

Chapter 19

Setting Up Printing with CUPS

In the early days of Linux, printing was difficult to set up and equally difficult to manage. The print system was known as LPD (line printer daemon). Just as with X configuration, in the early days, at least, grown men wept. I still have bitter memories from 1997 of trying to make sense of the Linux Printing HOWTO and then, when I thought I had cracked it, ending up with a huge stack of paper covered in apparent garbage (raw PostScript code).

Fortunately, those days are gone. The standard now is CUPS (the Common Unix Print System), which implements (among other protocols) IPP (the Internet Printing Protocol). CUPS is also used by Mac OS X and is available for other forms of Unix.

A CUPS server can act as a print server for clients running all operating systems, including Windows. This means that it is not necessary for a Linux server to run Samba (see Chapter 18) to offer printing services to Windows clients.

On SUSE Linux, as one would expect, the configuration of printing has been integrated into YaST. In most cases, YaST's printer configuration tool is all you need to set up printing — both for a single machine and for a print server for a small local network. If you need a print server with more complex requirements, then a knowledge of the CUPS configuration files and possibly also its web interface is useful.

SUSE's default CUPS setup differs only slightly from the standard CUPS defaults. This means that a simple setup for a locally attached printer or a simple network print server works using YaST — you don't need to use the CUPS administration tools.

IN THIS CHAPTER

Setting up local printers
with YaST

Printing to remote printers

Setting up a print server for
the local network

Setting up clients on Linux or
Windows

Using the CUPS web interface

Controlling printing with
command-line tools

CUPS logs

Documentation

Printing in Linux: The Role of PostScript

As we hinted previously, the essentials of printing in Linux have their origins in the way printing was handled in traditional Unix systems, which originally printed to line printers that were only capable of printing lines of text characters.

The first printers with graphical capabilities used the PostScript language. As a result PostScript became a standard page description language. PostScript was introduced by Adobe Systems and dates back to 1982. It is both a page description language and a printer control language. (That is, it describes the layout of the text and graphical elements on the page and can directly control the printer if the printer uses PostScript as its "native language"). PostScript is actually a full programming language: It is possible to include loops and other programming constructs in PostScript files, which can make the output arbitrarily complicated.

When other printer languages became common, PostScript was still used for the internal representation of the page on Unix systems: The PostScript file was then sent to the printer through a filter that converted it into the printer's native printer language.

Printers from different manufacturers use different control languages to describe the pages they are going to print. In the case of HP printers, this will be one of the versions of PCL (Printer Control Language, which is currently at version 6). Epson printers use a language called ESC/P. Whatever the "native language" of the printer is, when an application wants to print, the PostScript code that the application outputs will need to be converted from PostScript to that native language.

This basic method still applies: Normally the Ghostscript program together with the PPD files (PostScript Printer Description files, which describe the capabilities of the printer and how the conversion is to be made) provide the method whereby the PostScript is transformed into the printer's own language.

PostScript as a file format (normally PostScript files have names with the extension .ps) is widely used in Linux documentation and as a way of exchanging formatted printable materials. The programs `gv`, `kghostview`, `ggv`, `evince`, `okular`, and others will display PostScript on the screen. A PostScript file is, in fact, simply a text file with drawing instructions and text characters in it; if you are curious, view a PostScript file with a pager (such as `less`) or a text editor.

A full description of PostScript is available in documents published by Adobe, including: *PostScript Language Reference* (known as the Red Book) and *PostScript Language Tutorial and Cookbook* (known as the Blue Book). These are available as printed books and also in PDF format at www.adobe.com/products/postscript/pdfs/PLRM.pdf and http://partners.adobe.com/public/developer/ps/sdk/sample/index_psbooks.html, respectively.

Setting Up a Locally Connected Printer

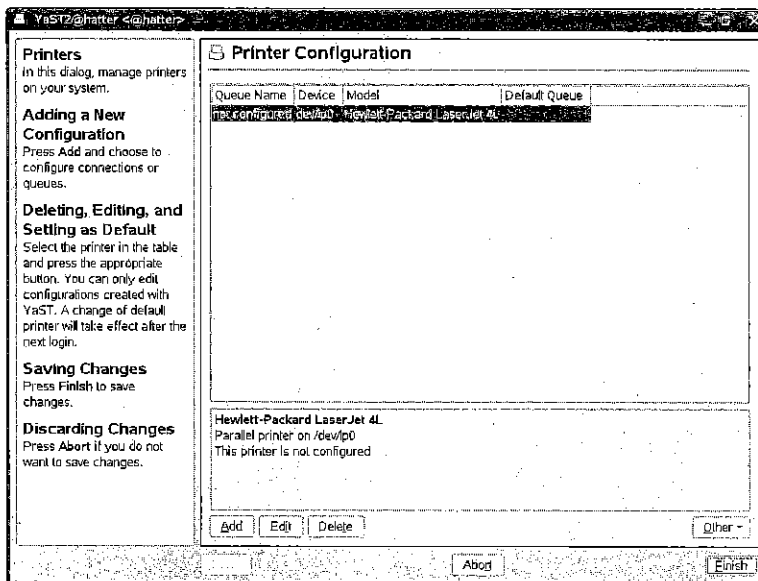
To begin setting up a printer using YaST, start YaST and from the Hardware menu, select Printer, or from the command line type:

```
yast2 printer
```

You will see the window shown in Figure 19-1.

