NAME

wait3, wait4 − wait for process to change state, BSD style

LIBRARY

Standard C library (*libc*, *−lc*)

SYNOPSIS

```
#include <sys/wait.h>

pid_t wait3(int *_Nullable wstatus, int options,
        struct rusage *_Nullable rusage);
pid_t wait4(pid_t pid, int *_Nullable wstatus, int options,
        struct rusage *_Nullable rusage);
```

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

wait3():
- Since glibc 2.26:
  - _DEFAULT_SOURCE
    || (_XOPEN_SOURCE >= 500 &&
    ! (_POSIX_C_SOURCE >= 200112L
    || _XOPEN_SOURCE >= 600))
- From glibc 2.19 to glibc 2.25:
  - _DEFAULT_SOURCE || _XOPEN_SOURCE >= 500
- glibc 2.19 and earlier:
  - _BSD_SOURCE || _XOPEN_SOURCE >= 500

wait4():
- Since glibc 2.19:
  - _DEFAULT_SOURCE
- glibc 2.19 and earlier:
  - _BSD_SOURCE

DESCRIPTION

These functions are nonstandard; in new programs, the use of **waitpid**(2) or **waitid**(2) is preferable.

The **wait3**() and **wait4**() system calls are similar to **waitpid**(2), but additionally return resource usage information about the child in the structure pointed to by *rusage*.

Other than the use of the *rusage* argument, the following **wait3**() call:

```
wait3(wstatus, options, rusage);
```

is equivalent to:

```
waitpid(−1, wstatus, options);
```

Similarly, the following **wait4**() call:

```
wait4(pid, wstatus, options, rusage);
```

is equivalent to:

```
waitpid(pid, wstatus, options);
```

In other words, **wait3**() waits of any child, while **wait4**() can be used to select a specific child, or children, on which to wait. See**wait**(2) for further details.

If *rusage* is not NULL, the *struct rusage* to which it points will be filled with accounting information about the child. See **getrusage**(2) for details.

RETURN VALUE

As for **waitpid**(2).

ERRORS

As for **waitpid**(2).

STANDARDS

4.3BSD.

SUSv1 included a specification of **wait3**(); SUSv2 included **wait3**(), but marked it LEGACY; SUSv3 removed it.

NOTES

Including <*sys/time.h*> is not required these days, but increases portability. (Indeed,<*sys/r esource.h*> defines the *rusage* structure with fields of type *struct timeval* defined in <*sys/time.h*>.)

C library/kernel differences

On Linux, **wait3**() is a library function implemented on top of the **wait4**() system call.

SEE ALSO

fork(2), **getrusage**(2), **sigaction**(2), **signal**(2), **wait**(2), **signal**(7)