

ESUP-Portail printing system installation guide

Date: 2005-06-29

Version: 0.3

Status: draft (this document should be validated by an install-party at University of Rennes 1 on June 29th).

Author: Pascal Aubry <pascal.aubry@univ-rennes1.fr>

Contributors:

Katy Santerre <katy.santerre@univ-rennes1.fr> (Samba and Windows clients).

Ambroise Diascorn <ambroise.diascorn@univ-rennes1.fr> (Linux FC3 clients).

Add-ons for release 0.4 (planned on 2005-07-04):

Installing server certificates for safe remote web administration of printers (Pascal Aubry).

Configuring Solaris 9 clients (Stéphane Ayello).

Looking for contributors:

Configuring Windows 2000 clients

Configuring Solaris 9/10 clients

Add-ons planned for future releases:

Installing and configuring `esup-print-channel` (Pascal Aubry)

The last release of this document can be found at <http://perso.univ-rennes1.fr/pascal.aubry/doc/printing>.

Discussions on the installation, the configuration and the usage of `esup-print-system` should take place in the (French-speaking) mailing list print-utilisateurs@esup-portail.org.

To join the list, go to <http://listes.esup-portail.org/sympa/info/print-utilisateurs>.

Summary

1	INTRODUCTION.....	4
1.1	Presentation.....	4
1.2	Architecture.....	4
2	LICENSE & COPYRIGHT.....	5
2.1	esup-cups-backend & esup-print-channel.....	5
2.2	Documentation.....	5
3	ESUP-PRINT-SERVER.....	5
3.1	Java.....	5
3.1.1	<i>Installing the JVM.....</i>	<i>5</i>
3.1.2	<i>Installing Apache ant.....</i>	<i>5</i>
3.1.3	<i>Creating /etc/profile.d/java.sh.....</i>	<i>6</i>
3.2	CUPS.....	6
3.2.1	<i>Installing CUPS.....</i>	<i>6</i>
3.2.2	<i>/etc/cups/cupsd.conf.....</i>	<i>6</i>
3.2.3	<i>/etc/pam.d/cups.....</i>	<i>7</i>
3.2.4	<i>/etc/ldap.conf.....</i>	<i>8</i>
3.2.5	<i>Adding printer drivers to CUPS.....</i>	<i>8</i>
3.2.6	<i>Troubleshooting.....</i>	<i>8</i>
3.3	Samba.....	9
3.3.1	<i>Installing Samba.....</i>	<i>9</i>
3.3.2	<i>/etc/smb.conf.....</i>	<i>9</i>
3.3.3	<i>/etc/pam.d/samba.....</i>	<i>10</i>
3.4	Adding a printer.....	10
3.4.1	<i>Creating a printer in CUPS.....</i>	<i>10</i>
3.4.2	<i>Adding a CUPS printer to Samba.....</i>	<i>14</i>
3.5	Configuring Unix clients.....	16
3.5.1	<i>Linux.....</i>	<i>16</i>
3.5.2	<i>Solaris (to be completed).....</i>	<i>18</i>
3.6	Configuring Windows clients.....	18
3.6.1	<i>Windows XP.....</i>	<i>18</i>
3.6.2	<i>Windows 2000 (to be completed).....</i>	<i>21</i>
3.7	Security concerns (to be completed).....	22
3.7.1	<i>CUPS requests encryption.....</i>	<i>22</i>
3.8	esup-cups-backend.....	22
3.8.1	<i>Installing the sources.....</i>	<i>22</i>
3.8.2	<i>Configuring the behaviour of the backend.....</i>	<i>22</i>
3.8.3	<i>Deploying the backend.....</i>	<i>22</i>
3.8.4	<i>Using the backend.....</i>	<i>22</i>
4	ESUP-PRINT-CHANNEL (TO BE COMPLETED).....	23
5	REFERENCES.....	23

6 CHANGELOG 23

1 Introduction

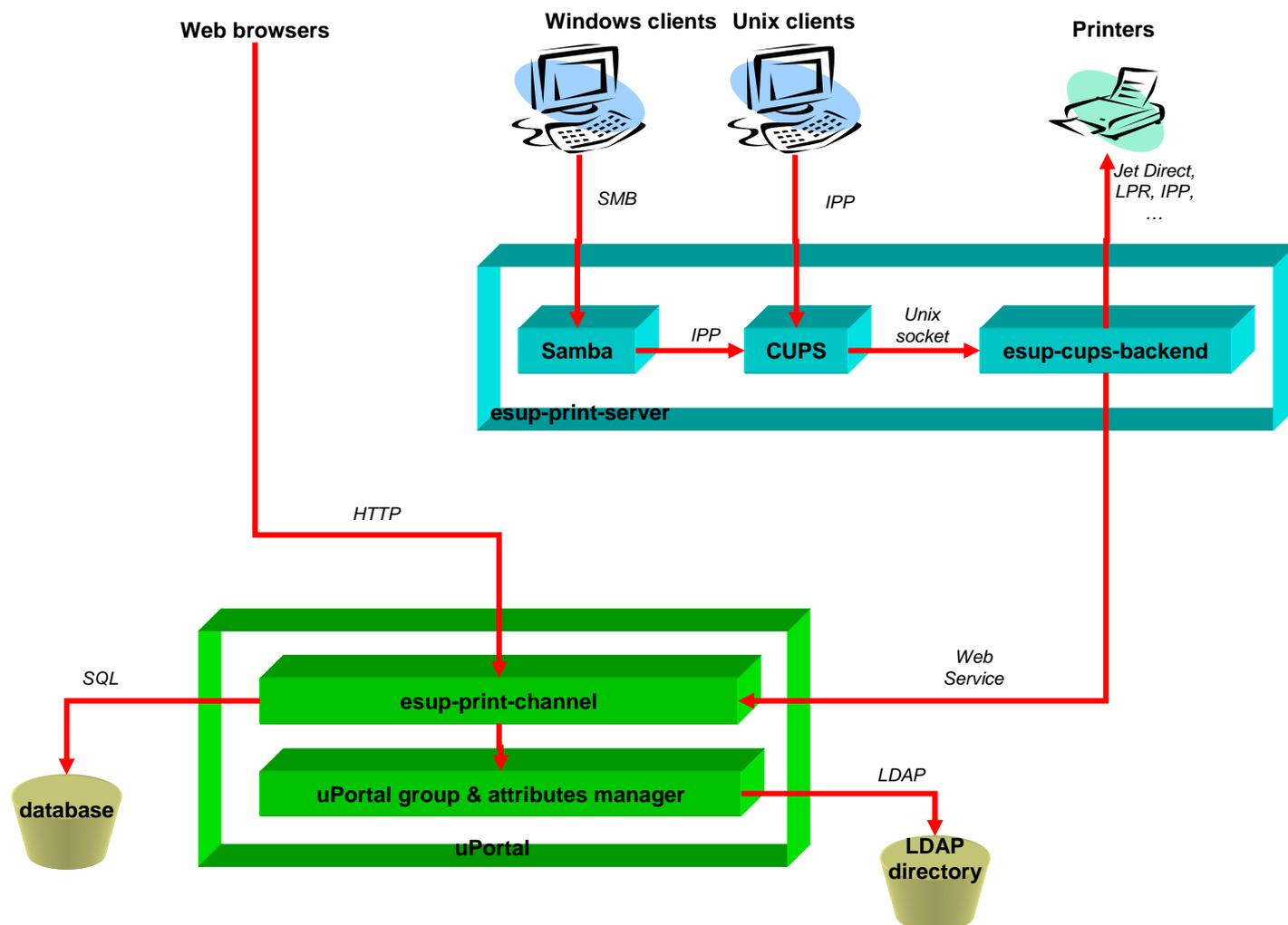
This document shows how to install and configure the printing system distributed by project ESUP-Portail.

