

NAME

sctp_bindx - bind or unbind an SCTP socket to a list of addresses

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

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

int

```
sctp_bindx(int s, struct sockaddr *addrs, int num, int type);
```

DESCRIPTION

The **sctp_bindx()** call binds or unbinds a address or a list of addresses to an SCTP endpoint. This allows a user to bind a subset of addresses. The **sctp_bindx()** call operates similarly to **bind()** but allows a list of addresses and also allows a bind or an unbind. The argument *s* must be a valid SCTP socket descriptor. The argument *addrs* is a list of addresses (where the list may be only 1 in length) that the user wishes to bind or unbind to the socket. The argument *type* must be one of the following values.

SCTP_BINDX_ADD_ADDR This value indicates that the listed address(es) need to be added to the endpoint.

SCTP_BINDX_REM_ADDR This value indicates that the listed address(es) need to be removed from the endpoint.

Note that when a user adds or deletes an address to an association if the dynamic address flag *net.inet.sctp.auto_asconf* is enabled any associations in the endpoint will attempt to have the address(es) added dynamically to the existing association.

RETURN VALUES

The call returns 0 on success and -1 upon failure.

ERRORS

The **sctp_bindx()** function can return the following errors:

[EINVAL]	This value is returned if the <i>type</i> field is not one of the allowed values (see above).
----------	---

[ENOMEM] This value is returned if the number of addresses being added causes a memory allocation failure in the call.

[EBADF] The argument *s* is not a valid descriptor.

[ENOTSOCK] The argument *s* is not a socket.

SEE ALSO

bind(2), sctp(4)