NAME

send, sendto, sendmsg – send a message from a socket

SYNOPSIS

cc [*flag* ...] *file* ... **–lsocket –lnsl** [*library* ...]

#include <sys/types.h>
#include <sys/socket.h>

int send(int *s*, **const char** **msg*, **int** *len*, **int** *flags***);**

int sendto(int *s*, **const char** **msg*, **int** *len*, **int** *flags*, **const struct sockaddr** **to*,
int *tolen***);**

int sendmsg(int *s*, **const struct msghdr** **msg*, **int** *flags***);**

MT-LEVEL

Safe

DESCRIPTION

send( ), **sendto( )**, and **sendmsg( )** are used to transmit a message to another transport end-point. **send( )** may be used only when the socket is in a *connected* state, while **sendto( )** and **sendmsg( )** may be used at any time. *s* is a socket created with **socket**(3N).

The address of the target is given by *to* with *tolen* specifying its size. The length of the message is given by *len*. If the message is too long to pass atomically through the underlying protocol, then the error **EMSGSIZE** is returned, and the message is not transmitted.

A return value of –1 indicates locally detected errors only. It does not implicitly mean the message was not delivered.

If the socket does not have enough buffer space available to hold the message being sent, **send( )** blocks, unless the socket has been placed in non-blocking I/O mode (see **fcntl**(2)). The **select**(3C) or **poll**(2) call may be used to determine when it is possible to send more data.

The *flags* parameter is formed from the bitwise OR of zero or more of the following:

MSG_OOB Send "out-of-band" data on sockets that support this notion. The underlying protocol must also support "out-of-band" data. Only **SOCK_STREAM** sockets created in the **AF_INET** address family support out-of-band data.

MSG_DONTROUTE The **SO_DONTROUTE** option is turned on for the duration of the operation. It is used only by diagnostic or routing programs.

See **recv**(3N) for a description of the **msghdr** structure.

RETURN VALUES

These calls return the number of bytes sent, or –1 if an error occurred.

ERRORS

The calls fail if:

EBADF *s* is an invalid file descriptor.

EINTR The operation was interrupted by delivery of a signal before any data could be buffered to be sent.

EINVAL *tolen* is not the size of a valid address for the specified address family.

EMSGSIZE The socket requires that message be sent atomically, and the message was too long.

ENOMEM There was insufficient memory available to complete the operation.

ENOSR There were insufficient STREAMS resources available for the operation to complete.

ENOTSOCK *s* is not a socket.

EWOULDBLOCK The socket is marked non-blocking and the requested operation would block.

SEE ALSO

fcntl(2), **poll**(2), **write**(2), **connect**(3N), **getsockopt**(3N), **recv**(3N), **select**(3C), **socket**(3N)