

Setting Up and Using Network Printers with CUPS

Serving Up CUPS

CUPS not only supports direct printing from a computer to an attached printer, in heterogeneous networks you can use the CUPS printing utility suite to allow Windows boxes to print to a printer attached to a Linux machine. We show you how.

BY MARTIN LOSCHWITZ



Printing with Linux used to be a disaster, with every distribution doing its own thing, and generally driving people to distraction, as successfully setting up a printer on SUSE Linux was no guarantee that it would work with Red Hat or Mandrake Linux.

Happily, this has all changed: the Common Unix Printing System, or *CUPS* for short, is a completely new approach to providing a universal printing system for Linux. CUPS was the project that unified the major Linux distros. SUSE, Fedora, or Mandrake may have a different GUI front-end, but what really matters is the tool running in the background, and that is CUPS in every case.

CUPS also solved another problem. Because you can run it either with Samba or as a stand-

alone print server, you can access a printer attached to a Linux machine from a Windows desktop. This makes it easy to set up a Linux computer with CUPS as a print server in a heterogeneous network environment.

And as an added bonus, the Linux box can use special drivers to add direct

printing of PostScript files besides the standard range of functions.

Depending on your environment, the configuration steps for the individual programs in the mixed environment with Windows desktops and a Linux print server can vary: Windows 95, Windows 98 and Windows ME need a Samba connection to print across the network. Windows 2000 and Windows XP can use the Internet Printing Protocol, or *IPP* for short, to access CUPS.

And then there is the Adobe PostScript driver as a special option for any operating system. You can install the driver as a pseudo-printer on your Windows systems and access the CUPS printer across the network.

On the following pages, we will be looking at how to set up your printer for

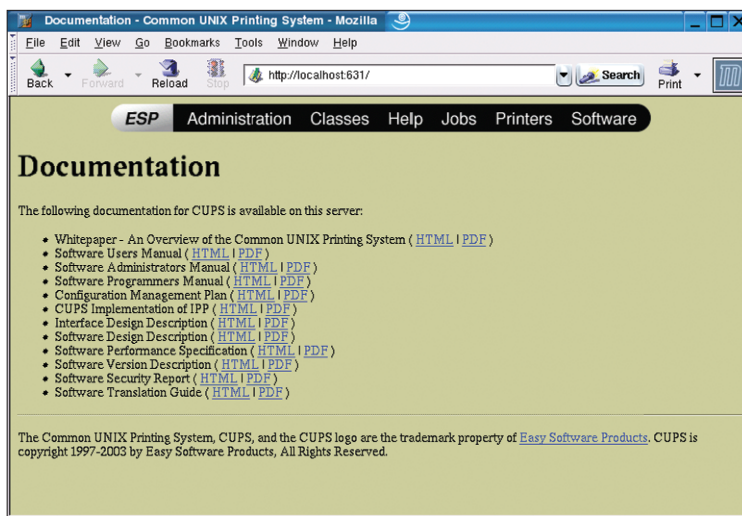


Figure 1: The CUPS Web interface after logging in over HTTP. Most or all of the CUPS configuration can be accomplished through a web browser.

a CUPS print server and how you access the printer from a Windows-based client. To cap this off, we will be introducing Adobe's PostScript driver.

Half Way There: CUPS Setup

The first item on the list is setting up CUPS and configuring the printer. Our milestone at this stage is outputting a successful test page on the computer to which the printer is attached. In other words, we will not be using the network for the time being.

The following howto provides a generic method for setting up a CUPS-based printer with any Linux distribution. As distributions such as SUSE, Fedora, and Mandrake install CUPS by default, you will have all the tools you need to automatically detect and configure your printer. You have the option of following the approach described in this article or using your distro's tools to configure the CUPS environment; the decision is yours. The following steps will work for any distribution that uses CUPS.

