



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 'setsockopt.3p' command

\$ man setsockopt.3p

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

setsockopt ? set the socket options

SYNOPSIS

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

int setsockopt(int socket, int level, int option_name,
               const void *option_value, socklen_t option_len);
```

DESCRIPTION

The `setsockopt()` function shall set the option specified by the `option_name` argument, at the protocol level specified by the `level` argument, to the value pointed to by the `option_value` argument for the socket associated with the file descriptor specified by the `socket` argument.

The `level` argument specifies the protocol level at which the option resides. To set options at the socket level, specify the `level` argument as `SOL_SOCKET`. To set options at other levels, supply the appropriate level identifier for the protocol controlling the option. For example, to indicate that an option is interpreted by the TCP (Transport Control

Protocol), set level to IPPROTO_TCP as defined in the <netinet/in.h> header.

The option_name argument specifies a single option to set. It can be one of the socket-level options defined in <sys_socket.h> and described in Section 2.10.16, Use of Options. If option_name is equal to SO_RCVTIMEO or SO_SNDTIMEO and the implementation supports setting the option, it is unspecified whether the struct timeval pointed to by option_value is stored as provided by this function or is rounded up to align with the resolution of the clock being used. If setsockopt() is called with option_name equal to SO_ACCEPTCONN, SO_ERROR, or SO_TYPE, the behavior is unspecified.

RETURN VALUE

Upon successful completion, setsockopt() shall return 0. Otherwise, -1 shall be returned and errno set to indicate the error.

ERRORS

The setsockopt() function shall fail if:

EBADF The socket argument is not a valid file descriptor.

EDOM The send and receive timeout values are too big to fit into the timeout fields in the socket structure.

EINVAL The specified option is invalid at the specified socket level or the socket has been shut down.

EISCONN

The socket is already connected, and a specified option cannot be set while the socket is connected.

ENOPROTOOPT

The option is not supported by the protocol.

ENOTSOCK

The socket argument does not refer to a socket.

The setsockopt() function may fail if:

ENOMEM There was insufficient memory available for the operation to complete.

ENOBUFS

Insufficient resources are available in the system to complete

the call.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

The `setsockopt()` function provides an application program with the means to control socket behavior. An application program can use `setsockopt()` to allocate buffer space, control timeouts, or permit socket data broadcasts. The `<sys/socket.h>` header defines the socket-level options available to `setsockopt()`.

Options may exist at multiple protocol levels. The `SO_` options are always present at the uppermost socket level.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.10, Sockets, `bind()`, `endprotoent()`, `getsockopt()`, `socket()`

The Base Definitions volume of POSIX.1-2017, `<netinet_in.h>`, `<sys_socket.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.ker?>

