



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

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

\$ man clearenv.3

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

NAME

clearenv - clear the environment

SYNOPSIS

```
#include <stdlib.h>

int clearenv(void);
```

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

```
clearenv():

/* Glibc since 2.19: */ _DEFAULT_SOURCE

|| /* Glibc versions <= 2.19: */ _SVID_SOURCE || _BSD_SOURCE
```

DESCRIPTION

The clearenv() function clears the environment of all name-value pairs and sets the value of the external variable environ to NULL. After this call, new variables can be added to the environment using putenv(3) and setenv(3).

RETURN VALUE

The clearenv() function returns zero on success, and a nonzero value on failure.

VERSIONS

Available since glibc 2.0.

ATTRIBUTES

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

????????????????????????????????????????????????????????????????

?Interface ? Attribute ? Value ?

????????????????????????????????????????????????????????????????

?clearenv() ? Thread safety ? MT-Unsafe const:env ?

????????????????????????????????????????????????????????????

CONFORMING TO

Various UNIX variants (DG/UX, HP-UX, QNX, ...). POSIX.9 (bindings for FORTRAN77). POSIX.1-1996 did not accept clearenv() and putenv(3), but changed its mind and scheduled these functions for some later issue of this standard (see ?B.4.6.1). However, POSIX.1-2001 adds only putenv(3), and rejected clearenv().

NOTES

On systems where clearenv() is unavailable, the assignment

```
environ = NULL;
```

will probably do.

The clearenv() function may be useful in security-conscious applications that want to precisely control the environment that is passed to programs executed using exec(3). The application would do this by first clearing the environment and then adding selected environment variables.

Note that the main effect of clearenv() is to adjust the value of the pointer environ(7); this function does not erase the contents of the buffers containing the environment definitions.

The DG/UX and Tru64 man pages write: If environ has been modified by anything other than the putenv(3), getenv(3), or clearenv() functions, then clearenv() will return an error and the process environment will remain unchanged.

SEE ALSO

getenv(3), putenv(3), setenv(3), unsetenv(3), environ(7)

COLOPHON

This page is part of release 5.10 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/>.