



Red Hat Enterprise Linux Release 9.2 Manual Pages on 'psiginfo.3p' command

\$ man psiginfo.3p

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

psiginfo, psignal ? write signal information to standard error

SYNOPSIS

```
#include <signal.h>

void psiginfo(const siginfo_t *pinfo, const char *message);

void psignal(int signum, const char *message);
```

DESCRIPTION

The psiginfo() and psignal() functions shall write a language-dependent message associated with a signal number to the standard error stream as follows:

- * First, if message is not a null pointer and is not the empty string, the string pointed to by the message argument shall be written, followed by a <colon> and a <space>.
- * Then the signal description string associated with signum or with the signal indicated by pinfo shall be written, followed by a <new? line>.

For psiginfo(), the application shall ensure that the argument pinfo

references a valid `siginfo_t` structure. For `psignal()`, if `signum` is not a valid signal number, the behavior is implementation-defined.

The `psiginfo()` and `psignal()` functions shall not change the orientation of the standard error stream.

The `psiginfo()` and `psignal()` functions shall mark for update the last data modification and last file status change timestamps of the file associated with the standard error stream at some time between their successful completion and `exit()`, `abort()`, or the completion of `fflush()` or `fclose()` on `stderr`.

The `psiginfo()` and `psignal()` functions shall not change the setting of `errno` if successful.

On error, the `psiginfo()` and `psignal()` functions shall set the error indicator for the stream to which `stderr` points, and shall set `errno` to indicate the error.

Since no value is returned, an application wishing to check for error situations should set `errno` to 0, then call `psiginfo()` or `psignal()`, then check `errno`.

RETURN VALUE

These functions shall not return a value.

ERRORS

Refer to `fputc()`.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

As an alternative to setting `errno` to zero before the call and checking if it is non-zero afterwards, applications can use `ferror()` to detect whether `psiginfo()` or `psignal()` encountered an error.

An application wishing to use this method to check for error situations should call `clearerr(stderr)` before calling `psiginfo()` or `psignal()`, then if `ferror(stderr)` returns non-zero, the value of `errno` indicates which error occurred.

RATIONALE

System V historically has `psignal()` and `psiginfo()` in `<siginfo.h>`. However, the `<siginfo.h>` header is not specified in the Base Definitions volume of POSIX.1-2017, and the type `siginfo_t` is defined in `<signal.h>`.

FUTURE DIRECTIONS

None.

SEE ALSO

`fputc()`, `perror()`, `strsignal()`

The Base Definitions volume of POSIX.1-2017, `<signal.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.