FIGURE 19-1

YaST detecting the locally attached printer



YaST usually detects any directly attached local printers (such as parallel or USB printers) immediately. For example, in Figure 19-1 a Laserjet 4L was detected on the parallel port.

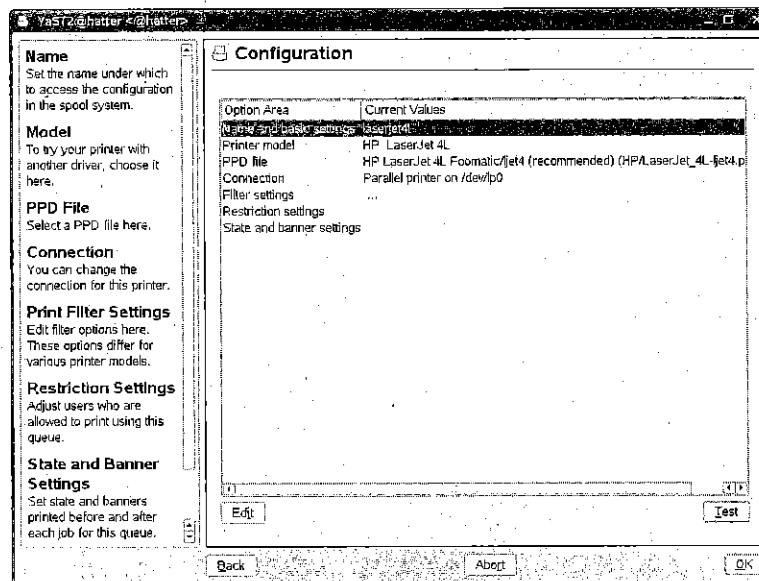
If the printer (or perhaps more than one printer) has been correctly detected, select the printer you wish to configure and choose Edit. Figure 19-2 shows YaST's suggested configuration. If it is not detected automatically, you may need to choose the model manually from a long list of manufacturers and types. A particular printer may be supported in more than one way by CUPS.

PPD Files

PostScript Printer Description files (PPD files) control how information is sent to your printer in its native format. These files were originally relevant only to printers requiring output in Adobe's PostScript printer language. However, the CUPS system uses PPD files to describe the capabilities of non-PostScript printers; PPD files for all printers supported by your SUSE version will be found on the system in compressed form under the directory `/usr/share/cups/model/`. In some cases, more than one PPD file is available for a given printer model, and using a different file may sometimes give different results in terms of functionality (including such things as whether two-sided printing is supported). It is possible to add PPD files to your system. You can find more information at www.linux-foundation.org/en/OpenPrinting/Database/PPDDocumentation.

FIGURE 19-2

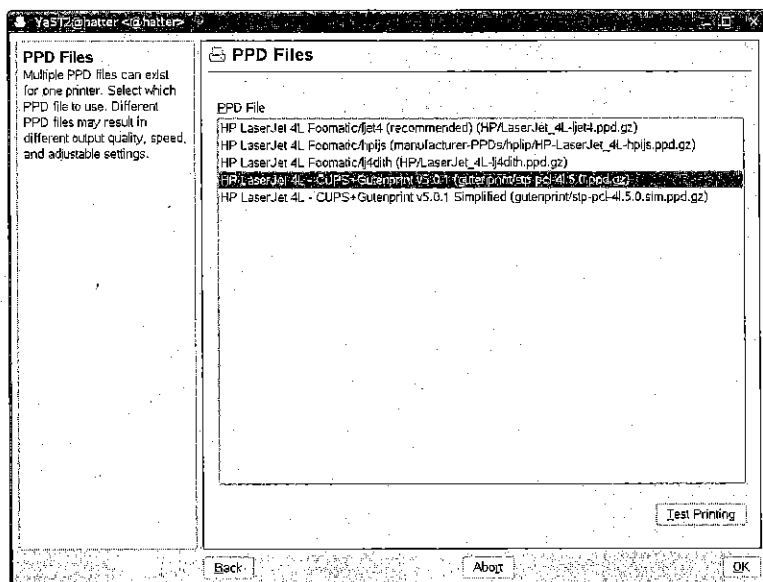
YaST's printer module: suggested configuration



You will find some options here that you can change. In particular, you can change the name that YaST has chosen for the printer (based on its model — in our example `laserjet4l`) to something more descriptive (like `reception` or `marketing`), and you can change the suggested PPD file that will be used. In general, you should accept the recommendation offered by YaST, but you may be able to change the quality of printing, particularly for graphics, by experimenting with the alternative PPD files (see Figure 19-3).

