

A simple, generic print server for distributed LPD queues. Installation and operation guide

Francesco Prelz, Stefania Alborghetti, Mauro Campanella, Luca Carbone,
Giuseppe Lo Biondo, Marcello Meroni
INFN Milano, Computing Services

June 6th, 1998

Abstract:

The installation and operation of a UNIX-based print server and multi-OS print client designed to provide distributed, LPD-based print services to various client platforms is described. The development of this tool is still in progress: the distribution structure and the installation details may be changed in future releases.

1 Introduction and hardware requirements

The requirements, installation and configuration details for a print server/client pair developed to meet the specifications set out in the companion design document (*Requirements and design specifications*) are described here.

This tool is meant to simplify the task of configuring, administering and maintaining a central print server for multiple, network-accessible queues. It also provides a number of features that are not available through standard, RFC 1179 LPD servers. These are described in brief detail in the next section.

Almost any commercially available printer can be plugged into the print server system we describe here, provided the following two conditions are met:

- The printer must be accessible via an LPD/RFC 1179 server, be it resident on the printer (in the case of a network printer), or on a remote host where the printer is directly attached.
Alternatively (if no LPD server is available), the printer must provide a passthru TCP port that will cause any data sent to the port to be printed.
- The printer must be *PostScript* or *GhostScript* compatible. This means that either a resident PostScript interpreter has to be available on the printer, or the printer can accept one of the many print data formats that can be generated by the publicly available *GhostScript* (gs) utility.

See the configuration file description (section ??) for details on how to configure a queue for TCP port access or *GhostScript* filtering.

2 Special (non-LPD) features

2.1 Access control

Access to a print queue can be currently granted or denied on a per-user or a per-host/domain/network basis. This allows to create private print queues, to exclude noisy users and to limit access to the queues from the outside world. See the *allow_user*, *deny_user*, *allow_host* and *deny_host* options in Section ??.

2.2 Job reordering by size

A queue can optionally be instructed to rearrange the print jobs so that shorter ones are printed first. This is not a perfectly fair criterion (there are short PostScript files that can require a long processing time), but it has a very strong sociological impact. See the *reorder* option in Section ??.

2.3 Queue reservation

It can sometimes be necessary to allow users of distributed printers a short amount of time to manually feed special printing media such as letterhead paper or transparent film into the printer before their job is printed. Many printers provide a "manual feed" special tray, that can be selected via appropriate options in the print file (see for instance the **-e** option in Section ??). For printers that do *not* provide such option, a reservation mechanism is available. A print job can be submitted with a flag (see for instance **-r**

in Section ??) that will cause the print server to hold the queue when the job reaches the top of the queue. The print user will then be able to insert the required special media, and to “release” the queue with an appropriate print command (see the **-R** flag in Section ??). This will cause the job to be actually printed.

2.4 Sanity checks

A couple of small features with the purpose of reducing the amount of paper wasted because of print mistakes were also added. A queue can be configured so that *only* files with the PostScript magic signature are actually uploaded to the printer (see the *just_postscript* option in Section ??). Also, only files where the **file** command output matches a certain substring can be printed. This allows for example to only print text files (including PostScript files but not binary files). See the *check_file* option in Section ?? for details.

2.5 Accounting

The *account* flag in the configuration file activates print accounting for a given queue. This will cause the number of pages printed to be logged, along with usernames and hostnames. This feature relies on the fact that all files are converted into PostScript files by the server, and that almost all PostScript files generated by recent drivers comply with the PostScript Document Structuring Conventions (DSC).

3 Software requirements and server installation

The print *server* is available for “generic” UNIX, and should be run on two different hosts (even though one host will work), so that the “secondary” print server host can take over in case of failure of the primary host.

The print server relies on a couple of widely available public-domain tools: *psnup* (from the *psutils* package) and *mpage*. It is possible that these are already installed on the system. The print server matches v2.4 of *mpage* and v1pl16 of *psnup*. For convenience, these versions are bundled together with the distribution archive of the print server.

