

**NAME**

pthread\_sigqueue – queue a signal and data to a thread

**SYNOPSIS**

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

```
#include <pthread.h>
```

```
int pthread_sigqueue(pthread_t thread, int sig,
                    const union sigval value);
```

Compile and link with *-pthread*.

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

```
pthread_sigqueue(): _GNU_SOURCE
```

**DESCRIPTION**

The **pthread\_sigqueue()** function performs a similar task to **sigqueue(3)**, but, rather than sending a signal to a process, it sends a signal to a thread in the same process as the calling thread.

The *thread* argument is the ID of a thread in the same process as the caller. The *sig* argument specifies the signal to be sent. The *value* argument specifies data to accompany the signal; see **sigqueue(3)** for details.

**RETURN VALUE**

On success, **pthread\_sigqueue()** returns 0; on error, it returns an error number.

**ERRORS****EAGAIN**

The limit of signals which may be queued has been reached. (See **signal(7)** for further information.)

**EINVAL**

*sig* was invalid.

**ENOSYS**

**pthread\_sigqueue()** is not supported on this system.

**ESRCH**

*thread* is not valid.

**VERSIONS**

The **pthread\_sigqueue()** function first appeared in glibc 2.11.

**ATTRIBUTES**

For an explanation of the terms used in this section, see **attributes(7)**.

Interface	Attribute	Value
<b>pthread_sigqueue()</b>	Thread safety	MT-Safe

**CONFORMING TO**

This function is a GNU extension.

**NOTES**

The glibc implementation of **pthread\_sigqueue()** gives an error (**EINVAL**) on attempts to send either of the real-time signals used internally by the NPTL threading implementation. See **nptl(7)** for details.

**SEE ALSO**

**rt\_tgsigqueueinfo(2)**, **sigaction(2)**, **pthread\_sigmask(3)**, **sigqueue(3)**, **sigwait(3)**, **pthread(7)**, **signal(7)**

**COLOPHON**

This page is part of release 5.05 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/>.