FIGURE 19-3

Choosing a different PPD file in YaST



You can now print a test page. If all is well, the printer will now produce a nice page with a SUSE logo at the top, a photograph of a real chameleon and various colored test images. If nothing comes out of the printer, check that the cable is properly connected. Particularly in the case of a parallel printer, be sure that the cable is not damaged at either end (if a pin is missing or broken on the connector, you can spend a long time wondering why the printer is detected, but no test page is emerging).

In almost all cases, setting up a local printer is as simple as this. There are two problems that might occur. Your printer might not be listed by YaST, and worse, it might not be supported at all.

About CUPS

The Common Unix Printing System (CUPS) is a network printing system. Whether or not you are printing to a local or a remote printer using CUPS, you will be using a network printing protocol (typically IPP, around which CUPS is based) to connect to cupsd, the CUPS daemon. By

continued

(continued)

default, CUPS offers a network service on port 631. IPP is essentially an extension of HTTP. Hence the CUPS configuration files look rather like Apache's configuration files.

CUPS offers an administrative interface over HTTP that can be accessed through a URL, such as `http://localhost:631/`. (The SUSE documentation and default settings do not particularly encourage the use of the web administration interface in CUPS, and for simple setups you will not need it.)

A CUPS server that is attached to a printer can advertise its presence on the network by putting out broadcasts; the clients that will print through it learn about the existence of the server from these broadcasts, and if the server allows them to do so, they can print through it without further configuration.

A CUPS server can also print to remote printers using a variety of protocols, including Windows printer shares and direct printing to a network printer.

The CUPS project web pages are at `www.cups.org`.

Printers Not Listed by YaST

The fact that the printer is not listed does not necessarily mean that it is not going to work. If the printer is a new one from one of the major vendors such as HP or Epson, there's actually a very good chance that it is a variant of one of the printers that is listed by YaST, but is a slightly changed model that has come on the market recently. So the best strategy is to select the closest model that you can find manually. Then try printing a test page and see what happens. In many cases this will work fine. The information about your printer at `www.linux-foundation.org/en/OpenPrinting` may help with this process.

Unsupported Printers

Unfortunately, some printers just won't work on Linux. There is a class of printers known as GDI printers, which tend to be cheap and frankly rather nasty printers; they are designed to interact directly with the Windows graphics device interface, and they do not have an independent open printer control language. These printers are either unsupported or poorly supported in Linux. (This is, in a way, the printing equivalent of the Winmodem problem.) The best place to get good information about whether particular printers will work is in the OpenPrinting pages on the Linux Foundation site referred to previously, where there is a database of printers currently supported by CUPS, as well as good general information and a buyer's guide that has good advice about which printers to avoid. There is a good discussion of GDI printers at `www.linuxprinting.org/show_printer.cgi?recnum=Generic-GDI_Printer`.

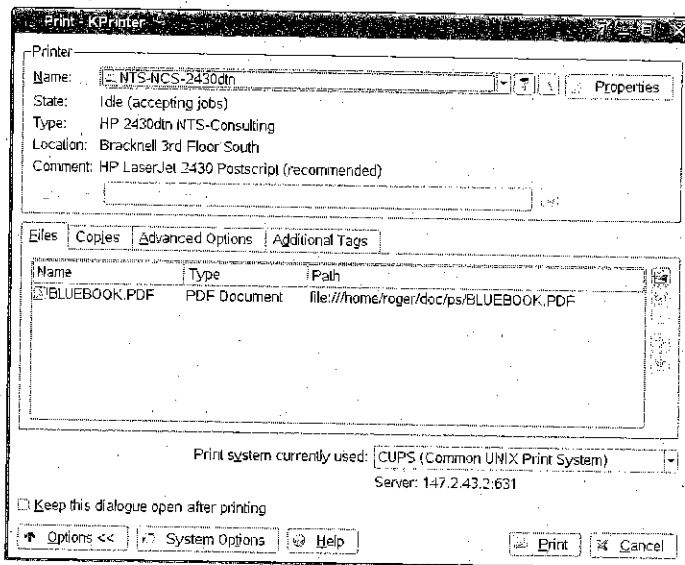
Printing from Applications

After you have set up a printer, printing from any application should work without problems. In the case of KDE applications, printing is handled through Kprinter (see Figure 19-4), which

is a common printing interface with nice features such as the ability to print multiple pages on one page. Kprinter both acts as a common print dialog box in all KDE applications and can be used as a standalone printing application. In KDE, you can print files simply by dragging them from Konqueror and dropping them into the Kprinter window. In addition to printing to available printers that have been set up, Kprinter supports printing to files in PostScript and PDF.

