



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

Rocky Enterprise Linux 9.2 Manual Pages on command 'man.1'

\$ man man.1

MAN(1) Manual pager utils MAN(1)

NAME

man - an interface to the system reference manuals

SYNOPSIS

man [man options] [[section] page ...] ...

man -k [apropos options] regexp ...

man -K [man options] [section] term ...

man -f [whatis options] page ...

man -l [man options] file ...

man -w|-W [man options] page ...

DESCRIPTION

man is the system's manual pager. Each page argument given to man is normally the name of a program, utility or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if page exists in several sections.

The table below shows the section numbers of the manual followed by the types of pages they contain.

- 1 Executable programs or shell commands
- 2 System calls (functions provided by the kernel)
- 3 Library calls (functions within program libraries)
- 4 Special files (usually found in /dev)

- 5 File formats and conventions, e.g. /etc/passwd
- 6 Games
- 7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7), man-pages(7)
- 8 System administration commands (usually only for root)
- 9 Kernel routines [Non standard]

A manual page consists of several sections.

Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.

The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.

bold text type exactly as shown.

italic text replace with appropriate argument.

[-abc] any or all arguments within [] are optional.

-a|-b options delimited by | cannot be used together.

argument ... argument is repeatable.

[expression] ... entire expression within [] is repeatable.

Exact rendering may vary depending on the output device. For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined or coloured text instead.

The command or function illustration is a pattern that should match all possible invocations. In some cases it is advisable to illustrate several exclusive invocations as is shown in the SYNOPSIS section of this manual page.

EXAMPLES

man ls

Display the manual page for the item (program) ls.

man man.7

Display the manual page for macro package man from section 7. (This is an alternative spelling of "man 7 man".)

man 'man(7)'

Display the manual page for macro package man from section 7. (This is another alternative spelling of "man 7 man". It may be more convenient when copying and pasting

cross-references to manual pages. Note that the parentheses must normally be quoted to protect them from the shell.)

`man -a intro`

Display, in succession, all of the available intro manual pages contained within the manual. It is possible to quit between successive displays or skip any of them.

`man -t bash | lpr -Pps`

Format the manual page for bash into the default troff or groff format and pipe it to the printer named ps. The default output for groff is usually PostScript. `man --help` should advise as to which processor is bound to the -t option.

`man -l -Tdvi ./foo.1x.gz > ./foo.1x.dvi`

This command will decompress and format the nroff source manual page `./foo.1x.gz` into a device independent (dvi) file. The redirection is necessary as the -T flag causes output to be directed to stdout with no pager. The output could be viewed with a program such as `xdvi` or further processed into PostScript using a program such as `dvips`.

`man -k printf`

Search the short descriptions and manual page names for the keyword `printf` as regular expression. Print out any matches. Equivalent to `apropos printf`.

`man -f smail`

Lookup the manual pages referenced by `smail` and print out the short descriptions of any found. Equivalent to `whatis smail`.

OVERVIEW

Many options are available to `man` in order to give as much flexibility as possible to the user. Changes can be made to the search path, section order, output processor, and other behaviours and operations detailed below.

If set, various environment variables are interrogated to determine the operation of `man`.

It is possible to set the "catch-all" variable `$MANOPT` to any string in command line for `man`, with the exception that any spaces used as part of an option's argument must be escaped (preceded by a backslash). `man` will parse `$MANOPT` prior to parsing its own command line. Those options requiring an argument will be overridden by the same options found on the command line. To reset all of the options set in `$MANOPT`, `-D` can be specified as the initial command line option. This will allow `man` to "forget" about the options specified in `$MANOPT`, although they must still have been valid.

Manual pages are normally stored in nroff(1) format under a directory such as

/usr/share/man. In some installations, there may also be preformatted cat pages to improve performance. See manpath(5) for details of where these files are stored.

This package supports manual pages in multiple languages, controlled by your locale. If your system did not set this up for you automatically, then you may need to set `$LC_MESSAGES`, `$LANG`, or another system-dependent environment variable to indicate your preferred locale, usually specified in the POSIX format:

```
<language>[_<territory>[.<character-set>[,<version>]]]
```

If the desired page is available in your locale, it will be displayed in lieu of the standard (usually American English) page.

