

# A CUPS PDF Printer for your SUSE Linux Enterprise Desktop

Prepared By Axel Schmidt <[axel.schmidt@novell.com](mailto:axel.schmidt@novell.com)>  
Version 1.1  
10 July, 2006

## INTRODUCTION

Although everybody has access to OpenOffice (<http://www.openoffice.org>) most of the users are stuck with any other application that is not capable of producing a PDF version of their documents without external tools. Since printing is handled through CUPS a PDF printer provided by CUPS which performs the “ps2pdf” conversion automatically in the background was on top of my wish list for SLED. Furthermore beside Linux, Win clients are able to use the CUPS PDF printer via IPP or Samba.

Searching the web I found some information about this issues and a script of a CUPS backend which nearly meets my requirements. Adapting this solution results in the following CUPS PDF printer for your SUSE Linux Enterprise Desktop.

To provide a CUPS PDF printer complete the following:

- Create a CUPS PDF printer backend
- Select an PPD (PostScript Printer Description) file
- Add the new CUPS PDF printer with lpadmin
- Create a device URI / printer queue
- Print test page

### Prerequisites

I assume you got a good working properly installed SUSE Linux Enterprise Desktop 10. Other packages you need, like CUPS, ps2pdf and gs (Ghostscript) should be installed on your system with a normal SLED10 deployment. The Postscript to PDF converter “ps2pdf” is contained in the ghostscript-library rpm.

## CUPS CONFIGURATION

CUPS backends can be run as the root or lp user. On SUSE the standard default settings of CUPS in the configuration file “/etc/cups/cups.conf” is ...

```
User lp
Group lp
RunAsUser Yes

<Location /admin>
AuthType BasicDigest
```

... which runs CUPS as user “lp”.

The pdf-writer CUPS backend provides a so so-called user mode which needs CUPS to be run as root. In user mode all created PDF files will reside in a PDF directory, which will be created in the home directory of the user.

To set SUSE to the standard default settings of CUPS change the lines above to:

```
User lp
Group lp
RunAsUser No

<Location /admin>
AuthType Basic
```

## CUPS BACKENDS

The CUPS backend interface provides a standard method for sending document files to different physical interfaces. Backends must be capable of reading from a filename on the command-line or from the standard input, copying the standard input to a temporary file if required by the physical interface.

Backends must support DEVICE DISCOVERY. When run with no arguments, the backend should list the devices and schemes it supports or is advertising to stdout. The output consists of zero or more lines consisting of any of the following forms:

```
device-class scheme "Unknown" "device-info"
device-class device-uri "device-make-and-model" "device-info"
device-class device-uri "device-make-and-model" "device-info" "device-id"
```

Every backend should accept 5 or 6 command-line arguments:

```
job user title copies options [filename]
```

For further information see backend(1) man page or CUPS documentation at (<http://www.cups.org>).

### CUPS PDF-WRITER Backend

For providing a CUPS PDF printer backend, you have to add the following script “pdf-writer” below the path:

#### **/usr/lib/cups/backend**

To create “pdf-writer” use the editor of your choice and perform the following steps:

- Login as root
- e.g. “vi /usr/lib/cups/backend/pdf-writer”
- cut and paste the following lines below
- save file
- chmod 755 /usr/lib/cups/backend/pdf-writer

```
#!/bin/sh
#####
#
#   File:           pdf-writer
#
#   Description:    A simple PDF Writer for CUPS
#
#   Copyright:      (c) 2006 Axel Schmidt, SUSE LINUX GmbH, Nuernberg, Germany
#                   (c) 2001 Michael Goffioul (kdeprint <at> swing <dot> be)
#
#   License:        GPL
#
#####

PS2PDF=`which ps2pdf`
DATE=`date +%Y-%m-%d-%H:%M.%S`

# USERMODE="on" changes PDFPATH to "$HOME/<user-name>/PDF"
#
USERMODE="off"
DEFGROUP="users"
HOME="/home"
JOB=$1
PRTUSER=$2

# Create user based file name
```

```

#
create_user_fname()
{
    if [ "$PRTUSER" != "" ]; then
        FILENAME="$PDFPATH/$PRTUSER-$DATE.pdf"
        logger "pdf-writer: \"$PRTUSER-$DATE.pdf\" was placed in: $PDFPATH"
    else
        FILENAME="$PDFPATH/$DATE.pdf"
        logger "pdf-writer: \"$DATE.pdf\" was placed in: $PDFPATH"
    fi
}

# Check DEVICE DISCOVERY + test PS2PDF
#
if [ "$JOB" = "" ]; then
    if test -f "$PS2PDF" ; then
        echo "network pdf-writer:/export/share/pdf \"unkown\" \"pdf writer\" "
        logger "pdf-writer: INFO: USERMODE=\"$USERMODE\""
        if [ "$USERMODE" = "on" ]; then
            logger "pdf-writer: CUPS user must be set to root with USERMODE=\"on\""
        fi
        exit 0
    else
        echo "Error: $0 - ps2pdf is not available!"
        logger "pdf-writer: ERROR: ps2pdf is not available!"
        exit 1
    fi
fi

logger "pdf-writer started: $1 $2 $3 $4 $5 $6"

# Check number of command line arguments
#
if [ $# -ne 5 -a $# -ne 6 ]; then
    echo "Usage: $0 job-id user title copies options [file]"
    logger "pdf-writer: CRITICAL: Printer stopped !"
    exit 1
fi

# get PDF-WRITER directory from device URI
#
PDFPATH=${DEVICE_URI#pdf-writer;}

# Create output filename based on user name and user mode
#
if [ "$USERMODE" = "on" -a "$PRTUSER" != "" ]; then
    mkdir -p $HOME/$PRTUSER/PDF
    chown $PRTUSER $HOME/$PRTUSER/PDF

# check if cups run as lp or root
# user lp may not run chown
#
    if [ $? -eq 1 ]; then
        create_user_fname
    else
        chgrp $DEFGROUP /home/$PRTUSER/PDF
        PDFPATH=$HOME/$PRTUSER/PDF
        FILENAME="$PDFPATH/$DATE.pdf"
        logger "pdf-writer: \"$DATE.pdf\" was placed in: $PDFPATH"
    fi
else
    create_user_fname
fi

# Check write status
#
if [ ! -d "$PDFPATH" -o ! -w "$PDFPATH" ]; then
    logger "pdf-writer: ERROR: directory $PDFPATH not writeable"
    exit 1
fi

