The Linux Printing Usage HOWTO – svensk översättning

av Mark Komarinski <markk@auratek.com>. Svensk översättning: Peter Nilsson <pnidv96@student.hv.se>

1 Inledning

I detta dokument beskrivs den utskriftshanterare (printer spooling system), som följer med operativsystemet Linux. Denna "HOWTO" utgör ett komplement till "The Linux Printing Setup HOWTO" (svensk översättning saknas f.n.), vari installation och konfiguration av Linux utskriftshanterare behandlas. Den information som här presenteras, torde även vara relevant för operativsystemet BSD.

1.1 Dokumentets historia

Anmärkning från Mark Komarinski <markk@auratek.com>:

Jag vill tacka Matt Foster för hans hårda arbete med omarbetningen av denna HOWTO. Jag fortsätter i hans anda, och gör tillägg när så är nödvändigt för att hålla dokumentet aktuellt.

Anmärkning från Matt Foster <mwf@engr.uark.edu>

Denna version av "Linux HOWTO—Att Skriva Ut" är en fullständig omarbetning av den HOWTO som ursprungligen skrevs av Grant Taylor <grant@god.tufts.edu> och Brian McCauley <B.A.McCauley@bham.ac.uk>. Jag har försökt få med allt material som Grant och Brians HOWTO täcker, men jag har drastiskt förändrat det sätt på vilket materialet presenteras samt mängden information som tagits med. Jag tror mig härigenom ha gjort dokumentet mer lättläst och komplett. Jag kan bara hoppas att du, som läsare, instämmer i det.

1.2 Versionshistorik

v1.2.2

- Indexeringsändringar samt några ändringar för att passa in i den nya RedHat-dokumentationen.

v1.2.1

- Uppdateringar, några ändringar för publiceringen av Dr. Linux

v1.2

- Windows-skrivare
- Hur man ändrar maximal storlek för filer som skall skrivas ut

v1.11

- Ny ansvarig (för dokumentet)!
- Lade till information om lpc

- Lade till en del information om felsökning
- Påbörjade avsnitt om utskrift av grafikfiler!

v1.1

- Ändrade ordalydelsen på några ställen
- Skrev avsnittet om PostScript-utskrifter
- Försökte klargöra några exempel 8-)
- Utökade drastiskt diskussionen om de grundläggande utskriftsverktygen för Linux

v1.0

- Första versionen som släppts till allmänheten

1.3 Upphovsrätt och varumärken

1.4 Att hämta hem HOWTO-dokumenten om utskrift

Om du vill skriva ut detta dokument, rekommenderar jag dig att hämta PostScript-versionen. Den är utformad på ett estetiskt tilltalande sätt och dessutom mer lättläst. Det går att hämta PostScript-versionen från någon av de många platser, där Linux distribueras (som t.ex. SunSITE `<ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/>`).

1.5 Gensvar

Frågor, kommentarer eller rättelser gällande denna HOWTO kan skickas till <markk@auratek.com>. (Vad gäller i första hand språkliga kommentarer och förslag till förbättringar av den svenska utgåvan, tas dessa med glädje emot av Peter Nilsson `$<$pnidv96@student.hv.se$>$`.)

1.6 Tillkännagivanden

Ett stort tack till alla dem som läste alfa-versionen av denna HOWTO och gav många kommentarer och förslag, vilka varit till stor hjälp; några kan finna resultatet av sin medverkan i denna version.

Jag vill också rikta ett tack till Matt Foster, som gjorde den ursprungliga omarbetningen.

2 Att skriva ut under Linux

I detta kapitel beskrivs hur man skriver ut filer, hur man gör för att titta på skrivarkön, hur utskriftsjobb raderas från skrivarkön, hur filer formateras före utskrift samt hur “utskriftsmiljön” ställs in.

2.1 Historik

Den utskriftshanterare som används under Linux - det s.k. lp-systemet - är en anpassning av den källkod som utvecklats vid University of California till Berkley-distributionen av UNIX (BSD).

2.2 Att skriva ut en fil med hjälp av *lpr*

Det allra enklaste sättet att skriva ut något i Linux, är att sända den fil som skall skrivas ut direkt till skrivarenheten. Detta kan t.ex. göras genom att använda kommandot *cat*. Som `root`, skulle man kunna skriva

```
# cat thesis.txt > /dev/lp
```

I det här fallet, är `/dev/lp` en symbolisk länk till den verkliga skrivarenheten—må det vara en matrisskrivare, laserskrivare, typsättningsmaskin eller plotter. (Se *ln*(1) för information om symboliska länkar.)

