

NAME

chroot - change root directory

SYNOPSIS

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

```
int
```

```
chroot(const char *dirname);
```

DESCRIPTION

Dirname is the address of the pathname of a directory, terminated by an ASCII NUL. **Chroot()** causes dirname to become the root directory, that is, the starting point for path searches of pathnames beginning with `/'.

In order for a directory to become the root directory a process must have execute (search) access for that directory.

If the program is not currently running with an altered root directory, it should be noted that **chroot()** has no effect on the process's current directory.

If the program is already running with an altered root directory, the process's current directory is changed to the same new root directory. This prevents the current directory from being further up the directory tree than the altered root directory.

This call is restricted to the super-user.

RETURN VALUES

Upon successful completion, a value of 0 is returned. Otherwise, a value of -1 is returned and errno is set to indicate an error.

ERRORS

Chroot() will fail and the root directory will be unchanged if:

- | | |
|----------------|--|
| [ENOTDIR] | A component of the path name is not a directory. |
| [ENAMETOOLONG] | A component of a pathname exceeded {NAME_MAX} characters, or an entire path name exceeded {PATH_MAX} characters. |
| [ENOENT] | The named directory does not exist. |
| [EACCES] | Search permission is denied for any component of the path name. |
| [ELOOP] | Too many symbolic links were encountered in translating the pathname. |

[EFAULT] Path points outside the process's allocated address space.

[EIO] An I/O error occurred while reading from or writing to the file system.

SEE ALSO

chdir(2)

WARNINGS

There are ways for a root process to escape from the chroot jail.

HISTORY

The **chroot()** function call appeared in 4.2BSD.

4.2 Berkeley Distribution

June 4, 1993

4.2 Berkeley Distribution