At the time of writing, little has been done to insure plug&play installation of the utility, and there still are few hardcoded elements that may require some working around. Here are the installation directions for the server:

NOTE: Re-organizing the print services on a UNIX host is a reasonably advanced system administration task. The following assumes the reader is familiar with UNIX administrative and housekeeping procedures and commands.

1. Obtain the archive file. It is currently available as `http://www.mi.infn.it/printmi.tar.gz`.
2. Unwind the tar file in an area of your choice.
3. Create a user `printmi` on the system. This user will be the owner of the spool files and of the daemon processes. A different, existing user can be used, but its name must be replaced in the `PRINTD_REGULAR_USER` macro in `printd.h`.
4. Make sure that the `mpage` and `psnup` utilities are available on the system. You can build them in the distribution tree by changing directory to `src/mpage-2.4` and `src/psutils` and issuing a `make` command.
5. Create a spool area. The default area is `/var/spool/printmi`. There should be enough room in this filesystem to accommodate all the files that are submitted to all printer queues at any given time. A link to the `mpage` and `psnup` executables should be placed in this area. You can choose a different area, but in this case the `PRINT_PROGRAM_ASCII` and `PRINT_PROGRAM_POST` macros in `printd.h` must be modified accordingly. The spool area should be owned (or at least writable) by the `printmi` user (see above).
6. Build the print server and the print client by issuing a `make` command in the `src` directory. At the end of the compilation the `printd` and `print` executables should be available in the distribution `bin` directory.
7. Create a symbolic link of `printd` in the spool area.
8. Create a configuration file (the default name is `queues.db`) in the spool area. This should be owned (or readable) by the `printmi` user. The format of this file and the configuration options are detailed in Section ??.
9. Repeat all previous steps on the secondary server host, if any.
10. In order for the print clients to work correctly, the primary and secondary print servers should be be aliased in the Domain Name Server as `printmi1.domain` and `printmi2.domain`. This can be changed as

well if the `SERV_HOST_NAME_I` and `SERV_HOST_NAME_II` macros in the `print.h` header file are suitably modified and the print client is rebuilt (`make client`).

11. Start the primary server. The first time it can be started by hand. Be sure you have root privileges (the LPD client needs to open a privileged socket). Move into the spool area and issue a `./printd -p secondary.host.domain` command. The output can be redirected to a suitable log file, and the job can be put into background (how to do this depends on which shell is being used).
12. Check whether the server is working: move back into the distribution `bin` directory and issue a `./lpq all`. This should produce a listing of the currently configured queues.
13. At this point, the appropriate commands can be added into the host boot configuration (`rc.d`, `init.d` or the like), so that the server will be started when the host is booted.
14. Build and install the print clients. On UNIX, a print client is built by issuing a `make client` command in the `src` build area. A print client for a VAX/VMS (with its Multinet) can be built by issuing a `@COMPILE_M` command. There is only *one* executable for the print client. This executable does all the functions of the ordinary `lpr`, `lpq`, `lprm`, `lp` and `cancel` print commands, and determines which function it has to perform by its invocation name. It will work as `lpr` if it is invoked via a symbolic link whose name ends in “`lpr`”; it will work as `lpq` if it is invoked via a symbolic link whose name ends in “`lpq`”, and so on for the other supported functions. A `sh` script (`install_client`) is provided to replace the existing print commands with links to the new code. It can simply be invoked from the `bin` distribution area, after building the print client. Its effects are undone by `uninstall_client`.
15. Each print client host may also have the print command man pages installed. They are found in the `man` distribution directory, and can be either pointed to via the system default `MANPATH` or copied to `/usr/local/man`.

4 Print server configuration options

The format of the print server configuration file is described here. The configuration file name defaults to `queues.db`, or is found in the `PRINTMI_QUEUE_CONFIG`

environment variable. All configuration details for the print queues are found in this file.

