

### 第21章 打印机

( 欢迎提出意见和建议：frebsdhandbook@163.com )

在这一章，我们将看看一些在FreeBSD中使用打印机的问题。作为一个用户，你不需要直接访问打印机。而是，用一连串叫做*spooler*的进程管理打印数据。一个进程，*lpr*，把用户的打印数据写到磁盘上，另一个，*lpd*，拷贝打印数据到打印机。这个方法启用进程来写入打印数据，即使打印机很忙，仍然可以确保打印机的可用性。

在这节，我们将主要看看设置打印机时你需要做些什么。更多细节，看看有关打印的在线手册，这节也是直接源自于那儿。

*lpd*是重要的*spooler*进程。它负责很多事情：

- 它控制访问连接的本地和远程的打印机，
- 它允许用户提交文件进行打印。这些提交被叫做*job*。
- 它阻止用户通过为每个打印机维护一个队列来同时访问打印机。
- 它可以打印*header page*，也叫做*banner*或*burst page*，所以用户可以很容易地从一堆打印输出中找到他们打印的工作
- 它获取连接到串行口的打印机的通信参数。
- 它可以通过网络发送工作到另一个主机的另一个*spooler*。
- 它可以运行特殊的过滤器来格式化将要打印的工作，以适应几种打印语言或打印兼容性。
- 它可以计算打印机的使用情况。

通过一个配置文件，和通过提供特殊的过滤程序，你可以启用*spooler*来处理所有的东西或上面提到的中的一部分，以支持很多的打印机硬件。

如果你是系统的唯一用户，这可能听起来有点过头。它可以直接访问打印机，但那不是一个好主意：

- *Spooler*打印工作在后台运行。你不必等待数据被拷贝到打印机。
- *Spooler*可以方便地通过过滤器运行一个将被打印的工作来添加*header*或转换特殊格式（如*PostScript*）成一个打印机能够理解的格式。
- 绝大多数程序提供了一个除了在你的系统上呼叫*spooler*以外的打印特性。

#### 打印机配置

现在，绝大多数打印机通过并口来连接。并口能够与打印机快速通信，最高达到每秒100,000字节，而串行打印机很少能够传输超过每秒1,920个字符的速度。

比较老的UNIX系统经常使用串行打印机，但它们不再经常使用。看看手册中有关串行打印机特性的文章。现代的打印机可以有一个以太网接口，可以用它来立刻把打印机连接到几台机器上。

直接连接一台并口打印机是很简单的：在打印机和计算机之间连上电缆线。你不需要作任何调整。如果有不止一个并行接口，你必须决定使用哪一个。并口打印设备被叫做`/dev/lptn`，这里n是数字，从0开始。看看表15-4了解更多细节。

你不需要任何特别的事情来配置行式打印机驱动程序`lpt`：它默认是在内核中。当然，如果你先前删除了它，你将必须添加进去。看看联机手册连接更多细节。

### 测试打印机

当你已经连接到打印机并且已经打开电源，如果有测试程序，可以运行厂商提供的测试程序。典型的，有一个产生一个描述打印机特性的输出的功能。接着，检查计算机与打印机之间的通信。

```
# lptest > /dev/lpt0
```

如果有一个PostScript打印机，就不能打印任何其他东西，你不能得到任何输出。但你会在状态显示上看到一些反应。看看在线手册中有关测试PostScript打印机的描述。

### 配置/etc/printcap

下一步是配置重要的配置文件`/etc/printcap`。这个文件不容易读，但不久，你就可以习惯使用它了。这儿是一些典型的记录：

```
lp|lj|ps|local LaserJet 6MP printer:\
:lp=/dev/lpt0:sd=/var/spool/output/lpd:lf=/var/log/lpd-errs:sh:mx#0:\
:if=/usr/local/libexec/lpfilter:\
rlp|sample remote printer:\
:rm=freebie:sd=/var/spool/output/freebie:lf=/var/log/lpd-errs:\
:rp=lp:
```

