NAME

chflags - change file flags

SYNOPSIS

chflags [**-fhvx**] [**-R** [**-H** | **-L** | **-P**]] *flags file ...*

DESCRIPTION

The **chflags** utility modifies the file flags of the listed files as specified by the *flags* operand.

The options are as follows:

-f Do not display a diagnostic message if **chflags** could not modify the flags 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 traversal are not followed.)

-h If the *file* is a symbolic link, change the file flags of the link itself rather than the file to which it points.

-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 file flags 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 **chflags** to be verbose, showing filenames as the flags are modified. If the **-v** option is specified more than once, the old and new flags of the file will also be printed, in octal notation.

-x Do not cross mount points.

The flags are specified as an octal number or a comma separated list of keywords. The following keywords are currently defined:

arch, **archived**
set the archived flag (super-user only)

nodump

set the nodump flag (owner or super-user only)

opaque set the opaque flag (owner or super-user only)

sappnd, **sappend**
set the system append-only flag (super-user only)

schg, **schange**, **simmutable**
set the system immutable flag (super-user only)

snapshot
set the snapshot flag (filesystems do not allow changing this flag)

sunlnk, **sunlink**
set the system undeletable flag (super-user only)

uappnd, **uappend**
set the user append-only flag (owner or super-user only)

uarch, **uarchive**
set the archive flag (owner or super-user only)

uchg, **uchange**, **uimmutable**
set the user immutable flag (owner or super-user only)

uhidden, **hidden**
set the hidden file attribute (owner or super-user only)

uoffline, **offline**
set the offline file attribute (owner or super-user only)

urdonly, **rdonly**, **readonly**
set the DOS, Windows and CIFS readonly flag (owner or super-user only)

usparse, **sparse**
set the sparse file attribute (owner or super-user only)

usystem, **system**
set the DOS, Windows and CIFS system flag (owner or super-user only)

ureparse, **reparse**
set the Windows reparse point file attribute (owner or super-user only)

uunlnk, **uunlink**
set the user undeletable flag (owner or super-user only)

Putting the letters "*no*" before or removing the letters "*no*" from a keyword causes the flag to be cleared. For example:

nouchg	clear the user immutable flag (owner or super-user only)
dump	clear the nodump flag (owner or super-user only)

A few of the octal values include:

0	Clear all file flags.
1	Translates to the **nodump** keyword.
2	Translates to the **uchg** keyword.
3	Translates to the **uchg**, **nodump** keywords.
4	Translates to the **uappnd** keyword.
10	Translates to the **opaque** keyword.
20	translates to the **uunlnk** keyword.

Other combinations of keywords may be placed by using the octets assigned; however, these are the most notable.

Unless the **-H**, **-L**, or **-h** options are given, **chflags** on a symbolic link always succeeds and has no effect. 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.

You can use "ls -lo" to see the flags of existing files.

Note that the ability to change certain flags is dependent on the current kernel *securelevel* setting. See security(7) for more information on this setting.

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

EXIT STATUS

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

EXAMPLES

Recursively clear all flags on files and directories contained within the *foobar* directory hierarchy:

```
chflags -R 0 foobar
```

SEE ALSO

ls(1), chflags(2), stat(2), fts(3), security(7), symlink(7)

HISTORY

The **chflags** command first appeared in 4.4BSD.

BUGS

Only a limited number of utilities are **chflags** aware. Some of these tools include ls(1), cp(1), find(1), install(1), dump(8), and restore(8). In particular a tool which is not currently **chflags** aware is the pax(1) utility.