Av säkerhetsskäl, har endast `root` och de användare som är medlemmar av samma grupp som skrivar-demonen rätt att skriva direkt till skrivaren. Därför måste kommandon som *lpr*, *lprm* och *lpq* användas för att komma åt skrivaren.

lpr används för att skriva ut en fil. Kommandot *lpr* ansvarar för det inledande arbete, som krävs för att filen skall kunna skrivas ut, och överlämnar därefter kontrollen till ett annat program, *lpd*, skrivar-daemonen. Skrivar-daemonen “talar om” för skrivaren hur filen skall skrivas ut.

När *lpr* exekveras, kopieras först den angivna filen till en särskild katalog (spool-katalogen, eller “the spool directory”), där den sedan finns tills *lpd* skriver ut den. Då *lpd* får reda på att det finns en fil att skriva ut, exekverar den en kopia av sig själv (något som vi programmerare kallar “delning” eller “forking”). Denna kopia av *lpd* skriver ut filen, medan den ursprungliga kopian väntar på ytterligare förfrågningar. På detta sätt kan flera skrivarjobb köa samtidigt.

Syntaxen för *lpr*(1), är välkänd,

```
$ lpr [ val ] [ filnanm ... ]
```

Om inget `filnamn` anges, förväntas inmatningen till *lpr* komma från standardenheten för inmatning (vanligen tangentbordet, eller utmatningen från ett annat program). Detta ger användaren möjlighet att omdirigera utmatningen från ett kommando till utskriftshanteraren, t.ex. såhär:

```
$ cat thesis.txt | lpr
```

eller

```
$ pr -l60 thesis.txt | lpr
```

lpr förstår ett flertal kommandoradsargument, vilket möjliggör för användaren att kontrollera dess funktion. Några av de vanligaste argumenten är dessa: **-P**skrivare anger vilken skrivare som skall användas, **-h** gör att ingen titelsida skrivs ut, **-s** skapar en symbolisk länk istället för att kopiera filen till spool-katalogen (användbart för stora filer) och -#nummer anger antal kopior som skall skrivas ut. Ett exempel på hur *lpr* kan användas är:

```
$ lpr -#2 -sP dj thesis.txt
```

Detta kommando skapar en symbolisk länk till filen `thesis.txt` i spool-katalogen för skrivaren *dj*, där *lpd* tar hand om filen. Dessutom skrivs en andra kopia av filen `thesis.txt` ut.

Se *lpr*(1) för en lista med alla alternativ som *lpr* förstår.

2.3 Att titta på skrivarkön med *lpq*

Använd kommandot *lpq*, för att titta på innehållet i skrivarkön. Om det körs utan argument, visas standardskrivarens kö.

Utmatningen från *lpq* är användbar till mycket.

```
$ lpq
lp is ready and printing
Rank   Owner      Job  Files                              Total Size
active mwf        31   thesis.txt                         682048 bytes
```

2.4 Att avbryta ett skrivarjobb med *lprm*

En annan användbar finess hos en utskriftshanterare är möjligheten att avbryta ett skrivarjobb som är placerat i kö. Använd *lprm* för att göra det.

```
$ lprm -
```

Kommandot ovan avbryter alla skrivarjobb som ägs av den användare som kör det. Ett enstaka jobb kan avbrytas genom att först ta reda på jobbets ID-nummer med *lpq*, och därefter ange det på kommandoraden till *lprm*. Till exempel skulle

```
$ lprm 31
```

avbryta jobb nummer 31 (`thesis.txt`) på standardskrivaren.

2.5 Att styra *lpd* med *lpc*

Programmet *lpc* används för att styra de skrivare som *lpd* kontrollerar. Du kan aktivera eller deaktivera en skrivare eller dess köer, förändra ordningen mellan jobb i en kö samt erhålla en statusrapport över skrivare och köer. *lpc* används oftast när flera skrivare är kopplade till en och samma dator.

```
$ lpc
```

Detta startar normalt *lpc* i ett interaktivt läge, i vilket det går att ge kommandon till programmet. Du kan också ge *lpc*-kommandon direkt på kommandoraden:

```
$ lpc status all
```

En lista med tillgängliga kommandon finns på manualsidan till *lpd*, men här ges några viktiga kommandon, som du bör känna till. Om kommandot innehåller ordet *val*, kan det efterföljas av ett skrivarnamn (lp, print, etc.) eller ordet `all`, vilket betyder alla skrivare.