If you find that the translations supplied with this package are not available in your native language and you would like to supply them, please contact the maintainer who will be coordinating such activity.

Individual manual pages are normally written and maintained by the maintainers of the program, function, or other topic that they document, and are not included with this package.

If you find that a manual page is missing or inadequate, please report that to the maintainers of the package in question.

For information regarding other features and extensions available with this manual pager, please read the documents supplied with the package.

DEFAULTS

The order of sections to search may be overridden by the environment variable `$MANSECT` or by the `SECTION` directive in `/etc/manpath.config`. By default it is as follows:

```
1 n | 8 3 0 2 3posix 3pm 3perl 3am 5 4 9 6 7
```

The formatted manual page is displayed using a pager. This can be specified in a number of ways, or else will fall back to a default (see option `-P` for details).

The filters are deciphered by a number of means. Firstly, the command line option `-p` or the environment variable `$MANROFFSEQ` is interrogated. If `-p` was not used and the environment variable was not set, the initial line of the `nroff` file is parsed for a preprocessor string. To contain a valid preprocessor string, the first line must resemble

```
"<string>
```

where `string` can be any combination of letters described by option `-p` below.

If none of the above methods provide any filter information, a default set is used.

A formatting pipeline is formed from the filters and the primary formatter (`nroff` or `[tg]roff` with `-t`) and executed. Alternatively, if an executable program `mandb_nfmt` (or

mandb_tfmt with -t) exists in the man tree root, it is executed instead. It gets passed the manual source file, the preprocessor string, and optionally the device specified with -T or -E as arguments.

OPTIONS

Non-argument options that are duplicated either on the command line, in \$MANOPT, or both, are not harmful. For options that require an argument, each duplication will override the previous argument value.

General options

-C file, --config-file=file

Use this user configuration file rather than the default of ~/.manpath.

-d, --debug

Print debugging information.

-D, --default

This option is normally issued as the very first option and resets man's behaviour to its default. Its use is to reset those options that may have been set in \$MANOPT. Any options that follow -D will have their usual effect.

--warnings[=warnings]

Enable warnings from groff. This may be used to perform sanity checks on the source text of manual pages. warnings is a comma-separated list of warning names; if it is not supplied, the default is "mac". See the ?Warnings? node in info groff for a list of available warning names.

Main modes of operation

-f, --whatis

Equivalent to whatis. Display a short description from the manual page, if available. See whatis(1) for details.

-k, --apropos

Equivalent to apropos. Search the short manual page descriptions for keywords and display any matches. See apropos(1) for details.

-K, --global-apropos

Search for text in all manual pages. This is a brute-force search, and is likely to take some time; if you can, you should specify a section to reduce the number of pages that need to be searched. Search terms may be simple strings (the default), or regular expressions if the --regex option is used.

Note that this searches the sources of the manual pages, not the rendered text, and so may include false positives due to things like comments in source files.

Searching the rendered text would be much slower.

`-l, --local-file`

Activate "local" mode. Format and display local manual files instead of searching through the system's manual collection. Each manual page argument will be interpreted as an nroff source file in the correct format. No cat file is produced. If '-' is listed as one of the arguments, input will be taken from stdin. When this option is not used, and man fails to find the page required, before displaying the error message, it attempts to act as if this option was supplied, using the name as a filename and looking for an exact match.

`-w, --where, --path, --location`

Don't actually display the manual page, but do print the location of the source nroff file that would be formatted. If the `-a` option is also used, then print the locations of all source files that match the search criteria.

`-W, --where-cat, --location-cat`

Don't actually display the manual page, but do print the location of the preformatted cat file that would be displayed. If the `-a` option is also used, then print the locations of all preformatted cat files that match the search criteria.

If `-w` and `-W` are both used, then print both source file and cat file separated by a space. If all of `-w`, `-W`, and `-a` are used, then do this for each possible match.

