



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 'getsockopt.3p' command

\$ man getsockopt.3p

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

getsockopt ? get the socket options

SYNOPSIS

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

int getsockopt(int socket, int level, int option_name,
               void *restrict option_value, socklen_t *restrict option_len);
```

DESCRIPTION

The `getsockopt()` function manipulates options associated with a socket.

The `getsockopt()` function shall retrieve the value for the option specified by the `option_name` argument for the socket specified by the `socket` argument. If the size of the option value is greater than `option_len`, the value stored in the object pointed to by the `option_value` argument shall be silently truncated. Otherwise, the object pointed to by the `option_len` argument shall be modified to indicate the actual length of the value.

The `level` argument specifies the protocol level at which the option resides. To retrieve options at the socket level, specify the level argu?

ment as SOL_SOCKET. To retrieve 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 (Transmission Control Protocol), set level to IPPROTO_TCP as defined in the <netinet/in.h> header.

The socket in use may require the process to have appropriate privileges to use the getsockopt() function.

The option_name argument specifies a single option to be retrieved. It can be one of the socket-level options defined in <sys_socket.h> and described in Section 2.10.16, Use of Options.

RETURN VALUE

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

ERRORS

The getsockopt() function shall fail if:

EBADF The socket argument is not a valid file descriptor.

EINVAL The specified option is invalid at the specified socket level.

ENOPROTOOPT

The option is not supported by the protocol.

ENOTSOCK

The socket argument does not refer to a socket.

The getsockopt() function may fail if:

EACCES The calling process does not have appropriate privileges.

EINVAL The socket has been shut down.

ENOBUFS

Insufficient resources are available in the system to complete the function.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.10.16, Use of Options, `bind()`, `close()`, `endprotoent()`, `set?`

`sockopt()`, `socket()`

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

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Por? table Operating System Interface (POSIX), The Open Group Base Specifi? cations 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? nel.org/doc/man-pages/reporting_bugs.html .

IEEE/The Open Group

2017

GETSOCKOPT(3P)