NAME

mbchain, **mb_init**, **mb_initm**, **mb_done**, **mb_detach**, **mb_fixhdr**, **mb_reserve**, **mb_put_uint8**, **mb_put_uint16be**, **mb_put_uint16le**, **mb_put_uint32be**, **mb_put_uint32le**, **mb_put_int64be**, **mb_put_int64le**, **mb_put_mem**, **mb_put_mbuf**, **mb_put_uio** - set of functions to build an mbuf chain from various data types

SYNOPSIS

options LIBMCHAIN kldload libmchain

#include <sys/param.h>
#include <sys/uio.h>
#include <sys/mchain.h>

int
mb_init(*struct mbchain *mbp*);

void
mb_initm(*struct mbchain *mbp, struct mbuf *m*);

void
mb_done(*struct mbchain *mbp*);

*struct mbuf **
mb_detach(*struct mbchain *mbp*);

int
mb_fixhdr(*struct mbchain *mbp*);

caddr_t
mb_reserve(*struct mbchain *mbp, int size*);

int
mb_put_uint8(*struct mbchain *mbp, uint8_t x*);

int
mb_put_uint16be(*struct mbchain *mbp, uint16_t x*);

int
mb_put_uint16le(*struct mbchain *mbp, uint16_t x*);

int
mb_put_uint32be(*struct mbchain *mbp, uint32_t x*);

int
mb_put_uint32le(*struct mbchain *mbp, uint32_t x*);

int
mb_put_int64be(*struct mbchain *mbp, int64_t x*);

int
mb_put_int64le(*struct mbchain *mbp, int64_t x*);

int
mb_put_mem(*struct mbchain *mbp, c_caddr_t source, int size, int type*);

int
mb_put_mbuf(*struct mbchain *mbp, struct mbuf *m*);

int
mb_put_uio(*struct mbchain *mbp, struct uio *uiop, int size*);

DESCRIPTION

These functions are used to compose mbuf chains from various data types. The *mbchain* structure is used as a working context and should be initialized with a call to either **mb_init**() or **mb_initm**(). It has the following fields:

mb_top (*struct mbuf* *) A pointer to the top of constructed mbuf chain.

mb_cur (*struct mbuf* *) A pointer to the currently filled mbuf.

mb_mleft (*int*) Number of bytes left in the current mbuf.

mb_count (*int*) Total number of bytes placed in the mbuf chain.

mb_copy (*mb_copy_t* *) User-defined function to perform a copy into mbuf; useful if any unusual data conversion is necessary.

mb_udata (*void* *) User-supplied data which can be used in the *mb_copy* function.

mb_done() function disposes an mbuf chain pointed to by *mbp->mb_top* field and sets the field to

NULL.

mb_detach() function returns the value of *mbp->mb_top* field and sets its value to NULL.

mb_fixhdr() recalculates the length of an mbuf chain and updates the *m_pkthdr.len* field of the first mbuf in the chain. It returns the calculated length.

mb_reserve() ensures that the object of the length specified by the *size* argument will fit in the current mbuf (mbuf allocation is performed if necessary), and advances all pointers as if the real data was placed. Returned value will point to the beginning of the reserved space. Note that the size of the object should not exceed MLEN bytes.

All **mb_put_*****() functions perform an actual copy of the data into mbuf chain. Functions which have **le** or **be** suffixes will perform conversion to the little- or big-endian data formats.

mb_put_mem() function copies *size* bytes of data specified by the *source* argument to an mbuf chain. The *type* argument specifies the method used to perform a copy, and can be one of the following:

MB_MSYSTEM
: Use **bcopy**() function.

MB_MUSER
: Use copyin(9) function.

MB_MINLINE
: Use an "inline" loop which does not call any function.

MB_MZERO
: Do not copy any data, but just fill the destination with zero bytes.

MB_MCUSTOM
: Call function specified by the *mbp->mb_copy* field.

RETURN VALUES

All *int* functions except **mb_fixhdr**() return zero if successful and an error code otherwise.

Note: after failure of any function, an mbuf chain is left in the broken state, and only **mb_done**() function can safely be called to destroy it.

EXAMPLES

```
struct mbchain *mbp;
struct mbuf *m;
```

```
mb_init(mbp);
mb_put_uint8(mbp, 33);
mb_put_uint16le(mbp, length);
m = m_copym(mbp->mb_top, 0, M_COPYALL, M_WAIT);
send(m);
mb_done(mbp);
```

SEE ALSO

mbuf(9), mdchain(9)

AUTHORS

This manual page was written by Boris Popov <*bp@FreeBSD.org*>.