`-c, --catman`

This option is not for general use and should only be used by the catman program.

`-R encoding, --recode=encoding`

Instead of formatting the manual page in the usual way, output its source converted to the specified encoding. If you already know the encoding of the source file, you can also use `manconv(1)` directly. However, this option allows you to convert several manual pages to a single encoding without having to explicitly state the encoding of each, provided that they were already installed in a structure similar to a manual page hierarchy.

Consider using `man-recode(1)` instead for converting multiple manual pages, since it has an interface designed for bulk conversion and so can be much faster.

`-L locale, --locale=locale`

man will normally determine your current locale by a call to the C function `setlocale(3)` which interrogates various environment variables, possibly including `$LC_MESSAGES` and `$LANG`. To temporarily override the determined value, use this option to supply a locale string directly to man. Note that it will not take effect until the search for pages actually begins. Output such as the help message will always be displayed in the initially determined locale.

`-m system[,...], --systems=system[,...]`

If this system has access to other operating systems' manual pages, they can be accessed using this option. To search for a manual page from NewOS's manual page collection, use the option `-m NewOS`.

The system specified can be a combination of comma delimited operating system names. To include a search of the native operating system's manual pages, include the system name `man` in the argument string. This option will override the `$SYSTEM` environment variable.

`-M path, --manpath=path`

Specify an alternate manpath to use. By default, man uses manpath derived code to determine the path to search. This option overrides the `$MANPATH` environment variable and causes option `-m` to be ignored.

A path specified as a manpath must be the root of a manual page hierarchy structured into sections as described in the `man-db` manual (under "The manual page system"). To view manual pages outside such hierarchies, see the `-l` option.

`-S list, -s list, --sections=list`

The given list is a colon- or comma-separated list of sections, used to determine which manual sections to search and in what order. This option overrides the `$MANSECT` environment variable. (The `-s` spelling is for compatibility with System V.)

`-e sub-extension, --extension=sub-extension`

Some systems incorporate large packages of manual pages, such as those that accompany the Tcl package, into the main manual page hierarchy. To get around the problem of having two manual pages with the same name such as `exit(3)`, the Tcl pages were usually all assigned to section I. As this is unfortunate, it is now possible to put the pages in the correct section, and to assign a specific "extension" to them, in this case, `exit(3tcl)`. Under normal operation, man will display `exit(3)`

in preference to `exit(3tcl)`. To negotiate this situation and to avoid having to know which section the page you require resides in, it is now possible to give `man` a sub-extension string indicating which package the page must belong to. Using the above example, supplying the option `-e tcl` to `man` will restrict the search to pages having an extension of `*tcl`.

`-i, --ignore-case`

Ignore case when searching for manual pages. This is the default.

`-I, --match-case`

Search for manual pages case-sensitively.

`--regex`

Show all pages with any part of either their names or their descriptions matching each page argument as a regular expression, as with `apropos(1)`. Since there is usually no reasonable way to pick a "best" page when searching for a regular expression, this option implies `-a`.

`--wildcard`

Show all pages with any part of either their names or their descriptions matching each page argument using shell-style wildcards, as with `apropos(1) --wildcard`. The page argument must match the entire name or description, or match on word boundaries in the description. Since there is usually no reasonable way to pick a "best" page when searching for a wildcard, this option implies `-a`.

`--names-only`

If the `--regex` or `--wildcard` option is used, match only page names, not page descriptions, as with `whatis(1)`. Otherwise, no effect.

`-a, --all`

By default, `man` will exit after displaying the most suitable manual page it finds. Using this option forces `man` to display all the manual pages with names that match the search criteria.

`-u, --update`

This option causes `man` to update its database caches of installed manual pages. This is only needed in rare situations, and it is normally better to run `mandb(8)` instead.