1.1 Presentation

ESUP-Portail [1] provides a fully-integrated solution for Universities to control printing on their network (access control and accounting). Based on Java standards (J2EE [2], database abstraction [7]), standard protocols (Web Services [4], IPP [5], SMB [6], ...) and open-source components, it is made up of two parts: **Spoolers**, located in each department, do the printing job. A spooler is based on CUPS (serving Unix clients), Samba (serving Windows clients) and esup-cups-backend (allowing/denying and logging jobs). A centralized **controller** made of a uPortal channel and web services. The channel is used to manage printing quotas and authorizations, while web services are used by the spoolers.

1.2 Architecture

The figure below shows the complete architecture of ESUP-Portail printing system.



The architecture presented here reflects the way ESUP-Portail printing system was set up at University of Rennes 1, where it was developed and installed for the first time. Everything should naturally be adapted to your local configuration, especially for the authentication policy of clients.

2 License & copyright

2.1 *esup-cups-backend & esup-print-channel*

`esup-print-system` is Copyright © 2004-2005 ESUP-Portail consortium.

ESUP-Portail is a French academic project developed under the GPL (General Public License) augmented according to the following:

A binary or source file developed by ESUP-Portail can be used or compiled with products under Apache license.

The official English text of the GPL can be found here: <http://www.gnu.org/licenses/gpl.html>.

A non official French translation can be found here: <http://www.linux-France.org/article/these/gpl.html>.

The different kinds of licenses governing the products developed by the Apache foundation can be found here: <http://www.apache.org/licenses>.

It follows that you can as well as use download contents as well modify and redistribute them provided you respect the GPL terms.

Downloading and using such contents do not provide any guaranty.

Be sure that you have well understood the terms of the license before using the contents it covers.

2.2 *Documentation*

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts and with no Back-Cover Texts. A copy of the license is available at <http://www.gnu.org/copyleft/fdl.html>.

3 `esup-print-server`

`esup-print-server` is made up of CUPS, Samba and `esup-cups-backend` (ESUP-Portail contribution).

3.1 *Java*

3.1.1 *Installing the JVM*

A JVM is needed to run `esup-cups-backend`, and a JDK to compile it. As the current release is not compliant with JDK 1.5, JDK 1.4 must be installed.

`Esup-print-server` has been validated with JDK 1.4.2_06, but you should get the latest stable 1.4 release to benefit from the last JDK improvements.

Download an RPM of JKD 1.4.2_06 for FC3 from <http://java.sun.com> into `/usr/local/src` and install:

```
cd /usr/local
./src/j2sdk-1_4_2_06-linux-i586.bin
```

This installs the JDK in `/usr/local/j2sdk-1_4_2_06`.

3.1.2 *Installing Apache ant*

Apache ant is needed to build and install `esup-cups-backend`.

Esup-print-server has been validated with ant 1.6.5, but you should get the latest stable release to benefit from the last ant improvements.

Download the latest release from <http://ant.apache.org> to /usr/local/src and install:

```
cd /usr/local
tar zxf src/apache-ant-1.6.5-bin.tar.gz
```

This installs ant in /usr/local/apache-ant-1.6.5.

3.1.3 Creating /etc/profile.d/java.sh

This file, automatically loaded by *bash*, should contain the following lines:

```
export JAVA_HOME=/usr/local/src/j2sdk1.4.2_06
export ANT_HOME=/usr/local/apache-ant-1.6.5
export PATH=$PATH:$JAVA_HOME/bin:$ANT_HOME/bin
```

This file will be sourced by esup-cups-backend.

3.2 CUPS

esup-print-server has been validated with CUPS 1.1.23, but you should get the latest stable release to benefit from the last CUPS improvements.

Note: the sources of CUPS are needed by esup-cups-backend, so you should not use binaries.

3.2.1 Installing CUPS

Get latest CUPS sources from Easy Software web site:

```
cd /usr/local/src
wget http://ftp.easysw.com/ftp/pub/cups/1.1.23/cups-1.1.23-source.tar.gz
tar zxf cups-1.1.23-source.tar.gz
cd cups-1.1.23
```

To be sure that CUPS starts before Samba, execute the following commands:

```
chkconfig --level 235 cups on
chkconfig --level 0 cups off
```

Compile and intall:

```
./configure --enable-ssl
make
make install
```

Note: installing CUPS may remove existing printing solutions, such as LPD or lprNG.

3.2.2 /etc/cups/cupsd.conf

Default ports should not be changed. Simply add the following line to on your LAN interface (use your own IP address), on port 631 (IPP default port):

```
Listen 148.60.4.119
```

By default, all URLs should be accessible only by users declared in your LDAP directory. Make IPP authentication rely on PAM by default, except from localhost (no authentication needed):

```
<Location />
  Order Deny,Allow
  Allow from localhost
  Deny from all
  AuthType Basic
  AuthClass User
```

```
Satisfy any
</Location>
```

Note: there is no need to set specific control access on printers, since access control will be performed by esup-cups-backend.

Administration pages should be accessible only by root, from localhost:

```
<Location /admin>
  Order Deny,Allow
  Allow from all
  AuthType Basic
  AuthClass System
  SystemGroup root
</Location>
```

Announce all your printers:

```
Browsing On
BrowseProtocols cups
BrowseOrder Deny,Allow
BrowseAllow from all
```

Finally set i18n directives for your comfort, for instance:

```
DefaultCharset iso-8859-1
DefaultLanguage fr
```

Important: CUPS daemon must be restarted each time the configuration file is changed:

```
service cups restart
```

Important: admin URLs should be accessible only through an encrypted tunnel (HTTPS). This part of the documentation will be added in a future release.

