

Linux Printing Tutorial at Linux-Kongress 2002 Cologne, Germany:

(IX.) Playing Nice with MS Windows Systems:

Printing with CUPS in Samba 2.2.x

Table of Contents

(IX.) Playing Nice with MS Windows Systems: Printing with CUPS in Samba 2.2.x.....	1
1. Configuring smb.conf for CUPS.....	1
2. Using CUPS as a mere spooling print server -- "raw" printing with vendor drivers download.....	2
3. CUPS as a network PostScript RIP -- CUPS drivers working on server, Adobe PostScript driver with CUPS-PPDs downloaded to clients.....	2
4. Windows Terminal Servers (WTS) as CUPS clients.....	3
5. Setting up CUPS for driver download.....	3
6. Sources of CUPS drivers / PPDs.....	4
7. cupsaddsmb.....	5
About the Author:.....	8
Links:.....	8



(IX.) SAMBA and CUPS



(IX.) Playing Nice with MS Windows Systems: Printing with CUPS in Samba 2.2.x

CUPS is a newcomer in UNIX printing. It convinced many people upon first trial already. It is easy to manage and run. However, quite a few new features in CUPS make it different from other, more traditional printing systems. Therefore a few things need to be learned before you master CUPS.

1. Configuring `smb.conf` for CUPS

Printing with CUPS in the most basic `smb.conf` setup in Samba 2.2.x only needs two settings: **printing = cups** and **printcap = cups**. While CUPS itself doesn't need a `printcap` anymore, the `cupsd.conf` configuration file knows two directives (example: **Printcap /etc/printcap** and **PrintcapFormat BSD**), which control if such a file should be created for the convenience of third party applications. Make sure it is set! For details see **man cupsd.conf** and other CUPS-related documentation.

