

**NAME**

BIO\_find\_type, BIO\_next, BIO\_method\_type - BIO chain traversal

**SYNOPSIS**

```
#include <openssl/bio.h>

BIO * BIO_find_type(BIO *b,int bio_type);
BIO * BIO_next(BIO *b);

#define BIO_method_type(b) ((b)->method->type)

#define BIO_TYPE_NONE 0
#define BIO_TYPE_MEM (1|0x0400)
#define BIO_TYPE_FILE (2|0x0400)

#define BIO_TYPE_FD (4|0x0400|0x0100)
#define BIO_TYPE_SOCKET (5|0x0400|0x0100)
#define BIO_TYPE_NULL (6|0x0400)
#define BIO_TYPE_SSL (7|0x0200)
#define BIO_TYPE_MD (8|0x0200)
#define BIO_TYPE_BUFFER (9|0x0200)
#define BIO_TYPE_CIPHER (10|0x0200)
#define BIO_TYPE_BASE64 (11|0x0200)
#define BIO_TYPE_CONNECT (12|0x0400|0x0100)
#define BIO_TYPE_ACCEPT (13|0x0400|0x0100)
#define BIO_TYPE_PROXY_CLIENT (14|0x0200)
#define BIO_TYPE_PROXY_SERVER (15|0x0200)
#define BIO_TYPE_NBIO_TEST (16|0x0200)
#define BIO_TYPE_NULL_FILTER (17|0x0200)
#define BIO_TYPE_BER (18|0x0200)
#define BIO_TYPE_BIO (19|0x0400)

#define BIO_TYPE_DESCRIPTOR 0x0100
#define BIO_TYPE_FILTER 0x0200
#define BIO_TYPE_SOURCE_SINK 0x0400
```

**DESCRIPTION**

The *BIO\_find\_type()* searches for a BIO of a given type in a chain, starting at BIO **b**. If **type** is a specific type (such as **BIO\_TYPE\_MEM**) then a search is made for a BIO of that type. If **type** is a general type (such as **BIO\_TYPE\_SOURCE\_SINK**) then the next matching BIO of the given general type is searched for. *BIO\_find\_type()* returns the next matching BIO or NULL if none is found.

Note: not all the **BIO\_TYPE\_\*** types above have corresponding BIO implementations.

*BIO\_next()* returns the next BIO in a chain. It can be used to traverse all BIOs in a chain or used in conjunction with *BIO\_find\_type()* to find all BIOs of a certain type.

*BIO\_method\_type()* returns the type of a BIO.

**RETURN VALUES**

*BIO\_find\_type()* returns a matching BIO or NULL for no match.

*BIO\_next()* returns the next BIO in a chain.

*BIO\_method\_type()* returns the type of the BIO **b**.

**NOTES**

*BIO\_next()* was added to OpenSSL 0.9.6 to provide a 'clean' way to traverse a BIO chain or find multiple matches using *BIO\_find\_type()*. Previous versions had to use:

```
next = bio->next_bio;
```

**BUGS**

*BIO\_find\_type()* in OpenSSL 0.9.5a and earlier could not be safely passed a NULL pointer for the **b** argument.

**EXAMPLE**

Traverse a chain looking for digest BIOs:

```
BIO *btmp;
btmp = in_bio; /* in_bio is chain to search through */

do {
    btmp = BIO_find_type(btmp, BIO_TYPE_MD);
    if(btmp == NULL) break; /* Not found */
    /* btmp is a digest BIO, do something with it ...*/
    ...

    btmp = BIO_next(btmp);
} while(btmp);
```

**SEE ALSO**

TBA