3.2.3 /etc/pam.d/cups

As CUPS now relies on PAM for user authentication, tell PAM to rely on your LDAP directory and local users:

```
auth      required      /lib/security/$ISA/pam_env.so
auth      sufficient    /lib/security/$ISA/pam_unix.so likeauth nullok
auth      sufficient    /lib/security/$ISA/pam_ldap.so use_first_pass
auth      required      /lib/security/$ISA/pam_denial.so

account   required      /lib/security/$ISA/pam_unix.so broken_shadow
account   sufficient    /lib/security/$ISA/pam_succeed_if.so uid < 100 quiet
account   [default=bad success=ok user_unknown=ignore] \
                                                /lib/security/$ISA/pam_ldap.so
account   required      /lib/security/$ISA/pam_permit.so
```

Important: we show in this part how to set up PAM authentication for CUPS. We assume here that configured your server in such a way that LDAP users are not able to login to your server. For instance, /etc/pam.d/system-auth could look like:

```
auth      required      /lib/security/$ISA/pam_env.so
auth      sufficient    /lib/security/$ISA/pam_unix.so likeauth nullok
auth      required      /lib/security/$ISA/pam_denial.so

account   required      /lib/security/$ISA/pam_unix.so broken_shadow
account   sufficient    /lib/security/$ISA/pam_succeed_if.so uid < 100 quiet
account   required      /lib/security/$ISA/pam_permit.so

password  requisite      /lib/security/$ISA/pam_cracklib.so retry=3
password  sufficient    /lib/security/$ISA/pam_unix.so \
                                                nullok use_authok md5 shadow
```

```
password    required    /lib/security/$ISA/pam_deny.so
```

```
session     required    /lib/security/$ISA/pam_limits.so
```

```
session     required    /lib/security/$ISA/pam_unix.so
```

Note: `/etc/nsswitch.conf` should contain something like `passwd: files ldap`.

At this step, you should be able to add a printer, but LDAP users will not be able to print until you configure `pam_lpad` (see below).

3.2.4 /etc/ldap.conf

The configuration of `pam_ldap` is not fully described here; as an example, at University of Rennes 1, `/etc/ldap.conf` looks like:

```
host ldapglobal.univ-rennes1.fr
base ou=people,dc=univ-rennes1,dc=fr
ssl no
pam_password md5
```

You should be able at this step to add a printer and print from Unix clients.

3.2.5 Adding printer drivers to CUPS

Drivers are provided by CUPS for common printers (see `/usr/share/cups/model/*.ppd`). This part explains how you can add other drivers.

Remarks:

It is not necessary to add printers of your clients already pre-format jobs, since the printers model to use by the server should be “Raw” in this case.

To be sure to have the drivers exactly corresponding to your printers (and thus ease your choice when adding a printer), all the printers provided by CUPS can be deleted prior to installing yours:

```
cd /usr/share/cups/model
mkdir /tmp/models.org
mv * /tmp/models.org/
```

For instance, let us add the driver of an HP 8150.

Download the driver from CUPS web site (<http://www.cups.org/ppd.php>):

```
cd /usr/local/src
mkdir ppd
cd ppd
wget http://www.cups.org/ppd/hp/en/hpb81507.ppd.gz
```

Unzip and install into folder `/usr/share/cups/model`:

```
gunzip hpb81507.ppd.gz
cp hpb81507.ppd /usr/share/cups/model/
```

Restart CUPS daemon:

```
Service cups restart
```

3.2.6 Troubleshooting

CAS logs are in `/var/log/cups`, log level can be increased for instance by adding the following line to `/etc/cups/cupsd.conf` (`debug2` can also be used for more verbosity):

```
LogLevel debug
```

PAM logs can generally be found in `/var/log/secure` or `/var/log/message`, and its verbosity can be increased by adding parameter `debug` to `pam_ldap` in `/etc/pam.d/cups`:

```
auth    sufficient    /lib/security/$ISA/pam_ldap.so use_first_pass debug
```

```
account      [default=bad success=ok user_unknown=ignore] \  
              /lib/security/$ISA/pam_ldap.so debug
```

3.3 Samba

Samba needs to be installed only if you have Windows clients. CUPS alone is sufficient to print from Unix clients. Samba should be installed on the server where you installed CUPS (you could install it on another server but access control will be much more complex, and you better not bother me with this... ;-).

Esup-print-server has been validated with Samba 3.0.8, but you should get the latest stable release to benefit from the last Samba improvements.

3.3.1 Installing Samba

As no development is needed around Samba, you can install binaries or, the easiest way, RPM packages `samba`, `samba-common` and `samba-client`.

3.3.2 /etc/smb.conf

[global]

Choose a significant name for your server's workgroup:

```
workgroup = CUPS
```

Choose the string your server will be announced to clients:

```
server string = Samba Server for CUPS
```

Set up a stand-alone server:

```
security = user
```

Allow the clients of your network, as well as your server:

```
hosts allow = 148.60. 127.
```

Tell samba to act as a print server, relying on CUPS (`/etc/printcap` will not be used, and the jobs will be passed as-is to CUPS):

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

Choose the verbosity and the location of the logs:

```
log level = 5  
log file = /var/log/samba/%m.log  
max log size = 50
```

Choose not to encrypt passwords to be able to rely on PAM:

```
encrypt passwords = no
```

Delegate authentication to PAM:

```
obey pam restrictions = yes
```

This should be added for better performance:

```
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
```

Tell your server to resolve clients by relying on DNS (`/etc/nsswitch.conf` should contain something like

```
host: files dns)  
name resolve order = host
```

[printers]

Share all your printers in the [printers] section.

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

This way, all CUPS printers will be automatically available to your clients.

[pGina]

We finally add a share that will be used by clients to authenticate just after pGina authentication (see further).

```
comment = used by pGina
path = /tmp/pGina
read only = no
public = yes
```

Do not forget to create the share:

```
mkdir /tmp/pGina
```

Important: Samba daemon must be restarted each time the configuration file is changed:

```
service samba restart
```

3.3.3 /etc/pam.d/samba

As Samba now relies on PAM for user authentication, tell PAM to rely on your LDAP directory and local users by copying `/etc/pam.d/cups` to `/etc/pam.d/samba`.

At this step, you should be able to share your printers to your Windows clients.

3.4 Adding a printer

We explain here how printers can be added to CUPS with an example. We stand here that the printer is a LaserJet HP 8150 DN (network-attached), Postscript3 and PCL6 capable. The configuration should be adapted for other printers, but should not differ much from what is presented here.

Two ways are possible to add a printer:

- Use the web interface

- Edit `/etc/cups/printers.conf`

We recommend newbie users to use the first method, described below. Advanced users can easily edit the configuration file, at their own risks.

