



*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'ASN1\_item\_i2d\_mem\_bio.3ossl'***

***\$ man ASN1\_item\_i2d\_mem\_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\_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. 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

Copyright 2021 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>.

