

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);
int BIO_method_type(const BIO *b);
```

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. If *type* is **BIO_TYPE_NONE** it will not find a match.

The following general types are defined: **BIO_TYPE_DESCRIPTOR**, **BIO_TYPE_FILTER**, and **BIO_TYPE_SOURCE_SINK**.

For a list of the specific types, see the `<openssl/bio.h>` header file.

BIO_next() returns the next **BIO** in a chain. It can be used to traverse all **BIO**s in a chain or used in conjunction with **BIO_find_type()** to find all **BIO**s 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*.

EXAMPLES

Traverse a chain looking for digest **BIO**s:

```
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);
```

COPYRIGHT

Copyright 2000–2024 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy in the file LICENSE in the source distribution or at <https://www.openssl.org/source/license.html>.