In any case, never edit `/etc/printcap`, as this file will be overwritten by CUPS.

3.4.1 Creating a printer in CUPS

Launch a web browser on your print server and access <http://127.0.0.1:631>:

ESP Administration Classes Help Jobs Printers Software

[Do Administration Tasks](#)

[Manage Printer Classes](#)

[On-Line Help](#)

[Manage Jobs](#)

[Manage Printers](#)

[Download the Current CUPS Software](#)

Choose “*Manage Printers*”:

Printer

No printers

Add Printer

Choose “*Add Printer*”:

Admin

Add New Printer

Name: lwa045

Location: Admin IFSIC

Description: reserved to system administrators

Continue

Name will be the logical name of the queue, while *Location* and *Description* are only informal fields. Fill-in the form and choose “*Continue*”:

Admin

Device for lwa045

Device: AppSocket/HP JetDirect

Continue

Choose the *Device* (in our case “AppSocket/HP JetDirect”) and “*Continue*”:

Admin

Device URI for lwa045

Device URI: socket:lwa045.ifsic.univ-rennes1.fr:9100

Examples:

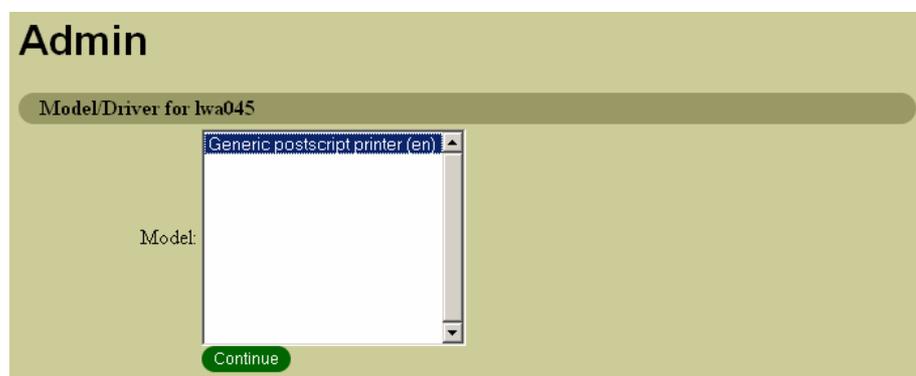
```
file:/path/to/filename.prn
http://hostname:631/ipp/
http://hostname:631/ipp/port1
ipp://hostname/ipp/
ipp://hostname/ipp/port1
lpd://hostname/queue
socket://hostname
socket://hostname:9100
```

Continue

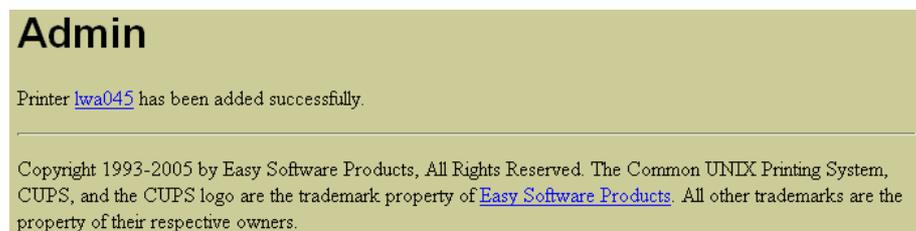
Set the *Device URI*; in our example socket : (indicating the device), lwa045.ifsic.univ-rennes1.fr (the FQDN of the printer, or its IP address), and : 9100 (telling the port the printer listens to). Then choose “Continue”:



As we expect our printer to be used by Unix clients, we set it as a “*Postscript*” printer. If you only have Windows clients, you will probably choose “*Raw*”. Select the driver of the printer and choose “*Continue*”:



Finally select the driver to use (in our case, only one is proposed) and choose “*Continue*”:

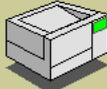


Click on the link to check the properties of your printer.

Iwa045

Default Destination: [lwa045](#)

[lwa045](#) Generic postscript printer



Description: reserved to system administrators
Location: Admin IFSIC
Printer State: idle, accepting jobs.
Device URI: socket://lwa045.ifsic.univ-rennes1.fr:9100

[Print Test Page](#) [Stop Printer](#) [Reject Jobs](#) [Modify Printer](#) [Configure Printer](#)
[Delete Printer](#) [Set As Default](#)

[Add Printer](#)

No Active Jobs

[Show Completed Jobs](#)

You can change these properties at any time by choosing “*Modify Printer*”.

Print a test page by choosing “*Print Test Page*”:

Iwa045

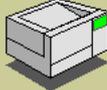
Test page sent, job ID is [lwa045-23](#).

Follow the link to come back to the printer’s page:

Iwa045

Default Destination: [lwa045](#)

[lwa045](#) Generic postscript printer



Description: reserved to system administrators
Location: Admin IFSIC
Printer State: idle, accepting jobs.
“*Received 75 bytes of back-channel data!*”
Device URI: socket://lwa045.ifsic.univ-rennes1.fr:9100

[Print Test Page](#) [Stop Printer](#) [Reject Jobs](#) [Modify Printer](#) [Configure Printer](#)
[Delete Printer](#) [Set As Default](#)

[Add Printer](#)

No Active Jobs

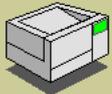
[Show Completed Jobs](#)

Choose “*Show Completed Jobs*” and you will normally see that your job has been completed. You can also run at your printer and get the wonderful page CUPS did for you, and drink a beer since this is the first page you ever printed with CUPS.

Iwa045

Default Destination: [Iwa045](#)

[Iwa045](#) Generic postscript printer



Description: reserved to system administrators
Location: Admin IFSIC
Printer State: idle, accepting jobs.
"Received 75 bytes of back-channel data!"
Device URI: socket://Iwa045.ifsic.univ-rennes1.fr:9100

Print Test Page Stop Printer Reject Jobs Modify Printer Configure Printer
Delete Printer Set As Default

Add Printer

ID	Name	User	Size	State	Control
Iwa045 -23	Test Page	paubry	15k	completed at Thu 16 Jun 2005 09:43:26 PM CEST	Restart Job

Show Active Jobs

Your printer is now properly set up for Unix clients.

Important: a physical printer should not be used by several CUPS queues for the accounting performed by esup-cups-backend to work correctly.

3.4.2 Adding a CUPS printer to Samba

Nothing has to be done: with the configuration shown above, Samba will automatically share all the printers defined in CUPS.

