Using the spooler to print to email

By Dirk Hart

The other day I got a call from a client asking if they could have a report sent by email rather than have it printed out each night. The thing was that the report was only interesting under specific conditions so it was thrown out most of the time.

After some experimentation and some useful advice I came across this useful formula.

First define a new spooler destination using the dumb interface script and /dev/null for the device. Give your new spooler destination a name - I called mine email displaying a deficit of imagination. Next, change directory to /usr/spool/lp/admins/lp/interfaces and edit email if that's what you called it. I use and like vi for editing but you can use whatever editor you like. Finally, I added a line, changing the FILTER to get the emailing done.

Immediately after this bit:

```
  #If we are not using a filter, use the default one.
 if [ -z "${FILTER}" ] then
       FILTER="${LPCAT}"
 fi
```

I added this bit:

```
 FILTER="/usr/bin/mail -s report boopy"
```

Where boopy is a valid user on the system. If you have sendmail or some other mail program running you can email to remote users with boopy@yourdomain.com.

Since my client wanted more than one user to receive this report by email I changed the user (boopy) to an alias and made a new alias in /usr/lib/mail/aliases. Don't forget to run newaliases if you do this, although I discovered that on Solaris it's unnecessary.

I had a little bit of help from JP Radley (jpr), which amounted to a whack on the side of the head, but it turns out he implements this with a different modification. He likes to apply this bit around the part of the script that does the actual printing: { ... } | mail -s report boopy.

```
 { while [ $i -le $copies ] do
       for file in $files do
             0<${file} eval ${FILTER} 2>&1 echo "\014\c"
       done i=`expr $i + 1`
 done            } | mail -s report boopy
```

Scott McMillan dispenses with the 'dumb' interface script altogether preferring to roll his own:

```
shift; shift; shift; shift
files="$*"
for file in $files
do
   mail -s "testing through lp" root <${file}
done

exit 0
```

Using any of these methods you can send email simply by invoking the spooler as you normally do, e.g., lp -demail /etc/hosts.

This works fine for plain text reports. Reports that have imbedded postscript or pcl are not candidates for emailing unless they can be converted to html, pcl or pdf. In this case you might prefer a solution where you can send your print jobs as an attachment.

Thanks to JP Radley and Scott McMillan for their ideas.