First of all, ensure that you have the full set of CUPS packages on your machine. The packages typically have *cupsys* and *libcups* in their names. Also make sure you have a collection of drivers; look out for *foomatic* in the driver names. If you have a Hewlett-Packard DeskJet or OfficeJet type printer, you should additionally install the *hpijs* pack-

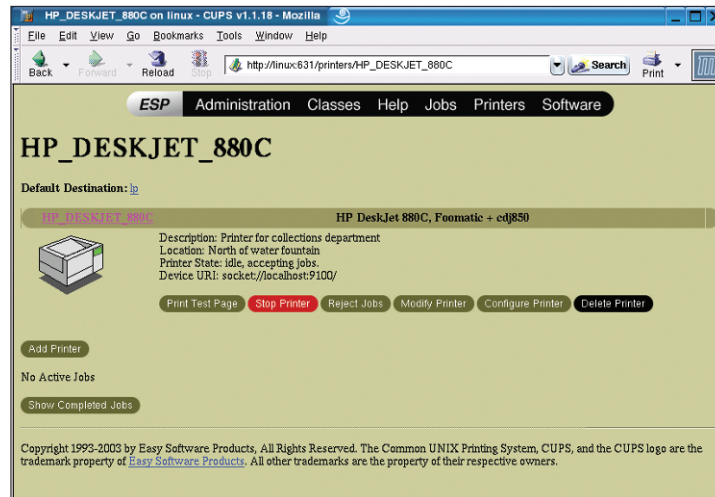


Figure 2: After setting up the printer in CUPS, you can select "Manage Printers" to view your options.

age. You can use your distro's Package Manager (YaST2 for SUSE, RPMdrake for Mandrake) to check if you have the required packages. If not, simply install the packages before you continue.

You next need to ensure that you have loaded all the required kernel modules to allow your Linux machine to detect the attached printer. Pop up a console window, and type

```
su
```

to become *root*. If you have attached the printer to the parallel port on your machine, enter *modprobe lp* to load the kernel module; for USB printers, type *modprobe printer* (for kernel 2.4), and *modprobe usblp* (for kernel 2.6) instead.

Finally, you need to locate a so-called PPD file for your printer. PPD means *PostScript Printer Description* and refers to a file with information on the best way of controlling the printer, the best

choice of printer driver, and the capabilities of the printer. The PPD file is the object that represents your printer in the CUPS system, so you will definitely need a PPD file that matches your printer to allow your printer to talk to CUPS. The best place to locate a PPD file is the LinuxPrinting project homepage at <http://www.linuxprinting.org>. Select the link for your printer below *Printer Listings*, and then click on *download PPD* for the *Recommended* entry. Store the PPD file in

your home directory for the time being, then, working as *root*, copy the file to the CUPS PPD directory (typically */usr/share/cups/model*). Then send a signal to CUPS (*killall -HUP cupsd*) to tell it to reread its configuration files.

Testing CUPS

After completing these preparatory steps, it's time to test CUPS. To do so, type <http://localhost:631/> in your browser's location box. If you see the CUPS intro page shown in Figure 1, you are up and running. The language that CUPS uses should reflect your current locale setting, but the links you need to follow will be in the same place, no matter what language you have.

In the CUPS Web user interface, click the *Do Administration Tasks* entry at the top of the list of links on this page. You are prompted for your password. Enter *root* as your username, and supply the password for the superuser, *root*. This

SSV

gives you access to the complete CUPS configuration and takes you to the *Admin* page.

Configuring a Printer

The admin page has three sections. Right now, we are interested in the section at the bottom *Printers*. Click *Add Printer* to open up the first page in the printer dialog. Type a name for the printer in the *Name* field. Note that you cannot use blanks in this name, but you can use an underscore (“_”) to separate the individual sections in the name. For example, if you have a HP DeskJet 880C, you could type “HP_DESKJET_880C”. You can add entries for *Location*, and *Description*, using blanks and special characters if needed.

In the next dialog, you need to select the interface to which the printer is attached. Click on the dropdown menu, and check if CUPS has detected your printer on one of the interfaces. If so, select the appropriate entry and click *Continue*.

Choose your printer manufacturer, and then click *Continue* again. In the next dialog, select the appropriate driver for your printer. This should be the driver you imported into CUPS as a PPD file. Select this entry. The success message in the next window tells you that you have completed the printer installation. Now click the *Manage Printers* entry top right to display the printer you just configured. You might like to print a test sheet, just to make sure that the printer works.

Windows 95 and Windows 98: Samba Olé!

Windows 95, 98, and ME do not support IPP-based printing, in contrast to Windows 2000 or Windows XP, for example. This does not mean you cannot use these Windows versions to print on a printer attached to a Linux machine; it simply means a short detour via Samba.