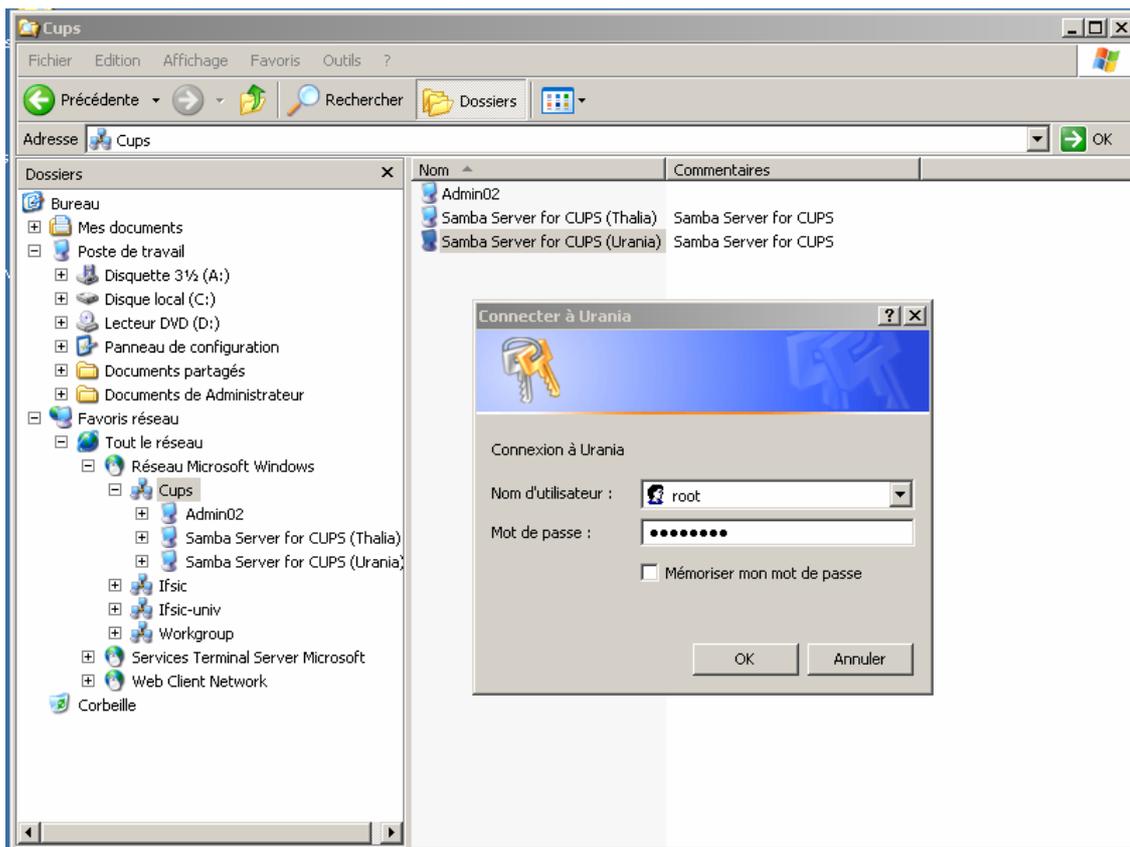
3.4.3 Share the printer's driver to Windows clients

Create folder `/etc/samba/printers` (that will be used to share the printers' drivers):
`mkdir /etc/samba/printers`

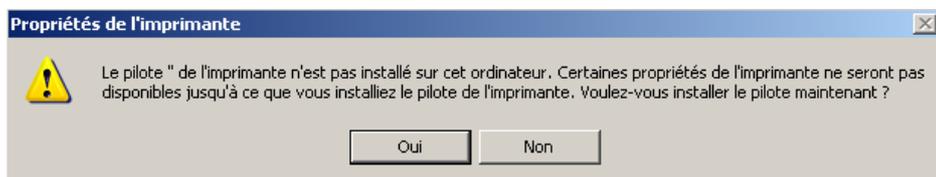
Then create subfolder `/etc/samba/printers/W32X86` (specifically used by Windows XP clients):
`mkdir /etc/samba/printers/W32X86`

Adding a Windows driver can be done locally on the server (using the `rpcclient` command) or from a Windows XP client. As the second is way is much easier, we show it here (people interested in the first way shall refer to <file:///usr/share/doc/samba-3.0.8/htmldocs/Samba-HOWTO-Collection/printing.html#inst-rpc>).

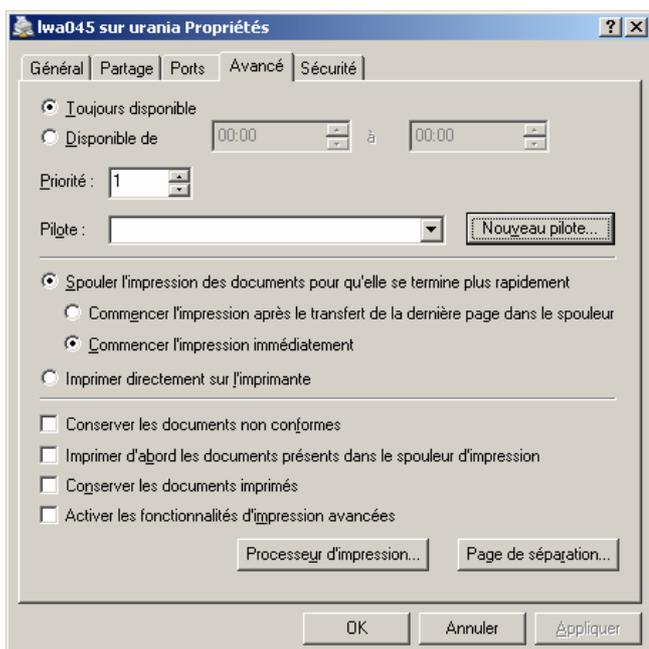
Connect on a Windows XP client using an account with administrator privileges (to be able to add a printer on the client), open the Network Neighbourhood and browse the Samba server using a root account (on the server):



Open the Printers folder, select the printer of which you want to install the driver, right-click and open the Properties Popup:



At this time, no driver is installed. **Choose not to install it now** (click on “No” on the error dialog).



Instead choose “*New driver*” (tab “*Advanced*”) and install the driver as you would do for a normal printer. The only difference is that files will be copied into the [print\$] share of the server instead of the local system.

That’s all: the printer’s driver will be automagically installed for all the Windows XP clients mounting this printer.

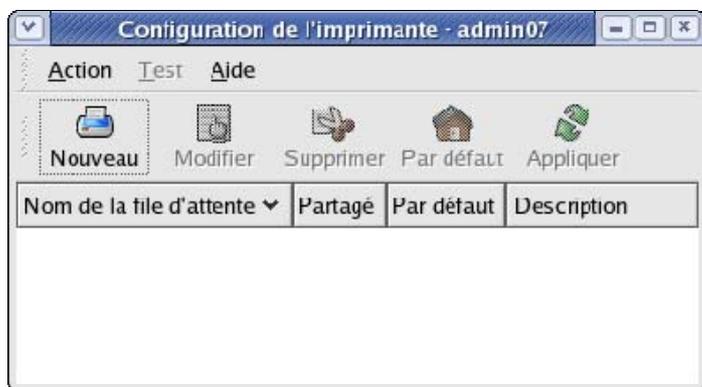
3.5 Configuring Unix clients

Unix clients should used IPP (they could also use SMB, but why getting into trouble when native protocols just work fine).

