

**NAME**

killpg – send signal to a process group

**LIBRARY**

Standard C library (*libc*, *-lc*)

**SYNOPSIS**

```
#include <signal.h>
```

```
int killpg(int pgrp, int sig);
```

Feature Test Macro Requirements for glibc (see **feature\_test\_macros(7)**):

```
killpg():
_XOPEN_SOURCE >= 500
    /* Since glibc 2.19: */ _DEFAULT_SOURCE
    /* glibc <= 2.19: */ _BSD_SOURCE
```

**DESCRIPTION**

**killpg()** sends the signal *sig* to the process group *pgrp*. See **signal(7)** for a list of signals.

If *pgrp* is 0, **killpg()** sends the signal to the calling process's process group. (POSIX says: if *pgrp* is less than or equal to 1, the behavior is undefined.)

For the permissions required to send a signal to another process, see **kill(2)**.

**RETURN VALUE**

On success, zero is returned. On error, *-1* is returned, and *errno* is set to indicate the error.

**ERRORS****EINVAL**

*sig* is not a valid signal number.

**EPERM**

The process does not have permission to send the signal to any of the target processes. For the required permissions, see **kill(2)**.

**ESRCH**

No process can be found in the process group specified by *pgrp*.

**ESRCH**

The process group was given as 0 but the sending process does not have a process group.

**STANDARDS**

POSIX.1-2001, POSIX.1-2008, SVr4, 4.4BSD (**killpg()** first appeared in 4BSD).

**NOTES**

There are various differences between the permission checking in BSD-type systems and System V-type systems. See the POSIX rationale for **kill(3p)**. A difference not mentioned by POSIX concerns the return value **EPERM**: BSD documents that no signal is sent and **EPERM** returned when the permission check failed for at least one target process, while POSIX documents **EPERM** only when the permission check failed for all target processes.

**C library/kernel differences**

On Linux, **killpg()** is implemented as a library function that makes the call *kill(-pgrp, sig)*.

**SEE ALSO**

**getpgrp(2)**, **kill(2)**, **signal(2)**, **capabilities(7)**, **credentials(7)**