NAME

BIO_new_CMS - CMS streaming filter BIO

SYNOPSIS

    #include <openssl/cms.h>

    BIO *BIO_new_CMS(BIO *out, CMS_ContentInfo *cms);

DESCRIPTION

BIO_new_CMS() returns a streaming filter BIO chain based on **cms**. The output of the filter is written to **out**. Any data written to the chain is automatically translated to a BER format CMS structure of the appropriate type.

NOTES

The chain returned by this function behaves like a standard filter BIO. It supports non blocking I/O. Content is processed and streamed on the fly and not all held in memory at once: so it is possible to encode very large structures. After all content has been written through the chain *BIO_flush()* must be called to finalise the structure.

The **CMS_STREAM** flag must be included in the corresponding **flags** parameter of the **cms** creation function.

If an application wishes to write additional data to **out** BIOs should be removed from the chain using *BIO_pop()* and freed with *BIO_free()* until **out** is reached. If no additional data needs to be written *BIO_free_all()* can be called to free up the whole chain.

Any content written through the filter is used verbatim: no canonical translation is performed.

It is possible to chain multiple BIOs to, for example, create a triple wrapped signed, enveloped, signed structure. In this case it is the applications responsibility to set the inner content type of any outer CMS_ContentInfo structures.

Large numbers of small writes through the chain should be avoided as this will produce an output consisting of lots of OCTET STRING structures. Prepending a *BIO_f_buffer()* buffering BIO will prevent this.

BUGS

There is currently no corresponding inverse BIO: i.e. one which can decode a CMS structure on the fly.

RETURN VALUES

BIO_new_CMS() returns a BIO chain when successful or NULL if an error occurred. The error can be obtained from *ERR_get_error(3)*.

SEE ALSO

ERR_get_error(3), *CMS_sign(3)*, *CMS_encrypt(3)*

HISTORY

BIO_new_CMS() was added to OpenSSL 1.0.0