

**NAME**

getuid, geteuid – get user identity

**SYNOPSIS**

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

uid_t getuid(void);
uid_t geteuid(void);
```

**DESCRIPTION**

**getuid()** returns the real user ID of the calling process.

**geteuid()** returns the effective user ID of the calling process.

**ERRORS**

These functions are always successful.

**CONFORMING TO**

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

**NOTES****History**

In UNIX V6 the **getuid()** call returned  $(euid \ll 8) + uid$ . UNIX V7 introduced separate calls **getuid()** and **geteuid()**.

The original Linux **getuid()** and **geteuid()** system calls supported only 16-bit user IDs. Subsequently, Linux 2.4 added **getuid32()** and **geteuid32()**, supporting 32-bit IDs. The glibc **getuid()** and **geteuid()** wrapper functions transparently deal with the variations across kernel versions.

On Alpha, instead of a pair of **getuid()** and **geteuid()** system calls, a single **getxuid()** system call is provided, which returns a pair of real and effective UIDs. The glibc **getuid()** and **geteuid()** wrapper functions transparently deal with this. See **syscall(2)** for details regarding register mapping.

**SEE ALSO**

**getresuid(2)**, **setreuid(2)**, **setuid(2)**, **credentials(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/>.