

WORKSHOP

PRINT SERVERS

Windows To Linux Printing (and Vice Versa)

By Pete Payne

With Linux reaching servers and catching on at the desktop, more enterprises must ensure interoperability among multiple versions of Windows and Linux. This isn't easy, even for one of the most basic tasks: printing.

If you don't have the proper cross-platform print tools, Linux clients will be restricted to Linux print servers, and Windows clients to Windows print servers, potentially doubling the cost of your network printing hardware and maintenance. Older methods of network printing, such as LPD and LPRng, won't always help—vendor extensions to these technologies poorly support or don't support the many features a Windows-Linux environment needs, such as printer browsing, remote print-server administration, failover, clustering, and integration with authentication and authorization systems.

But two technologies provide printing solutions for a mixed Linux-Windows environment: Samba and CUPS (Common Unix Printing System). You can use one, or combine the two for a robust cross-platform print setup.

Do That Samba

Samba, an open-source software suite, uses the SMB (Server Message Block) network protocol so that a non-Windows machine, such as a PC with Linux, can communicate with a Windows box using Microsoft's preferred networking protocol. To the Windows client, it's just networked to another Windows box, not to Linux or another operating system. Samba provides many of the services of a Windows server, including file sharing, authentication, authorization, and name and print serving. Samba version 3 and higher also integrate well with Windows domain-control methods, including NT ACLs (access-control lists), Active Directory and LDAP.

CUPS uses IPP (Internet Printing Protocol) rather than SMB for network communication. An extension to HTTP, IPP is supported by all major Linux distributions

and Windows 2000 and above. The protocol enjoys broad vendor support and is becoming the standard network-printing technology for major OSs. Although Windows 9x doesn't natively support IPP, you can download the necessary software from Microsoft.

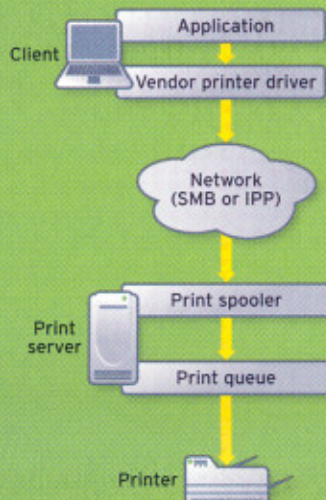
But CUPS is more than a simple print spooler: The software suite's API handles print commands from third-party software and provides print-queue management, print filters, a scheduler and a Web admin interface. Unlike Samba, which relies on a separate printing subsystem to do the printing, CUPS handles all aspects of local or network printing, from receiving the print job to printing, administration and statistics-gathering.

Refresher Course

Printing across a network is simple: The client machine typically prepares the print job using the vendor-specific printer driver for the target printer. Once the print job is complete, the print file is sent to the print server, which spools the file. When the printer is available, the print spooler forwards the print job to the printer. "In the Raw" (below) shows a print job being sent for printing without conversion. The print subsystem, which can be LPD, LPRng, a Windows print spooler or any suitable print spooler, acts as simple pass-through for the print job.

This spooler-only setup is easy to create in most enter-

In the Raw



With a raw, or pass-through, print architecture, the unmodified print file is sent to the printer, and the print subsystem acts as a simple pass-through for the print job.

prises, but difficult to administer. It makes most sense for home networks and small offices, where printers are rarely replaced or upgraded. Spooler-only setups don't work well in larger installations because each client must be configured separately with its vendor-specific printer driver. At best, this is a headache, but it's a nightmare if the printer driver gets a new update or you upgrade your target printer. To maintain these drivers, you must visit each desktop or push the drivers to the clients using remote-management software.

And the spooler-only setup doesn't support failover or clustering. The clients must "know" the print servers by name, and if a print server is not available, the print job won't automatically be sent to another server. Finally, not all print spoolers support the latest authentication and authorization methods. The print server controls access to printing resources, and each client must be explicitly granted access. So this step is often skipped, and the print server becomes an unsecured, centralized print server with no control over printer resources.

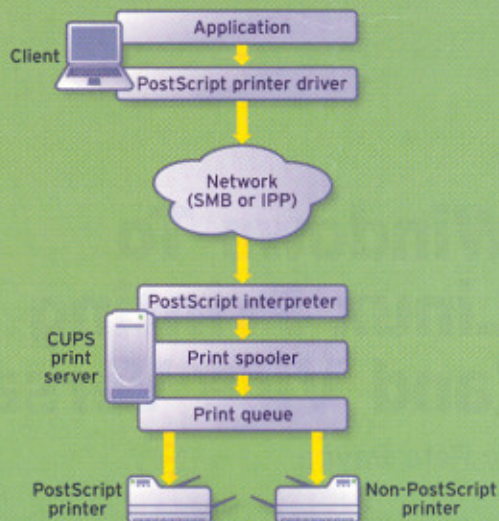
You can use Samba networking software in a spooler-only configuration. More involved installations of Samba can overcome these security problems and provide printer-driver uploads to clients. The trade-off is that Samba print spoolers don't support failover or clus-

When Trouble Strikes

Uh-oh. Something's wrong and you can't print. To troubleshoot, try rolling back each new feature, such as authentication and authorization, you've enabled to find out if it's causing the problem. Here are some other things to look for:

- » **Samba is "overconfigured."** The `smb.conf` file has lots of options—many of them aren't relevant to printing but still impact how clients access the print server. Start with the most basic information necessary to enable printing, then add configuration options as needed.
- » **You can't access CUPS.** By default, CUPS installs with Web admin access to the print server only from the local machine. You won't be able to access the administration section of the Web interface if you're connecting from a remote machine. Turn off this feature in the `cupsd.conf` configuration file and allow admin access from anywhere. Then restore this restriction once your configuration is complete.
- » **A firewall block.** Allow access via TCP Port 631 for CUPS, and UDP Ports 137 and 138 and TCP Port 139 for Samba.
- » **Allow and deny problems.** Verify the allow and deny directives for the CUPS server contained in `cupsd.conf`. These are similar to the access-control conventions used by Apache and other software. Be sure the clients and their subnets have access to the print server.