We are assuming that you have a Samba machine running, that the Samba machine is attached to your network, and that you have the Samba user management function working to assign access privileges. Samba is quite easy to set up using the distribution tools. For more on configuring Samba, you can also check out the documentation at the Samba website (www.samba.org).

To configure Samba for use with CUPS, open the Samba configuration file in a console window; you will need to be *root* to do this. Most distributions store the file as */etc/samba/smb.conf*. Locate the *[global]* entry, and add the following lines in that section:

```
printing = cups
printcap name = cups
load printers = yes
```

If these entries already exist, change their values to the ones shown above. Then look for the *[printers]* entry, and add the following lines, or change the existing lines to the following:

```
comment = All Printers
path = /var/spool/samba
browseable = no
public = yes
guest ok = yes
writable = no
printable = yes
printer admin = root
```

Then look for the line that starts with *invalid users =* in *smb.conf*. There is normally an entry for *root*. To allow Samba to associate with CUPS, you need to run the *cupsaddsmb -a* command as *root*. And that means that the superuser needs to log on to your Samba server. Remove the entry for *root* to allow this to happen.

Then add the following entry at the end of the file:

```
[print$]
comment = Printer Drivers
path = /etc/samba/drivers
```

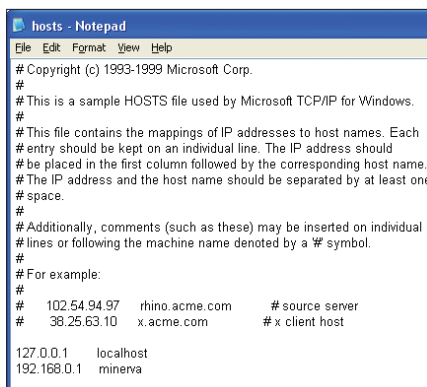


Figure 3: Like Linux, Windows uses the familiar “hosts” file approach for resolving hostnames and IP addresses.

```
browseable = yes
guest ok = yes
read only = yes
write list = root
```

After modifying *smb.conf* as described, first save your changes, and close the file, then relaunch Samba. You can then run the command we mentioned earlier: working as *root*, run *cupsaddsmb -a* to associate the CUPS printer with Samba. The printer should appear in your Samba configuration after doing so.

By default, CUPS is configured to accept only raw data. CUPS handles formatting itself prior to printing. As you have to install a printer driver for Windows 98 and Windows ME, any data you send to CUPS will be formatted rather than raw. A workaround allows CUPS to print your data. Working as *root*, open the MIME settings files for *cupsd* (most distributions store these files in */etc/cups/mime.convs*, and */etc/cups/mime.types*). *mime.convs* has a line that looks like the following line:

```
#application/octet-stream
application/vnd.cups-raw 0 -
```

You also need to watch out for the following line in *mime.types*:

```
#application/octet-stream
```

Remove the # sign at the beginning of both lines in both files. Then relaunch CUPS; if you don’t know how to do this, you can simply reboot your computer.

When you are finished, you should harden your Samba installation once more. To do so, open *smb.conf*, and reinstate the *invalid users* entry for *root*.

The Other Side: Windows Printers

Follow the standard Windows procedure to set up a printer on Windows 95, 98 or ME. Open the *Control Panel* and click *Printers*. Then select *New Printer* and follow the Wizard. Select *Network printer* and click on *Browse*. Locate the computer to which the printer is attached, and select this computer. Ignore any messages that the printer driver on the Samba machine will now be installed on your computer; you will be using the Windows driver, so this feature

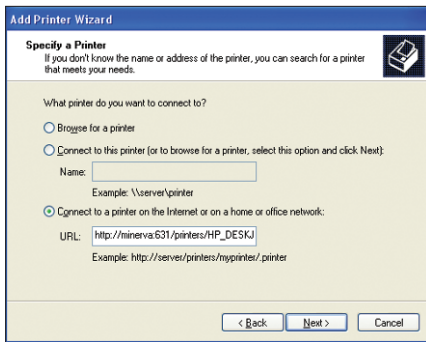


Figure 4: The printer setup dialog for a network printer under Windows XP. Make sure you select the “Connect to printer ...” entry, as this is the only type that uses IPP.

is of no use to you. Make sure you have your printer driver CD nearby, as Windows will ask you for a driver next. Of course, you can use an existing driver, assuming your printer is on the list of available printers. When you are done, opt to print a test page.