```

```

# Run ps2pdf (ghostscript)
if [ $# -eq 6 ]; then
    $PS2PDF $6 $FILENAME >& /dev/null
else
    $PS2PDF - $FILENAME >& /dev/null
fi

# Chown + set permissions for the user
# Note: this will no work if CUPS runs as lp
if [ "$PRTUSER" != "" ]; then
    chmod 644 $FILENAME
    chown $PRTUSER:$DEFGROUP $FILENAME
fi
exit 0

#
##### CUPS pdf-writer ends here #####
#

```

**Note:** To enable the new CUPS backend you have to restart cups by executing "rccups restart"

### Selecting a PPD File

Finally a PPD (PostScript Printer Description) file is needed by CUPS. The available models can be found below the path: **/usr/share/cups/model**

Here we select a generic PostScript printer for as driver for the CUPD-PDF printer, e.g. Postscript-level2.ppd.

After the base installation of SLED10 you maybe have to unzip the PPD file.

- login as root
- cd /usr/share/cups/model
- gunzip -d Postscript-level2.ppd.gz

For achieving the best results in PDF printing, you also can download the free Adobe™ Distiller ppd file here:

<http://www.adobe.com/support/downloads/thankyou.jsp?ftpID=204&fileID=204>

### Adding the CUPS PDF Printer

CUPS provides two methods for adding printers: a command-line program called `lpadmin` and a Web interface. The `lpadmin` command allows you to perform most printer administration tasks from the command-line and is located in `/usr/sbin`. The Web interface is located at: <http://localhost:631/admin>

To add the CUPS PDF printer perform the following steps:

- Login as root
- Add printer URI: `mkdir -p /export/share/pdf`
- `chmod -R 777 /export`
- Using PPD file: Postscript-level2 execute the following:

```

lpadmin -p CUPS-PDF -v pdf-writer:/export/share/pdf/ -E -P \
/usr/share/cups/model/Postscript-level2.ppd -D "PDF Writer for CUPS" \
-L "PDF Backend /usr/lib/backend/pdf-writer"

```

Using the PPD file ADIST5.PPD from Adobe execute the following:

- Login as root
- `unzip -d adobe.zip`
- `cp Adobe/ADIST5.PPD /usr/share/cups/model/Acrobat-Distiller.ppd`

- `lpadmin -p CUPS-PDF -v pdf-writer:/export/share/pdf/ -E -P \`  
`/usr/share/cups/model/Acrobat-Distiller.ppd -D "PDF Writer for CUPS" \`  
`-L "PDF Backend /usr/lib/backend/pdf-writer"`

Launching the CUPS GUI will show the result:

All printed PDF file can be found at: `/export/share/pdf`

```
axels@dhcp234:/export/share/pdf> ls -l
```

```
-rw-r----- 1 lp      lp      51357 2006-07-06 15:35 axels-2006-07-06-15:35.48.pdf
-rw-r----- 1 lp      lp      35707 2006-07-06 15:38 axels-2006-07-06-15:38.09.pdf
-rw-r----- 1 axels   users   3283  2006-07-06 15:54 axels-2006-07-06-15:54.57.pdf
-rw-r----- 1 axels   users  106546 2006-07-06 16:05 axels-2006-07-06-16:05.23.pdf
-rw-r----- 1 lp      lp      52550 2006-07-06 15:30 2006-07-06-15:30.53.pdf
-rw-r----- 1 root    users   52550 2006-07-06 16:04 2006-07-06-16:04.49.pdf
```

Note: Depending on if CUPS runs as root or user lp the owner and group of the generated PDF may differ.

### Switching USERMODE to "on"

The CUPS pdf-writer backend supports direct printing to the users home directory. To enable this feature edit `/usr/lib/cups/backend/pdf-writer` and set the following:

```
USERMODE="on"
```

Furthermore CUPS must be configured to run as user root. Refer to chapter "CUPS CONFIGURATION". After changing `/etc/cups/cupsd.conf` don't forget to restart the CUPS (`rc cups restart`).

As shown in the picture below, all user related PDF file will reside in the users home directory, e.g. at `/home/axels/PDF` running pdf-writer in user mode.