FIGURE 19-4

Using Kprinter



KDE also has a print monitor. When Kprinter has sent a job for printing, the print status monitor Kjobviewer appears (initially as an icon in the notification area system tray) to show the progress of the print jobs.

Other applications may print directly to the print system rather than going through Kprinter.

GNOME has its own print tools: the print dialog boxes in the native GNOME applications, and `gnome-cups-manager`, which includes both a print queue viewer and a tool for adding printers. We advise sticking to YaST for that job, however.

Printing from the Command Line

After printing is set up, you can print various types of files from the command line. The basic command to print a file is `lpr` (or `lp` — they are essentially equivalent, but there are two commands for historical reasons):

```
lpr bible.ps
```

The preceding command prints the postscript file `bible.ps` to the default printer.

If more than one printer is available, use a command such as this:

```
lpr -P laserjet41 bible.ps
```

The file is interpreted as a PostScript file and rendered correctly on the page. In the same way, you can print other standard file types directly from the command line because appropriate filters to convert them are included in the print system. For example, the following commands will all work (assuming the files really are in the formats suggested by their filenames):

```
lpr bible.pdf
lpr bible.png
lpr bible.jpg
lpr bible.html
```

In the case of PostScript and PDF files, this method is entirely safe and will always produce the desired result. Other file types might sometimes be detected incorrectly. In that case, there is a risk that it might try to render the result as plain text on the paper, which is not what you want. In general for HTML and graphics file types, it makes more sense to open them in a screen viewer first (Konqueror or GIMP, for example) and then print using that application's print dialog box. This usually gives you some additional control over the appearance of the final printed copy in any case.

Canceling a Print Job from the Command Line

Sometimes you want to cancel a job that has already been submitted. You need to look at the print queue, select a job, and then remove it.

lpq

The command `lpq` shows the default print queue — for example:

```
roger@bible:~> lpq
laserjet41 is ready
no entries
```

This means there's nothing left in the print queue. However, if there are jobs in the queue, you will see something like this:

Rank	Owner	Job	File(s)	Total Size
			Laserjet41 is ready and printing	
Active	roger	118	(stdin)	2725888 bytes
active	roger	119	bible.pdf	1354558 bytes

NOTE

The command `lpq` shows the default print queue. If there is more than one queue and you want to display another queue, you need to use the option `-P` to specify it,

like this:

```
lpq -P laserjet41
```

If there is more than one printer available, you can set the default printer in YaST in the main screen that displays all configured printers. Select the **Other** button and choose "Set as default."

The Commands `cancel` and `lprm`

You could now type **cancel 119** or **lprm 119** to remove the second of these jobs from the queue. Of course, you must be the owner of the job to do this. You may often find that you're too late — the job has gone to the printer and is in the printer's memory. Whether you can easily cancel the job at that stage depends on the printer.

Setting Up a Simple Print Server on the Local Network

If you have successfully set up a printer on the local machine, you have also successfully set up a print server for the local network. The SUSE default settings in the CUPS configuration file `/etc/cups/cupsd.conf` ensure this. In the Browsing Options section of `/etc/cups/cupsd.conf`, this is controlled by the lines:

```
BrowseAllow @LOCAL  
BrowseDeny All
```

The CUPS server advertises itself on the local network and is ready to accept jobs from other machines after they have been set up correctly to print to it. Of course, you can change this behavior if you want to, but for the time being we assume that you have the default configuration on the machine that has the printer attached.

In this example, the printer is attached to the machine `hatter`, and we are setting up printing on another SUSE Linux machine. Do the following:

1. Start the YaST printer module. If no printers have been set up before, and no local printers are detected, choose **Add**. You will now see a screen where you choose between adding local or network printers. Choose **Network Printers** and a dialog appears, similar to the one in Figure 19-5.
2. Select **Print via CUPS Network Server** and then, in the next screen, choose the CUPS **Client-Only** configuration. You will be asked to give the server name, as shown in Figure 19-6 (or you can allow YaST to detect it, or choose it from a list of all hosts on the network).
3. After you have clicked **OK** and **Finish**, you should be able to print across the network to the printer attached to the server named `hatter`.