If SAMBA is compiled against `libcups`, then **printcap = cups** uses the CUPS API to list printers, submit jobs, etc. Otherwise it maps to the System V commands with an additional `-oraw` option for printing. On a Linux system, you can use the **ldd** command to find out details (`ldd` may not be present on other OS platforms, or its function may be embodied by a different command):

```
transmeta:/home/kurt # ldd `which smbd`
libssl.so.0.9.6 => /usr/lib/libssl.so.0.9.6 (0x4002d000)
libcrypto.so.0.9.6 => /usr/lib/libcrypto.so.0.9.6 (0x4005a000)
libcups.so.2 => /usr/lib/libcups.so.2 (0x40123000)
libdl.so.2 => /lib/libdl.so.2 (0x401e8000)
libnsl.so.1 => /lib/libnsl.so.1 (0x401ec000)
libpam.so.0 => /lib/libpam.so.0 (0x40202000)
libc.so.6 => /lib/libc.so.6 (0x4020b000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```



(IX.) SAMBA and CUPS

The line "libcups.so.2 => /usr/lib/libcups.so.2 (0x40123000)" shows there is CUPS support compiled into this version of Samba. If this is the case, and **printing = cups** is set, then any otherwise manually set print command in smb.conf is ignored.

2. Using CUPS as a mere spooling print server -- "raw" printing with vendor drivers download

You can setup Samba and your Windows clients to use the CUPS print subsystem just as you would with any of the more traditional print subsystems: that means the use of vendor provided, native Windows printer drivers for each target printer. If you setup the [print\$] share to download these drivers to the clients, their GDI system (Graphical Device Interface) will output the Windows EMF (Enhanced MetaFile) and convert it -- with the help of the printer driver -- locally into the format the printer is expecting. Samba and the CUPS print subsystem will have to treat these files as raw print files -- they are already in the shape to be digestible for the printer. This is the same traditional setup for Unix print servers handling Windows client jobs. It does not take much CPU power to handle this kind of task efficiently.

3. CUPS as a network PostScript RIP -- CUPS drivers working on server, Adobe PostScript driver with CUPS-PPDs downloaded to clients

CUPS is perfectly able to use PPD files (PostScript Printer Descriptions). PPDs can control all print device options. They are usually provided by the manufacturer -- if you own a PostScript printer, that is. PPD files are always a component of PostScript printer drivers on MS Windows or Apple Mac OS systems. They are ASCII files containing user-selectable print options, mapped to appropriate PostScript, PCL or PDL commands for the target printer. Printer driver GUI dialogs translate these options "on-the-fly" into buttons and drop-down lists for the user to select.

CUPS can load, without any conversions, the PPD file from any Windows (NT is recommended) PostScript driver and handle the options. There is a web browser interface to the print options (select <http://localhost:631/printers/> and click on one "Configure Printer" button to see it), a commandline interface (see **man lptions** or try if you have **lphelp** on your system) plus some different GUI frontends on Linux UNIX, which can present PPD options to the users. PPD options are normally meant to become evaluated by the PostScript RIP on the real PostScript printer.

CUPS doesn't stop at "real" PostScript printers in its usage of PPDs. The CUPS developers have extended the PPD concept, to also describe available device and driver options for non-PostScript printers through CUPS-PPDs.

This is logical, as CUPS includes a fully featured PostScript interpreter (RIP). This RIP is based on Ghostscript. It can process all received PostScript (and additionally many other file formats) from clients. All CUPS-PPDs geared to non-PostScript printers contain an additional line, starting with the keyword **cupsFilter*. This line tells the CUPS print system which printer-specific filter to use for the interpretation of the accompanying PostScript. Thus CUPS lets all its printers appear as PostScript devices to its clients, because it can act as a PostScript RIP for those printers, processing the received PostScript code into a proper raster print format.

CUPS-PPDs can also be used on Windows-Clients, on top of a PostScript driver (recommended is the Adobe one).

(IX.) SAMBA and CUPS

This feature enables CUPS to do a few tricks no other spooler can do:

- act as a networked PostScript RIP (Raster Image Processor), handling printfiles from all client platforms in a uniform way;
- act as a central accounting and billing server, as all files are passed through the **pstops** Filter and are therefor logged in the CUPS `page_log`. – *NOTE*: this can not happen with "raw" print jobs, which always remain unfiltered per definition;
- enable clients to consolidate on a single PostScript driver, even for many different target printers.

4. Windows Terminal Servers (WTS) as CUPS clients

This setup may be of special interest to people experiencing major problems in WTS environments. WTS need often a multitude of non-PostScript drivers installed to run their clients' variety of different printer models. This often imposes the price of much increased instability. In many cases, in an attempt to overcome this problem, site administrators have resorted to restrict the allowed drivers installed on their WTS to one generic PCL- and one PostScript driver. This however restricts the clients in the amount of printer options available for them -- often they can't get out more then simplex prints from one standard paper tray, while their devices could do much better, if driven by a different driver!

Using an Adobe PostScript driver, enabled with a CUPS-PPD, seems to be a very elegant way to overcome all these shortcomings. The PostScript driver is not known to cause major stability problems on WTS (even if used with many different PPDs). The clients will be able to (again) chose paper trays, duplex printing and other settings. However, there is a certain price for this too: a CUPS server acting as a PostScript RIP for its clients requires more CPU and RAM than just to act as a "raw spooling" device. Plus, this setup is not yet widely tested, although the first feedbacks look very promising...

5. Setting up CUPS for driver download

The **cupsadsmb** utility (shipped with all current CUPS versions) makes the sharing of any (or all) installed CUPS printers very easy. Prior to using it, you need the following settings in `smb.conf`:

```
[global]
    load printers = yes
    printing = cups
    printcap name = cups

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

[print$]
    comment = Printer Drivers
    path = /etc/samba/drivers
    browseable = yes
    guest ok = no
    read only = yes
    write list = root
```



(IX.) SAMBA and CUPS

For licensing reasons the necessary files of the Adobe Postscript driver can not be distributed with either Samba or CUPS. You need to download them yourself from the Adobe website. Once extracted, create a `drivers` directory in the CUPS data directory (usually `/usr/share/cups/`). Copy the Adobe files using UPPERCASE filenames, to this directory as follows:

