra questo ed il modo in cui la porta e' inizializzata al boot e' data da -clocal, -crtscts, e ixon. Per voi potrebbe essere anche differente, a seconda del modo in cui la vostra stampante fa il controllo di flusso.

In realta' l'uso di stty e' un po' strano. Infatti stty opera sul terminale collegato al suo standard input, lo si usa per manipolare una data porta seriale utilizzaando il carattere '<' come sopra.

Dopo aver configurato correttamente i settaggi di stty, in modo che '*cat file > /dev/ttyS2*' (nel mio caso) invia il file alla stampante, guardate il file /usr/src/linux/include/linux/termios.h. Contiene molti #defines ed alcune strutture (potreste farne il cat sulla stampante (ormai funziona, no?)). Andate alla sezione che comincia con

```
/* c_cflag bit meaning */
 #define CBAUD   0000017
```

Questa sezione elenca i significati dei bit *fc#* ed *fs#*. Noterete che i nomi (dopo il baud rate) coincidono con una delle linee dell'output di stty. Non e' forse facile?

Annotate quali di questi settaggi sono preceduti da un - nell'output di stty. Sommate tutti questi numeri (sono in ottale). Questo rappresenta i bit che dovete azzerare, quindi il risultato e' la vostra capacita' di *fc#*. Naturalmente ricordate che poi setterete i vari bit direttamente dopo averli azzerati, pertanto potete semplicemente usare '`fc#0177777`' (io lo faccio).

Ora fate lo stesso per quei settaggi (elencati in questa sezione) che non hanno un - davanti nell'output di stty. Nel mio esempio i piu' importanti sono CS8 (0000060), HUPCL (0002000), e CREAD (0000200). Controllate anche la flag relativa al baud rate (la mia e' 0000015). Sommate tutti questi numeri, nel mio esempio ottenendo 0002275. Questo numero viene messo nelle vostre capacita' di *fs#* ('`fs#02275`' funziona correttamente per me).

Fata la stessa cosa di azzeramento e settaggio dei bit nella successiva sezione del file di include, "c_lflag bits". Nel mio caso io non devo settare niente, quindi uso '`xc#0157777`' e '`xs#0`'.

7.2 Stampanti seriali antiquate che perdono caratteri.

Jon Luckey ha evidenziato come alcune stampanti seriali con interfacce da due lire e buffer molto piccoli, intendono veramente stop quando indicano cio' attraverso i meccanismi di controllo di flusso. Jon ha scoperto che disabilitando il buffer FIFO della porta seriale 16550 con *setserial* ha risolto il problema (in pratica e' come se si indicasse a Linux che la porta seriale e' una 8250).

8 Soluzioni commerciali.

Questa sezione e', per definizione, incompleta. Sentitevi liberi di inviare i dettagli della vostra distribuzione preferita.

8.1 RedHat 2.x

La RedHat fornisce un tool di amministrazione dotato di interfaccia grafica che puo aggiungere stampanti remote e stampanti locali. Permette di scegliere tra le stampanti supportate dal Ghostscript e i dispositivi su cui stampare, e quindi installa una coda di stampa in */etc/printcap* e un filtro magico PostScript-e-Ascii basato su `gs` e *nenscript* . Questa soluzione funziona piuttosto bene, ed e' semplicissima da configurare.

8.2 Altre distribuzioni

Per favore fatemi sapere cosa fanno le altre distribuzioni.

9 Come stampare su una stampante attraverso la rete.

Una delle capacita' di lpd e' che supporta la stampa attraverso la rete su stampanti fisicamente connesse ad altre macchine. Con una attenta combinazione di filtri e utility assortite, si puo' far stampare lpd su stampanti connesse a qualsiasi tipo di rete, in modo del tutto trasparente.

9.1 Su un host Unix/lpd

Per abilitare una macchina remota a stampare sulla vostra stampante, dovete elencare tale macchina in */etc/hosts.equiv* o */etc/hosts.lpd* (notate che mettere una macchina in *hosts.equiv* ha molti altri effetti; assicuratevi di sapere cosa state facendo prima di mettere una macchina qui). Potete anche abilitare solo certi utenti della macchina remota a stampare sulla vostra stampante utilizzando l'attributo *rs*; leggete la man page di *lpd* per informazioni.