FIGURE 19-5

Choosing a printer type: remote CUPS server

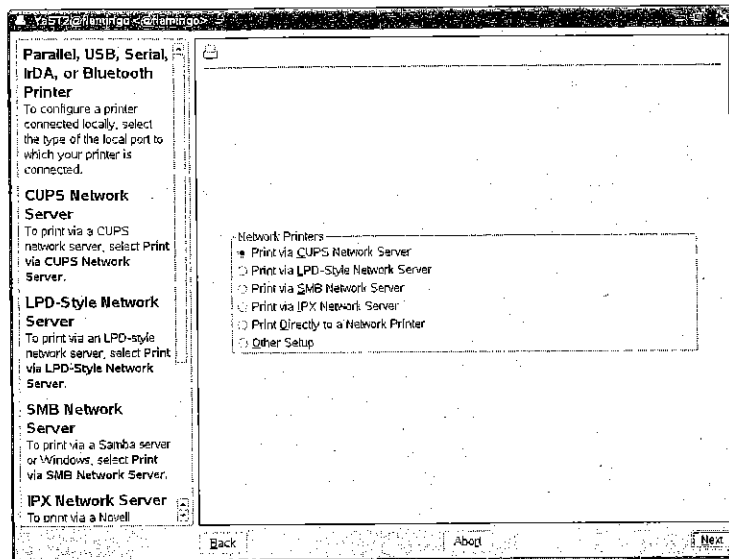
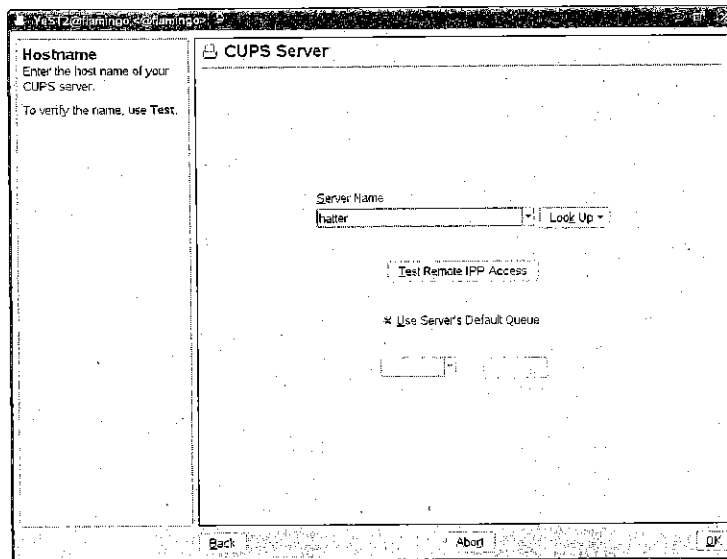


FIGURE 19-6

Setting up the remote CUPS server in YaST



Starting and Stopping the CUPS Server

On SUSE, CUPS is started and stopped by the commands `rc cups start` and `rc cups stop`. It is just another service controlled in the usual way. If you make changes to the CUPS configuration files, you will need to restart CUPS for the changes to take effect.

You can check that CUPS is running with the command `rc cups status`. By default, CUPS will be started in runlevels 2, 3, and 5.

Checking That the Remote CUPS Server Is Available

From the client machine, you may want to be sure that CUPS is running on the server and available across the network. As with other services, you can check this by seeing if you can connect to the relevant port, using a command like:

```
telnet hatter 631
```

If you see output like this, CUPS is available across the network:

```
Trying 192.168.2.6 ...
Connected to hatter.
Escape character is '^['.
```

Now use `Ctrl+] and type quit to disconnect.`

If there is a problem you may see this:

```
Trying 192.168.2.6 ...
telnet: connect to address 192.168.2.6: Connection refused
```

Such code appears for a number of reasons: There could be a problem with the network, CUPS might not be running, or there could be a firewall rule on `hatter` blocking the connection.

Setting Up a Windows Client to Print to the CUPS Server

Recent versions of Windows support the IPP protocol, so you can set them up to print to a CUPS server.

