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 'gethostname.2' command

$ man gethostname.2

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

NAME

gethostname, sethostname - get/set hostname

SYNOPSIS

#include <unistd.h>

int gethostname(char *name, size_t len);

int sethostname(const char *name, size_t len);

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

gethostname():

Since glibc 2.12: _BSD_SOURCE || _XOPEN_SOURCE >= 500

|| /* Since glibc 2.12: */ _POSIX_C_SOURCE >= 200112L

sethostname():

Since glibc 2.21:

_DEFAULT_SOURCE

In glibc 2.19 and 2.20:

_DEFAULT_SOURCE || (_XOPEN_SOURCE && _XOPEN_SOURCE < 500)

Up to and including glibc 2.19:

_BSD_SOURCE || (_XOPEN_SOURCE && _XOPEN_SOURCE < 500)

DESCRIPTION

These system calls are used to access or to change the system hostname.

More precisely, they operate on the hostname associated with the call?

ing process's UTS namespace.

sethostname() sets the hostname to the value given in the character ar?

ray name. The len argument specifies the number of bytes in name. (Thus, name does not require a terminating null byte.)

gethostname() returns the null-terminated hostname in the character ar? ray name, which has a length of len bytes. If the null-terminated hostname is too large to fit, then the name is truncated, and no error is returned (but see NOTES below). POSIX.1 says that if such trunca? tion occurs, then it is unspecified whether the returned buffer in? cludes a terminating null byte.

RETURN VALUE

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

ERRORS

EFAULT name is an invalid address.

EINVAL len is negative or, for sethostname(), len is larger than the maximum allowed size.

ENAMETOOLONG

(glibc gethostname()) len is smaller than the actual size. (Be? fore version 2.1, glibc uses EINVAL for this case.)

EPERM For sethostname(), the caller did not have the CAP_SYS_ADMIN ca? pability in the user namespace associated with its UTS namespace (see namespaces(7)).

CONFORMING TO

SVr4, 4.4BSD (these interfaces first appeared in 4.2BSD). POSIX.1-2001 and POSIX.1-2008 specify gethostname() but not sethost? name().

NOTES

SUSv2 guarantees that "Host names are limited to 255 bytes". POSIX.1 guarantees that "Host names (not including the terminating null byte) are limited to HOST_NAME_MAX bytes". On Linux, HOST_NAME_MAX is de? fined with the value 64, which has been the limit since Linux 1.0 (ear? lier kernels imposed a limit of 8 bytes).

C library/kernel differences

The GNU C library does not employ the gethostname() system call; in?

stead, it implements gethostname() as a library function that calls un?
ame(2) and copies up to len bytes from the returned nodename field into
name. Having performed the copy, the function then checks if the
length of the nodename was greater than or equal to len, and if it is,
then the function returns -1 with errno set to ENAMETOOLONG; in this
case, a terminating null byte is not included in the returned name.
Versions of glibc before 2.2 handle the case where the length of the
nodename was greater than or equal to len differently: nothing is
copied into name and the function returns -1 with errno set to ENAME?
TOOLONG.

SEE ALSO

hostname(1), getdomainname(2), setdomainname(2), uname(2), uts_name?
spaces(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/.

Linux 2019-10-10 GETHOSTNAME(2)