让我们仔细地看看这个：

- 所有的字段是由冒号（:）隔开。
- 继续连接到下一行要求一个反斜杠字符（\）。特别注意在继续行的结尾和下一行的开头要有一个冒号。
- 当与`lpr`或`lpd`会话时，每个记录的第一行指定许多你可以用来指定这个打印机的名称。名称是由竖线条隔开的。通常，最后一个名称是一个更详细的描述，你通常不需要使用它与程序会话。
- 下一个字段描述功能，描述如何做事。功能是由一个两个字母的关键字加上一个可选的参数来描述的，这通过一个分割符来单独地指出参数的类型。如果字段获得一个字符参数，分割符是=，如果它获得一个数字值，分割符是#。你将在联机手册中发现一个完整的描述。
- 第一个记录定义了一个本地打印机，叫做`lp`、`lj`、`ps`和`local LaserJet 6MP printer`。为什么有这么长的名称呢？`lp`是默认的，所以你将通常使用它。`Lj`是当前被用来与理解HP LaserJet语言(现在是PCL)的打印机进行会话，`ps`可能被用来与理解PostScript的打印机进行

会话。最后的名称是一个更详细的描述。

- 记录 `lp=/dev/lpt0` 告诉 `spooler` 打印机连接的物理设备的名称。远程打印机没有物理设备。
- `sd` 告诉 `spooler` 存储等待打印工作的目录。这个目录必须存在，`spooler` 不会自己创建它。
- `lf=/var/log/lpd-errs` 指定存储日志错误的文件名。
- `sh` 是一个告诉 `lpd` 忽略 header page 的标记。如果你没有，每个工作将产生一个描述页。在一个小环境中，这没有大的意义，只是浪费一些纸而已。
- 参数 `mx` 用千字节告诉 `lpd` 一个 `spool` 工作的最大大小。如果工作比这个值大，`lpd` 拒绝打印它。在我们的例子中，我们没有限制这个值。我们通过把 `mx` 设置成 0 来这样做。
- `if` 告诉 `lpd` 在打印之前给工作应用一个过滤器。我们将在下面看到。
- 在远程打印机记录中，`rm=freebie` 告诉 `lpd` 发送数据给叫做 `freebie` 的机器。当然，这可能是一个完整的域名。
- 在远程打印机记录中，`rp=lp` 告诉 `lpd` 在远程机器上的打印机的名称。这不一定要与本地机器的名称一样。

### 远程打印

在一个网络中，你不需要在每一台机器上都有一台打印机。你可以在同样的网络上在另一台机器上打印。正如我们从上面的例子 `/etc/printcap` 中看到的，`spooler` 支持这个配置。

有很多事情需要考虑：

- 有两台机器与远程打印有关，客户端机器与服务端机器（远程）。我们已经在上面看到了一台典型的客户端记录。你用 `rm` 功能来指定服务端机器的名称，你用 `rp` 功能来指定打印机的名称。你不能用 `lp` 功能来指定设备的名称。
- 在服务端机器上，你不需要用 `/etc/printcap` 文件做任何特殊的事情。当然，你需要在客户端机器的 `rp` 记录中为打印机指定一个记录。
- 在服务器机器上，你必须允许 `spooler` 从客户端机器上访问。在 `/etc/hosts.lpd` 文件中添加一行机器的名称。

这样就可以了！

### Spooler过滤器

大概在先前配置文件中的最不能理解的记录是 `if` 记录。它指定了一个输入过滤器的名称，在打印之前通过 `lpd` 来传送完整的打印数据。

这样做是为什么呢？可能有很多原因。也许你有不适合打印的数据格式。例如，它可能是 PostScript，而你的打印机可能不理解 PostScript。或它可能是用其他方式：你的打印机只理解 PostScript，而输入不是 PostScript。

## The Complete FreeBSD(4th)

有很多原因要求使用一个过滤器：绝大多数打印机仍模拟旧的电传打字机，所以它们要求有一个返回字符( Ctrl-M或`^M`)来重起一行，一个新行的字符( Ctrl-J或`^J`)提前到下一行。UNIX只使用`^J`，所以如果你拷贝数据到那儿，你很容易看到一个阶梯效应。例如，`ps`可能会显示：

```
$ ps
PID TT STAT TIME COMMAND
2252 p1 Ss 0:01.35 /bin/bash
2287 p1 IW 0:04.77 e /etc/printcap
2346 p1 R+ 0:00.05 ps
```

