NAME

process-keyring − per-process shared keyring

DESCRIPTION

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

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

From the **keyctl**(1) utility, '**@p**' can be used instead of a numeric key ID in much the same way, but since **keyctl**(1) is a program run after forking, this is of no utility.

A thread created using the **clone**(2) **CLONE_THREAD** flag has the same process keyring as the caller of **clone**(2). When a new process is created using **fork**() it initially has no process keyring. A process's process keyring is cleared on **execve**(2). The process keyring is destroyed when the last thread that refers to it terminates.

If a process doesn't have a process keyring when it is accessed, then the process keyring will be created if the keyring is to be modified; otherwise, the error **ENOKEY** results.

SEE ALSO

keyctl(1), **keyctl**(3), **keyrings**(7), **persistent−keyring**(7), **session−keyring**(7), **thread−keyring**(7), **user−keyring**(7), **user−session−keyring**(7)