CUPS Serves Up Printing



CUPS handles all aspects of printing—from receiving the print job to printing, administration and gathering statistics.

tering. (You could force CUPS into a spooler-only configuration, but that's not the best use of the software.)

Put the Server to Work

A more flexible network-printing architecture has the print server perform some work on print jobs. In "CUPS Serves Up Printing" (above), the client doesn't format the print job using a vendor-specific printer driver. Instead, it converts the print job to PostScript format, and the job is forwarded to the print server. The print spooler is configured based on the targeted printer's PostScript capability. If the printer supports PostScript natively, the print spooler sends the print job directly to the printer with no further modification. If the targeted printer doesn't support PostScript, the printing subsystem applies one or more filters to the print job. These filters process the job by converting the incoming PostScript commands into whatever language the printer understands.

The major benefit of this architecture is that printer-specific commands, such as PCL, ESC/P2 and others, are no longer created by vendor-specific printer drivers on the client—they reside on the print server. The client's printer driver, which replaces the vendor-supplied driver normally installed on the Windows client, generates the print job with PostScript commands. If you upgrade the printer on the print server, you can configure the printing subsystem on the print server to use a new set of filters. The clients would be modified only if the existing print drivers were missing a feature or found faulty.

CUPS uses this architecture, and provides robust authentication and authorization using basic or digest authentication and/or certificate-based security. It also offers printer browsing over a Web browser. CUPS sup-

step by step

ports remote print-server administration through a standard Web browser, automatic failover and clustering, using implicit or explicit classes (groupings of identically named printers on a network), proxy printing, encryption, and an ever-growing library of print filters.

CUPS also provides multiple back ends. These let the output be sent to different targets, including parallel and serial devices, USB ports, Samba shares for printing to Windows print servers, and other CUPS servers for enhanced failover and clustering.

More Options

An even better way to execute network printing is to use both Samba and CUPS. Samba acts as the print spooler, which lets the print server join AD domains or authenticate using LDAP and other methods. This enhances and strengthens the built-in CUPS authentication. Plus, you can configure Samba to provide printer-driver uploads to clients, so you don't have to touch each client machine. These uploads can be scripted and executed when the user logs on. Samba also can publish printers to AD or LDAP directories, making their discovery and use across networks easier. This combined approach is the cream of mixed-network printing and gives you a flexible, configurable and stable network-printing architecture that ensures Windows and Linux play nicely together.

If you use Windows print servers, you can run CUPS locally on your Linux clients, which—when configured properly—converts PostScript print commands into those the targeted printer requires. The CUPS software then sends the print job to the Windows print server. So if you run CUPS on the Linux clients, you can run Windows print servers as well.

Another approach leverages Windows print servers by deploying a Linux machine running CUPS. This Linux box can be used by both Linux and Windows clients for network printing. And rather than doing the actual printing, the Linux machine forwards the print jobs to a Windows print server, which then sends the jobs to the printer. This way, you get to keep your Windows print servers plus gain the benefits of CUPS.

The network appliance is yet another option for Linux-Windows printing across a network. Akin to a hardware-based firewall or router, this appliance connects to the existing network and to one or more printers. It advertises the printers to the network and provides network printing. Using a Web browser, you can configure the printers, modify security and so on. But be sure the appliance you select supports all the operating systems from which you'll be printing. **NWC**

Pete Payne is a software programming consultant with a nationwide consulting firm. Write to him at ppayne@nwc.com. Post a comment or question on this story at www.nwc.com/go/ask.html.

TIPS FOR SETTING UP A WINDOWS-LINUX PRINT ARCHITECTURE

1 **Decide which architecture is best for your needs.** If your organization or office requires only basic printing services, a simple Samba server might be all you require. For more robust services and administration, go with the mixed Samba-CUPS server.

2 **Install the print-server software.** Many Linux distributions ship with Samba and CUPS already installed. But if the print-server software isn't included, download it.

3 **Make sure the print server can print to the local printer.** Use whichever local printing subsystem you're most familiar with. Both Samba and CUPS work with older printing subsystems, such as LPD. If you're building a new print server, use CUPS as your local printing subsystem.

4 **Use the most basic configuration settings that allow network printing.** For Samba, this involves editing the `smb.conf` file, adding only what's necessary to allow printing. For CUPS, the `cupsd.conf` file is where the CUPS daemon configuration settings are stored. An alternative to editing the `cupsd.conf` file manually is to use the Web interface the software provides. Point your browser to `localhost:631` to access the Web administration interface.

5 **Add features.** Add clustering, failover, authentication, authorization, load balancing, driver uploading, directory-service integration and browsing services as you need them.