当你设法打印它的时候，你得到这样的结果：

```
PID TT STAT TIME COMMAND
                2252 p1 Ss 0:01.35 /bin/bash
                2287 p1 IW 0
```

这页的其余部分是空的：你已经跑到右边去了。有很多方法可以来解决这个问题：

- 你可以配置你的打印机来中断Ctrl-J产生新行或回车，避免使用Ctrl-M。检查你的打印机手册。
- 你可以给出一个控制顺序给你的打印机告诉它中断Ctrl-J避免产生新行和回车到行的开头，避免使用Ctrl-M。例如，如果你给HP LaserJet和兼容打印机发送控制顺序ESC&k2G，它们将这样做。
- 你可以写一个转换打印工作成打印机能够理解的格式的输入过滤器。我们将在下面看看这个。

对于打印机过滤器有很多选项。它们中有一个，可以从在线手册获得，在每个工作之前发送一个LaserJet控制顺序。在`/usr/local/libexec/lpfilter`中放入下面的shell脚本：

```
#!/bin/sh
printf "\033&k2G" && cat && printf "\f" && exit 0
exit 2
```

图21-1 简单打印过滤

这个方法在诸如我的HP LaserJet 6MP打印机上工作得不是很好，它可能会胡乱地打印PostScript 和LaserJet格式。它们通过在工作开始确认文本来这样做。这个特殊的过滤器通过发送一个LaserJet命令代码把我们给搞糊涂了，所以打印机打印PostScript就好象它是纯文本。源文件`/scripts/lpfilter.c`包含一个用Ctrl-J Ctrl-M代替所有Ctrl-J的过滤器。编译这个程序，然后存储在`/usr/local/libexec/lpfilter`中：

```
# cc /scripts/lpfilter.c -o /usr/local/libexec/lpfilter
```

那就是你通常必须设置你的打印机的所有工作。接着，你应当测试它们。

### 启动spooler

正如我们在上面看到的，行式打印机守护程序`lpd`是与打印spool工作相对应的。默认的，在启动时它不会启动。如果你是root，你可以通过键入`lpd`来启动它：

## The Complete FreeBSD(4th)

### # lpd

通常，你需要在系统启动时它能自动启动。你可以在`/etc/rc.conf`文件中通过设置

`lpd_enable`变量来完成：

```
lpd_enable="YES" # Run the line printer daemon
```

看看第271页的`/etc/rc.conf`了解更多细节。

在`/etc/rc.conf`中的另一行引用了行式打印机守护程序：

```
lpd_flags="" # Flags to lpd (if enabled).
```

你通常不需要修改这行。看看`lpd`的联机手册连接更多细节。

### 测试spooler

要测试spooler，你再可以运行`lptest`程序。这时，代替直接发送给打印机，你可以把它发送给spooler:

```
$ lptest 20 5 | lpr
```

结果看起来是这样的：

```
!"#$%&'()*+,-./01234
"#$$%&'()*+,-./012345
#$%&'()*+,-./0123456
$%&'()*+,-./01234567
%&'()*+,-./012345678
```

### 问题解答

这里有一个最普通的问题和如何解决的列表：

表21-2 普通的打印机问题

问题	原因
打印机打印，但最后一页并不出现。状态显示打印机仍有数据在缓存中。几分钟以后，最后一页可能会出现。	你的输出数据没有打出最后一页。打印机被配置来等待一个外部的打印（ASCII形式的feed字符，Ctrl-L）或一段时间以后打印。 你可以选择做些什么呢。通常你可以配置打印机，或你可以在工作的结尾配置一个打印过滤起来打印feed形式的字符。图21-1已经这样做了——那是 <code>printf "\f"</code> 。
行直接跑到了页的右边，再也看不见了。	这是我们在第384页看到的阶梯效应。看看那儿的解决办法。
独立的字符或完整的一段文本丢失了。	这个问题只可能是由串行打印机引起的。它是不正确的按键的结果——看看在线手册了解更多细节。
输出完全是些难以理解的随意字	在一个串行打印机上，如果字符出现很慢，有一些字符