NOTE

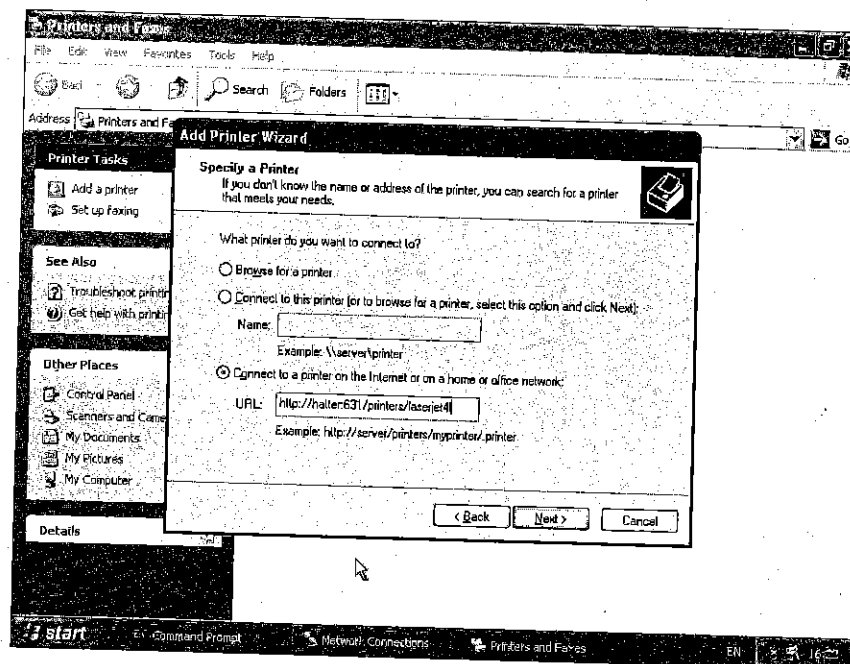
Older versions of Windows require an additional program to make use of IPP printing. To use IPP printing on Windows 95 and 98, you need to download the file `wpnpins.exe` from www.microsoft.com/windows98/downloads/contents/WUPreviews/IPP/.

Printing from a Windows client using IPP is an alternative to using Samba as the print server on Linux. The printer does not appear as a Windows shared printer from the point of view of the Windows client, and unfortunately this means that you can't just browse for it as you would for a Windows shared printer or Samba printer. You have to enter the specific URL in the Windows Add Printer Wizard dialog box. Figure 19-7 shows what to expect. The URL that is required is

made up of the name or IP number of the CUPS server followed by :631 (the IPP port number), followed by /printers/, and finally the name of the print queue on the server. Hence, you see in our example `http://hatter:631/printers/laserjet41`.

FIGURE 19-7

Setting up printing to a remote CUPS server in Windows



Printing from Linux to Other Types of Remote Printers

When you add a printer using YaST, a bewildering variety of alternatives are presented. In most cases, if you need the particular protocol or connection type, you will know about it in advance. The most common cases that you are likely to need to set up are printing to a Windows or Samba shared printer and printing directly to a network printer. In YaST these correspond to the Print via SMB Network Server and Print Directly to a Network Printer options.

Printing to an SMB Network Server

This could be a Windows machine that is sharing its printer or a Samba server. From the client's point of view they look just the same.

One thing to note here is that YaST wants to be able to find the server on the network by DNS; it won't automatically use Windows network names. So it is possible in some circumstances that you might need to add an entry to `/etc/hosts` for the SMB print server before YaST will be able to find it. Otherwise, this will just work.

Printing Directly to a Network Printer

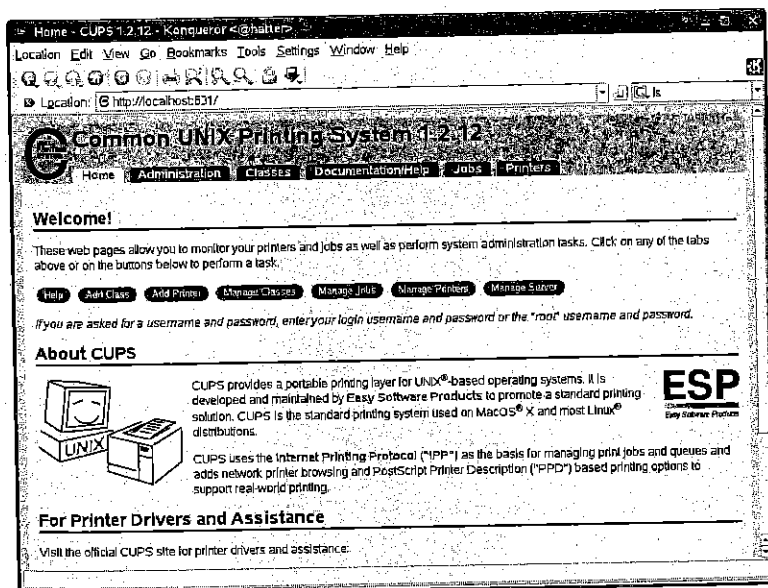
Printers that are directly attached to the network will usually be doing direct TCP port printing and will be listening on port 9100 (the first of the options that YaST's dialog box offers). Setting this up is simple provided you know which printer you want to print to. YaST will scan the network and display a list of available printers for you to choose from in any case.

Using the CUPS Web Interface

The CUPS web interface can be viewed from a browser using port 631 (see Figure 19-8). By default, SUSE's settings allow only administrative changes through the browser interface when connecting from the local machine. This can be changed in the `cupsd.conf` file, but for now we will look at administering the server from a browser running on itself. So from the local machine, you need to browse to `http://localhost:631`.

