Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'lp.1p' command

$ man lp.1p

LP(1P) POSIX Programmer's Manual LP(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

lp ? send files to a printer

SYNOPSIS

lp [-c] [-d dest] [-n copies] [-msw] [-o option]... [-t title] [file...]

DESCRIPTION

The lp utility shall copy the input files to an output destination in an unspecified manner. The default output destination should be to a hardcopy device, such as a printer or microfilm recorder, that produces non-volatile, human-readable documents. If such a device is not avail? able to the application, or if the system provides no such device, the lp utility shall exit with a non-zero exit status.

The actual writing to the output device may occur some time after the lp utility successfully exits. During the portion of the writing that corresponds to each input file, the implementation shall guarantee ex? clusive access to the device.

The lp utility shall associate a unique request ID with each request.

Normally, a banner page is produced to separate and identify each print

job. This page may be suppressed by implementation-defined conditions, such as an operator command or one of the -o option values.

OPTIONS

The lp utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following options shall be supported:

-c Exit only after further access to any of the input files is no longer required. The application can then safely delete or modify the files without affecting the output operation. Nor? mally, files are not copied, but are linked whenever possi? ble. If the -c option is not given, then the user should be careful not to remove any of the files before the request has been printed in its entirety. It should also be noted that in the absence of the -c option, any changes made to the named files after the request is made but before it is printed may be reflected in the printed output. On some implementations, -c may be on by default.

-d dest Specify a string that names the destination (dest). If dest is a printer, the request shall be printed only on that spe? cific printer. If dest is a class of printers, the request shall be printed on the first available printer that is a member of the class. Under certain conditions (printer un? availability, file space limitation, and so on), requests for specific destinations need not be accepted. Destination names vary between systems.

If -d is not specified, and neither the LPDEST nor PRINTER environment variable is set, an unspecified destination is used. The -d dest option shall take precedence over LPDEST, which in turn shall take precedence over PRINTER. Results are undefined when dest contains a value that is not a valid destination name.

-m Send mail (see mailx) after the files have been printed. By default, no mail is sent upon normal completion of the print

request.

-n copies Write copies number of copies of the files, where copies is a positive decimal integer. The methods for producing multiple copies and for arranging the multiple copies when multiple file operands are used are unspecified, except that each file shall be output as an integral whole, not interleaved with portions of other files.

-o option Specify printer-dependent or class-dependent options. Sev? eral such options may be collected by specifying the -o op? tion more than once.

-s Suppress messages from lp.

-t title Write title on the banner page of the output.

-w Write a message on the user's terminal after the files have been printed. If the user is not logged in, then mail shall be sent instead.

OPERANDS

The following operand shall be supported:

file A pathname of a file to be output. If no file operands are specified, or if a file operand is '-', the standard input shall be used. If a file operand is used, but the -c option is not specified, the process performing the writing to the output device may have user and group permissions that differ from that of the process invoking lp.

STDIN

The standard input shall be used only if no file operands are speci? fied, or if a file operand is '-'. See the INPUT FILES section.

INPUT FILES

The input files shall be text files.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of lp:

LANG Provide a default value for the internationalization vari? ables that are unset or null. (See the Base Definitions vol? ume of POSIX.1?2017, Section 8.2, Internationalization Vari?

ables for the precedence of internationalization variables used to determine the values of locale categories.)

LC_ALL If set to a non-empty string value, override the values of all the other internationalization variables.

LC_CTYPE Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments and input files).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error and informative messages written to standard output.

LC_TIME Determine the format and contents of date and time strings displayed in the lp banner page, if any.

LPDEST Determine the destination. If the LPDEST environment variable is not set, the PRINTER environment variable shall be used. The -d dest option takes precedence over LPDEST. Results are undefined when -d is not specified and LPDEST contains a value that is not a valid destination name.

NLSPATH Determine the location of message catalogs for the processing of LC_MESSAGES.

PRINTER Determine the output device or destination. If the LPDEST and PRINTER environment variables are not set, an unspecified output device is used. The -d dest option and the LPDEST en? vironment variable shall take precedence over PRINTER. Re? sults are undefined when -d is not specified, LPDEST is un? set, and PRINTER contains a value that is not a valid device or destination name.

TZ Determine the timezone used to calculate date and time strings displayed in the lp banner page, if any. If TZ is un? set or null, an unspecified default timezone shall be used.

ASYNCHRONOUS EVENTS

Default.

STDOUT

The lp utility shall write a request ID to the standard output, unless -s is specified. The format of the message is unspecified. The request ID can be used on systems supporting the historical cancel and lpstat utilities.

