

## NAME

faked - daemon that remembers fake ownership/permissions of files manipulated by fakeroot processes.

## SYNOPSIS

**faked** [**--debug**] [**--foreground**] [**--cleanup**] [**--key** *msg-key*] [**--load**] [**--save-file** *save-file*] [**--port** *TCP-port*]

## DESCRIPTION

If a fakeroot process wants to change the ownership of a file, then **faked** is the process that remembers that new owner. If later the same fakeroot process does a `stat()` for that filename, then the libfakeroot wrapped `stat()` call will first ask faked for the fake ownership etc of that file, and then report it.

## OPTIONS

**--debug**

Print debugging information on stderr.

**--foreground**

Don't fork into the background.

**--cleanup** **number**

Cleanup the semaphores.

**--key** **key-number**

Don't allocate a new communication channel, but use channel specified by key. (If the specified channel doesn't exist, it's created).

**--save-file** **save-file**

Save the environment to save-file on exit.

**--load** Load a previously saved environment from the standard input.

**--unknown-is-real**

Use real ownership of previously-unknown files instead of setting them to root:root.

**--port** **tcp-port**

Use TCP port tcp-port.

## BUGS

None so far. Be warned, though: although I've written quite a few much larger (and smaller) programs, I've never written anything that was as tiny as **fakeroot**, had as many bugs as **fakeroot**, and still was as usable as, say, **fakeroot** version 0.0\_3, the first version that could be used to build itself.

## COPYING

**fakeroot** is distributed under the GNU General Public License. (GPL 2.0 or greater).

## AUTHORS

joost witteveen

<[joostje@debian.org](mailto:joostje@debian.org)>

Clint Adams

<[clint@debian.org](mailto:clint@debian.org)>

Timo Savola

## MANUAL PAGE

mostly by J.H.M. Dassen <[jdassen@debian.org](mailto:jdassen@debian.org)> mods/additions by joost and Clint.

## SEE ALSO

[fakeroot\(1\)](#), [dpkg-buildpackage\(1\)](#), [debuid\(1\)](#) /usr/share/doc/fakeroot/DEBUG