- disable *val* - gör att inga nya skrivarjobb kan placeras i kö
- down *val* - deaktiverar denna skrivare
- enable *val* - tillåter nya skrivarjobb att placeras i kön
- quit (eller exit) - avslutar *lpc*
- restart *val* - startar om *lpd* för skrivaren
- status *val* - visar skrivarens status
- up *val* - slår på utskrift och startar lpd på nytt

2.6 RedHats printtool

Bara en kort kommentar här angående RedHats förbluffande printtool-program. Det verkar göra allt som ett magiskt filter gör. RedHat installerar redan många av de program, som utför själva filtreringen. Jag beskriver här hur mina skrivarinställningar är under RH 4.0 med en HP LJ 4L ansluten till parallellporten (samma sak torde gälla för andra versioner av RedHat också).

- Bli root och plocka fram printtool (om du använde su, glömde du inte SETENV DISPLAY :0.0 och xhost +, eller?)
- Klocka på ”Add” och välj ”OK” för en lokal skrivare.
- Fyll i skrivarenheten (/dev/lp1 för mig).
- Fyll i infiltret; välj skrivartyp, upplösning och pappersstorlek (ljet4, 300x300 och letter).
- Välj ”OK” hela vägen tillbaka och starta om lpd.

Precis som när man pillar med en /etc/printcap för hand, går det att ha flera skrivardefinitioner för varje fysisk skrivare. En för annan upplösning, pappersstorlek etcetra.

3 Utskrift av filer

Detta avsnitt innehåller information om hur du skriver ut olika filer, som du kommer att komma i kontakt med i ett Linuxsystem.

3.1 Att skriva ut grafik

Hur man gör för att skriva ut grafikfiler beror dels på vilken typ av grafik som används och dels på typen av skrivare. Matrisskrivare är oftast inte användbara i detta sammanhang, p.g.a. deras sätt att hantera grafik på. Möjligen kan du, om skrivaren är kompatibel med en Epson eller IBM ProPrinter, konvertera filen till PostScript-format och därefter använda Ghostscript (se nästa avsnitt) för att skriva ut den.

Om du har en laserskrivare, blir saker och ting betydligt enklare, eftersom sådana vanligen är kompatibla med PCL. Detta ger dig ett par olika möjligheter. Vissa program kan direkt göra om filer till PCL. Om så inte är fallet, kan program som *NetPBM* användas för att konvertera till PCL. Slutligen kan du använda Ghostscript (se nästa avsnitt).

Det bästa är att installera programpaket som NetPBM eller Ghostscript och använda ett s.k. magiskt filter ("magic filter" på engelska) som tar hand om grafikfilerna automagiskt.

3.2 Utskrift av PostScript-filer

Att skriva ut PostScript på en skrivare som har en inbyggd PostScript-tolk, är inte svårt; använd bara *lpr*, så sköter skrivaren detaljerna. Vi andra, som inte har en PostScript-skrivare, får välja ett annat tillvägagångssätt. Lyckligtvis finns det program som förstår PostScript och kan översätta det till ett språk som de flesta skrivare förstår. Troligen är det mest kända av dessa program Ghostscript.

Ghostscripts uppgift är att översätta styrkoderna i en PostScript-fil till kommandon som skrivaren förstår. För att skriva ut en fil med Ghostscript, kan du skriva ungefär såhär:

```
$ gs -dSAFER -dNOPAUSE -sDEVICE=deskjet -sOutputFile=\|lpr thesis.ps
```

Observera att utmatningen från Ghostscript faktiskt sänds till *lpr* i exemplet ovan. Det beror på att flaggan **sOutputFile** används.

Ghostview är ett gränssnitt till Ghostscript för X Window System. Detta program möjliggör förhandsgranskning av en PostScript-fil innan den skrivs ut. Ghostview och Ghostscript kan båda hämtas från `<ftp://prep.ai.mit.edu/pub/gnu/>`.

3.3 Att skriva ut PDF-filer

Adobe har släppt en Acrobat-läsare för Linux. Denna är tillgänglig på Adobes hemsida, `<http://www.adobe.com>`. Dess företrädare, xpdf, finns även att tillgå. Båda torde skriva till en PostScript-enhet.