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

0 All input files were processed successfully.

>0 No output device was available, or an error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

The pr and fold utilities can be used to achieve reasonable formatting for the implementation's default page size.

A conforming application can use one of the file operands only with the -c option or if the file is publicly readable and guaranteed to be available at the time of printing. This is because POSIX.1?2008 gives the implementation the freedom to queue up the request for printing at some later time by a different process that might not be able to access the file.

EXAMPLES

1. To print file file:

   lp -c file

2. To print multiple files with headers:

   pr file1 file2 | lp

RATIONALE

The lp utility was designed to be a basic version of a utility that is already available in many historical implementations. The standard de? velopers considered that it should be implementable simply as:

```
cat "$@" > /dev/lp
```

after appropriate processing of options, if that is how the implementa? tion chose to do it and if exclusive access could be granted (so that two users did not write to the device simultaneously). Although in the future the standard developers may add other options to this utility, it should always be able to execute with no options or operands and send the standard input to an unspecified output device.

This volume of POSIX.1?2017 makes no representations concerning the format of the printed output, except that it must be ``human-readable'' and ``non-volatile''. Thus, writing by default to a disk or tape drive or a display terminal would not qualify. (Such destinations are not prohibited when -d dest, LPDEST, or PRINTER are used, however.)

This volume of POSIX.1?2017 is worded such that a ``print job'' con? sisting of multiple input files, possibly in multiple copies, is guar? anteed to print so that any one file is not intermixed with another, but there is no statement that all the files or copies have to print out together.

The -c option may imply a spooling operation, but this is not required.

The utility can be implemented to wait until the printer is ready and then wait until it is finished. Because of that, there is no attempt to define a queuing mechanism (priorities, classes of output, and so on).

On some historical systems, the request ID reported on the STDOUT can be used to later cancel or find the status of a request using utilities not defined in this volume of POSIX.1?2017.

Although the historical System V lp and BSD lpr utilities have provided similar functionality, they used different names for the environment variable specifying the destination printer. Since the name of the utility here is lp, LPDEST (used by the System V lp utility) was given precedence over PRINTER (used by the BSD lpr utility). Since environ? ments of users frequently contain one or the other environment vari?

able, the lp utility is required to recognize both. If this was not done, many applications would send output to unexpected output devices when users moved from system to system.

Some have commented that lp has far too little functionality to make it worthwhile. Requests have proposed additional options or operands or both that added functionality. The requests included:

* Wording requiring the output to be ``hardcopy''
* A requirement for multiple printers
* Options for supporting various page-description languages

Given that a compliant system is not required to even have a printer, placing further restrictions upon the behavior of the printer is not useful. Since hardcopy format is so application-dependent, it is diffi? cult, if not impossible, to select a reasonable subset of functionality that should be required on all compliant systems.

The term unspecified is used in this section in lieu of implementation-defined as most known implementations would not be able to make defini? tive statements in their conformance documents; the existence and usage of printers is very dependent on how the system administrator config? ures each individual system.

Since the default destination, device type, queuing mechanisms, and ac? ceptable forms of input are all unspecified, usage guidelines for what a conforming application can do are as follows:

* Use the command in a pipeline, or with -c, so that there are no permission problems and the files can be safely deleted or modi? fied.
* Limit output to text files of reasonable line lengths and printable characters and include no device-specific formatting information, such as a page description language. The meaning of ``reasonable'' in this context can only be answered as a quality-of-implementation issue, but it should be apparent from historical usage patterns in the industry and the locale. The pr and fold utilities can be used to achieve reasonable formatting for the default page size of the implementation.

Alternatively, the application can arrange its installation in such a way that it requires the system administrator or operator to provide the appropriate information on lp options and environment variable val? ues.

At a minimum, having this utility in this volume of POSIX.1?2017 tells the industry that conforming applications require a means to print out? put and provides at least a command name and LPDEST routing mechanism that can be used for discussions between vendors, application develop? ers, and users. The use of ``should'' in the DESCRIPTION of lp clearly shows the intent of the standard developers, even if they cannot man? date that all systems (such as laptops) have printers.

This volume of POSIX.1?2017 does not specify what the ownership of the process performing the writing to the output device may be. If -c is not used, it is unspecified whether the process performing the writing to the output device has permission to read file if there are any re? strictions in place on who may read file until after it is printed.

Also, if -c is not used, the results of deleting file before it is printed are unspecified.

FUTURE DIRECTIONS

None.

SEE ALSO

mailx

The Base Definitions volume of POSIX.1?2017, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guidelines

COPYRIGHT

at http://www.opengroup.org/unix/online.html .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.ker?nel.org/doc/man-pages/reporting_bugs.html .