```
ADFONT.S.MFM
ADOBEPS4.DRV
ADOBEPS4.HLP
ADOBEPS5.DLL
ADOBEPSU.DLL
ADOBEPSU.HLP
DEFPRT2.PPD
ICONLIB.DLL
```

Users of the ESP Print Pro software are able to install their "Samba Drivers" package for this purpose with no problem.

6. Sources of CUPS drivers / PPDs

On the internet you can find now many thousand CUPS-PPD files (with their companion filters), in many national languages, supporting more than 1.000 non-PostScript models.

- **ESP PrintPro** (<http://www.easysw.com/printpro/>) (commercial, non-Free) is packaged with more than 3.000 PPDs, ready for successful usage "out of the box" on Linux, IBM-AIX, HP-UX, Sun-Solaris, SGI-IRIX, Compaq Tru64, Digital Unix and some more commercial Unices (it is written by the CUPS developers themselves and its sales help finance the further development of CUPS, as they feed their creators)
- the **Gimp-Print-Project** (<http://gimp-print.sourceforge.net/>) (GPL, Free Software) provides around 120 PPDs (supporting nearly 300 printers, many driven to photo quality output), to be used alongside the Gimp-Print CUPS filters;
- **TurboPrint** (<http://www.turboprint.com/>) (Shareware, non-Free) supports roughly the same amount of printers in excellent quality;
- **OMNI** (<http://www-124.ibm.com/developerworks/oss/linux/projects/omni/>) (LPGL, Free) is a package made by IBM, now containing support for more than 400 printers, stemming from the inheritance of IBM OS/2 KnowHow ported over to Linux (CUPS support is in a Beta-stage at present);
- **HPIJS** (<http://hpinkjet.sourceforge.net/>) (BSD-style licences, Free) supports around 120 of HP's own printers and is also providing excellent print quality now;
- **Foomatic/cupsomatic** (<http://www.linuxprinting.org/>) (LPGL, Free) from Linuxprinting.org are providing PPDs for practically every Ghostscript filter known to the world, now usable with CUPS.

NOTE: the cupsomatic trick from Linuxprinting.org is working different from the other drivers. While the other drivers take the generic CUPS raster (produced by CUPS' own pstoraster PostScript RIP) as their input, cupsomatic "kidnaps" the PostScript inside CUPS, before RIP-ping, deviates it to an external Ghostscript installation (which now becomes the RIP) and gives it back to a CUPS backend once Ghostscript is finished. -- CUPS versions from 1.1.15 and later will provide their pstoraster PostScript RIP function again inside a system-wide Ghostscript installation rather than in "their own" pstoraster filter. (This CUPS-enabling Ghostscript version may be installed either as a



(IX.) SAMBA and CUPS

patch to GNU or AFPL Ghostscript, or as a complete ESP Ghostscript package). However, this will not change the cupsomatic approach of guiding the printjob along a different path through the filtering system than the standard CUPS way...

Once you installed a printer inside CUPS with one of the recommended methods (the lpadmin command, the web browser interface or one of the available GUI wizards), you can use **cupsaddsmb** to share the printer via Samba. **cupsaddsmb** prepares the driver files for comfortable client download and installation upon their first contact with this printer share.

7. cupsaddsmb

The **cupsaddsmb** command copies the needed files for convenient Windows client installations from the previously prepared CUPS data directory to your [print\$] share. Additionally, the PPD associated with this printer is copied from /etc/cups/ppd/ to [print\$].

```
root# cupsaddsmb -U root infotec_IS2027
Password for root required to access localhost via SAMBA:
[type in password 'secret']
```

To share all printers and drivers, use the **-a** parameter instead of a printer name.

Probably you want to see what's going on. Use the **-v** parameter to get a more verbose output:

```
root# cupsaddsmb -v -U root infotec_IS2027
Password for root required to access localhost via SAMBA:
Running command: smbclient //localhost/print/$ -N -U'root%secret' \
-c 'mkdir W32X86;put
/var/spool/cups/tmp/3cd1cc66376c0
W32X86/infotec_IS2027.PPD;put
/usr/share/cups/drivers/ADOBEPS5.DLL
W32X86/ADOBEPS5.DLL;put
/usr/share/cups/drivers/ADOBEPSU.DLL
W32X86/ADOBEPSU.DLL;put
/usr/share/cups/drivers/ADOBEPSU.HLP
W32X86/ADOBEPSU.HLP'
added interface ip=10.160.16.45 bcast=10.160.31.255 nmask=255.255.240.0
added interface ip=192.168.182.1 bcast=192.168.182.255 nmask=255.255.255.0
added interface ip=172.16.200.1 bcast=172.16.200.255 nmask=255.255.255.0
Domain=[TUX-NET] OS=[Unix] Server=[Samba 2.2.3a.200204262025cvs]
NT_STATUS_OBJECT_NAME_COLLISION making remote directory \W32X86
putting file /var/spool/cups/tmp/3cd1cc66376c0 as \
\W32X86/infotec_IS2027.PPD (17394.6 kb/s) \
(average 17395.2 kb/s)
putting file /usr/share/cups/drivers/ADOBEPS5.DLL as \
\W32X86/ADOBEPS5.DLL (10877.4 kb/s) \
(average 11343.0 kb/s)
putting file /usr/share/cups/drivers/ADOBEPSU.DLL as \
\W32X86/ADOBEPSU.DLL (5095.2 kb/s) \
(average 9260.4 kb/s)
putting file /usr/share/cups/drivers/ADOBEPSU.HLP as \
W32X86/ADOBEPSU.HLP (8828.7 kb/s) \
(average 9247.1 kb/s)

Running command: smbclient //localhost/print/$ -N -U'root%secret' \
-c 'mkdir WIN40;put
/var/spool/cups/tmp/3cd1cc66376c0
WIN40/infotec_IS2027.PPD;put
```



(IX.) SAMBA and CUPS

```

        /usr/share/cups/drivers/ADFFONTS.MFM \
        WIN40/ADFFONTS.MFM;put \
        /usr/share/cups/drivers/ADOBEPS4.DRV \
        WIN40/ADOBEPS4.DRV;put \
        /usr/share/cups/drivers/ADOBEPS4.HLP \
        WIN40/ADOBEPS4.HLP;put \
        /usr/share/cups/drivers/DEFPRTR2.PPD \
        WIN40/DEFPRTR2.PPD;put \
        /usr/share/cups/drivers/ICONLIB.DLL \
        WIN40/ICONLIB.DLL;put \
        /usr/share/cups/drivers/PSMON.DLL \
        WIN40/PSMON.DLL;'
added interface ip=10.160.16.45 bcast=10.160.31.255 nmask=255.255.240.0
added interface ip=192.168.182.1 bcast=192.168.182.255 nmask=255.255.255.0
added interface ip=172.16.200.1 bcast=172.16.200.255 nmask=255.255.255.0
Domain=[TUX-NET] OS=[Unix] Server=[Samba 2.2.3a.200204262025cvs]
NT_STATUS_OBJECT_NAME_COLLISION making remote directory \WIN40
putting file /var/spool/cups/tmp/3cd1cc66376c0 as \
        WIN40/infotec_IS2027.PPD (26091.5 kb/s) \
        (average 26092.8 kb/s)
putting file /usr/share/cups/drivers/ADFFONTS.MFM as \
        WIN40/ADFFONTS.MFM (11241.6 kb/s) \
        (average 11812.9 kb/s)
putting file /usr/share/cups/drivers/ADOBEPS4.DRV as \
        WIN40/ADOBEPS4.DRV (16640.6 kb/s) \
        (average 14679.3 kb/s)
putting file /usr/share/cups/drivers/ADOBEPS4.HLP as \
        WIN40/ADOBEPS4.HLP (11285.6 kb/s) \
        (average 14281.5 kb/s)
putting file /usr/share/cups/drivers/DEFPRTR2.PPD as \
        WIN40/DEFPRTR2.PPD (823.5 kb/s) \
        (average 12944.0 kb/s)
putting file /usr/share/cups/drivers/ICONLIB.DLL as \
        WIN40/ICONLIB.DLL (19226.2 kb/s) \
        (average 13169.7 kb/s)
putting file /usr/share/cups/drivers/PSMON.DLL as \
        WIN40/PSMON.DLL (18666.1 kb/s) \
        (average 13266.7 kb/s)

Running command: rpcclient localhost -N -U'root%secret' \
        -c 'adddriver "Windows NT x86" \
        "infotec_IS2027:ADOBEPS5.DLL:infotec_IS2027.PPD: \
        ADOBEPSU.DLL:ADOBEPSU.HLP:NULL:RAW:NULL"' \
cmd = adddriver "Windows NT x86" "infotec_IS2027:ADOBEPS5.DLL: \
        infotec_IS2027.PPD:ADOBEPSU.DLL:ADOBEPSU.HLP:NULL:RAW:NULL"
Printer Driver infotec_IS2027 successfully installed.

