Printing with CUPS via netatalk

Rainer Ruprechtsberger 0056402

$Id: pap - docu.tex, v1.12002/05/2217: 39: 43 rupiExp$

Contents

1 Introduction

The old BSD-style lpd has a lot of troubles, frequently vulnerabilities and some conceptional limits. There are two different approaches to solve this problems. The Common Unix Printing System (CUPS) and lpr New Generation (lprNG). While lprNG stays close to the concept of the BSD implementation CUPS implements the Internet Printing Protocol (as described in RFC 2568) and a (at least for UN*X like operating systems, the MacOS use this techniques since the late 80s) new printer driver system the PostScript Printer Description files.

This offers a easy to use configuration, because the driver knows about maximum page margins and other things hard to implement in old lpd. Unfortunately currently there is no public available back-end for printing to an AppleTalk printer. Apple(inc.) talked about publishing their implementation "as soon as possible" but in the meanwhile we have to use scripts like mine to do that job.

2 Where to get the Software and Documentation

CUPS

CUPS is available for various flavors of UN*X. It is published under the GPL and can be found under the following URL: www.cups.org. Installation and a more in depth administration manual is the CUPS Software Administrators Manual on-line at www.cups.org/sam.html. This document refers to version 1.1.14 of CUPS.

netatalk

Netatalk works under Linux, BSD* and Sun-OS/Solaris. You can get it under sourceforge.net/projects/netatalk. Unfortunately its not well documented, but the installed default configuration files are commented and most of the utilities come with man-pages. The pap back-end introduced here is tested with version 1.5 only, but it should work with 1.4 too.

pap back-end

The pap back-end is available on-line under www.oeh.uni-linz.ac.at/%7Erupi/pap and this document is currently the only available documentation. The back-end itself is a small shellscript which uses various tools usually existing on a UNIX system: cat, rm, and basename. Currently I'm using perl to do some regex wizardry, but it should not be that problem porting them to sed/awk.

3 Short Introduction to CUPS

This section gives only a brief introduction. For a more in depth discussion take a look in the CUPS documentation.

CUPS take care of filtering any transmitted job. The information needed to do this comes with the ppd file. The filtered data is given to the back-end. This back-end actually takes care of transmitting the data to the printer. So all you need besides CUPS and netatalk to print to an AppleTalk printer is the back-end and the correct ppd files.

3.1 A note to ppd files

There are a lot of Apple Printer ppd-files for the so called "foomatic" floating around in the net. You *cannot* use this together with this pap back-end. Unfortunately the only source I know about is a MacOS X installation. But you can use this pap back-end without a proper ppd. Simply install it as a "raw" device in CUPS. The pap back-end checks if the input is in PostScript and filters through psf (which comes with netatalk) if it isn't. Of course you loose the main advantage of CUPS if you're doing so.

4 `pap` back-end

4.1 Install procedure

This is a simple task: copy the pap script into the back-end directory of your CUPS installation (usually `/usr/var/cups/backend`). Since the pap script supports the cups device detection this should work.

4.2 Configuration

Follow the normal CUPS procedure. The available printer listing should show you a line like:
`AppleTalk LaserWriter (foobar)`
where foobar is the AppleTalk name of the printer. The pap-back-end has to be configured with a Device URI like:
`pap://foobar/`

4.3 Limits and known problems

Currently there is no error checking implemented. So you're not informed if a paper jam or postscript error occurs. Even worse cups thinks all is £ne and your print jobs are spooled to /dev/null.

AppleTalk Zones aren't tested.