Every line of the configuration file which is non-empty and does not begin with `#` (which identifies comments) describes a single print queue. When multiple lines are required to describe a single print queue the “\” continuation mark can be used. The backslash needs to be the *last* character of the line to be continued.

The first three whitespace-separated tokens of each configuration line are required and contain the following information:

- **Queue name** - this is the name that identifies the queue to the print server and the print clients. It doesn't need to be the same as the LPD queue name.
- **Remote LPD server name** - This is the host name or IP address of the LPD server that will carry out the actual print service for this queue.
- **Queue name on the remote LPD server** - This is the name that identifies the print queue on the remote LPD server described in the previous token.

Note: If the queue name is in the form `tcp_port:port_#` no LPD server will be used on the printer, but all print data will be sent directly to the printer via TCP port `port_#`. This is useful for printers with buggy LPD servers, which provide such a passthru port.

Any number of whitespace-separated options, taken from the following list, can then be added for each queue on the same line (using continuation marks if needed):

Option	Description
reorder	Enable (for this queue) the optional rearrangement of print jobs by size, so that shorter jobs are printed first.
reserve	Enable the “queue reservation” option for this queue. This allows to hold the queue for a short while before a given print job is actually printed, so that special print media, such as letterhead paper or transparent film, can be loaded into the printer.
banner	Set the “banner” option in the lpr print requests sent to this queue. The handling of this option is up to the lpd server, but will generally cause a banner page to be printed before or after the print job.
duplex	“Paper-saving” option: force all print jobs sent to this queue to be printed on two sides by default. If this option is set, the one-side print flag <i>must</i> be set in print requests in order to obtain one-side printouts.
account	Cause accounting information to be output by the print server. Accounting information is printed to the standard output, each line starting with “ACCT:”, and includes the username and host that submitted the job, the queue name and the number of pages printed.

Option	Description
nostatus	This option needs to be specified for printers or lpd servers that do <i>not</i> return the standard “no entries” string when the queue is empty. It will cause the server to accept an empty status string as evidence that the queue is idle.
just_postscript	If this option is present, only files that start with the standard PostScript signature (!PS-Adobe) will actually be sent to the printer. All other files will be dequeued and discarded. This prevents non-PostScript or corrupted files to be sent to the printer, and avoids wasting paper.
idlemon <i><number_of_minutes></i>	If present, this option will cause an external program or script (pointed to by the environment variable PRINTMI_IDLEMON_SCRIPT) to be executed every time the queue has not been able to print a submitted job for <i>number_of_minutes</i> minutes. This may be used for early trouble reporting.
allow_user <i><username1>[,username2,...]</i>	If this option is present, only the user, or users who match the <i>username1,username2...</i> list will be allowed to send printer jobs to the queue. All other users will be denied access. The username list needs to be specified as a single token, without embedded whitespace. This option is overridden by deny_user , if present.
deny_user <i><username1>[,username2,...]</i>	If this option is present, the user, or users whose name matches the <i>username1,username2...</i> list are denied access to the printer. All other users are allowed access. If this option is present, the allow_user option is ignored. The username list needs to be specified as a single token, without embedded whitespace.