9.2 Con lpd

Per stampare su un'altra macchina, dovete aggiungere ad */etc/printcap* una voce di questo tipo:

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

Notate che c'e' ancora la coda di stampa sulla macchina locale, gestita da *lpd*, in modo che se la macchina remota e' occupata o offline, i lavori di stampa possono essere accodati nell'area di spool fino a che possono essere inviati.

9.3 Con rlpr

Si puo' anche usare *rplr* per mandare un lavoro di stampa direttamente in una coda su una macchina remota, senza doversi preoccupare di configurare lpd per supportare la stampa remota. Cio' e' piu' utile quando stampate su un gran numero di stampanti solo occasionalmente. Dall'annuncio di *rlpr*:

Rlpr usa il TCP/IP per mandare lavori di stampa a server lpd ovunque su una rete.

Diversamente da lpr, non necessita che la stampante remota sia esplicitamente conosciuta alla macchina da cui si vuole stampare (ad esempio tramite */etc/printcap*) e pertanto e' molto piu' flessibile e richiede meno amministrazione.

Rlpr puo' essere usato ovunque si puo' usare un normale lpr, ed ' compatibile all'indietro con l'lpr BSD.

Il maggior vantaggio che si ottiene con rlpr e' la capacita' di stampare remotamente da qualsiasi macchina a qualsiasi macchina, senza preoccuparsi per come e' configurato il sistema su cui si desidera stampare. Inoltre puo' lavorare come un filtro, in modo che applicazioni cliente in esecuzione su una macchina remota, come Netscape, xemacs, etc, possono stampare sulla macchina locale senza sforzo.

Rlpr e' disponibile presso *SunSite* `ftp://sunsite.unc.edu/pub/Linux/system/Printing/` .

9.4 Su una stampante Win95, WinNT, LanManager o Samba.

E' possibile dirigere una coda di stampa di lpd, attraverso il programma *smbclient* (della suite samba), ad un servizio di stampa SMB basato su TCP/IP. Samba comprende uno script chiamato smbprint per fare questo. In breve, inserite un file di configurazione per la stampante in questione nella directory di spool, e installate *smbprint* come *if*.

In */etc/printcap* ci sara' qualcosa del genere:

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

Dovete leggere la documentazione all'interno dello script `smbprint` per ulteriori informazioni su come configurarlo.

E' anche possibile usare `smbclient` per passare un file direttamente al servizio di stampa SMB senza coinvolgere `lpd`. Controllate la man page.

9.5 Ad una stampante NetWare

La suite ncpfs comprende una utility chiamata `nprint` che fornisce le stesse funzionalita' di `smbprint`, ma per NetWare. Potete ottenere ncpfs da

`ftp://linux01.gwdg.de/pub/ncpfs/` . Dalla LSM per ncpfs versione 0.16:

> Con ncpfs potete montare volumi del vostro server NetWare sotto Linux. Inoltre potete stampare verso code di stampa NetWare e fare lo spool di code di stampa NetWare verso sistemi Linux. Avete bisogno del kernel 1.2.x, o 1.3.54 o superiori. ncpfs non funziona con kernel della serie 1.3.x inferiori alla 1.3.54.

Per far funzionare `nprint` attraverso lpd, serve un piccolo script della shell che stampa lo stdin sulla stampante NetWare, e che va installato come *if* per una coda di stampa di lpd. Dovreste avere qualcosa del tipo

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

mentre lo script per nprint sara' qualcosa del genere

```
#! /bin/sh
 |/usr/local/bin/nprint -S net -U name -P passwd -q printq-q -
```

9.6 Ad una stampante EtherTalk (Apple)

Il package netatalk comprende qualcosa sul genere di `nprint` e `smbclient`. Werner Eugster ha documentato la procedura per stampare da e verso una rete Apple molto meglio di quanto io potrei mai fare; date un'occhiata alla sua

web page `http://garnet.berkeley.edu/~weugster/appleprint.html` .

