



Rocky Enterprise Linux 9.2 Manual Pages on command 'verrx.3'

C:\>man verrx.3

ERR(3) Linux Programmer's Manual ERR(3)

NAME

err, verr, errx, verrx, warn, vwarn, warnx, vwarnx - formatted error messages

SYNOPSIS

```
#include <err.h>

void err(int eval, const char *fmt, ...);
void errx(int eval, const char *fmt, ...);
void warn(const char *fmt, ...);
void warnx(const char *fmt, ...);

#include <stdarg.h>

void verr(int eval, const char *fmt, va_list args);
void verrx(int eval, const char *fmt, va_list args);
void vwarn(const char *fmt, va_list args);
void vwarnx(const char *fmt, va_list args);
```

DESCRIPTION

The `err()` and `warn()` family of functions display a formatted error message on the standard error output. In all cases, the last component of the program name, a colon character, and a space are output. If the `fmt` argument is not `NULL`, the `printf(3)`-like formatted error message is output. The output is terminated by a newline character.

The `err()`, `verr()`, `warn()`, and `vwarn()` functions append an error message obtained from `strerror(3)` based on the global variable `errno`, preceded by another colon and

space unless the fmt argument is NULL.

The errx() and warnx() functions do not append an error message.

The err(), verr(), errx(), and verrx() functions do not return, but exit with the value of the argument eval.

ATTRIBUTES

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

??

?Interface ? Attribute ? Value ?

??

?err(), errx(), ? Thread safety ? MT-Safe locale ?

?warn(), warnx(), ? ? ?

?verr(), verrx(), ? ? ?

?vwarn(), vwarnx() ? ? ?

??

CONFORMING TO

These functions are nonstandard BSD extensions.

EXAMPLE

Display the current errno information string and exit:

```
p = malloc(size);
if (p == NULL)
    err(1, NULL);
fd = open(file_name, O_RDONLY, 0);
if (fd == -1)
    err(1, "%s", file_name);
```

Display an error message and exit:

```
if (tm.tm_hour < START_TIME)
    errx(1, "too early, wait until %s", start_time_string);
```

Warn of an error:

```
fd = open(raw_device, O_RDONLY, 0);
if (fd == -1)
    warnx("%s: %s: trying the block device",
        raw_device, strerror(errno));
fd = open(block_device, O_RDONLY, 0);
```

```
if (fd == -1)
```

```
    err(1, "%s", block_device);
```

SEE ALSO

error(3), exit(3), perror(3), printf(3), strerror(3)

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/>.

Linux

2017-09-15

ERR(3)