



*Full credit is given to the above companies including the OS that this PDF file was generated!*

## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'chdir.3p' command***

### ***\$ man chdir.3p***

CHDIR(3P)                    POSIX Programmer's Manual                    CHDIR(3P)

### PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

### NAME

chdir ? change working directory

### SYNOPSIS

```
#include <unistd.h>

int chdir(const char *path);
```

### DESCRIPTION

The `chdir()` function shall cause the directory named by the `pathname` pointed to by the `path` argument to become the current working directory; that is, the starting point for path searches for `pathnames` not beginning with `'/'`.

### RETURN VALUE

Upon successful completion, 0 shall be returned. Otherwise, -1 shall be returned, the current working directory shall remain unchanged, and `errno` shall be set to indicate the error.

### ERRORS

The `chdir()` function shall fail if:

EACCES Search permission is denied for any component of the `pathname`.

**ELOOP** A loop exists in symbolic links encountered during resolution of the path argument.

#### **ENAMETOOLONG**

The length of a component of a pathname is longer than {NAME\_MAX}.

**ENOENT** A component of path does not name an existing directory or path is an empty string.

#### **ENOTDIR**

A component of the pathname names an existing file that is neither a directory nor a symbolic link to a directory.

The `chdir()` function may fail if:

**ELOOP** More than {SYMLOOP\_MAX} symbolic links were encountered during resolution of the path argument.

#### **ENAMETOOLONG**

The length of a pathname exceeds {PATH\_MAX}, or pathname resolution of a symbolic link produced an intermediate result with a length that exceeds {PATH\_MAX}.

The following sections are informative.

### **EXAMPLES**

#### **Changing the Current Working Directory**

The following example makes the value pointed to by `directory`, `/tmp`, the current working directory.

```
#include <unistd.h>

...

char *directory = "/tmp";

int ret;

ret = chdir (directory);
```

### **APPLICATION USAGE**

None.

### **RATIONALE**

The `chdir()` function only affects the working directory of the current process.

### **FUTURE DIRECTIONS**

None.

## SEE ALSO

`getcwd()`

The Base Definitions volume of POSIX.1?2017, <unistd.h>

## COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html> .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see [https://www.kernel.org/doc/man-pages/reporting\\_bugs.html](https://www.kernel.org/doc/man-pages/reporting_bugs.html) .

IEEE/The Open Group

2017

CHDIR(3P)