NAME

bio - I/O abstraction

SYNOPSIS

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

TBA

DESCRIPTION

A BIO is an I/O abstraction, it hides many of the underlying I/O details from an application. If an application uses a BIO for its I/O it can transparently handle SSL connections, unencrypted network connections and file I/O.

There are two type of BIO, a source/sink BIO and a filter BIO.

As its name implies a source/sink BIO is a source and/or sink of data, examples include a socket BIO and a file BIO.

A filter BIO takes data from one BIO and passes it through to another, or the application. The data may be left unmodified (for example a message digest BIO) or translated (for example an encryption BIO). The effect of a filter BIO may change according to the I/O operation it is performing: for example an encryption BIO will encrypt data if it is being written to and decrypt data if it is being read from.

BIOs can be joined together to form a chain (a single BIO is a chain with one component). A chain normally consist of one source/sink BIO and one or more filter BIOs. Data read from or written to the first BIO then traverses the chain to the end (normally a source/sink BIO).

SEE ALSO

BIO_ctrl(3), *BIO_f_base64(3)*, *BIO_f_buffer(3)*, *BIO_f_cipher(3)*, *BIO_f_md(3)*, *BIO_f_null(3)*, *BIO_f_ssl(3)*, *BIO_find_type(3)*, *BIO_new(3)*, *BIO_new_bio_pair(3)*, *BIO_push(3)*, *BIO_read(3)*, *BIO_s_accept(3)*, *BIO_s_bio(3)*, *BIO_s_connect(3)*, *BIO_s_fd(3)*, *BIO_s_file(3)*, *BIO_s_mem(3)*, *BIO_s_null(3)*, *BIO_s_socket(3)*, *BIO_set_callback(3)*, *BIO_should_retry(3)*