3.5.1 Linux

We show here how a printer can be used by a FC3 client. The configuration of any Linux client should be very close.

Launch `printconf` as root:



Choose “*Add*”:



Type the name of the CUPS queue you want to print on, as well as a short comment:



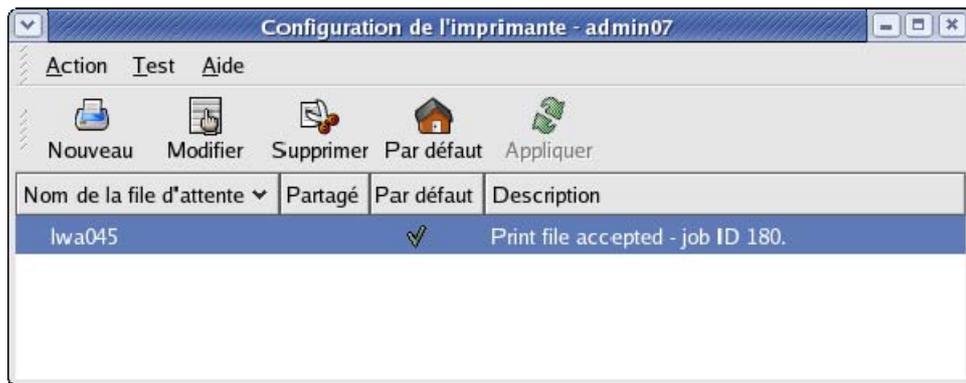
Choose an IPP queue, and fill-in the “Server” and “Path” inputs (there is no need to precise port 631):



Choose the model of the printer (Postscript):



Check and validate if everything is ok. You can then print a test page and run to the printer:



3.5.2 Solaris (to be completed)

Solaris 8 (to be completed)

To be completed!

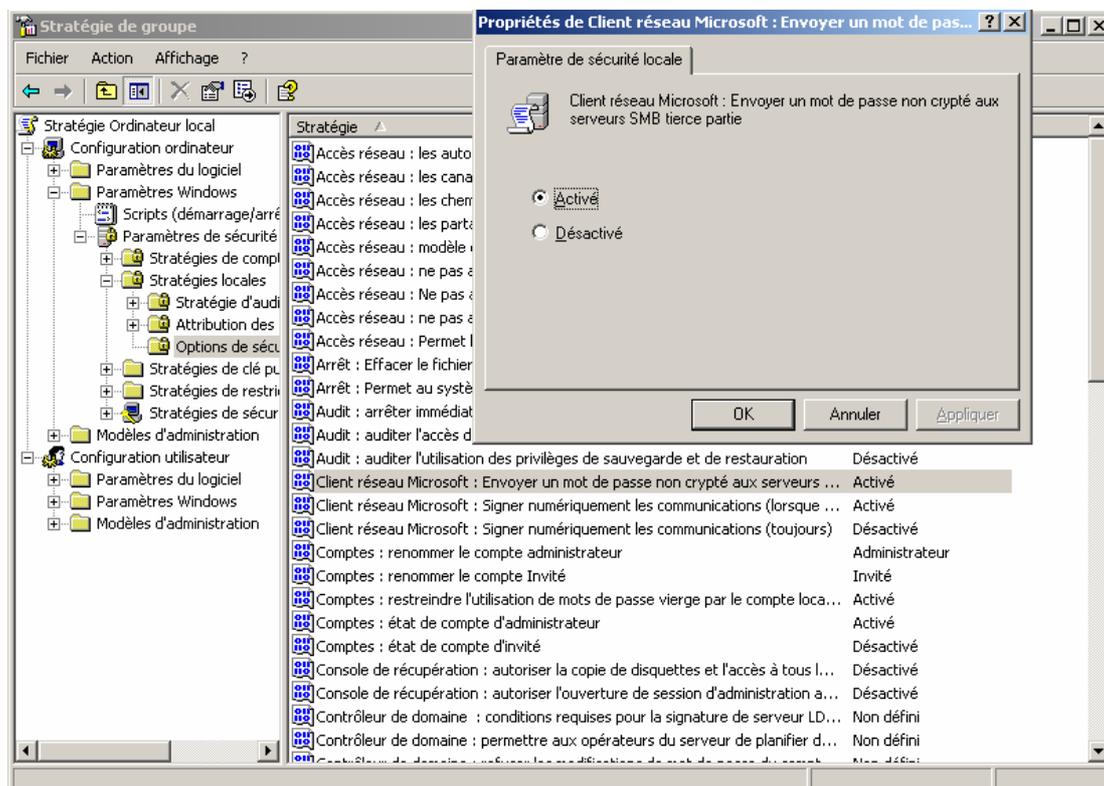
Solaris 9 (to be completed)

To be completed!

