

**NAME**

**udplite** - Lightweight User Datagram Protocol

**SYNOPSIS**

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/udplite.h>
```

*int*

```
socket(AF_INET, SOCK_DGRAM, IPPROTO_UDPLITE);
```

**DESCRIPTION**

The UDP-Lite protocol provides a partial checksum which allows corrupted packets to be transmitted to the receiving application. This has advantages for some types of multimedia transport that may be able to make use of slightly damaged datagrams, rather than having them discarded by lower-layer protocols.

UDP-Lite supports a number of socket options which can be set with `setsockopt(2)` and tested with `getsockopt(2)`:

**UDPLITE\_SEND\_CSCOV** This option sets the sender checksum coverage. A value of zero indicates that all sent packets will have full checksum coverage. A value of 8 to 65535 limits the checksum coverage of all sent packets to the value given.

**UDPLITE\_RECV\_CSCOV**

This option is the receiver-side analogue. A value of zero instructs the kernel to drop all received packets not having full checksum coverage. A value of 8 to 65535 instructs the kernel to drop all received packets with a partial checksum coverage smaller than the value specified.

**ERRORS**

A socket operation may fail with one of the following errors returned:

[EISCONN]           when trying to establish a connection on a socket which already has one, or when trying to send a datagram with the destination address specified and the socket is already connected;

[ENOTCONN]         when trying to send a datagram, but no destination address is specified, and the socket has not been connected;

[ENOBUFS]           when the system runs out of memory for an internal data structure;

[EADDRINUSE] when an attempt is made to create a socket with a port which has already been allocated;

[EADDRNOTAVAIL] when an attempt is made to create a socket with a network address for which no network interface exists.

**SEE ALSO**

getsockopt(2), recv(2), send(2), socket(2)