Problema misterioso della settimana: Netatalk non funziona con una scheda SMP Etherpower PCI dotata di chip DEC tulip.

9.7 Ad una stampante HP o altre stampanti ethernet.

Le HP ed alcune altre stampanti sono dotate di un'interfaccia ethernet verso la quale si puo' stampare direttamente usando lpd. Dovete seguire le istruzioni allegate alla stampante, o all'interfaccia di rete, ma in generale tali stampanti "eseguono" lpd, e forniscono una o piu' code su cui stampare. Una HP, per esempio, dovrebbe funzionare con un printcap del tipo

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

In una installazione su larga scala, specialmente in un ambito in cui alcune stampanti non supportano il PostScript, puo' essere conveniente configurare un server di stampa dedicato, verso il quale stampano tutte le macchine, e che esegue le varie istanze di ghostscript.

9.7.1 Sulle vecchie HP

Alcune stampanti (e stampanti messe in rete come "scatole nere") supportano solo una specie di non-protocollo basato su connessioni TCP. In questa categoria rientrano le prime schede JetDirect (comprese alcune JetDirecEx). Fondamentalmente, per stampare su stampanti di questo tipo, si deve aprire una connessione TCP su una porta specifica (di solito 9100) e trasmetterci il documento da stampare. Cio' puo' essere fatto, tra gli altri modi, in Perl:

```
#!/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;
}
```

9.8 Esecuzione di if per una stampante remota

Un difetto di lpd e' che l'*if* non e' eseguito per le stampanti remote. Se avete bisogno di eseguire un *if*, potete farlo configurando una coda di stampa doppia, e riaccodando il lavoro di stampa. Come esempio prendiamo questo printcap:

```
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:
```

alla luce di questo script di filtraggio filter-lj-5:

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

L'opzione *-U* di lpr funziona solamente se lpr e' esguito come demone, e pone il nome d'utente corretto nella seconda coda di stampa. Probabilmente e' meglio usare un metodo piu' robusto per determinare il nome dell'utente, perche' in alcuni casi non e' il quinto argomento. Verificate la man page di printcap .

10 Come stampare su un fax

10.1 Usando un faxmodem

Ci sono o una quantita' di programmi per fax che permettono di spedire e ricevere fax. Uno dei piu' complessi e' *HylaFax* di Sam Leffler, disponibile presso `ftp.sgi.com` Supporta di tutto, dall'uso di piu' modem al broadcasting.

E' anche disponibile, e probabilmente e' anche una scelta migliore per macchine Linux, ***efax*** , un semplice programma che spedisce fax. Il programma di getty `mgetty` puo' ricevere i fax (ed anche gestire la posta vocale su alcuni modem!).

10.2 Usare il servizio di Stampa Remota.

Esiste un servizio sperimentale che permette di spedire un messaggio di email contenente qualcosa che vorreste stampare, in modo che questo compaia su un fax da qualche altra parte. Essendo supportati anche formati come il postscript, questo e' un servizio che puo' essere utile, anche se la copertura mondiale e' a macchie. Guardate la *FAQ* `http://www.town.hall.org/fax/faq.html` .

11 Come generare qualcosa che valga la pena stampare

Qui abbiamo veramente un sacco di programmi. Basicamente Linux puo' eseguire quattro tipi di binari con vari livelli di successo: Linux, iBCS, Win16/win32s (con dosemu, e, un giorno, con Wine), e Mac/68k (con Executor). Tratteremo solo di applicazioni native per Linux e programmi comuni per Unix, facendo solo notare che WordPerfect per SCO, e molto probabilmente altri programmi commerciali di word processing, funzionano perfettamente sotto l'emulazione iBCS di Linux.

