Session 5540
Linux Printing

Adam Thornton
Sine Nomine Associates
25 July 2001
How do I print in Linux for System/390?

- Linux does not yet have any drivers for channel-attached printers
- You could always write one
  - Linux is Open Source
  - Simply write the device driver and drop it into the kernel; how hard can that be?
- However, it’s easier to just tell Linux you have a remote printer and let something else drive it

© 2001 Sine Nomine Associates
Printing in Linux

- Direct-connected printers often connect to /dev/lpX via a device driver
  - In the PC world, typically a parallel port
  - The port is usually not used raw but exported to applications via the lpr/lpd facility
- Two different methods of remote printing
  - lpr/lpd within Unix world
  - Samba within Windows world
Traditional Unix Printing

- **lpd** is a service daemon
  - Runs on TCP port 515
  - Accepts print jobs from the network
- **lpr** is the client
  - Accepts print jobs from user
  - Submits them to lpd on the network
  - Controlled by /etc/printcap
- RFC 1179; not all implementations really follow RFC
Unix Printing

- PostScript printers are generally assumed
  - In the absence of other information, spooled print data is assumed by Linux to be PostScript
- `/etc/printcap` controls `lpd` as well as `lpr`
- `/etc/hosts.lpd` defines host access to a particular machine’s print subsystem
  - `hosts.equiv` works too, but is more dangerous
Lpd on the mainframe

- VM/ESA’s TCP/IP contains LPSERVE, which acts as an lp daemon
  - This is the older interface and probably shouldn’t be used if you can use RSCS instead
- RSCS allows you to define an LPD-type-link
  - Preferred way of using lp under VM/ESA or z/VM
- Z/OS Network Print Facility can speak LPD too
  - JES or VTAM can access LPD functionality

© 2001 Sine Nomine Associates
S/390 Examples

- I’m going to use z/VM or VM/ESA for my examples
  - If you’re running Linux, it’s likely you have some version of VM in the picture
  - I’m more familiar with it than with z/OS
  - All of this should work exactly the same if you can get lpd running under z/OS and pointed at your Linux LPAR
Configuring /etc/printcap

- /etc/printcap contains queue name, spool file locations, and input and output filters
- A simple network printer entry might be

```
lp:\n:rm=printer.barfoo.cx:\
:rp=lp:\n:sd=/var/spool/lp:\n:lf=/var/spool/lp/lp-errs:
```
Scenario I: Printing from Linux

- With the /etc/printcap from the last slide, our default printer ("lp") became the PostScript printer named printer.barfoo.cx
- To print on it, we just do
  \[\text{lpr \textit{filename}}\]
  which is the same as
  \[\text{lpr –Plp \textit{filename}}\]
Scenario II: VM/ESA, S/390 printer

- We have a printer controlled by our System/390, which is running VM/ESA
- We have a Linux guest under VM/ESA that wishes to print to that printer
- We will set up lpd on the VM/ESA side to accept jobs from our Linux guest and print them on its locally-attached printer
Linux preparation

- VM/ESA’s TCP/IP stack is at vm.barfoo.cx
- We will call the printer “lp” under Linux
- /etc/printcap:

```
lp:
:rm=vm.barfoo.cx:\
:rp=SYSTEM:\
:sd=/var/spool/lp:\
:lf=/var/spool/lp/lp-errs:
```

© 2001 Sine Nomine Associates
VM/ESA Preparation

- Use RSCS LPD-type-link, not TCP/IP LPSERVE
- Add FILEDEFS to GLOBAL LOADLIB in PROFILE GCS
- The RSCS documentation explains this pretty thoroughly
- Add your appropriate LINKDEFINEs and their corresponding PARMs to RSCS CONFIG
- Then START your links

© 2001 Sine Nomine Associates
Network-attached printers

- The mainframe side of things can speak lpr in addition to lpd, so mainframe users can act as print clients.
- So we can configure lpr-type links to point at any host and printer that can do lpr/lpd printing.
- And we can use this, plus Linux, to print to printers otherwise unsupported in a mainframe environment.
- Linux simply presents all the printers it knows about as PostScript devices.

© 2001 Sine Nomine Associates
Printing to non-PS printers

- Linux distributions come with a great utility called “ghostscript”
- Ghostscript translates PostScript into other printer definition languages
- This is where we use the input and output filter fields in /etc/printcap
  - Use the :if= directive to run ghostscript to change its input to something the printer understands

© 2001 Sine Nomine Associates
Ghostscript Print Devices

- All kinds of stuff is supported
  - LaserJet models
  - Common laser and inkjet printers
  - Several fax resolutions
  - Dot-matrix printers (Proprinter, Imagewriter)
  - JPEG/PDF/raster graphic formats
Non-PostScript printers

• You can do this all manually by directly giving Ghostscript command-line parameters
  – Set :if= to something like
    • gs –dNOPAUSE –slaserjet –sOutputFile=- -r300
• Or you can just use apsfilter (or magicfilter), which figures it out for you and creates the appropriate printcap entries on your behalf
With apsfilter, it gets really easy: you say, “I have a LaserJet 4 on /dev/lp0”, and it builds the appropriate /etc/printcap entries for you.