Windows 2000 and Windows XP need hosts

Printer sharing between Linux and Windows became a lot easier with the introduction of Windows 2000; Windows 2000 speaks IPP, and of course, XP understands this new standard too. Although the Samba workaround is history, you still need a few tricks to get your printers working.

Before you can even launch into the Windows 2000 and Windows XP printer setup, you need to fix a bug. To install the printer, you should theoretically be able to specify the IP address of the CUPS machine as the network path to the printer. Unfortunately, Windows 2000 and Windows XP display an error message if you attempt to specify an IP address at this stage. You can work around this issue by specifying the hostname for your CUPS machine.

As most home networks will not have a name server, and thus be incapable of resolving the hostname to an IP address, you have to tell Windows the IP address for the CUPS server hostname. Like Linux, Windows uses a *hosts* file to specify hostnames.

Using the Windows Explorer, navigate to the *C:\WINNT\SYSTEM32\DRIVERS\ETC* directory for Windows 2000, or *C:\WINDOWS\SYSTEM32\DRIVERS\ETC* for Windows XP. You should find a *hosts*

file below this directory. Open the file in a text editor. Add a line for your CUPS server at the end of this file. For example, if the machine is called “minerva”, and its IP is *192.168.0.1*, the entry would be:

```
192.168.0.1 minerva
```

Make sure you spell this entry correctly, as it is vital to setting up the network printer in the next step.

CUPS: Guarding the Door to IPP

IPP involves the printer client using HTTP to connect to the CUPS server. By default, CUPS selects a secure option and only permits connections of this type on *localhost*, that is, the machine running CUPS. This setting is not much help if you need to print to a network printer via the CUPS server. In this case, you need to set up CUPS to accept connections from other computers. To do so, make sure you are *root*, and open the */etc/cups/cupsd.conf* file for editing. Look for the line that starts with *<Location />*. The directive looks like this under Debian:

```
<Location />
Order Deny,Allow
Deny From All
Allow From 127.0.0.1
</Location>
```

To allow clients on your local network to print, you need to add an “Allow” entry for the network. The entry should look like the entry in our example, but use the IP range for your network. You can use wildcards, if needed. If you use the *192.168.1.** address range, the appropriate “Allow” entry would be as follows:

```
Allow From
192.168.0.*
```

Add this line to the end of the directive in *cupsd.conf* but before the line with *</Location >*. Save

your changes when you have finished.

Finally, there are two aspects of the CUPS system you need to change to allow the printer to identify files sent to it by Windows clients. Modify the two lines in *mime.convs* and *mime.types* as described earlier in the Windows 95, 98, and ME section. Relaunch your CUPS system after doing so.

Windows 2000: Class Struggle

To work around the bug in Windows 2000’s handling of IPP, use the Web interface shown in Figure 1 to assign the computer to a “Printer Class”. Launch the Web interface, click on *Do Administration Tasks*, and then *Add Class*. Fill out the required fields.

Note that blanks and special characters are not permitted in the *Name* box (though you could simply type “Class,” for example). You can add arbitrary entries for all the other fields. In the dialog that then appears, select the printer you configured earlier (this should be the only one in the list for a new installation). Then click on *Continue* to complete the procedure.

The next step is to configure the printer under Windows 2000. To do so, open the add printer dialog. Select *Network printer* as the printer type, and in the dialog that then appears, opt for *Con-*