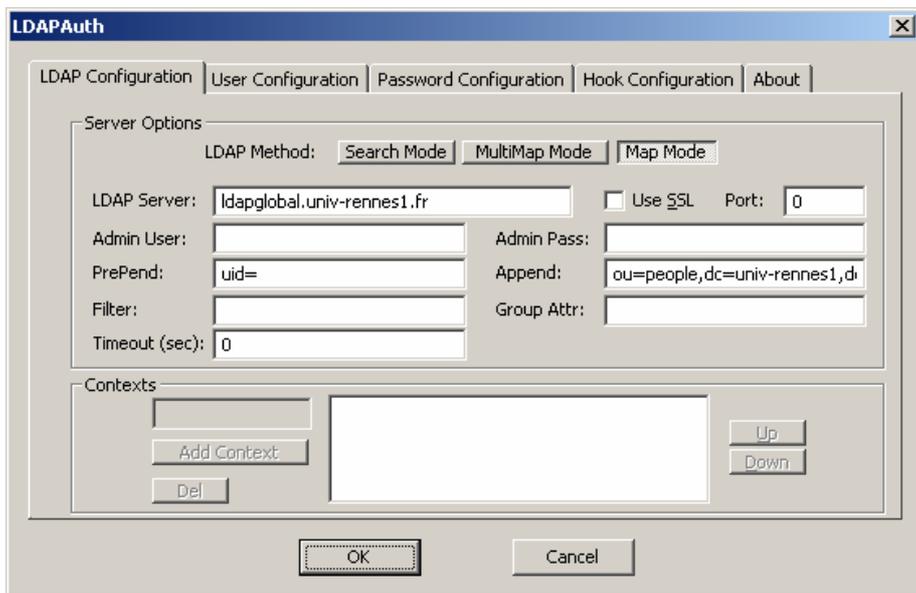
3.6 Configuring Windows clients

3.6.1 Windows XP

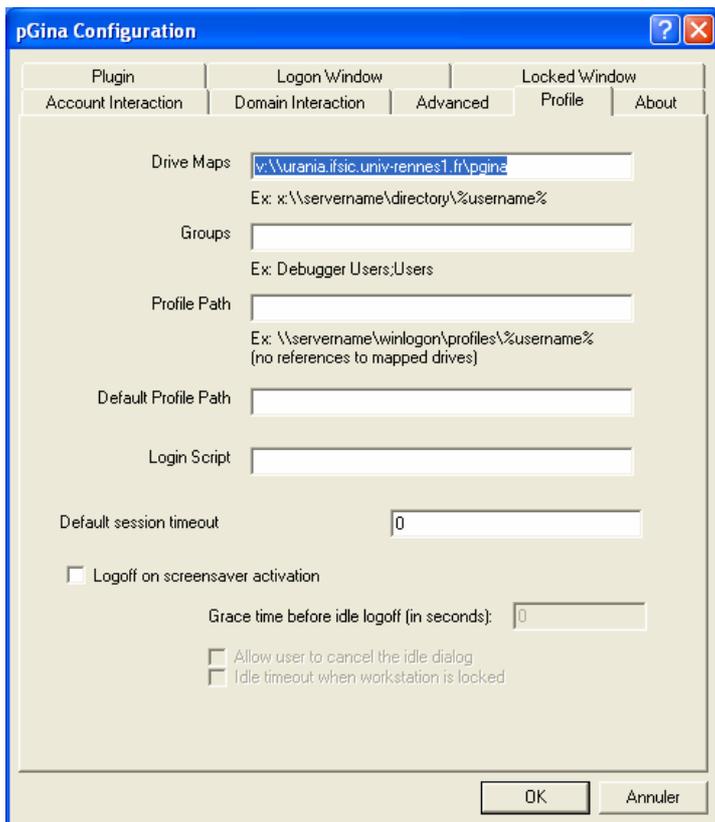
As Samba relies on PAM for authentication, tell Windows XP not to encrypt passwords when talking to Samba server. This can be done either by editing the registry (set HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation\parameters\enableplaintextpassword: : Reg_Dword to 1) or with help of the group strategy manager:



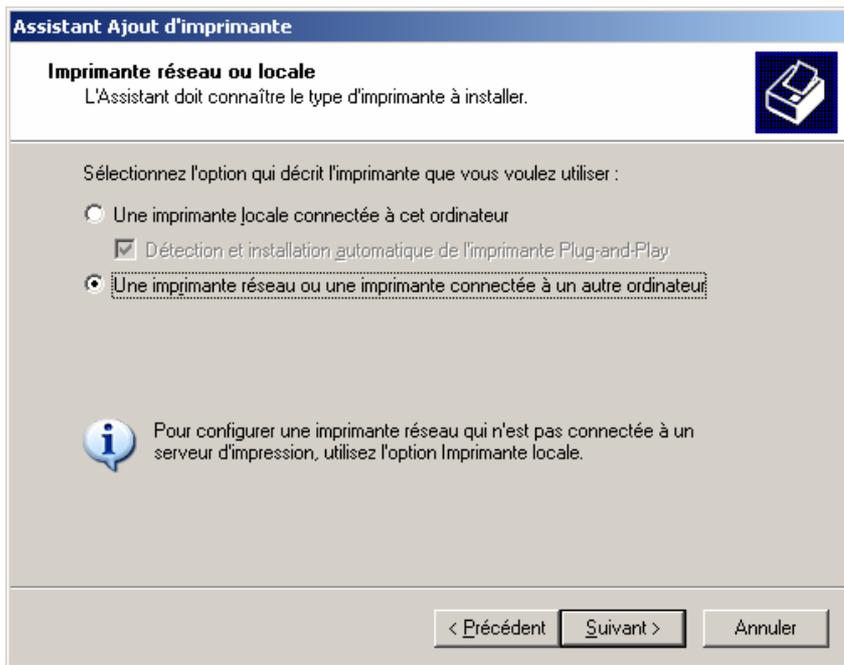
Replace Windows XP Gina by Pgina to handle session authentication. Download pGina from <http://pgina.xpasystems.com/>, as well as the LDAP plugin, and install. Configure pGina to rely on your LDAP directory:



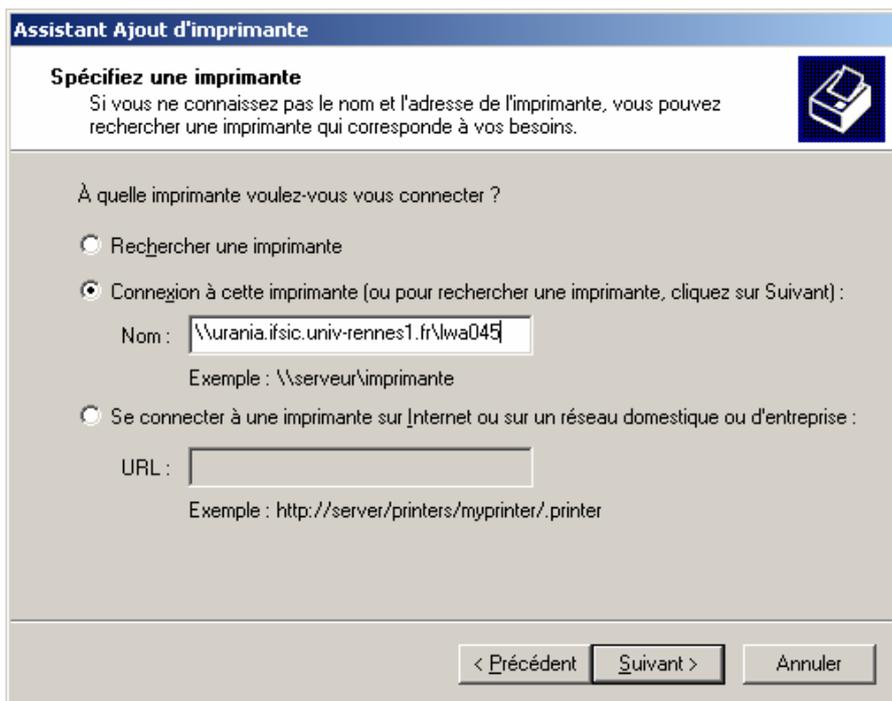
An IP channel must be set prior to using a printer for credentials to be automatically passed to the print server. This is achieved by mounting a dummy share (the pGina share we declared previously in Samba):



