NAME

updatedb − update a database for mlocate

SYNOPSIS

updatedb [*OPTION*]...

DESCRIPTION

updatedb creates or updates a database used by **locate**(1). If the database already exists, its data is reused to avoid rereading directories that have not changed.

updatedb is usually run daily by **cron**(8) to update the default database.

EXIT STATUS

updatedb returns with exit status 0 on success, 1 on error.

OPTIONS

The **PRUNE_BIND_MOUNTS**, **PRUNEFS**, **PRUNENAMES** and **PRUNEPATHS** variables, which are modified by some of the options, are documented in detail in **updatedb.conf**(5).

−f, **−−add-prunefs** *FS*

Add entries in white-space-separated list *FS* to **PRUNEFS**.

−n, **−−add-prunenames** *NAMES*

Add entries in white-space-separated list *NAMES* to **PRUNENAMES**.

−e, **−−add-prunepaths** *PATHS*

Add entries in white-space-separated list *PATHS* to **PRUNEPATHS**.

−U, **−−database−root** *PATH*

Store only results of scanning the file system subtree rooted at *PATH* to the generated database. The whole file system is scanned by default.

locate(1) outputs entries as absolute path names which don't contain symbolic links, regardless of the form of *PATH*.

−−debug−pruning

Write debugging information about pruning decisions to standard error output.

−h, **−−help**

Write a summary of the available options to standard output and exit successfully.

−o, **−−output** *FILE*

Write the database to *FILE* instead of using the default database.

−−prune−bind−mounts *FLAG*

Set **PRUNE_BIND_MOUNTS** to *FLAG*, overriding the configuration file.

−−prunefs *FS*
Set **PRUNEFS** to *FS*, overriding the configuration file.

−−prunenames *NAMES*
Set **PRUNENAMES** to *NAMES*, overriding the configuration file.

−−prunepaths *PATHS*
Set **PRUNEPATHS** to *PATHS*, overriding the configuration file.

−l, **−−require−visibility** *FLAG*
Set the "require file visibility before reporting it" flag in the generated database to *FLAG*.

If *FLAG* is **0** or **no**, or if the database file is readable by "others" or it is not owned by **mlocate**, **locate**(1) outputs the database entries even if the user running **locate**(1) could not have read the directory necessary to find out the file described by the database entry.

If *FLAG* is **1** or **yes** (the default), **locate**(1) checks the permissions of parent directories of each entry before reporting it to the invoking user. To make the file existence truly hidden from other users, the database group is set to **mlocate** and the database permissions prohibit reading the database by users using other means than **locate**(1), which is set-gid **mlocate**.

Note that the visibility flag is checked only if the database is owned by **mlocate** and it is not readable by "others".

−v, **−−verbose**
Output path names of files to standard output, as soon as they are found.

−V, **−−version**
Write information about the version and license of **locate** on standard output and exit successfully.

EXAMPLES

To create a private mlocate database as an user other than **root**, run

updatedb -l 0 −o *db_file* **−U** *source_directory*

Note that all users that can read *db_file* can get the complete list of files in the subtree of *source_directory*.

FILES

/etc/updatedb.conf
A configuration file. See **updatedb.conf**(5).

/var/lib/mlocate/mlocate.db
The database updated by default.

SECURITY

Databases built with **−−require−visibility no** allow users to find names of files and directories of other users, which they would not otherwise be able to do.

NOTES

The accompanying **locate**(1) utility was designed to be compatible to **slocate** and attempts to be compatible to GNU **locate** where possible. This is not the case for **updatedb**.

AUTHOR

Miloslav Trmac <mitr@redhat.com>

SEE ALSO

locate(1), **mlocate.db**(5), **updatedb.conf**(5)