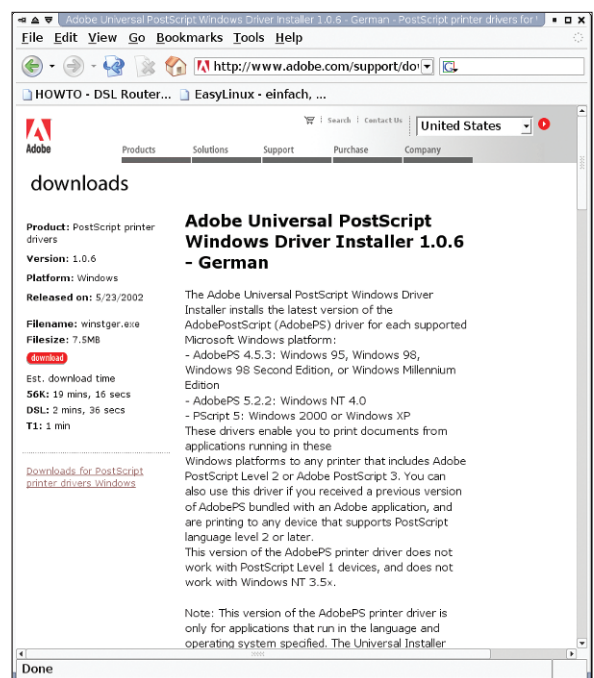


Figure 5: Adobe’s PostScript driver for Windows: a good idea, but the driver has some restrictions.

nect to a printer on the Internet or on a home or office network. Enter the URL for the printer attached to your CUPS machine into the URL box. You need the following syntax: "http://Hostname:631/classes/classname".

For example, if your CUPS machine is called "minerva," and you called the printer class you just set up on CUPS "Class," the correct entry would read `http://minerva:631/classes/Class`. Keep the printer driver handy, as Windows will prompt you for the driver in the next step. At the end of this process, you should be able to print the test page without any trouble.

Windows XP: Against All Odds!

Windows XP is the only Windows version that supports IPP-based printing, except for the mini-bug with the `hosts` file. Setting up IPP printing should be trivial. After preparing CUPS, as described earlier, launch the Add Printer dialog. Select *Network printer* as the printer type, and, as you did for Windows 2000, go on to select *Connect to a printer on the Internet or on a home or office network*. Enter the path to the printer attached to your CUPS machine into the URL field. For Windows XP, the path name is composed as follows "http://hostname_of_cups_printer:631/printers/printer".

For example, if your CUPS server is called "minerva," and you called the printer `HP_DESKJET_880C` in Step 1, the entry would read `http://minerva:631/printers/HP_DESKJET_880C`. Again, make sure you have the Windows driver. Finish the printer Wizard -again you should be able to print a test page at the end of this process.

Coup de Grace: Adobe's PostScript Driver

Adobe's PostScript driver provides another way of allowing any Windows operating system to use IPP for CUPS-based printing. Just to explain: PostScript is an internationally standardized printer file format that is typically used by professional and industrial printers. The PostScript driver allows users to send PostScript files, such as posters and high-definition images, directly to industrial printers, thus avoiding loss of



Figure 6: linuxprinting.org has PPD files for most printers and information on Linux printer support.

quality due to format conversions. In our scenario, this is not the recommended approach.

There are several disadvantages: for example, the Adobe driver does not support the full range of functionality that the PostScript standard defines. Additionally, the driver is only available across the Internet, whereas any Windows version can use the standard approach. And finally, the Adobe driver does not seem to support Windows XP; Windows recognized the pseudo-printer we created as a monochrome device but refused to print in color. If you still want to try out the PostScript driver because you want to print a PostScript file (and your normal, low budget printer does not support PostScript), follow these steps:

- First, you need to set up CUPS on the print server, and Windows on the client, as described earlier in the section on Windows XP and Windows 2000. As the Adobe driver also supports older versions of Windows, you can even try this with Windows 95, 98, and ME. The `hosts` file for these versions is at the same location as under Windows XP.
- After configuring your CUPS server, and the client computer, you can download the Adobe driver off the

Internet. Look for the *PostScript printer drivers* link on the Adobe website at `http://www.adobe.com/support/downloads/main.html`.

- Download the file and launch the setup. In the *Printer connection type* dialog, select *Printer is attached to network (network printer)*. In the dialog that follows, you need to specify the path to the printer on the CUPS server as follows: "http://server_name/printers/printer_name". Assuming that your CUPS computer is called "minerva," and you called the printer you set up in Step 1 `HP_DESKJET_880C`, the correct path would be `http://minerva:631/printers/HP_DESKJET_880C`. In the windows that follow, simply confirm the defaults. After completing the installation, you need to reboot your Windows computer. The PostScript printer should appear in the Control Panel after rebooting. If everything goes according to plan, you should now be able to print a PostScript file with the Adobe driver. ■

THE AUTHOR

Martin Loschwitz is from a small German town called Niederkrüchten and a developer for Debian GNU/Linux. Martin's leisure time is mainly pre-occupied with activities in the Debian or GNU community.