

# Printing from Unix to Windows Printers

Using nothing more than the software that comes free as part of most Unix and Windows systems, you can successfully print from Unix to Windows printers. There are several steps involved on both, the Windows system and the Unix system. I have done my best to accurately compile the process in this article.

## From the Windows system that is hosting the printer –

The Windows portion of this procedure was prepared on a Windows 2000 SP4 system. Other Windows versions may vary somewhat.

### Install the Microsoft LPD print server.

- Go to "Control Panel / Add/Remove Programs / Add/Remove Windows Components / Components / Other Network File and Print Services / Details...".
- Click the check-box next to "Print Services for Unix", then click "OK / Next > / Finish / Close".

### Start the newly-installed LPD print server.

- Go to "Start / Settings / Control Panel / Administrative Tools / Services / TCP/IP Print Server / Properties / Startup type:" and change it to "Automatic". Click the "Apply" button.
- If necessary, click the "Start" button to start the service, then click "OK" to close the Properties dialog.
- Leave the Services window open for now. We will return to it in a moment.

### Configure the Windows printer.

- Go to "Start / Settings / Printers".
- Double-click "Add Printer" to start the "Add Printer Wizard".
- Use the wizard to add a Windows printer. It does not matter if the printer is a local or network printer. The manufacturer of the printer should be specified as "Generic" and the printer model should be "Generic / Text Only". The name of the new printer should be Unix-friendly (i.e., all lowercase, no spaces). Printer sharing is not necessary, and the share name does not matter. Unix will refer to the printer by its Windows printer name, not its share name.
- Once the "Add Printer Wizard" completes, right-click the newly-added printer and choose "Properties" from the context menu.
- On the "General" tab, click "Printing Preferences... / Advanced..." and specify the "Paper Size:" as "US Std Fanfold".
- On the "Advanced" tab, ensure that the check-box next to "Enable advanced printing features" is selected, then click the "Printing Defaults... / Advanced..." and specify the "Paper Size:" as "US Std Fanfold".
- On the "Device Settings" tab, under "Generic / Text Only Device Settings / Form To Tray Assignment", specify the "Cont. Feed - No Break:", "Cut Sheet:", and "Cont. Feed - With Break:" settings as "US Std Fanfold".

- Click “OK” to close the printer properties dialog.

#### **Enable Windows Passthrough Printing.**

- Start the Windows Registry Editor by clicking “Start / Run...”. Specify to open “regedit” and click “OK”.
- From within the Registry Editor, go to “HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\LPDSVC\Parameters”. Click “Edit / New / DWORD Value” and name the new value “SimulatePassThrough”. Change the value to “1”.
- Close the Registry Editor window.

#### **Restart the Print Spooler service.**

- Return to the Services window that we left open previously.
- Right-click the “Print Spooler” service and choose “Restart” from the context menu.
- When prompted, confirm that you wish to restart other services also by clicking “Yes”.
- Close the “Services” window.

### **From the Unix system that will be printing to the Windows printer –**

(OpenServer systems older than 5.0.6 do not support the “of” (output filter) clause in /etc/printcap, so this solution is not suitable for them. Instead, you should investigate setting-up a virtual local printer to act as a “wrapper” for the remote printer. This will give you a printer interface script that can act as your output filter and then forward your print jobs to the actual remote printer via the “lp” command. Here is a link to an article detailing how to configure a virtual local printer:

[http://aplawrence.com/SCOFAQ/FAQ\\_scotec7virtualprint.html](http://aplawrence.com/SCOFAQ/FAQ_scotec7virtualprint.html).)

#### **Add the remote printer.**

The process for OpenServer 5.0.6 will be detailed below. Adjust the procedure as appropriate for your system.

- Go to “scoadmin / Printers / Printer Manager / Printer / Add Remote / Unix...”.
- Specify the “Host” as the IP address or host name of the Windows system that is hosting the printer.
- Specify the “Printer” as the Windows printer name that you specified when you added the printer to the Windows system. (Once again, the Windows share name is not what we want here.)
- Uncheck the “Use extended remote printing protocol ...” check-box.
- Select “OK” to complete the addition of the new printer.

#### **Create an output filter for the remote printer.**

Create a new output filter with your favorite text editor. It should reside in a directory that is in root’s default PATH. The output filter should accomplish the following:

- Prepend the print stream with any appropriate printer control codes.
- Pass the print stream (i.e., standard input) through to standard output.

- Append to the print stream any required control codes (e.g., form feed).
- Translate Unix line-feeds to DOS/Windows carriage-return/line-feeds.

Below are two examples for PCL printers. The first example will print in portrait mode, and the second in landscape. Both examples specify a non-default *vertical motion index* (VMI) in order to allow printing 66 lines of text to a page. Both examples append form-feeds to the end of the print stream. Both examples translate Unix line terminators to DOS/Windows compatible format.

