



## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'setuid.3p' command***

***\$ man setuid.3p***

SETUID(3P)                    POSIX Programmer's Manual                    SETUID(3P)

### PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

### NAME

setuid ? create session and set process group ID

### SYNOPSIS

```
#include <unistd.h>

pid_t setuid(void);
```

### DESCRIPTION

The setuid() function shall create a new session, if the calling process is not a process group leader. Upon return the calling process shall be the session leader of this new session, shall be the process group leader of a new process group, and shall have no controlling terminal. The process group ID of the calling process shall be set equal to the process ID of the calling process. The calling process shall be the only process in the new process group and the only process in the new session.

### RETURN VALUE

Upon successful completion, setuid() shall return the value of the new process group ID of the calling process. Otherwise, it shall return -1

and set `errno` to indicate the error.

## ERRORS

The `setsid()` function shall fail if:

**EPERM** The calling process is already a process group leader, or the process group ID of a process other than the calling process matches the process ID of the calling process.

The following sections are informative.

## EXAMPLES

None.

## APPLICATION USAGE

None.

## RATIONALE

The `setsid()` function is similar to the `setpgrp()` function of System V.

System V, without job control, groups processes into process groups and creates new process groups via `setpgrp()`; only one process group may be part of a login session.

Job control allows multiple process groups within a login session. In order to limit job control actions so that they can only affect processes in the same login session, this volume of POSIX.1-2017 adds the concept of a session that is created via `setsid()`. The `setsid()` function also creates the initial process group contained in the session.

Additional process groups can be created via the `setpgid()` function. A System V process group would correspond to a POSIX System Interfaces session containing a single POSIX process group. Note that this function requires that the calling process not be a process group leader.

The usual way to ensure this is true is to create a new process with `fork()` and have it call `setsid()`. The `fork()` function guarantees that the process ID of the new process does not match any existing process group ID.

## FUTURE DIRECTIONS

None.

## SEE ALSO

`getsid()`, `setpgid()`, `setpgrp()`

The Base Definitions volume of POSIX.1-2017, <sys\_types.h>, <unistd.h>

## COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html> .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see [https://www.kernel.org/doc/man-pages/reporting\\_bugs.html](https://www.kernel.org/doc/man-pages/reporting_bugs.html) .

IEEE/The Open Group

2017

SETSID(3P)