Running command: rpcclient localhost -N -U'root%secret' \
        -c 'adddriver "Windows 4.0" \
        "infotec_IS2027:ADOBEPS4.DRV:infotec_IS2027.PPD: \
        NULL:ADOBEPS4.HLP: \
        :PSMON.DLL:RAW:ADFFONTS.MFM,DEFPRTR2.PPD,ICONLIB.DLL"' \
cmd = adddriver "Windows 4.0" "infotec_IS2027:ADOBEPS4.DRV: \
        infotec_IS2027.PPD:NULL:ADOBEPS4.HLP:PSMON.DLL: \
        RAW:ADFFONTS.MFM,DEFPRTR2.PPD,ICONLIB.DLL"
Printer Driver infotec_IS2027 successfully installed.

Running command: rpcclient localhost -N -U'root%secret' \
        -c 'setdriver infotec_IS2027 infotec_IS2027' \
cmd = setdriver infotec_IS2027 infotec_IS2027
Sucessfully set infotec_IS2027 to driver infotec_IS2027.

root#
```



(IX.) SAMBA and CUPS

If you look closely, you'll discover your root password was transferred unencrypted over the wire, so beware! Also, if you look further her, you'll discover error messages like `NT_STATUS_OBJECT_NAME_COLLISION` in between. They occur, because the directories `WIN40` and `W32X86` already existed in the `[print$]` driver download share (from a previous driver installation). They are harmless here.

Now your printer is prepared for the clients to use. From a client, browse to the CUPS/Samba server, open the "Printers" share, right-click on this printer and select "Install.." or "Connect.." (depending on the Windows version you use). Now there should be a new printer in your client's local "Printers" folder, named (in my case) "infotec_IS2027 on kde-bitshop"

NOTE: **cupsaddsmb** will only reliably work with CUPS version 1.1.15 or higher and Samba from 2.2.4. If it doesn't work, or if the automatic printer driver download to the clients doesn't succeed, you can still manually install the CUPS printer PPD on top of the Adobe PostScript driver on clients and then point the client's printer queue to the Samba printer share for connection, should you desire to use the CUPS networked PostScript RIP functions.



About the Author:

Kurt Pfeifle works in Stuttgart as a System Specialist for Danka Deutschland GmbH, one of the largest manufacturer-independent providers for sales and service of system solutions in the digital printing market, integrating hardware as well as software components, covering heterogeneous network environments. He may be contacted via kpfeifle.at.danka.de.

Links:

<http://www.danka.de/printpro/faq.html>

CUPS-FAQ by Kurt Pfeifle, more than 150 Questions (and presently a bit less Answers ;-)

<http://www.pwg.org/ipp/>

First hand infos about Internet Printing Protocol

<http://www.cups.org/software.html>

CUPS download

<http://www.cups.org/documentation.html>

First hand infos about CUPS

<http://www.easysw.com/software.html>

ESP PrintPro download

<http://www.easysw.com/documentation.html>

First hand infos about ESP PrintPro

<http://www.linuxprinting.org/>

Grant Taylor and Till Kämpeter: Linux Printing HOWTO, Linux Printing Database, PPD-O-Matic and PDQ-O-Matic and much more

<http://www.pwg.org/ipp/IPP-Products.html>

List of IPP-aware products at the Printer Working Group website

<http://us1.samba.org/samba/docs/>

One of the many mirrors where you can access Samba documentation

Annotation: this paper is based on a chapter Kurt wrote for the "Samba-HOWTO-Collection", shipped as part of the current Samba 2.2.x sources. The Samba-HOWTO-Collection is also online available at all mirrors of the Samba Project, f.e. at <http://us1.samba.org/samba/docs/Samba-HOWTO-Collection.html>.