## The Complete FreeBSD(4th)

符。	{ }~，这大概意味着你设置的通信参数不正确。检查一下在线手册来解决这个问题。确信你不会与下面一个问题混淆。
文本是清晰的，但它与你所要打印的东西没有关系。	一个可能性是你正在发送PostScript输出给你的打印机。看看第389页的例子检查一下它是否是PostScript。如果它是，你的打印机就不要中断它，因为它不理解PostScript，或因为它已经搞糊涂了。我们将在第389页看看PostScript的细节。
在打印机上的显示显示出数据正在到达，但打印机没有打印任何东西。	你可能正在发送普通的文本到一个不理解普通文本的PostScript打印机。在这种情况下，你需要一个过滤器来转换文本为PostScript——与先前的问题相反。 另外一个办法，你的打印机端口可能没有正确中断。这将阻止打印机打印，但它可能会花费20分钟的时间来打印一页。你可以通过使用下面的命令来修复这个，它使打印机/dev/lpt0进入polled模式：  # <b>lptcontrol -p</b>
你得到信息lpr： cannot create /var/spool/output/ freebie/.seq	你已经忘记了创建spool目录/var/spool/output/freebie。

### 使用spooler

使用spooler很简单。代替输出数据到打印机，你用管道把它传输到spooler lpr命令。例如，这儿是一样的打印命令，首先直接打印到打印机，接着通过spooler。

```
# ps waux > /dev/lpt0  
# ps waux | lpr
```

spooler创建从这个数据创建一个工作。你可以用lpq程序看看当前的打印队列

```
$ lpq  
waiting for lp to become ready (offline ?)  
Rank Owner Job Files Total Size  
1st grog 313 (standard input) 9151 bytes  
2nd grog 30 (standard input) 3319 bytes  
3rd yvonne 31 (standard input) 3395 bytes  
4th root 0 (standard input) 2611 bytes
```

第一行是一个lpd不能当前打印的警告。你应当认真地对待它。在这个例子中，打印机故意地被关闭，以至队列不会从一个例子变成另一个。

## The Complete FreeBSD(4th)

---

通常，工作编号会继续增加：这个特殊的例子来自三个不同的机器。你可以用-l选项来得到更多细节：

```
$ lpq -l
waiting for lp to become ready (offline ?)
grog: 1st [job 313freebie.example.org]
(standard input) 9151 bytes
grog: 2nd [job 030presto.example.org]
(standard input) 3319 bytes
yvonne: 3rd [job 031presto.example.org]
(standard input) 3395 bytes
root: 4th [job 000bumble.example.org]
(standard input) 2611 bytes
```

### 删除打印工作

有时，你可能需要删除一些没有打印的spool输出。因为一个打印机的配置错误，你不一定要这样做：只需要关闭打印机，修复配置错误，然后再打开打印机。工作然后就会正确地打印出来。但如果你发现打印工作本身包含一些乱码，你可以用lprm程序删除它。首先，你必须知道工作编号。假定我们看到的是上面列出的，我们可能要删除工作30：

```
# lprm 30
dfA030presto.example.org dequeued
cfA030presto.example.org dequeued
# lpq
waiting for lp to become ready (offline ?)
Rank Owner Job Files Total Size
1st grog 313 (standard input) 9151 bytes
2nd yvonne 31 (standard input) 3395 bytes
3rd root 0 (standard input) 2611 bytes
```

如果打印机离线了，可能要给lprm花一些时间来完成。

### PostScript

我们已经好几次遇到了术语*PostScript*。它是一个强大的页面描述语言。用它，你可以传输细致的如这本书的电子文档，然后用同样的格式打印出来。*PostScript*是一个万维网上非常流行的格式，像Netscape浏览器通常可以用*PostScript*格式来打印。

绝大多数其他文档格式用换码顺序 (*escape sequences*) 来描述特殊的打印特性，HP LaserJet and PCL格式使用ASCII ESC字符(0x1b)来指出一个换码顺序的开始。PostScript使用相反的方法：除非定义其他方面，一个PostScript文件的内容是一些命令，可打印数据被包含在圆括号中。

PostScript文档是像这样来开始的：

```
%!PS-Adobe-3.0
%%Creator: groff version 1.10
%%CreationDate: Fri Oct 31 18:36:45 1997
%%DocumentNeededResources: font Symbol
%%+ font Courier
%%+ font Times-Roman
%%DocumentSuppliedResources: file images/vipw.ps
%%Pages: 32
%%PageOrder: Ascend
```

