

NAME

chmod - change file modes

SYNOPSIS

chmod [-fhv] [-R [-H | -L | -P]] *mode file ...*

DESCRIPTION

The **chmod** utility modifies the file mode bits of the listed files as specified by the *mode* operand.

The options are as follows:

- f** Do not display a diagnostic message if **chmod** could not modify the mode for *file*, nor modify the exit status to reflect such failures.
- H** If the **-R** option is specified, symbolic links on the command line are followed and hence unaffected by the command. (Symbolic links encountered during tree traversal are not followed.)
- h** If the file is a symbolic link, change the mode of the link itself rather than the file that the link points to.
- L** If the **-R** option is specified, all symbolic links are followed.
- P** If the **-R** option is specified, no symbolic links are followed. This is the default.
- R** Change the modes of the file hierarchies rooted in the files, instead of just the files themselves. Beware of unintentionally matching the "." hard link to the parent directory when using wildcards like ".*".
- v** Cause **chmod** to be verbose, showing filenames as the mode is modified. If the **-v** flag is specified more than once, the old and new modes of the file will also be printed, in both octal and symbolic notation.

The **-H**, **-L** and **-P** options are ignored unless the **-R** option is specified. In addition, these options override each other and the command's actions are determined by the last one specified.

If **chmod** receives a SIGINFO signal (see the **status** argument for stty(1)), then the current filename as well as the old and new modes are displayed.

Only the owner of a file or the super-user is permitted to change the mode of a file.

EXIT STATUS

The **chmod** utility exits 0 on success, and >0 if an error occurs.

MODES

Modes may be absolute or symbolic. An absolute mode is an octal number constructed from the sum of one or more of the following values:

- 4000 (the setuid bit). Executable files with this bit set will run with effective uid set to the uid of the file owner. Directories with this bit set will force all files and sub-directories created in them to be owned by the directory owner and not by the uid of the creating process, if the underlying file system supports this feature: see `chmod(2)` and the **suid** option to `mount(8)`.
- 2000 (the setgid bit). Executable files with this bit set will run with effective gid set to the gid of the file owner.
- 1000 (the sticky bit). See `chmod(2)` and `sticky(7)`.
- 0400 Allow read by owner.
- 0200 Allow write by owner.
- 0100 For files, allow execution by owner. For directories, allow the owner to search in the directory.
- 0040 Allow read by group members.
- 0020 Allow write by group members.
- 0010 For files, allow execution by group members. For directories, allow group members to search in the directory.
- 0004 Allow read by others.
- 0002 Allow write by others.
- 0001 For files, allow execution by others. For directories allow others to search in the directory.

For example, the absolute mode that permits read, write and execute by the owner, read and execute by group members, read and execute by others, and no set-uid or set-gid behaviour is `755` ($400+200+100+040+010+004+001$).

The symbolic mode is described by the following grammar:

```

mode      ::= clause [, clause ...]
clause    ::= [who ...] [action ...] action
action    ::= op [perm ...]
who       ::= a | u | g | o
op        ::= + | - | =
perm      ::= r | s | t | w | x | X | u | g | o

```

The *who* symbols “u”, “g”, and “o” specify the user, group, and other parts of the mode bits, respectively. The *who* symbol “a” is equivalent to “ugo”.

The *perm* symbols represent the portions of the mode bits as follows:

r	The read bits.
s	The set-user-ID-on-execution and set-group-ID-on-execution bits.
t	The sticky bit.
w	The write bits.
x	The execute/search bits.
X	The execute/search bits if the file is a directory or any of the execute/search bits are set in the original (unmodified) mode. Operations with the <i>perm</i> symbol “X” are only meaningful in conjunction with the <i>op</i> symbol “+”, and are ignored in all other cases.
u	The user permission bits in the original mode of the file.
g	The group permission bits in the original mode of the file.
o	The other permission bits in the original mode of the file.

The *op* symbols represent the operation performed, as follows:

- + If no value is supplied for *perm*, the “+” operation has no effect. If no value is supplied for *who*, each permission bit specified in *perm*, for which the corresponding bit in the file mode creation mask (see `umask(2)`) is clear, is set. Otherwise, the mode bits represented by the specified *who* and *perm* values are set.
- If no value is supplied for *perm*, the “-” operation has no effect. If no value is supplied for *who*, each permission bit specified in *perm*, for which the corresponding bit in the file mode creation mask is set, is cleared. Otherwise, the mode bits represented by the specified *who* and *perm* values are cleared.
- = The mode bits specified by the *who* value are cleared, or, if no *who* value is specified, the owner, group and other mode bits are cleared. Then, if no value is supplied for *who*, each permission bit specified in *perm*, for which the corresponding bit in the file mode creation mask is clear, is set. Otherwise, the mode bits represented by the specified *who* and *perm* values are set.

Each *clause* specifies one or more operations to be performed on the mode bits, and each operation is applied to the mode bits in the order specified.

Operations upon the other permissions only (specified by the symbol “o” by itself), in combination with the *perm* symbols “s” or “t”, are ignored.

The “w” permission on directories will permit file creation, relocation, and copy into that directory. Files created within the directory itself will inherit its group ID.

EXAMPLES

- 644 make a file readable by anyone and writable by the owner only.
- go-w deny write permission to group and others.
- =rw,+X set the read and write permissions to the usual defaults, but retain any execute permissions that are currently set.
- +X make a directory or file searchable/executable by everyone if it is already searchable/executable by anyone.
- 755
- u=rwx,go=rx
- u=rwx,go=u-w make a file readable/executable by everyone and writable by the owner only.
- go= clear all mode bits for group and others.
- g=u-w set the group bits equal to the user bits, but clear the group write bit.

COMPATIBILITY

The `-v` option is non-standard and its use in scripts is not recommended.

SEE ALSO

`chflags(1)`, `install(1)`, `setfacl(1)`, `chmod(2)`, `stat(2)`, `umask(2)`, `fts(3)`, `setmode(3)`, `sticky(7)`, `symlink(7)`, `chown(8)`, `mount(8)`

STANDARDS

The `chmod` utility is expected to be IEEE Std 1003.2 (“POSIX.2”) compatible with the exception of the *perm* symbol “t” which is not included in that standard.

HISTORY

A `chmod` command appeared in Version 1 AT&T UNIX.

BUGS

There is no *perm* option for the naughty bits of a horse.