Per Linux, le scelte sono limitate a quelle disponibili per Unix in generale:

11.1 Linguaggi di Markup

La maggior parte dei linguaggi di markup serve per lavori ripetitivi, dove si desidera che sia il computer a controllare l'aspetto del testo, per rendere le cose uniformi. Fare un a bella firma in un linguaggio di markup e' sicuramente difficile...

nroff

Questo e' stato uno dei primi linguaggi di markup per Unix. Le man page sono l'esempio piu' comune di testo formattato con le macro *roff; molte persone lo gradiscono, ma per me nroff ha una sintassi piu' arcana del necessario, che lo rende inadeguato per nuovi lavori. Vale la pena sapere, comunque, che potete comporre una man page direttamente in PostScript con groff. La maggior parte dei comandi man lo fanno per voi con `man -t foo | lpr`.

TeX

TeX, ed il package di macro LaTeX, e' uno dei piu' usati linguaggi di markup in ambiente Unix. Molti lavori tecnici sono scritti in LaTeX perche' semplifica notevolmente i problemi di aspetto della pagina ed e' ancora uno dei pochi sistemi di text processing che supporta bene e completamente la simbologia matematica. TeX produce file `dvi`, che possono essere convertiti in PostScript oppure nel PCL dell'HP con `dvips` o `dvilj`.

SGML

C'e' almeno un parser sgml free per Unix e Linux; esso forma la base del sistema di documentazione Linuxdoc-SGML, ma supporta anche altri DTD.

HTML

Qualcuno sostiene che per lavori semplici potrebbe essere sufficiente scrivere in HTML e stampare usando Netscape. Non sono d'accordo, ma voi potreste pensarla diversamente.

11.2 WYSYWYG

C'e' una generale mancanza di programmi di produzione di testi WYSYWYG. Sentitevi liberi di scriverne qualcuno e fatemi sapere se ho omesso qualcosa.

Lyrix

Lyrix e' un'interfaccia (front-end) a LaTeX che sembra molto promettente.

The Andrew User Interface System

AUIS include `ez`, un editor in stile WYSYWYG con molte capacita' basiche per un word processor, capacita' HTML, e pieno supporto dei newsgroup e email MIME.

Proposte commerciali

Almeno Caldera e Red Hat distribuiscono package contenenti le solite applicazioni per ufficio, come un word processor WYSYWYG ed un foglio elettronico. Suppongo che facciano un buon lavoro, ma non li ho mai usati. (Credo anche che caldera fornisca le WABI della Sun, in tal modo potete probabilmente eseguire qualcosa tipo MS Office se dovete integrare tali tipi di files).

Jeff Phillips <`jeff@I_RATUS.org`> utilizza WordPerfect per Linux di Caldera, e dice che funziona bene. Apparentemente include il supporto per la stampante, come e' logico aspettarsi. Caldera dovrebbe avere delle informazioni su `http://www.caldera.com/` .

RedHat distribuisce una suite chiamata *Applixware*; potete trovare il loro sito web a `http://www.redhat.com/` .

12 Ghostscript

Ghostscript e' un programma incredibilmente importante per la stampa sotto Linux. La maggior parte del software di stampa per Unix genera PostScript, che e' un opzione da oltre mezzo milione su una stampante.

Ghostscript invece e' gratis (free) e genera il linguaggio della vostra stampante dal PostScript. Quando viene collegato con il filtro di ingresso di `lpd`, vi da' una stampante PostScript virtuale e semplifica la vita immensamente.

Ghostscript e' disponibile in due forme. La versione commerciale di Ghostscript, detta Aladdin Ghostscript, puo' essere usata liberamente per scopi personali, ma non puo' essere distrubuita nelle distribuzioni commerciali di Linux. E' ira un anno avanti alla versione free di Ghostscript; al momento supporta il Portable Document Format dell'Adobe Acrobat, mentre le versioni precedenti non lo fanno.

La versione free di Ghostscript e' il Ghostscript GNU, ed e' semplicemente una versione piu' vecchia del ghostscript Alladin gentilmente fornita alla GNU. (Grazie alla Alladin per questo accordo; piu' produttori di software dovrebbero supportare il software free in questo modo).