## The Complete FreeBSD(4th)

```
%%Orientation: Portrait
%%EndComments
%%BeginProlog
```

这是这章的PostScript输出的开头。如果它包含内嵌的字体或图像，这样一个程序的开头可能有几百千字长。一个更加典型的大小是大约500行。

你可以用PostScript做很多事情：

- 你可以用`ghostscript`或`ghostview`来阅读它，这两个都是在Ports Collection中。
- 许多打印机理解PostScript，可以直接打印它。你应当知道这个，因为它是一个昂贵的选项，但万一出现疑问，一定要认真检查打印机手册。
- 如果你的打印机不理解PostScript，你可以用`ghostscript`来打印它。

用`ghostview`来查看

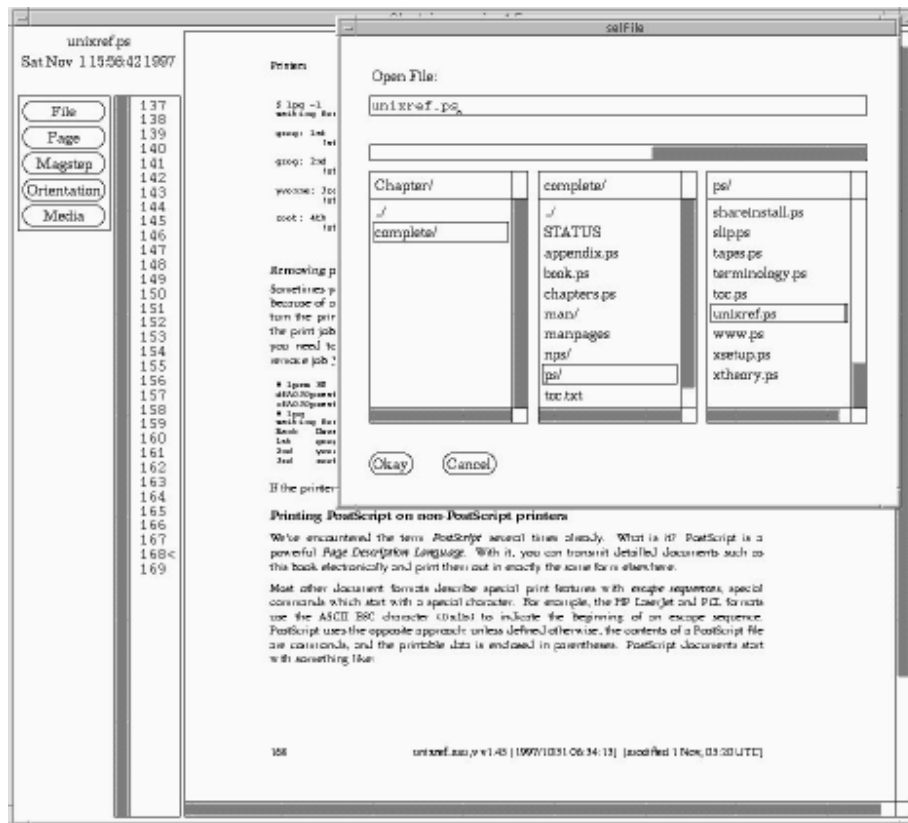


图21-3 `ghostview`显示

`ghostscript`和`gv`都是我们在第117页讨论的工作站安装方式中一个组件。要用`ghostview`看一个文件，简单地启动它：

```
$ ghostview &
```

你将得到一个空白显示，但你可以通过键入`o`来打开一个文件窗口，然后选择文件并显示它们。图21-3在顶部左面在一个覆盖的窗口下显示了这页的一个草稿版。`selFile`窗口顶部包含了一

## The Complete FreeBSD(4th)

个你可以键入一个文件名的区域。另外，下面有三个选项，用滚动条允许你浏览当前的目录和前面的目录。接口看起来相当原始，但它能够工作。

在右边，窗口下面粗略地显示了以前页面的文本。用中间的按钮来拖动滚动条。在左上角是五个椭圆形的按钮，你可以用鼠标左键来选择。特别注意Magstep按钮，它可以设置显示的大小。