Option	Description
allow_host <i><host1.domain or IP address></i> <i>[,host2.domain,...]</i>	<p>Only hosts whose name matches one the host names in the list, or whose resolved address matches a listed IP address or subnet are allowed access to the queue. All other hosts are denied access. Hosts can be listed by fully qualified domain name, or by IP address. If an IP address with less than 4 octets is present in the comma-separated list (e.g. 207.68.156, or 207.68, or 207 only) the directive will apply to all the hosts where the first octets in the resolved IP address match the specified octets. Hosts that don't match a listed name or IP address are denied access to the queue. Hosts whose name cannot be resolved are denied access. The hostname/IP address list needs to be specified as a single token, with no embedded whitespace. This option is overridden by deny_host.</p>
deny_host <i><host1.domain or IP address></i> <i>[,host2.domain,...]</i>	<p>Hosts whose name or IP address matches a listed name, IP address or subnet (the list is in the same format as for allow_host, see above) are denied access to the queue. All other hosts are allowed access. If this option is present, allow_host will be ignored.</p>
use_gs <i><name of a valid Ghostscript device></i>	<p>This option will instruct the queue manager to process every print job through gs (GhostScript) before sending it to the printer. This function is useful for printers with a slow or buggy resident PostScript interpreter, as it will shift the load of interpreting PostScript to the print server CPU. GhostScript must be already installed on the system and has to be accessible via the PATH available to the print server. gs is invoked for the output device specified as an argument. The device must obviously be available (compiled into) GhostScript.</p>

Option	Description
<code>check_file</code> < <i>substring_to_match</i> >	This option will cause the print server to invoke the <code>file</code> utility and discard any print job where the output from the <code>file</code> command does not contain <i>substring_to_match</i> . This may be useful to eliminate binary and special files that can be mistakenly sent to a printer queue. The <code>file</code> command must be accessible via the PATH available to the print server. Since its output format varies for different UNIX flavours, this setup option has to be tailored on different print servers.

Finally, a description of the printer, meaningful enough for the users of the local system (printer description, location, etc.) can be appended to the configuration line: everything after a ‘|’ on the queue configuration line (with the appropriate continuation marks, if applicable) is stored as the printer description. This description is returned in the `lpq` command output.

Here’s an example `queues.db` configuration file, where various options are shown:

```
#
# This is an example queues.db file that demonstrates the usage
# of some of the available options.
#
#name server queue      options
#
# This is a generic queue on a remote LPD server. We enable job reservation
# and job reordering by size
printer1 lpd_server.domain printer1      reorder reserve |\
This is the queue description (e.g. 1st floor Building Q)
#
# This is a queue that causes a banner to be printed, forces all jobs
# to be printed on two sides and prints an accounting log of all jobs
printer2 lpd_server.domain printer2      banner duplex account
#
# Access control examples. Here we explicitly deny users chip&dale from
# accessing the queue printer3, which points to the raw LPD queue of a
# remote printer.
printer3      printer3.domain      raw      deny_user chip,dale
```

```

#
# Or we allow only user big_boss to access printer4:
printer4          printer4.domain          raw_cr          allow_user big_boss
#
# Here we limit access to printer5 from two C-class IP networks only
# (note that in this case all data will be sent to TCP port 2000,
# instead of the LPD server):
printer5          printer5.domain          tcp_port:2000    allow_host 192.1.1,192.1.2
#
# Or we ban access from host hacker.domain
printer6          lpd_server.domain         printer6         deny_host hacker.domain
#
# This is a color PaintJet, for which we'd like to process data with gs:
# note that the GhostScript driver (pjxl300) must be compiled into the
# installed gs.
printer7          color_printer.domain      raw              use_gs pjxl300
#

```

5 Print client: lpr command

This is the main command used to submit a print request to a print queue. It comprises all the regular options for the BSD-style print command, with a few extra options. Here's the command synopsis:

lpr [*options*] *filename*

where *filename* is the file that needs to be submitted to the printer (will default to standard input if omitted or equal to "-"), and *options* can be a combination of options taken from the following list:

5.1 Lpr options valid for any file

- b y|n** This switch causes a border (outline) to be drawn around each printed page. The border will be drawn around each sub-page when multiple pages are drawn on the same sheet of paper. This makes n-up printouts more readable.
- e tray-# | M** This option allows to select the physical tray where the printer obtains the print media (paper, special paper or film). The number assigned to each tray differs from printer to printer, and may not correspond to the number marked on the trays or printer. If **-eM** is used, the printer will process the job taking the print media from its

manual feed slot or tray (if present). This option can be used to print on special supports such as letterhead paper or transparent films, that need to be fed manually.

