NAME

lockfile−progs − command−line programs to safely lock and unlock files and mailboxes (via liblockfile).

SYNOPSIS

mail−lock [−−use−pid] [−−retry *retry−count*]
mail−unlock
mail−touchlock [−−oneshot]

lockfile−create [−−use−pid] [−−retry *retry−count*] [-−lock−name] *filename*
lockfile−remove [-−lock−name] *filename*
lockfile−touch [−−oneshot] [-−lock−name] *filename*
lockfile−check [−−use-pid] [-−lock−name] *filename*

DESCRIPTION

Lockfile−progs provides a set a programs that can be used to lock and unlock mailboxes and files safely (via liblockfile):

mail−lock - lock the current user’s mailbox
mail−unlock - unlock the current user’s mailbox
mail−touchlock - touch the lock on the current user’s mailbox

lockfile−create - lock a given file
lockfile−remove - remove the lock on a given file
lockfile−touch - touch the lock on a given file
lockfile−check - check the lock on a given file

By default, the *filename* argument refers to the name of the file to be locked, and the name of the lockfile will be *filename* .lock. However, if the −−lock-name argument is specified, then *filename* will be taken as the name of the lockfile itself.

Each of the mail locking commands attempts to lock /var/spool/mail/<user>, where <user> is the name associated with the effective user ID, as determined by via geteuid(2).

Once a file is locked, the lock must be touched at least once every five minutes or the lock will be considered stale, and subsequent lock attempts will succeed. Also see the **−−use−pid** option and the **lockfile_create**(3) manpage.

The **lockfile−check** command tests whether or not a valid lock already exists.

OPTIONS

−q, **−−quiet**
Suppress any output. Success or failure will only be indicated by the exit status.

−v, **−−verbose**
Enable diagnostic output.

−l, **−−lock−name**
Do not append .lock to the *filename*. This option applies to**lockfile−cr eate**, **lockfile−remove**, **lockfile-touch**, or **lockfile-check**.

−p, **−−use−pid**
Write the parent process id (PPID) to the lockfile whenever a lockfile is created, and use that pid when checking a lock’s validity. See the**lockfile_cr eate**(3) manpage for more information. This option applies to **lockfile−create** and **lockfile-check**. NOTE: this option will not work correctly between machines sharing a filesystem.

−o, **−−oneshot**
Touch the lock and exit immediately. This option applies to**lockfile−touch** and **mail−touchlock**.

When not provided, these commands will run forever, touching the lock once every minute until killed.

−r *retry−count*, **−−retry** *retry−count*

Try to lock *filename retry−count* times before giving up. Each attempt will be delayed a bit longer than the last (in 5 second increments) until reaching a maximum delay of one minute between retries. If *retry−count* is unspecified, the default is 9 which will give up after 225 seconds if all 9 lock attempts fail.

EXAMPLES

Locking a file during a lengthy process:

```
lockfile−create /some/file
lockfile−touch /some/file &
# Save the PID of the lockfile−touch process
BADGER="$!"
do−something−important−with /some/file
kill "${BADGER}"
lockfile−remove /some/file
```

EXIT STATUS

0

For **lockfile−check** this indicates that a valid lock exists, otherwise it just indicates successful program execution.

Not 0

For **lockfile−check** a non-zero exit status indicates that the specified lock does not exist or is not valid. For other programs it indicates that some problem was encountered.

SEE ALSO

maillock(3)
touchlock(3)
mailunlock(3)
lockfile_create(3)
lockfile_remove(3)
lockfile_touch(3)
lockfile_check(3)

AUTHOR

Written by Rob Browning <rlb@defaultvalue.org>