3.4 Att skriva ut TeX-filer

Ett av de enklaste sätten att skriva ut TeX-filer, är att först göra om dem till PostScript och därefter skriva ut dem med hjälp av Ghostscript. För att göra det, måste de först konverteras från TeX till ett format som kallas DVI (vilket står för "device independent"). Det görs med kommandot *tex*(1). Sedan konverteras DVI-filen till PostScript med *dvips*. Detta ser ut som följer, när det skrivs in:

```
$ tex thesis.tex
$ dvips thesis.dvi
```

Nu kan resultatet skrivas ut på så sätt som beskrivits ovan.

3.5 Utskrift av filer som formateras med `troff`

```
$ groff -Tascii thesis.tr | lpr
```

eller, om du så vill,

```
$ groff thesis.tr > thesis.ps
```

Skriv sedan ut PostScriptfilen som ovan.

(Översättarens anmärkning: Om filen innehåller svenska tecken, skall flaggan "-Tlatin1" användas.)

3.6 Att skriva ut `manual`sidor

```
$ man man | col -b | lpr
```

Manualsidorna innehåller förformaterad `troff`-data, så fetstil, understrykningar etc. måste tas bort. Kommandot "col" gör detta på ett trevligt sätt, och eftersom vi sänder utmatningen från man direkt till ett annat program via ett rör, så körs inte more.

4 Diverse ting

Detta avsnitt innehåller information som inte finns någon annanstans.

4.1 Att formatera före utskrift

Då de flesta ASCII-filer ej är formaterade för utskrift, kan det vara bra att formatera dem på något sätt innan de skickas till skrivaren. Det kan t.ex. innebära att placera titel och sidnummer på varje sida, sätta marginaler, skriva texten med dubbelt radavstånd, formatera med indrag och/eller skriva texten i flera spalter. Ett vanligt sätt att göra detta, är att använda en s.k. förprocessor, som exempelvis *pr*.

```
$ pr +4 -d -h"Doktorsavhandling, andra utkastet" -l60 thesis.txt | lpr
```

I exemplet ovan tar *pr* innehållet i filen `thesis.txt` och hoppar över de tre första sidorna (+4), sätter sidlängden till 60 rader (-l60), använder dubbelt radavstånd (-d) och placerar texten "Doktorsavhandling, andra utkastet" överst på varje sida (-h). Utmatningen från pr placeras så i skrivarkön av *lpr*. Se manualsidan, om du vill ha ytterligare information om *pr*.

4.2 Miljövariabeln PRINTER

Alla kommandon som ingår i Linux utskriftshanterare accepterar flaggan -**P**. Denna flagga tillåter användaren att bestämma vilken skrivare som skall användas. Om skrivare inte anges, kommer utmatningen att skickas till standardskrivaren.

Istället för att var gång behöva ange skrivare, kan du sätta miljövariabeln PRINTER till namnet på den skrivare du vill använda. Detta görs på olika sätt i olika skal. I *bash* görs det med

```
$ PRINTER="skrivar_namn"; export PRINTER
```

och i *csh*, skriver du kommandot

```
% setenv PRINTER "skrivar_namn"
```

Dessa kommandon kan placeras i ditt loginskript (`.profile` för *bash*, eller `.cshrc` för *csh*), eller ges på kommandoraden. (Se *bash*(1) och *csh*(1) för ytterligare information om miljövariabler.)

5 Svar på ofta ställda frågor

F1. Hur undviker jag "trappstegseffekten" (staircase på engelska)?

S1. "Trappstegseffekten" orsakas av det sätt på vilket vissa skrivare vill att radslut skall markeras. Somliga skrivare vill ha radslut markerade med vagnretur/radmatning (DOS-varianten) istället för endast radmatning, som UNIX-system använder. Enklaste sättet att åtgärda problemet är att undersöka om skrivaren kan växla mellan de två varianterna på något sätt—antingen genom att ändra en DIP-switch, eller att skicka en kontrollsekvens i början av varje utskrift. För att göra det senare, måste du skapa ett filter (se F2).

Ett snabbt sätt att fixa det på är att använda ett filter på kommandoraden. Ett exempel är:

```
$ cat thesis.txt | todos | lpr
```

F2. Vad är ett filter?

S2. Ett filter är ett program som läser från standardinmatningen (*stdin*), bearbetar den inlästa datan och därefter skriver till standardutmatningen (*stdout*). Filter används till många saker, bl.a. bearbetning av text.

F3. Vad är ett *magiskt* filter?

S3. Ett magiskt filter är ett filter som utför olika saker beroende på filens typ. Om filen t.ex. är en textfil, skrivs den ut som vanligt. Om filen är en PostScript-fil, eller i något annat format, skrivs den ut på något annat sätt (Ghostscript). Två exempel på magiska filter är magicfilter och APSfilter. En nackdel med dessa är att de nödvändiga programmen måste intalleras före filtret.

