



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 'rmdir.2'

\$ man rmdir.2

RMDIR(2) Linux Programmer's Manual RMDIR(2)

NAME

rmdir - delete a directory

SYNOPSIS

```
#include <unistd.h>
```

```
int rmdir(const char *pathname);
```

DESCRIPTION

rmdir() deletes a directory, which must be empty.

RETURN VALUE

On success, zero is returned. On error, -1 is returned, and errno is set appropriately.

ERRORS

EACCES Write access to the directory containing pathname was not allowed, or one of the directories in the path prefix of pathname did not allow search permission. (See also path_resolution(7)).

EBUSY pathname is currently in use by the system or some process that prevents its removal. On Linux, this means pathname is currently used as a mount point or is the root directory of the calling process.

EFAULT pathname points outside your accessible address space.

EINVAL pathname has . as last component.

ELOOP Too many symbolic links were encountered in resolving pathname.

ENAMETOOLONG

pathname was too long.

ENOENT A directory component in pathname does not exist or is a dangling symbolic link.

ENOMEM Insufficient kernel memory was available.

ENOTDIR

pathname, or a component used as a directory in pathname, is not, in fact, a directory.

ENOTEMPTY

pathname contains entries other than . and .. ; or, pathname has .. as its final component. POSIX.1 also allows EEXIST for this condition.

EPERM The directory containing pathname has the sticky bit (S_ISVTX) set and the process's effective user ID is neither the user ID of the file to be deleted nor that of the directory containing it, and the process is not privileged (Linux: does not have the CAP_FOWNER capability).

EPERM The filesystem containing pathname does not support the removal of directories.

EROFS pathname refers to a directory on a read-only filesystem.

CONFORMING TO

POSIX.1-2001, POSIX.1-2008, SVr4, 4.3BSD.

BUGS

Infelicities in the protocol underlying NFS can cause the unexpected disappearance of directories which are still being used.

SEE ALSO

rm(1), rmdir(1), chdir(2), chmod(2), mkdir(2), rename(2), unlink(2), unlinkat(2)

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/>.