NAME

BIO_new_ex, BIO_new, BIO_up_ref, BIO_free, BIO_vfree, BIO_free_all − BIO allocation and freeing functions

SYNOPSIS

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

BIO *BIO_new_ex(OSSL_LIB_CTX *libctx, const BIO_METHOD *type);
BIO *BIO_new(const BIO_METHOD *type);
int BIO_up_ref(BIO *a);
int BIO_free(BIO *a);
void BIO_vfree(BIO *a);
void BIO_free_all(BIO *a);
```

DESCRIPTION

The **BIO_new_ex()** function returns a new BIO using method **type** associated with the library context *libctx* (see **OSSL_LIB_CTX**(3)). The library context may be NULL to indicate the default library context. *type* **MUST NOT** be NULL.

The **BIO_new()** is the same as **BIO_new_ex()** except the default library context is always used.

BIO_up_ref() increments the reference count associated with the BIO object.

BIO_free() frees up a single BIO, **BIO_vfree()** also frees up a single BIO but it does not return a value. If **a** is NULL nothing is done. Calling **BIO_free()** may also have some effect on the underlying I/O structure, for example it may close the file being referred to under certain circumstances. For more details see the individual BIO_METHOD descriptions.

BIO_free_all() frees up an entire BIO chain, it does not halt if an error occurs freeing up an individual BIO in the chain. If **a** is NULL nothing is done.

RETURN VALUES

BIO_new_ex() and **BIO_new()** return a newly created BIO or NULL if the call fails.

BIO_up_ref() and **BIO_free()** return 1 for success and 0 for failure.

BIO_free_all() and **BIO_vfree()** do not return values.

NOTES

If **BIO_free()** is called on a BIO chain it will only free one BIO resulting in a memory leak.

Calling **BIO_free_all()** on a single BIO has the same effect as calling **BIO_free()** on it other than the discarded return value.

HISTORY

BIO_set() was removed in OpenSSL 1.1.0 as BIO type is now opaque.

BIO_new_ex() was added in OpenSSL 3.0.

EXAMPLES

Create a memory BIO:

```
BIO *mem = BIO_new(BIO_s_mem());
```

COPYRIGHT