Rocky Enterprise Linux 9.2 Manual Pages on command 'ASN1_item_d2i_bio.3ossl'

$ man ASN1_item_d2i_bio.3ossl

ASN1_ITEM_D2I_BIO(3ossl) OpenSSL ASN1_ITEM_D2I_BIO(3ossl)

NAME

ASN1_item_d2i_ex, ASN1_item_d2i, ASN1_item_d2i_bio_ex,

ASN1_item_d2i_bio, ASN1_item_d2i_fp_ex, ASN1_item_d2i_fp,

ASN1_item_i2d_mem_bio - decode and encode DER-encoded ASN.1 structures

SYNOPSIS

#include <openssl/asn1.h>

ASN1_VALUE *ASN1_item_d2i_ex(ASN1_VALUE **pval, const unsigned char **in,

long len, const ASN1_ITEM *it,

OSSL_LIB_CTX *libctx, const char *propq);

ASN1_VALUE *ASN1_item_d2i(ASN1_VALUE **pval, const unsigned char **in,

long len, const ASN1_ITEM *it);

void *ASN1_item_d2i_bio_ex(const ASN1_ITEM *it, BIO *in, void *x,

OSSL_LIB_CTX *libctx, const char *propq);

```
void *ASN1_item_d2i_bio(const ASN1_ITEM *it, BIO *in, void *x);

void *ASN1_item_d2i_fp_ex(const ASN1_ITEM *it, FILE *in, void *x,
                          OSSL_LIB_CTX *libctx, const char *propq);
void *ASN1_item_d2i_fp(const ASN1_ITEM *it, FILE *in, void *x);

BIO *ASN1_item_i2d_mem_bio(const ASN1_ITEM *it, const ASN1_VALUE *val);
```

DESCRIPTION

ASN1_item_d2i_ex() decodes the contents of the data stored in *in of length len which must be a DER-encoded ASN.1 structure, using the ASN.1 template it. It places the result in *pval unless pval is NULL. If *pval is non-NULL on entry then the ASN1_VALUE present there will be reused. Otherwise a new ASN1_VALUE will be allocated. If any algorithm fetches are required during the process then they will use the OSSL_LIB_CTXprovided in the libctx parameter and the property query string in propq. See "ALGORITHM FETCHING" in crypto(7) for more information about algorithm fetching. On exit *in will be updated to point to the next byte in the buffer after the decoded structure.

ASN1_item_d2i() is the same as ASN1_item_d2i_ex() except that the default OSSL_LIB_CTX is used (i.e. NULL) and with a NULL property query string.

ASN1_item_d2i_bio_ex() decodes the contents of its input BIO in, which must be a DER-encoded ASN.1 structure, using the ASN.1 template it and places the result in *pval unless pval is NULL. If in is NULL it returns NULL, else a pointer to the parsed structure. If any algorithm fetches are required during the process then they will use the OSSL_LIB_CTX provided in the libctx parameter and the property query string in propq. See "ALGORITHM FETCHING" in crypto(7) for more information about algorithm fetching.

ASN1_item_d2i_bio() is the same as ASN1_item_d2i_bio_ex() except that the default OSSL_LIB_CTX is used (i.e. NULL) and with a NULL property query string.

ASN1_item_d2i_fp_ex() is the same as ASN1_item_d2i_bio_ex() except that a FILE pointer is provided instead of a BIO.

ASN1_item_d2i_fp() is the same as ASN1_item_d2i_fp_ex() except that the default OSSL_LIB_CTX is used (i.e. NULL) and with a NULL property query string.

ASN1_item_i2d_mem_bio() encodes the given ASN.1 value val using the ASN.1 template it and returns the result in a memory BIO.

RETURN VALUES

ASN1_item_d2i_bio() returns a pointer to an ASN1_VALUE or NULL.

ASN1_item_i2d_mem_bio() returns a pointer to a memory BIO or NULL on error.

HISTORY

The functions ASN1_item_d2i_ex(), ASN1_item_d2i_bio_ex(), ASN1_item_d2i_fp_ex() and ASN1_item_i2d_mem_bio() were added in OpenSSL 3.0.

COPYRIGHT