NAME

abort2 - abort process with diagnostics

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <stdlib.h>

void
abort2(*const char *why*, *int nargs*, *void **args*);

DESCRIPTION

The **abort2**() system call causes the process to be killed and the specified diagnostic message (with arguments) to be delivered by the kernel to the syslogd(8) daemon.

The *why* argument points to a NUL-terminated string specifying a reason of the program's termination (maximum 128 characters long). The *args* array contains pointers which will be logged numerically (with the kernel's '%p' printf(9) format). The *nargs* argument specifies the number of pointers in *args* (maximum 16).

The **abort2**() system call is intended for use in situations where continuation of a process is impossible or for other definitive reasons is unwanted, and normal diagnostic channels cannot be trusted to deliver the message.

RETURN VALUES

The **abort2**() function never returns.

The process is killed with SIGABRT unless the arguments to **abort2**() are invalid, in which case SIGKILL is used.

EXAMPLES

```
#include <stdlib.h>

if (weight_kg > max_load) {
        void *ptrs[3];

        ptrs[0] = (void *)(intptr_t)weight_kg;
        ptrs[1] = (void *)(intptr_t)max_load;
        ptrs[2] = haystack;
```

```
        abort2("Camel overloaded", 3, ptrs);
}
```

SEE ALSO

abort(3), exit(3)

HISTORY

The **abort2**() system call first appeared in FreeBSD 7.0.

AUTHORS

The **abort2**() system call was designed by Poul-Henning Kamp <*phk@FreeBSD.org*>. It was implemented by Wojciech A. Koszek <*dunstan@freebsd.czest.pl*>.