- j** This option will cause only the print request ID to be returned to the standard output, rather than the full message (*Job accepted...*). This option is useful when the print command is invoked by a script or some other automated procedure.
- k sides** This option causes the print server to add into the file directives for one side (**-K1**) or two sides (**-K2**) printing. The directives are added to the final PostScript file before it is sent to the printer (the server converts all ASCII files to PostScript before printing them).
Note: when an original PostScript file is printed, it may already contain directives for one/two sides printing (most commercial PostScript print drivers include this choice). The server will try to add another directive at the end of the file prologue (if the file complies with the PostScript Document Structuring Conventions, DSC), but will not try to remove existing commands. So, if the print mode is defined or modified in the file after the PostScript prologue, it will not be overridden by this option.
- Nboxes** This option will cause *boxes* pages to be printed on each sheet of paper. The argument *boxes* is an integer, and can be 1, 2, 4, 8 for text (ASCII) files, or any integer for PostScript files. The default value is 1 (one).
- O landscape|portrait** This option causes the printed output lines or images to be parallel to the short side of the page **-Oportrait** or to the long side of the page **-Olandscape**.
- P print_queue** This flag specifies the print job destination, that is, the printer queue where the job will be queued. For a list of available printers the **lpq all** command can be used. If this flag is not present, the printer specified in the PRINTER environment variable will be used. If this environment variable is not set, an error will be generated.
- r** Submit job reservation. The current print request is a printer reservation. This means that the printer queue will be held (for a maximum of 15 minutes) when this print request reaches the top of the print queue. Nothing will be printed until the reservation is released with the **-R** option. This option is useful to print on special print supports such as letterhead paper or transparent film on printers which do not provide

a manual feed slot (see the **-e** option). If the reservation is not released within 15 minutes, the job is cancelled.

- R** *print_job_id* Releases a previously submitted print reservation. This causes the the print data to be actually sent to the printer. *print_job_id* is the ID of the reservation (returned by **lpr** or found via **lpq**).
- t on|off** Tumble option: **-ton** will allow double-side (**-K2**) printouts to be bound on the shorter side. The default value **-toff** makes double-side printouts that can be bound on the longer side.
- V** Print the current client and server versions on standard output and exit.
- y range** This option allows to specify a page range to be printed. The range has to be specified in the format *start_page-end_page*[,*start_page-end_page*,...]. For example, to print all pages from page 5 to page 8, use **-y5-8**
- # number_of_copies** This options causes the print job to be re-submitted to the queue *number_of_copies* times. This causes the specified number of copies of the file to be printed.

5.2 Lpr options valid for PostScript files only

- a paper** Can be set in order to include in the PostScript file to be printed a directive that requests a specific type of Paper. Most multiple tray printers will hold until the requested paper size becomes available in one of the trays. This option can be useful when more than one paper size (e.g. A4 and A3 paper) is available in a given printer, since it will cause the printer to select the required size of paper. Valid values for *paper* are **a3**, **a4**, **a5**, **b5**, **letter**, **legal**, **tabloid**, **statement**, **executive**, **folio**, **quarto**, **10x14**. The default paper size is A4.
- c** This flag only applies when at least 4 pages are printed on the same sheet of paper. If present, it will cause successive pages to be printed down in vertical columns, rather than in horizontal rows.
- d scale_factor** This option will apply the specified *scale_factor* to the file. *scale_factor* is a floating point number: for example **-d 1.2** enlarges the output by 20%, while **-d .8** reduces the output by 20%.