*Example 1: Portrait; 17cpi; 66 lines/page; form-feed at end of job; DOS/Windows line terminators*

```
#!/bin/sh
(
  echo "\033E\c"          # Reset printer.
  echo "\033&l100\c"      # Portrait mode
  echo "\033&l7.27C\c"    # VMI of 7.27/48ths of an inch
  echo "\033(s16.67H\c"   # Compressed print (16.67 CPI)
  cat -                   # Relay print stream to standard output.
  echo "\f\c"            # Append form-feed.
) | /usr/lib/lp/lpr      # Translate line terminators.
```

*Example 2: Landscape; 17cpi; 66 lines/page; form-feed at end of job; DOS/Windows line terminators*

```
#!/bin/sh
(
  echo "\033E\c"          # Reset printer.
  echo "\033&l10\c"       # Landscape mode
  echo "\033&l5.45C\c"    # VMI of 5.45/48ths of an inch
  echo "\033(s16.67H\c"   # Compressed print (16.67 CPI)
  cat -                   # Relay print stream to standard output.
  echo "\f\c"            # Append form-feed.
) | /usr/lib/lp/lpr      # Translate line terminators.
```

Make the new output filter executable with the “`chmod ugo+rx filename`” command.

**Edit /etc/printcap to specify the output filter.**

Edit /etc/printcap with your favorite editor to add a clause specifying the output filter to use.

*Example of an /etc/printcap entry that specifies an output filter in “/usr/local/bin”*

```
laser:\
:lp=:rm=192.168.1.101:rp=laser:sd=/usr/spool/lpd/laser:mx#0:\
:of=/usr/local/bin/pcl_17cpi_66lines_port:
```

## Testing and Troubleshooting –

Problems that can occur with the configuration may exhibit some of the following symptoms:

- Nothing prints.
- Printer starts to print, but nothing comes out.
- Long lines are wrapped, even when there is room on the paper for them.
- Printer control codes are printed, instead of acted upon by the printer.
- Stair-step printing
- Only the first line prints.

To test the printer, you might want to use your favorite editor to create a test pattern that will allow you to easily troubleshoot any problems you might encounter. I generally use vi to create a file with 132 characters per line and 66 lines per page. Within the pattern, I number the lines and the character positions.

*Example printer test pattern*

```
01 -+----1----+----2----+---- ... ----+----9----+----0----+----1----+----2----+----3--
02 -+----1----+----2----+---- ... ----+----9----+----0----+----1----+----2----+----3--
    ...
65 -+----1----+----2----+---- ... ----+----9----+----0----+----1----+----2----+----3--
66 -+----1----+----2----+---- ... ----+----9----+----0----+----1----+----2----+----3--
```

Print the test pattern using a command similar to: `lp -dlaser test.prn`.

If nothing prints, consider whether there might be a partially rendered page in the printer, waiting for more text or a form-feed to push it out. You may have to add a trailing form-feed to your output filter, as in the examples above.

If still nothing prints, try setting the Windows printer to “Use printer offline”, then print again and see if the print job appears in the Windows queue for that printer. If the job never appears in the queue, check the syslog file on the Unix system for errors related to the problem. (On OpenServer, the syslog file is in `/usr/adm/syslog`.) If that doesn’t help, check the “Enable advanced printing features” setting on the Windows printer. Also, check the Windows services to ensure that the “Print Spooler” and “TCP/IP Print Server” services are running. If you still can’t print, try pinging the Windows system from Unix to ensure that the two systems can communicate with each other.

If printing wraps when there is still room on the line for more printing, check the paper size in the Windows printer properties.

If printing is in a “stair-stepping” pattern or only the first line prints, check the line termination conversion in your output filter.

If printer control codes print, instead of causing the printer to act on them, check the “SimulatePassThrough” registry entry on the Windows system. Also, check to ensure that you are using the “Generic / Text Only” printer driver.

Windows places spooled print jobs under the Windows system directory (usually "c:\windows" or "c:\winnt") in the "system32\spool\printers" folder with an extension of ".SPL". You can examine that file with a text or hex editor to determine if your data stream contains the expected content before it gets to the printer. If so, and you still aren't getting the expected results, suspect that Windows is tampering with the data stream. Check the printer driver and "SimulatePassThrough" registry entry again.

Use the Unix printer status command ("lpstat -t" on OpenServer) to see what information you can gather about any problems. If it hangs, it usually represents a communication problem between the two systems. Either the Windows services aren't running, or the two systems can't communicate via TCP/IP.

Also, ensure that the printer status as reported by Unix doesn't say anything about "not accepting requests" or "not enabled". On OpenServer, the "/usr/lib/accept *printername*" command will allow the printer to accept requests, and the "enable *printername*" command will allow spooled print requests to be delivered to the printer. In my experience, I always thought that those commands were not generally related to remote printing from OpenServer, but I recently noticed the lpstat command reporting "not accepting requests" and "not enabled" on a remote printer, and those two commands corrected the problem.

## Contact Me —

If you are still having problems, feel free to drop me an email or phone call. I'll be happy to help as much as time allows.

Lytle David Smith  
<http://LytleDavidSmith.DynDNS.org:56180/>  
[LytleDavidSmith@Gmail.com](mailto:LytleDavidSmith@Gmail.com)  
(502) 561-1111

Link to this article:  
<http://lytleidavidsmith.dyndns.org:56180/articles/PrintingFromUnixToWindowsPrinters.pdf>