Ghostscript and apsfilter both work on S/390—but ghostscript is slow on pre-G5 machines.

If possible, offload your ghostscript filtering to someplace (like x86) where cycles are cheap.
Scenario III: z/VM to network-attached HP LaserJet

- Set up z/VM lpd links as before
- Set up z/VM lpr links pointing to Linux guest
  - LINKDEFINEs in RSCS CONFIG
  - HOST parameter in LPRXFORM CONFIG
  - PRINTER defines Linux print queue: let’s call it pcl
- As far as z/VM knows, it’s simply printing on printer “pcl” on the Linux machine
• Printer is named ljet.printers.barfoo.cx
• It is a LaserJet 5 that can do 600 dpi
• We will print to the “raw” queue
• /etc/printcap needs an entry like
  pcl:\n    :rm=ljet.printers.barfoo.cx:\n    :rp=raw:\n    :if=/var/lib/apsfilter/bin/ljet-5-letter-auto-mono-600:
• apsfilter will set up printcap for us
Putting it all together

- CMS user prints PostScript file using RSCS (PPD EXEC); let RSCS handle carriage control
  - Continuous forms get tricky in page-oriented world
- RSCS uses LPR link to Linux lpd with “pcl” queue
- lpd consults /etc/printcap and runs its input filter, which hands file to apsfilter, which invokes ghostscript
- Ghostscript sends PCL back to Linux lpd
- PCL is sent over network to printer’s “raw” queue
Putting it all together

RSCS prints file using LPR link

LPR link

Lpd on Linux for S/390

lpr out to printer

apfilter
and
GhostScript

Input Filter to GS

PCL Output from GS
But what about Windows?

- So far, we have neglected a whole class of printing solutions
- Plenty of Windows machines whose users would like to print to big network-attached printers
- Plenty of machines out there with Windows-shared printers
Samba

- Samba provides SMB (Server Message Block) networking (Windows Networking)
- Open Source
- Has been built on all sorts of systems
- Standard with Linux distributions
- Controlled from /etc/smb.conf
Adding printers in Samba

- Add a [printers] section in smb.conf
  path = /var/spool/public
  printable = yes
- Add to [global] in smb.conf
  printing = bsd
  printcap name = /etc/printcap
  load printers = yes
- All printers in /etc/printcap will appear
• If we did that and started Samba, then…
• With *no further modification* Windows clients could print to our mainframe-attached printer in Scenario II
• With *no further modification* Windows clients could use our LaserJet as a PostScript printer in Scenario III
• Samba is also used for file sharing: see Rick Troth’s Session 5547, Monday at 1:30
You can use the “smbprint” script as an input filter in /etc/printcap
  – It lurks in /usr/share/doc/packages/samba/examples/printing/smbprint on SuSE—make it an executable in $PATH
Recent apsfilters will let you simply set up a Windows remote printer
  – It’s probably integrated into your system configuration tool as well; it certainly is for SuSE
Samba 2.2

- Samba 2.2 contains direct support for NT printing via MS-RPC rather than the LAN Manager printing interface
- Upload/download drivers on demand
- NT ACLs on printers
- Details are in the Samba 2.2 HOWTO
  - I haven’t implemented this myself, yet
Scenario IV: z/VM to Windows

- We want to let our CMS users print to Windows-managed, shared printers
- This is exactly the same as Scenario III
  - Except that we define a remote Samba printer to our Linux guest
  - Then we point our LPR link at the Linux queue that represents the Samba printer
Capabilities

- We can now implement a lot of the capabilities of something like Macro 4 for free
- By using a Linux machine as our printing hub, we can print from any platform to nearly any printer
- Linux is Linux
  - Our Linux printing hub could be Linux/390 in a virtual machine
  - Or it could be a standalone x86 box
Capabilities

• Moving beyond paper
  – With Ghostscript and apsfilter acting as translators, we can “print” all kinds of things:
    • Faxes
    • PDF file output
    • Various formats of graphical file output
• And if you’re really ambitious, netatalk will let you network AppleTalk printers into the mix too

© 2001 Sine Nomine Associates
Conclusion

- The combination of Linux, lpr/lpd, ghostscript, and apsfilter make a tremendously powerful printing solution
- With apsfilter and integrated system tools (like YaST or Linuxconf) setup is made pretty easy
- And on top of all that, it’s all free and Open Source!
Contact Information

• Adam Thornton
  – athornton@sinenomine.net
  – (314) 692-2046
• Sine Nomine Associates
  – info@sinenomine.net
  – (703) 723-6673