FIGURE 19-8

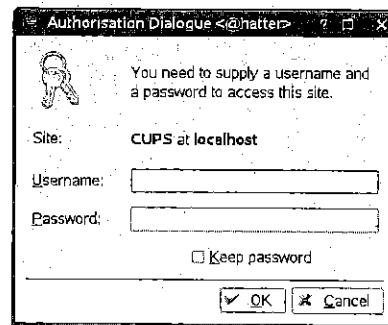
The CUPS web interface



Some of the functions that are available simply provide information. Others can change the set up. Administrative rights are required on the CUPS server and, when you access these functions, an authentication dialog box is displayed (see Figure 19-9).

FIGURE 19-9

The CUPS web interface with authentication dialog box



The SUSE default settings will accept the user root and the root password at this point. Alternatively you can create a CUPS user from the command line using a command like the following:

```
lppasswd -g sys -a roger
```

This adds the pre-existing system user roger as a CUPS user and prompts you for a password. After doing this, roger becomes a CUPS user with the ability to administer CUPS through the web interface. Now you can log into the administrative interface as this user and administer the CUPS server. Oddly, however, on recent SUSE versions, the authentication mechanism that is used is the system's standard authentication, so the password that you set with the `lppasswd` command is irrelevant. When you are logged in, you can perform all administration tasks from the web interface, including adding printers. The changes you make will then be written back to the CUPS configuration files, `/etc/cups/`.

The user information is specific to CUPS and is stored in the file `/etc/cups/passwd.md5`.

Working with Classes in CUPS

One of the nice features of CUPS is the fact that you can create a class of printers (for example, a group of printers in a certain physical location). The class will consist of a set of printers to which the CUPS server can print, whether locally connected or across the network. After a class is set up, users' print jobs will be printed on any one of the printers in the class to which they have access.

In the web interface, you can add a class (you will be prompted for its name, location, and description, which can all be arbitrary). You then add printers to the class from the list of printers that CUPS already knows about. Figure 19-10 shows you how to add a class in the CUPS web interface.

For example, if you are configuring a Windows system to print to the class queue called sales, you would choose `http://hatter:631/printers/sales`.

Allowing Remote Access to the CUPS Web Interface

As we mentioned earlier in the chapter, by default, CUPS does not allow you to log in to its web interface from a remote location. If you want to change this, you have to edit the file `/etc/cups/cupsd.conf`. In the section that begins

```
<Location /admin>
```

add a line

```
Allow from @LOCAL
```

Then save the file and restart CUPS:

```
rc cups restart
```

Now you can use the web interface from other machines on the local network. Of course you can restrict this to a single IP address if you want by replacing `@LOCAL` by that IP address.

The CUPS Command-Line Tools and Configuration Files

CUPS also provides a set of command-line tools that can do all the administration that the web interface allows. These are the commands provided by the `cups-client` package. In particular, the `lpinfo`, `lpadmin`, and `lptions` commands provide the functionality that the web interface provides, but from the command line.

In general, to avoid problems, you should use the available tools in the following order of preference:

- The YaST printer module for basic setup
- The CUPS web interface
- The CUPS command-line tools

In other words, do what you can with YaST. If you can't do a task with YaST, use the CUPS web interface and use the command-line tools only if you have to.

lpinfo

`lpinfo` shows the available printing options. `lpinfo -v` shows the potential devices that can be used for printing. `lpinfo -m` shows the available printer drivers.

ladmin

The command `ladmin` has a very large number of options to administer printing. You will probably prefer to use the web interface if you can, but here are one or two examples. The following:

```
ladmin -h hatter -p laserjet41 -u allow:roger,justin
```

allows the users `roger` and `justin` to print to the queue `laserjet41` on the CUPS server `hatter`. And this:

```
ladmin -h hatter -p laserjet41 -u deny:badman
```

prevents the user `badman` from printing to the queue `laserjet41` on the CUPS server `hatter`.

```
ladmin -p newprinter -c sales
```

Typing this command on `hatter` will add the printer `newprinter` to the class `sales` of printers that we set up earlier.