这些按钮右边的列是页码的列表。你可以用中间的鼠标按钮选择一个页码。你也可以通过键入左键在鼠标指针周围得到一个放大的文本显示区域。键入标记为Dismiss的区域来删除放大。

### 用ghostscript打印

如果你的打印机不支持PostScript，你可能仍可以在ghostscript的帮助下打印一些特别的文档。结果是可以被激光打印机和喷墨打印机接受，矩阵打印机比较少，即使是24针的版本。

要在你的特殊打印机上打印，你首先必须在ghostscript中为它寻找一个驱动程序（driver）。在这篇文章中，术语driver意思是包含在ghostscript中的用于转换数据成打印机能够打印的一些代码。不幸的是，联机手册没有相关帮助。要找出它支持哪个驱动程序，可以启动ghostscript，然后在文本窗口键入下面的这些指令，忽略它打开的显示窗口：

```
$ gs that's the name of the ghostscript program
Aladdin Ghostscript 5.03 (1997-8-8)
Copyright (C) 1997 Aladdin Enterprises, Menlo Park, CA. All rights reserved.
This software comes with NO WARRANTY: see the file PUBLIC for details.
GS>devicenames == list device names
[/tiff24nc /ppm /pcxgray /cgm8 /sgirgb /pnm /mgr8 /bmp16m /psmono /pgnm /mgrgray8
/bmp
16 /png256 /pgm /mgrgray2 /bitcmk /pnggray /pbm /miff24 /bit /nullpage /pkmraw /pcx24
b /jpeg /pdfwrite /tiff1zw /ppmraw /pcx16 /cgm24 /tiff12nc /pnmraw /pcxmono /cgmmo
n
psgray /pgnmraw /mgr4 /bmp256 /png16m /pgmraw /mgrgray4 /bpmmono /png16 /pbmraw
/mgrmo
no /bitrgb /pngmono /pcxcmyk /jpeggray /pswrite /tiffpack /pkm /pcx256 /cif /t4693d8
/
paintjet /ljet3d /iwlq /declj250 /appledmp /tiffg32d /t4693d2 /oki182 /ljet2p /iwhi
/c
dj850 /sxlcr /tiffcrle /st800 /necp6 /ln03 /ibmp /cdj500 /x11gray2 /faxg32d /r4081
/lp8000 /lbp8 /eps9high /cdjcolor /x11alpha /dfaxlow /pxlmono /lj5gray /la75 /epson
/c
cr /xes /pjxl /ljetplus /la50 /djet500c /bjc600 /tek4696 /pj /ljet4 /jetp3852 /deskjet
/bj10e /tiffg4 /t4693d4 /okiibm /ljet3 /iwlo /cp50 /ap3250 /tiffg3 /stcolor /oce9050
/lj250 /imagen /cdj550 /x11mono /faxg4 /sj48 /m8510 /lips3 /epsonc /cdjmono /x11cmyk
/
faxg3 /pxlcolor /lp2563 /la75plus /eps9mid /cdeskjet /x11 /dfaxhigh /pjxl300 /lj5mono
/la70 /dnj650c /bjc800 /uniprint /pjetxl /lj4dith /laserjet /djet500 /bj200
/epswrite]
GS>^D exit
```

不幸的是，不看源代码或至少Makefile很难决定这些驱动程序中哪些驱动程序做些什么，所以当知道绝大多数普通的非PostScript打印机，Hewlett-Packard LaserJet系列，使用前缀为ljet驱动程序时就给人一些安慰。现代版本将用ljet4运行。如果你遇到问题，试试旧一点的。特别是，最初的LaserJet要求驱动程序ljet。下面在线的脚本将打印PostScript文档到spooler。它

## The Complete FreeBSD(4th)

---

在CDROM中以`/scripts/gsp`出现。

```
#!/bin/sh
gs -dNOPAUSE -q -sDEVICE=ljet4 -sOutputFile=|lpr -- $*
```

注意打印机的输出是二进制数据，所以，在`/scripts/lpfilter.c`中显示的过滤器将损坏数据格式和产生很多毫无意义的输出。这不应当是一个问题，因为那个过滤器是专门针对PostScript打印机的，它不需要`ghostscript`。