

NAME

setuid, setegid – set effective user or group ID

SYNOPSIS

```
#include <sys/types.h>
#include <unistd.h>
```

```
int setuid(uid_t euid);
int setegid(gid_t egid);
```

Feature Test Macro Requirements for glibc (see **feature_test_macros(7)**):

```
setuid(), setegid():
    _POSIX_C_SOURCE >= 200112L
    || /* Glibc versions <= 2.19: */ _BSD_SOURCE
```

DESCRIPTION

setuid() sets the effective user ID of the calling process. Unprivileged processes may only set the effective user ID to the real user ID, the effective user ID or the saved set-user-ID.

Precisely the same holds for **setegid()** with "group" instead of "user".

RETURN VALUE

On success, zero is returned. On error, `-1` is returned, and *errno* is set appropriately.

Note: there are cases where **setuid()** can fail even when the caller is UID 0; it is a grave security error to omit checking for a failure return from **setuid()**.

ERRORS**EINVAL**

The target user or group ID is not valid in this user namespace.

EPERM

In the case of **setuid()**: the calling process is not privileged (does not have the **CAP_SETUID** capability in its user namespace) and *euid* does not match the current real user ID, current effective user ID, or current saved set-user-ID.

In the case of **setegid()**: the calling process is not privileged (does not have the **CAP_SETGID** capability in its user namespace) and *egid* does not match the current real group ID, current effective group ID, or current saved set-group-ID.

CONFORMING TO

POSIX.1-2001, POSIX.1-2008, 4.3BSD.

NOTES

Setting the effective user (group) ID to the saved set-user-ID (saved set-group-ID) is possible since Linux 1.1.37 (1.1.38). On an arbitrary system one should check **_POSIX_SAVED_IDS**.

Under glibc 2.0, **setuid(euid)** is equivalent to **setreuid(-1, euid)** and hence may change the saved set-user-ID. Under glibc 2.1 and later, it is equivalent to **setresuid(-1, euid, -1)** and hence does not change the saved set-user-ID. Analogous remarks hold for **setegid()**, with the difference that the change in implementation from **setregid(-1, egid)** to **setresgid(-1, egid, -1)** occurred in glibc 2.2 or 2.3 (depending on the hardware architecture).

According to POSIX.1, **setuid()** (**setegid()**) need not permit *euid* (*egid*) to be the same value as the current effective user (group) ID, and some implementations do not permit this.

C library/kernel differences

On Linux, **setuid()** and **setegid()** are implemented as library functions that call, respectively, **setreuid(2)** and **setregid(2)**.

SEE ALSO

geteuid(2), **setresuid(2)**, **setreuid(2)**, **setuid(2)**, **capabilities(7)**, **credentials(7)**, **user_namespaces(7)**

COLOPHON

This page is part of release 5.02 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/>.