lptions

The command `lptions` can be used to view the current options for a queue or to set new options:

```
root@hatter:~ # lptions -p laserjet41 -l
REt/REt Setting: Dark Light *Medium Off
TonerDensity/Toner Density: 1 2 *3 4 5
Dithering/Floyd-Steinberg Dithering: *Normal FSDithered
Manualfeed/Manual Feed of Paper: Off On
InputSlot/Media Source: *Default Tray1 Tray2 Tray3 Tray4 Envelope Manual Auto
Copies/Number of Copies: *1 2 3 4 5 6 7 8 9 10 11 12 13 [. . .] 100
PageSize/Page Size: *A4 Letter 11x17 A3 A5 B5 Env10 EnvC5 EnvDL EnvISOB5
EnvMonarch Executive Legal
PageRegion/PageRegion: A4 Letter 11x17 A3 A5 B5 Env10 EnvC5 EnvDL EnvISOB5
EnvMonarch Executive Legal
Resolution/Resolution: 75x75dpi 150x150dpi *300x300dpi
Economode/Toner Saving: *Off On
```

To change an option, do something like this:

```
lptions -p laserjet41 -o TonerDensity=5 -o Resolution=150x150dpi
```

This changes the Toner Density and Resolution to new settings.

lpstat

lpstat has many options to provide status information. For example:

```
roger@hatter:~ > lpstat -s
system default destination: laserjet41
members of class sales:
    laserjet41
device for laserjet41: parallel:/dev/lp0
```

This is a summary (-s) of information about the printing system.

lpstat -r tells you whether the CUPS scheduler is available. lpstat -p gives you a listing of printers and their current status. lpstat -t gives fuller status information. lpstat -R shows the current print jobs with their owners and ranking in the queue.

The CUPS Configuration Files

The configuration files are in the directory `/etc/cups/`. The most important files are `cupsd.conf`, `printers.conf`, `classes.conf`, and `lpoptions`. Apart from making changes to `cupsd.conf` to change authentication settings (which hosts can connect and so on), in general it is best to make changes from YaST or using the web interface or the command tools, and let CUPS write the configuration files itself. The danger is that otherwise you may find that the changes you have made will be overwritten when you use the graphical tools later.

The CUPS Logs

CUPS logs its activity in three log files: `/var/log/cups/access_log`, `/var/log/cups/page_log`, and `/var/log/cups/error_log`. The names are self-explanatory.

access_log

The file `access_log` shows access to the CUPS server in a rather similar way to the Apache web server logs. It shows the requesting host name and the date for each access to the CUPS server.

page_log

The file `page_log` shows the user and the job name, as well as the number of pages printed. Each page printed shows in the file as a serial number against the same job number. By analyzing the page log you can do user accounting. A number of tools are available for this, including PyKota (www.pykota.com/).

error_log

This file is not really so much an error log as a place where CUPS logs everything else about its activity. It is useful for debugging problems because it shows what CUPS is doing in the background — for example, which filters are being used to print files and so on.

Other Tools

There are some additional tools (not part of the CUPS package) that are useful for printing.

kprinter

The KDE interface includes the `kprinter` tool, which handles printing from all KDE applications. You can set other programs to print via `kprinter`, too. It has the ability to manage (among other things) a number of different paper sizes and printing multiple pages to a single physical page.

xpp

`xpp` is the X Printing Panel, a graphical application similar in conception to `kprinter`, but independent of KDE. You can use it to browse for files to print, choose print queues to send them to, and choose various options regarding the appearance of the output, the number of copies, and so on.

gtklp

The `gtklp` package provides the `gtklp` graphical printing tool. This tool is similar in functionality to `xpp` but has a cleaner interface. It allows, among other things, scaling and color adjustment when printing image files.

Documentation

As is always the case with SUSE Linux, you can find a great deal of documentation to support you as you set up your printing, both locally and on your networks.

CUPS Online Documentation

The CUPS documentation is installed on the system and can be viewed either through the CUPS web interface locally at <http://localhost:631/help/> or on the CUPS web site at www.cups.org/documentation.php. The commands `lp`, `lpr`, `lpadmin`, `lpoptions`, `lpstat`, `lpq`, and `lprm`, and other CUPS commands all have man pages.

The CUPS Book

There is an official CUPS book: *CUPS: Common UNIX Printing System* by Michael Sweet (Sams, 2001). The book is available on line at www.cups.org/book/index.php.

SUSE Printing Documentation

You can find a good many articles on printing on the openSUSE and Novell sites. In particular, see <http://en.opensuse.org/CUPS>, http://en.opensuse.org/SDB:CUPS_in_a_Nutshell, and the printing chapters in the openSUSE or SLES manuals.

IPP Documentation

You can find more information about IPP on the Printer Working Group's site at www.pwg.org/ipp/. This includes links to the relevant RFC documents.

CUPS and the SUSE YaST printer module make setting up printing relatively easy. CUPS has far more capabilities than can be covered here. If you intend to use CUPS in a large networked environment, you should read the full documentation to find out more. In particular, you should read about such features as page and job accounting.

There are also useful articles about CUPS on the openSUSE site at http://en.opensuse.org/SDB:CUPS_in_a_Nutshell and <http://en.opensuse.org/CUPS>.