NAME

setsid − creates a session and sets the process group ID

LIBRARY

Standard C library (*libc*, *−lc*)

SYNOPSIS

#include <unistd.h>

pid_t setsid(void);

DESCRIPTION

setsid() creates a new session if the calling process is not a process group leader. The calling process is the leader of the new session (i.e., its session ID is made the same as its process ID). The calling process also becomes the process group leader of a new process group in the session (i.e., its process group ID is made the same as its process ID).

The calling process will be the only process in the new process group and in the new session.

Initially, the new session has no controlling terminal. For details of how a session acquires a controlling terminal, see **credentials**(7).

RETURN VALUE

On success, the (new) session ID of the calling process is returned. On error, *(pid_t) −1* is returned, and *errno* is set to indicate the error.

ERRORS

EPERM

The process group ID of any process equals the PID of the calling process. Thus, in particular, **setsid**() fails if the calling process is already a process group leader.

STANDARDS

POSIX.1-2001, POSIX.1-2008, SVr4.

NOTES

A child created via **fork**(2) inherits its parent's session ID. The session ID is preserved across an **execve**(2).

A process group leader is a process whose process group ID equals its PID. Disallowing a process group leader from calling **setsid**() prevents the possibility that a process group leader places itself in a new session while other processes in the process group remain in the original session; such a scenario would break the strict two-level hierarchy of sessions and process groups. In order to be sure that **setsid**() will succeed, call **fork**(2) and have the parent **_exit**(2), while the child (which by definition can't be a process group leader) calls **setsid**().

If a session has a controlling terminal, and the **CLOCAL** flag for that terminal is not set, and a terminal hangup occurs, then the session leader is sent a **SIGHUP** signal.

If a process that is a session leader terminates, then a **SIGHUP** signal is sent to each process in the foreground process group of the controlling terminal.

SEE ALSO

setsid(1), **getsid**(2), **setpgid**(2), **setpgrp**(2), **tcgetsid**(3), **credentials**(7), **sched**(7)