`--no-subpages`

By default, `man` will try to interpret pairs of manual page names given on the command

mand line as equivalent to a single manual page name containing a hyphen or an underscore. This supports the common pattern of programs that implement a number of subcommands, allowing them to provide manual pages for each that can be accessed using similar syntax as would be used to invoke the subcommands themselves. For example:

```
$ man -aw git diff
/usr/share/man/man1/git-diff.1.gz
```

To disable this behaviour, use the `--no-subpages` option.

```
$ man -aw --no-subpages git diff
/usr/share/man/man1/git.1.gz
/usr/share/man/man3/Git.3pm.gz
/usr/share/man/man1/diff.1.gz
```

Controlling formatted output

`-P pager, --pager=pager`

Specify which output pager to use. By default, `man` uses `pager`, falling back to `cat` if `pager` is not found or is not executable. This option overrides the `$MANPAGER` environment variable, which in turn overrides the `$PAGER` environment variable. It is not used in conjunction with `-f` or `-k`.

The value may be a simple command name or a command with arguments, and may use shell quoting (backslashes, single quotes, or double quotes). It may not use pipes to connect multiple commands; if you need that, use a wrapper script, which may take the file to display either as an argument or on standard input.

`-r prompt, --prompt=prompt`

If a recent version of `less` is used as the pager, `man` will attempt to set its prompt and some sensible options. The default prompt looks like

```
Manual page name(sec) line x
```

where `name` denotes the manual page name, `sec` denotes the section it was found under and `x` the current line number. This is achieved by using the `$LESS` environment variable.

Supplying `-r` with a string will override this default. The string may contain the text `$MAN_PN` which will be expanded to the name of the current manual page and its section name surrounded by "(" and ")". The string used to produce the default could be expressed as

\ Manual\ page\ \\$MAN_PN\ ?ltline\ %lt?L/%L.:

byte\ %bB?s/%s..?\ (END):?pB\ %pB\\%..

(press h for help or q to quit)

It is broken into three lines here for the sake of readability only. For its mean?

ing see the less(1) manual page. The prompt string is first evaluated by the

shell. All double quotes, back-quotes and backslashes in the prompt must be es?

caped by a preceding backslash. The prompt string may end in an escaped \$ which

may be followed by further options for less. By default man sets the -ix8 options.

The \$MANLESS environment variable described below may be used to set a default

prompt string if none is supplied on the command line.

-7, --ascii

When viewing a pure ascii(7) manual page on a 7 bit terminal or terminal emulator,

some characters may not display correctly when using the latin1(7) device descrip?

tion with GNU nroff. This option allows pure ascii manual pages to be displayed in

ascii with the latin1 device. It will not translate any latin1 text. The follow?

ing table shows the translations performed: some parts of it may only be displayed

properly when using GNU nroff's latin1(7) device.

Description Octal latin1 ascii

??

continuation hyphen 255 ? -

bullet (middle dot) 267 ? o

acute accent 264 ? '

multiplication sign 327 ? x

If the latin1 column displays correctly, your terminal may be set up for latin1

characters and this option is not necessary. If the latin1 and ascii columns are

identical, you are reading this page using this option or man did not format this

page using the latin1 device description. If the latin1 column is missing or cor?

rupt, you may need to view manual pages with this option.

This option is ignored when using options -t, -H, -T, or -Z and may be useless for

nroff other than GNU's.

-E encoding, --encoding=encoding

Generate output for a character encoding other than the default. For backward com?

patibility, encoding may be an nroff device such as ascii, latin1, or utf8 as well

as a true character encoding such as UTF-8.

`--no-hyphenation, --nh`

Normally, `nroff` will automatically hyphenate text at line breaks even in words that do not contain hyphens, if it is necessary to do so to lay out words on a line without excessive spacing. This option disables automatic hyphenation, so words will only be hyphenated if they already contain hyphens.

If you are writing a manual page and simply want to prevent `nroff` from hyphenating a word at an inappropriate point, do not use this option, but consult the `nroff` documentation instead; for instance, you can put "\%" inside a word to indicate that it may be hyphenated at that point, or put "\%" at the start of a word to prevent it from being hyphenated.

`--no-justification, --nj`

Normally, `nroff` will automatically justify text to both margins. This option disables full justification, leaving justified only to the left margin, sometimes called "ragged-right" text.