Windows XP must be restarted for your changes to take effect. Once the client has been restarted, printers can be added. Choose to install a network printer:



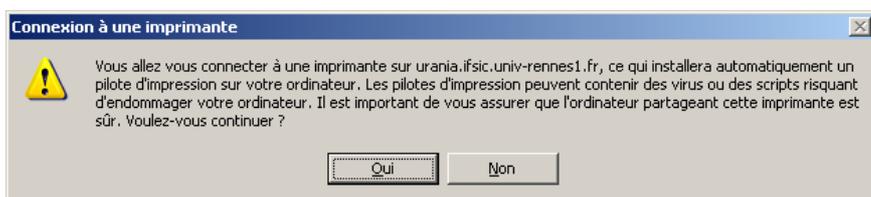
Then indicate the path of the printer:



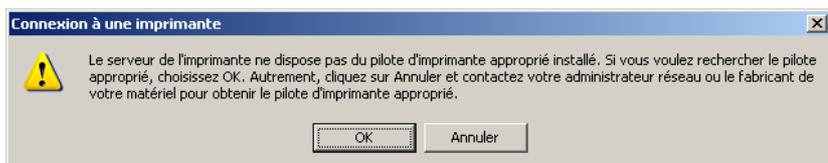
Use an LDAP account to connect for the first time (do not memorize the password):



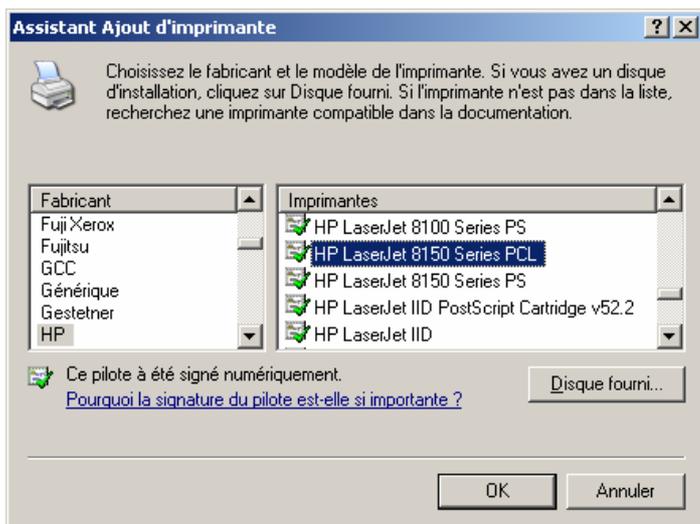
Accept the installation of the printer's driver:



Note: this release of the documentation does not show how to set up Samba to automatically send the printer's driver to the client when mounting the printer (next release will). Choose "ok" to install your driver:



Your driver should be provided with the documentation of the printer. Install it just as you would do for local printers:



That's all. Close your session, connect under another identity and check that you can print by sending a test page.

3.6.2 Windows 2000 (to be completed)

To be completed!

3.7 Security concerns (to be completed)

3.7.1 CUPS requests encryption

CUPS IPP requests should be performed through an encrypted tunnel for obvious security reasons. However, be sure everything works fine now before going on with security concerns.

To be completed!

3.8 esup-cups-backend

Once your clients work with “normal” printers (i.e. printers based on traditional backends, such as “socket”), it is time to use `esup-cups-backend` to gain powerful access control, accounting and logging features.

Note: as we write this part, the current release of `esup-cups-backend` does not handle access control, accounts jobs but only logs locally (future releases will log with the help of `esup-print-channel`, using web services).

3.8.1 Installing the sources

Download the latest release of `esup-cups-backend` and uncompress:

```
cd /usr/local/src
unzip -o esup-cups-backend-0.3.0.zip
```

This will create folder `esup-cups-backend-0.3.0/`.

3.8.2 Configuring the behaviour of the backend

CEsupCupsBackend.xml

The behaviour of the backend at runtime is set in file `properties/CEsupCupsBackend.xml`. Create this file by duplicating `CEsupCupsBackend-example.xml`, and edit it to reflect your own configuration. Most tags should not be changed since the default behaviour should fit to common needs.

For help on the configuration tags, please refer to the comments in file `CEsupCupsBackend-example.xml`.

log4j.properties

As the backend uses log4j, it is very easy to configure the logs of `esup-cups-backend`. You just have to write your own `properties/log4j.properties` file, by duplicating `log4j-example.properties` (for more information, please use the comments in this file).

3.8.3 Deploying the backend

Edit file `properties/build.properties` to fit to your configuration (refer to the comments of file `properties/build.properties` for more information) and run:

```
ant deploy
```

This will compile Java sources and deploy the classes and the configuration files in a proper place.

Important: this command should be ran every time your change any of the configuration files for your changes to be taken into account.

3.8.4 Using the backend

`Esup-cups-backend` should be used just as any other backend, with its own syntax.

Allowed patterns for the URIs are:

```
esup://address?backend=socket&port=xxx[&accounting=xxx]
esup://address?backend=ipp[&port=xxx]&path=xxx[&accounting=xxx]
esup://address?backend=lpd&path=xxx[&accounting=xxx]
```

At this time, only `accounting=hp` is allowed, corresponding to the way `esup-cups-backend` prompts HP network printers to get their status and the value of their page counter.

4 esup-print-channel (to be completed)

To be completed!

5 Références

- [1] ESUP-Portail project, <http://www.esup-portail.org>.
- [2] Java 2 platform, Enterprise Edition, <http://java.sun.com/j2ee>.
- [3] Java Data Objects, <http://www.jdocentral.com>.
- [4] Web Services, <http://www.w3.org/2002/ws/>.
- [5] Internet Printing Protocol, RFC2566, <http://www.faqs.org/rfcs/rfc2566.html>.
- [6] Server Message Block, <http://samba.anu.edu.au/cifs/docs/what-is-smb.html>.
- [7] Hibernate, Relational Persistence For Idiomatic Java, <http://www.hibernate.org>.
- [8] CUPS, Common Unix Printing System, <http://www.cups.org>.
- [9] Samba, Opening Windows to a wider world, <http://www.samba.org>.
- [10] NTLM authentication for Windows, <http://www.innovation.ch/java/ntlm.html>.

6 ChangeLog

Version 0.1

Initial version (Pascal Aubry).

Version 0.2

Added the configuration of Windows XP clients (Katy Santerre).
Added Samba instructions (Katy santerre and Pascal Aubry).

Version 0.3

Added the configuration of `esup-cups-backend` (Pascal Aubry).
Added the configuration of Linux FC3 clients (Ambroise Diascorn).
Added CUPS printer drivers (Pascal Aubry).
Added Samba printer drivers (Katy Santerre).