

NAME

thread-keyring – per-thread keyring

DESCRIPTION

The thread keyring is a keyring used to anchor keys on behalf of a process. It is created only when a thread requests it. The thread keyring has the name (description) *_tid*.

A special serial number value, **KEY_SPEC_THREAD_KEYRING**, is defined that can be used in lieu of the actual serial number of the calling thread's thread keyring.

From the **keyctl(1)** utility, '@t' can be used instead of a numeric key ID in much the same way, but as **keyctl(1)** is a program run after forking, this is of no utility.

Thread keyrings are not inherited across **clone(2)** and **fork(2)** and are cleared by **execve(2)**. A thread keyring is destroyed when the thread that refers to it terminates.

Initially, a thread does not have a thread keyring. If a thread doesn't have a thread keyring when it is accessed, then it will be created if it is to be modified; otherwise the operation fails with the error **ENOKEY**.

SEE ALSO

keyctl(1), **keyctl(3)**, **keyrings(7)**, **persistent-keyring(7)**, **process-keyring(7)**, **session-keyring(7)**, **user-keyring(7)**, **user-session-keyring(7)**

COLOPHON

This page is part of release 5.05 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.