If you are writing a manual page and simply want to prevent `nroff` from justifying certain paragraphs, do not use this option, but consult the `nroff` documentation instead; for instance, you can use the ".na", ".nf", ".fi", and ".ad" requests to temporarily disable adjusting and filling.

`-p string, --preprocessor=string`

Specify the sequence of preprocessors to run before `nroff` or `troff/groff`. Not all installations will have a full set of preprocessors. Some of the preprocessors and the letters used to designate them are: `eqn` (e), `grap` (g), `pic` (p), `tbl` (t), `vgrind` (v), `refer` (r). This option overrides the `$MANROFFSEQ` environment variable. `zsoe`? `lim` is always run as the very first preprocessor.

`-t, --troff`

Use `groff -mandoc` to format the manual page to `stdout`. This option is not required in conjunction with `-H`, `-T`, or `-Z`.

`-T[device], --troff-device[=device]`

This option is used to change `groff` (or possibly `troff`'s) output to be suitable for a device other than the default. It implies `-t`. Examples (provided with `Groff-1.17`) include `dvi`, `latin1`, `ps`, `utf8`, `X75` and `X100`.

`-H[browser], --html[=browser]`

This option will cause groff to produce HTML output, and will display that output in a web browser. The choice of browser is determined by the optional browser argument if one is provided, by the \$BROWSER environment variable, or by a compile-time default if that is unset (usually lynx). This option implies -t, and will only work with GNU troff.

-X[dpi], --gxditview[=dpi]

This option displays the output of groff in a graphical window using the gxditview program. The dpi (dots per inch) may be 75, 75-12, 100, or 100-12, defaulting to 75; the -12 variants use a 12-point base font. This option implies -T with the X75, X75-12, X100, or X100-12 device respectively.

-Z, --ditroff

groff will run troff and then use an appropriate post-processor to produce output suitable for the chosen device. If groff -mandoc is groff, this option is passed to groff and will suppress the use of a post-processor. It implies -t.

Getting help

-, --help

Print a help message and exit.

--usage

Print a short usage message and exit.

-V, --version

Display version information.

EXIT STATUS

- 0 Successful program execution.
- 1 Usage, syntax or configuration file error.
- 2 Operational error.
- 3 A child process returned a non-zero exit status.
- 16 At least one of the pages/files/keywords didn't exist or wasn't matched.

ENVIRONMENT

MANPATH

If \$MANPATH is set, its value is used as the path to search for manual pages.

See the SEARCH PATH section of manpath(5) for the default behaviour and details of how this environment variable is handled.

MANROFFOPT

Every time man invokes the formatter (nroff, troff, or groff), it adds the contents of \$MANROFFOPT to the formatter's command line.

MANROFFSEQ

If \$MANROFFSEQ is set, its value is used to determine the set of preprocessors to pass each manual page through. The default preprocessor list is system dependent.

MANSECT

If \$MANSECT is set, its value is a colon-delimited list of sections and it is used to determine which manual sections to search and in what order. The default is "1 n 1 8 3 0 2 3posix 3pm 3perl 3am 5 4 9 6 7", unless overridden by the SECTION directive in /etc/manpath.config.

MANPAGER, PAGER

If \$MANPAGER or \$PAGER is set (\$MANPAGER is used in preference), its value is used as the name of the program used to display the manual page. By default, pager is used, falling back to cat if pager is not found or is not executable.

The value may be a simple command name or a command with arguments, and may use shell quoting (backslashes, single quotes, or double quotes). It may not use pipes to connect multiple commands; if you need that, use a wrapper script, which may take the file to display either as an argument or on standard input.

MANLESS

If \$MANLESS is set, its value will be used as the default prompt string for the less pager, as if it had been passed using the -r option (so any occurrences of the text \$MAN_PN will be expanded in the same way). For example, if you want to set the prompt string unconditionally to ?my prompt string?, set \$MANLESS to ?-Psmyprompt string?. Using the -r option overrides this environment variable.