- m** *margin* This option allows to specify a white space margin to leave around the printed page. *margin* has to be specified as a floating-point number, followed by one of the following units: pt, in, cm, mm. For example: **-m 2.5cm**.
- n** This flag causes the print server to ignore every formatting option and to avoid passing the file through a formatter altogether. This option is useful when certain PostScript files (typically those produce by private applications) are incorrectly translated by the formatter because of an unusual prologue or graphical context handling within the file. If this flag is present, all the other formatting options (such as -K, -N, etc.) are ignored.
- q** Quiet output flag: causes the output response from the remote print server (usually the print job ID) *not* to be printed on the standard output.

5.3 Lpr options valid for plaintext (ASCII) files only

- f** *font_type* This option allows to select the font type used to format plain-text (ASCII) files. Available fonts are (for example) Times-Roman, Times-Italic, Times-Bold, Times-BoldItalic, Helvetica, Helvetica-Oblique, Helvetica-Bold, Helvetica-BoldOblique, Courier, Courier-Oblique, Courier-Bold, Courier-BoldOblique. The default font is Courier.
- p y|n** This option enables **-py** or disables **-pn** the automatic numbering of pages. A title is also printed on each page, whose contents are by default the file name, the date, and the user who submitted the job. A different title may be supplied with the **-T** option.
- s** *font_size* This option allows to choose the size (in points) of the characters used when printing the text file. *font_size* is an integer, whose default value is 12 (pt).
- T***title* Causes *title* to be printed on each page of formatted text output, instead of the file name. This option is only meaningful when **-py** is present.
- u** *lines_per_page* This option causes the file to be formatted with *lines_per_page* on each page.
- W** *width* Set the number of text columns printed on the page. *width* is an integer. The default value is 80 columns.

6 Print client: **lpq** command

This commands queries the print queue(s), and produces a listing of pending and printing jobs. There are two possible forms:

lpq *print_queue*

prints the queue listing for the specified queue. The forms **lpq -P** *print_queue* or **lpq -p** *print_queue* are also valid for compatibility. In order to get a listing for all the available queues the following form may be used:

lpq < **-j** > all

The **-j** flag will produce a short output, with the queue names and description only (this may be useful for external print scripts or tools).

7 Print client: **lprm** command

This command is used to remove a print job from the queue. Note that, if the job has already been submitted to its final LPD server destination, it will removed from the remote server as well.

The command synopsis is as follows:

lprm < **-S** > *print_job_id*

print_job_id is the ID which is returned by the *lpr* command, or by the *lpq* command.

The **-S** option directs the cancel request to the secondary (backup) server. This is necessary when the primary server has recovered from an interruption, and is waiting for the backup server to finish processing all of the pending print requests. A job ID assigned by the backup server may not be available in the primary server, which is now responding to the client requests. This flag will cause the job to be correctly removed from the backup server. A message recommending the use of the **-S** option is returned by the primary server whan a print remove request fails, but the backup server still has pending jobs.

8 Print client: SysV compatibility (**lp** and **cancel**)

In order to make the print service accessible by client who only know about the System V (*lp*) syntax, the print client is also able to understand the *lp* and *cancel* command syntax. However, very few “extended” options are supported in this mode, which is provided for compatibility only. The *lpr*

command should be considered the primary access to the print server, and all the possible options are available there.

Command synopsis:

lp [**options**] *file_to_print*

cancel *print_job_id*

cancel works the same as *lprm*. The options for *lp* are as follows (print data will be accepted from the standard input if *file_to_print* is missing or “-”):

- d** *print_queue* This flag specifies the print job destination, that is, the printer queue where the job will be queued. If this flag is not present, the printer specified in the PRINTER environment variable will be used. If this environment variable is not set, an error will be generated.
- n** *number_of_copies* This options causes the print job to be re-submitted to the queue *number_of_copies* times, so that the specified number of copies of the file is printed.
- r** Submit job reservation. The current print request is a printer reservation. Works the same as the same option in *lpr*.
- R** *print_job_id* Releases a previously submitted print reservation (same behaviour as for *lpr*). *print_job_id* is the ID of the reservation (returned by **lpr** or found via **lpq**).