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 'mkdir.1p' command

$ man mkdir.1p

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

mkdir ? make directories

SYNOPSIS

mkdir [-p] [-m mode] dir...

DESCRIPTION

The mkdir utility shall create the directories specified by the oper? ands, in the order specified.

For each dir operand, the mkdir utility shall perform actions equiva? lent to the mkdir() function defined in the System Interfaces volume of POSIX.1?2017, called with the following arguments:

1. The dir operand is used as the path argument.
2. The value of the bitwise-inclusive OR of S_IRWXU, S_IRWXG, and S_IRWXO is used as the mode argument. (If the -m option is speci? fied, the value of the mkdir() mode argument is unspecified, but the directory shall at no time have permissions less restrictive than the -m mode option-argument.)

OPTIONS

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

The following options shall be supported:

-m mode Set the file permission bits of the newly-created directory to the specified mode value. The mode option-argument shall be the same as the mode operand defined for the chmod util?ity. In the symbolic_mode strings, the op characters '+' and '-' shall be interpreted relative to an assumed initial mode of a=rwx; '+' shall add permissions to the default mode, '-' shall delete permissions from the default mode.

-p Create any missing intermediate pathname components. For each dir operand that does not name an existing direc?tory, before performing the actions described in the DESCRIP?TION above, the mkdir utility shall create any pathname com?ponents of the path prefix of dir that do not name an exist?ing directory by performing actions equivalent to first call?ing the mkdir() function with the following arguments:

1. A pathname naming the missing pathname component, ending with a trailing <slash> character, as the path argument
2. The value zero as the mode argument

and then calling the chmod() function with the following ar?guments:

1. The same path argument as in the mkdir() call
2. The value (S_IWUSR|S_IXUSR|~filemask)&0777 as the mode argument, where filemask is the file mode creation mask of the process (see the System Interfaces volume of POSIX.1?2017, umask())

Each dir operand that names an existing directory shall be ignored without error.

OPERANDS

The following operand shall be supported:

dir A pathname of a directory to be created.

STDIN

Not used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of mkdir:

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).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard er?ror.

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

ASYNCHRONOUS EVENTS

Default.

STDOUT

Not used.

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 the specified directories were created successfully, or the -p option was specified and all the specified directories either already existed or were created successfully.

>0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

The default file mode for directories is a=rwx (777 on most systems) with selected permissions removed in accordance with the file mode cre? ation mask. For intermediate pathname components created by mkdir, the mode is the default modified by u+wx so that the subdirectories can al? ways be created regardless of the file mode creation mask; if different ultimate permissions are desired for the intermediate directories, they can be changed afterwards with chmod.

Note that some of the requested directories may have been created even if an error occurs.

EXAMPLES

None.

RATIONALE

The System V -m option was included to control the file mode.

The System V -p option was included to create any needed intermediate directories and to complement the functionality provided by rmdir for removing directories in the path prefix as they become empty. Because no error is produced if any path component already exists, the -p op? tion is also useful to ensure that a particular directory exists.

The functionality of mkdir is described substantially through a refer? ence to the mkdir() function in the System Interfaces volume of POSIX.1?2017. For example, by default, the mode of the directory is af? fected by the file mode creation mask in accordance with the specified behavior of the mkdir() function. In this way, there is less duplica? tion of effort required for describing details of the directory cre?

ation.

FUTURE DIRECTIONS

None.

SEE ALSO

chmod, rm, rmdir, umask

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

The System Interfaces volume of POSIX.1?2017, mkdir(), umask()

COPYRIGHT