BROWSER

If \$BROWSER is set, its value is a colon-delimited list of commands, each of which in turn is used to try to start a web browser for man --html. In each command, %s is replaced by a filename containing the HTML output from groff, %% is replaced by a single percent sign (%), and %c is replaced by a colon (:).

SYSTEM If \$SYSTEM is set, it will have the same effect as if it had been specified as the argument to the -m option.

MANOPT If \$MANOPT is set, it will be parsed prior to man's command line and is expected to be in a similar format. As all of the other man specific environment variables can

be expressed as command line options, and are thus candidates for being included in \$MANOPT it is expected that they will become obsolete. N.B. All spaces that should be interpreted as part of an option's argument must be escaped.

MANWIDTH

If \$MANWIDTH is set, its value is used as the line length for which manual pages should be formatted. If it is not set, manual pages will be formatted with a line length appropriate to the current terminal (using the value of \$COLUMNS, and ioctl(2) if available, or falling back to 80 characters if neither is available).

Cat pages will only be saved when the default formatting can be used, that is when the terminal line length is between 66 and 80 characters.

MAN_KEEP_FORMATTING

Normally, when output is not being directed to a terminal (such as to a file or a pipe), formatting characters are discarded to make it easier to read the result without special tools. However, if \$MAN_KEEP_FORMATTING is set to any non-empty value, these formatting characters are retained. This may be useful for wrappers around man that can interpret formatting characters.

MAN_KEEP_STDERR

Normally, when output is being directed to a terminal (usually to a pager), any error output from the command used to produce formatted versions of manual pages is discarded to avoid interfering with the pager's display. Programs such as groff often produce relatively minor error messages about typographical problems such as poor alignment, which are unsightly and generally confusing when displayed along with the manual page. However, some users want to see them anyway, so, if \$MAN_KEEP_STDERR is set to any non-empty value, error output will be displayed as usual.

MAN_DISABLE_SECCOMP

On Linux, man normally confines subprocesses that handle untrusted data using a seccomp(2) sandbox. This makes it safer to run complex parsing code over arbitrary manual pages. If this goes wrong for some reason unrelated to the content of the page being displayed, you can set \$MAN_DISABLE_SECCOMP to any non-empty value to disable the sandbox.

PIPELINE_DEBUG

If the \$PIPELINE_DEBUG environment variable is set to "1", then man will print de?

bugging messages to standard error describing each subprocess it runs.

LANG, LC_MESSAGES

Depending on system and implementation, either or both of `$LANG` and `$LC_MESSAGES` will be interrogated for the current message locale. `man` will display its messages in that locale (if available). See `setlocale(3)` for precise details.

FILES

`/etc/manpath.config`

man-db configuration file.

`/usr/share/man`

A global manual page hierarchy.

SEE ALSO

`apropos(1)`, `groff(1)`, `less(1)`, `manpath(1)`, `nroff(1)`, `troff(1)`, `whatis(1)`, `zsoelim(1)`, `man?path(5)`, `man(7)`, `catman(8)`, `mandb(8)`

Documentation for some packages may be available in other formats, such as `info(1)` or HTML.

HISTORY

1990, 1991 ? Originally written by John W. Eaton (`jwe@che.utexas.edu`).

Dec 23 1992: Rik Faith (`faith@cs.unc.edu`) applied bug fixes supplied by Willem Kasdorp (`wkasdo@nikhef.nikef.nl`).

30th April 1994 ? 23rd February 2000: Wilf. (G.Wilford@ee.surrey.ac.uk) has been developing and maintaining this package with the help of a few dedicated people.

30th October 1996 ? 30th March 2001: Fabrizio Polacco (`fpolacco@debian.org`) maintained and enhanced this package for the Debian project, with the help of all the community.

31st March 2001 ? present day: Colin Watson (`cjwatson@debian.org`) is now developing and maintaining man-db.

BUGS

<https://gitlab.com/cjwatson/man-db/-/issues>

<https://savannah.nongnu.org/bugs/?group=man-db>

2.10.2

2022-03-17

MAN(1)