Il difetto principale di Ghostscript e' nei suoi font, che sono derivati dallo scan dei caratteri. Font piu' belli, ma che necessitano di licenza commerciale, possono essere ottenuti da diverse fonti, visto che Ghostscript puo' leggere i font Adobe tipo 1. La Adobe distribuisce dei font utilizzabili con il loro visualizzatore di PDF Acrobat, oppure potete usare i font della Adobe ATM o un fontpack.

Qualsiasi cosa facciate con *gs* , assicuratevi di eseguirlo con l'opzione che disabilita l'accesso ai file (`-dSAFER`). Il PostScript e' un linguaggio pienamente funzionale, ed un programma PostScript difettoso puo' procurarvi piu' di un mal di testa.

12.1 Regolazione dell'output di Ghostscript

Ci sono un sacco di cose che si possono fare se l'output di gs non e' soddisfacente (in realta' potete fare qualsiasi cosa, visto che avete i sorgenti).

12.1.1 Dimensioni e posizione dell'output

La posizione, la dimensione ed il rapporto altezza-larghezza dell'immagine sulla pagina dipendono dal driver specifico della stampante di ghostscript. Se vi accorgete che le vostre pagine sono troppo corte, o troppo lunghe, o troppo grandi di un fattore due, dovreste controllare il sorgente del driver della vostra stampante e modificare i parametri che vi sembrano pertinenti. Sfortunatamente ciascun driver e' differente, e quindi non posso dirvi esattamente cosa modificare, ma la maggior parte e' ragionevolmente ben commentata.

12.1.2 Gamma, dotsize, etc.

Molte stampanti non-laser hanno il problema che i singoli punti sono troppo grandi. Cio' da' luogo a stampe troppo scure. Se avete questo problema dovete usare una vostra funzione di trasferimento. Semplicemente create il seguente file nella lib-dir di ghostscript, ed aggiungete il suo nome alle chiamate di gs, appena prima del file vero e proprio. Potreste avere bisogno di modificare i valori per adeguarli alla vostra stampante. Valori piu' bassi danno stampe piu' chiare. Specialmente se il vostro driver usa l'algoritmo di Floyd-Steinberg per rasterizzare i colori, valori piu' bassi (0.2-0.15) sono una scelta migliore.

```
---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<---
```

E' anche possibile rimettere a posto stampanti che hanno qualche problema con il colore modificando questi valori. Se fate questo tipo di cose vi raccomando l'uso del file colorcir.ps, che viene fornito con ghostscript (nella sottodirectory example/), come pagina di prova.

13 Anteprima su schermo di immagini stampabili

Quasi tutto cio' che puo' essere stampato puo' anche essere visualizzato sullo schermo.

13.1 PostScript

Ghostscript ha un driver per X11, che viene sfruttato al meglio sotto il controllo di *Ghostview* . Le ultime versioni di questi programmi supportano anche i file PDF.

13.2 TeX dvi

I Device Indipendent file di TeX possono essere visti sotto X11 con *xdvi* . Le versioni moderne di xdvi chiamano ghostscript per rendere alcune particolarita' del PostScript.

Esiste un driver per VT100 che si chiama `dgvt`. `Tmview` lavora con Linux e la svgalib, se non avete niente di meglio.

14 Ringraziamenti

Le informazioni su smbprint provengono da un articolo di Marcel Roelofs <`marcel@paragon.nl`>.

Le informazioni su nprint per usare stampanti NetWare sono state fornite da Michael Smith <`mikes@bioch.ox.ac.uk`>.

La sezione sulle stampanti seriali proviene da Andrew Tefft <`teffta@engr.dnet.ge.com`>.

Il materiale su gamma e cose del genere per gs e' stato mandato da Andreas <`quasi@hub-fue.franken.de`>.

15 Note del traduttore.

Questa e' la prima traduzione di questo documento. E' quindi da considerarsi, per quanto riguarda lo stile e la lingua, poco piu' di una bozza. Le informazioni contenute dovrebbero comunque essere corrette.

Per commenti, critiche e quant'altro rivolegetevi direttamente al *PLUTO* `http://www.dei.unipd.it/it/linux/pluto/intro.html` , in quanto non mi sara' piu' possibile occuparmi di questo documento.