Anledningen till det är att filtret, när det installeras, undersöker vilka program som finns installerade (t.ex. Ghostscript—om det återfinns, vet filtret att PostScript kan hanteras). Filtret skapar därefter sig själv på grundval av de program som hittats:

- GhostScript
- TeX
- NetPBM
- jpeg-verktyg
- gzip

F4. Kommer Linux att kunna samarbeta med Windows utskriftshanterare?

S4. Kanske. Skrivare som endast hanterar WPS-kommandon, kommer inte att understödjas av Linux. Skrivare som hanterar WPS- och andra kommandon (vilket exempelvis Canon BJC 610 gör) fungerar, så länge de är inställda på något annat än WPS. Andra skrivare, som en del HP DeskJet 820C0i/Cse, kommer *inte* att fungera med Linux. Däremot kan Linux fungera som en skrivarserver (se Samba) för Win95-datorer, eftersom Win95 har drivrutiner för sådana skrivare.

F5. Vad är det här för jävla system? Jag kan inte skriva ut mer än sex sidor utan att få "file too large".

S5. En av inställningarna i filen `/etc/printcap` har att göra med den maximala storleken på en fil som skall skrivas ut. Standardvärdet är 1000 block på disken (ca 500 k?). Vad gäller PostScript-filer och liknande, innebär detta i storleksordningen 6–8 sidr (inklusive grafik). Lägg till följande rad i skrivardefinitionen:

```
mx=0
```

Huvudskälet till begränsningen är att den partition som spool-katalogen finns på inte skall bli full. Det finns ett ytterligare sätt att kringgå begränsningen, nämligen genom att låta *lpr* skapa en symbolisk länk från spool-katalogen till filen som skall skrivas ut. Du måste då komma ihåg flaggan `-s` var gång du kör *lpr*.

6 Felsökning

I detta avsnitt behandlas några vanliga fel som kan uppkomma i samband med användning av utskrift-shanteraren.

Om skrivaren inte fungerar:

- Fungerar andra skrivarjobb? (problem med applikationen?)
- Är lpd igång? (kontrollera med lpc) (print controller?)
- Kan `root` skicka något direkt till skrivaren? (utskriftsrutinerna?)
- Kan du göra utskrifter från DOS? (kabel-/skrivarproblem?)

Svaren på dessa frågor kan hjälpa dig att finna en lösning.

Skicka ytterligare förslag gällande detta avsnitt till <markk@auratek.com>.

7 Referenser

Detta avsnitt innehåller referenser vad gäller Linux utskriftshanterare. Jag har försökt hålla avsnittet så inriktat på ämnet som möjligt. Om du tycker att jag har glömt någon viktig referens, så tveka inte att kontakta mig.

Betänk följande, innan du postar en fråga till en USENET-grupp:

- Tar skrivaren emot skrivarjobb? (Använd *lpc*(8) för att kontrollera detta.)
- Kan du finna svaret på din fråga i denna HOWTO eller Grant Taylors “Printing HOWTO”?

Om någon av ovanstående frågor kan besvaras med “ja”, kanske du skall tänka efter två gånger innan du skickar frågan. Och när du slutligen skickar frågan till en nyhetsgrupp, försök då att få med all information som behövs. Fråga inte bara något i stil med “Jag har problem med lpr. Snälla, hjälp mig!” Den typen av brev, kommer de flesta definitivt att hoppa över. Ange även vilken version av kärnan du använder, på vilket sätt felet inträffade och, om det finns, exakt det felmeddelande som systemet gav.

Manualsidor

- *cat*(1) slå samman och skriv filer
- *dvips*(1) konvertera en DVI-fil från TeX till PostScript
- *ghostview*(1) Visa PostScript-dokument med hjälp av Ghostscript
- *groff*(1) “framsida” till formateringssystemet groff
- *gs*(1) Ghostscripttolk/-visare
- *lpc*(8) styrprogram för skrivarna

- *lpd*(8) skrivardemon
- *lpq*(1) program för att titta på skrivarköer
- *lpr*(1) skriv ut i bakgrunden
- *lprm*(1) ta bort jobb från skrivarköer
- *pr*(1) konvertera textfiler för utskrift
- *tex*(1) textformaterings- och typpsättningsprogram

USENET nyhetsgrupper

- `comp.os.linux.*` varierande information om Linux
- `comp.unix.*` UNIX-relaterade diskussioner