Rocky Enterprise Linux 9.2 Manual Pages on command 'ASN1_PRINT_ARG.3ossl'

$ man ASN1_PRINT_ARG.3ossl

ASN1_AUX_CB(3ossl) OpenSSL ASN1_AUX_CB(3ossl)

NAME

ASN1_AUX, ASN1_PRINT_ARG, ASN1_STREAM_ARG, ASN1_aux_cb,

ASN1_aux_const_cb - ASN.1 auxilliary data

SYNOPSIS

```
#include <openssl/asn1t.h>

struct ASN1_AUX_st {
    void *app_data;
    int flags;
    int ref_offset;           /* Offset of reference value */
    int ref_lock;             /* Offset to an CRYPTO_RWLOCK */
    ASN1_aux_cb *asn1_cb;
    int enc_offset;           /* Offset of ASN1_ENCODING structure */
    ASN1_aux_const_cb *asn1_const_cb; /* for ASN1_OP_I2D_ and ASN1_OP_PRINT_ */
};
```

```
typedef struct ASN1_AUX_st ASN1_AUX;

struct ASN1_PRINT_ARG_st {
    BIO *out;
    int indent;
    const ASN1_PCTX *pctx;
};
typedef struct ASN1_PRINT_ARG_st ASN1_PRINT_ARG;

struct ASN1_STREAM_ARG_st {
    BIO *out;
    BIO *ndef_bio;
    unsigned char **boundary;
};
typedef struct ASN1_STREAM_ARG_st ASN1_STREAM_ARG;

typedef int ASN1_aux_cb(int operation, ASN1_VALUE **in, const ASN1_ITEM *it,
                        void *exarg);
typedef int ASN1_aux_const_cb(int operation, const ASN1_VALUE **in,
                              const ASN1_ITEM *it, void *exarg);
```

DESCRIPTION

ASN.1 data structures can be associated with an ASN1_AUX object to supply additional information about the ASN.1 structure. An ASN1_AUX structure is associated with the structure during the definition of the ASN.1 template. For example an ASN1_AUX structure will be associated by using one of the various ASN.1 template definition macros that supply auxilliary information such as ASN1_SEQUENCE_enc(), ASN1_SEQUENCE_ref(), ASN1_SEQUENCE_cb_const_cb(), ASN1_SEQUENCE_const_cb(), ASN1_SEQUENCE_cb() or ASN1_NDEF_SEQUENCE_cb().

An ASN1_AUX structure contains the following information.

app_data

Arbitrary application data

flags

Flags which indicate the auxiliarly functionality supported.

The ASN1_AFLG_REFCOUNT flag indicates that objects support reference counting.

The ASN1_AFLG_ENCODING flag indicates that the original encoding of the object will be saved.

The ASN1_AFLG_BROKEN flag is a work around for broken encoders where the sequence length value may not be correct. This should generally not be used.

The ASN1_AFLG_CONST_CB flag indicates that the "const" form of the ASN1_AUX callback should be used in preference to the non-const form.

ref_offset

If the ASN1_AFLG_REFCOUNT flag is set then this value is assumed to be an offset into the ASN1_VALUE structure where a CRYPTO_REF_COUNT may be found for the purposes of reference counting.

ref_lock

If the ASN1_AFLG_REFCOUNT flag is set then this value is assumed to be an offset into the ASN1_VALUE structure where a CRYPTO_RWLOCK may be found for the purposes of reference counting.

asn1_cb

A callback that will be invoked at various points during the

processing of the the ASN1_VALLUE. See below for further details.

enc_offset

Offset into the ASN1_VALUE object where the original encoding of the object will be saved if the ASN1_AFLG_ENCODING flag has been set.

asn1_const_cb

A callback that will be invoked at various points during the processing of the the ASN1_VALLUE. This is used in preference to the asn1_cb callback if the ASN1_AFLG_CONST_CB flag is set. See below for further details.

During the processing of an ASN1_VALUE object the callbacks set via asn1_cb or asn1_const_cb will be invoked as a result of various events indicated via the operation parameter. The value of *in will be the ASN1_VALUE object being processed based on the template in it. An additional operation specific parameter may be passed in exarg. The currently supported operations are as follows. The callbacks should return a positive value on success or zero on error, unless otherwise noted below.

ASN1_OP_NEW_PRE

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure prior to an ASN1_VALUE object being allocated. The callback may allocate the ASN1_VALUE itself and store it in *pval. If it does so it should return 2 from the callback. On error it should return 0.

ASN1_OP_NEW_POST

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure after an ASN1_VALUE object has been allocated. The allocated object is in *pval.

ASN1_OP_FREE_PRE

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately before an ASN1_VALUE is freed. If the callback originally constructed the ASN1_VALUE via ASN1_OP_NEW_PRE then it should free it at this point and return 2 from the callback. Otherwise it should return 1 for success or 0 on error.

ASN1_OP_FREE_POST

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately after ASN1_VALUE sub-structures are freed.

ASN1_OP_D2I_PRE

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately before a "d2i" operation for the ASN1_VALUE.

ASN1_OP_D2I_POST

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately after a "d2i" operation for the ASN1_VALUE.

ASN1_OP_I2D_PRE

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately before a "i2d" operation for the ASN1_VALUE.

ASN1_OP_I2D_POST

Invoked when processing a CHOICE, SEQUENCE or NDEF_SEQUENCE structure immediately after a "i2d" operation for the ASN1_VALUE.

ASN1_OP_PRINT_PRE

Invoked when processing a SEQUENCE or NDEF_SEQUENCE structure immediately before printing the ASN1_VALUE. The exarg argument will be a pointer to an ASN1_PRINT_ARG structure (see below).

ASN1_OP_PRINT_POST

Invoked when processing a SEQUENCE or NDEF_SEQUENCE structure immediately after printing the ASN1_VALUE. The exarg argument will be a pointer to an ASN1_PRINT_ARG structure (see below).

ASN1_OP_STREAM_PRE

Invoked immediately prior to streaming the ASN1_VALUE data using indefinite length encoding. The exarg argument will be a pointer to a ASN1_STREAM_ARG structure (see below).

ASN1_OP_STREAM_POST

Invoked immediately after streaming the ASN1_VALUE data using indefinite length encoding. The exarg argument will be a pointer to a ASN1_STREAM_ARG structure (see below).

ASN1_OP_DETACHED_PRE

Invoked immediately prior to processing the ASN1_VALUE data as a "detached" value (as used in CMS and PKCS7). The exarg argument will be a pointer to a ASN1_STREAM_ARG structure (see below).

ASN1_OP_DETACHED_POST

Invoked immediately after processing the ASN1_VALUE data as a "detached" value (as used in CMS and PKCS7). The exarg argument will be a pointer to a ASN1_STREAM_ARG structure (see below).

ASN1_OP_DUP_PRE

Invoked immediate prior to an ASN1_VALUE being duplicated via a call to ASN1_item_dup().

ASN1_OP_DUP_POST

Invoked immediate after to an ASN1_VALUE has been duplicated via a call to ASN1_item_dup().

ASN1_OP_GET0_LIBCTX

Invoked in order to obtain the OSSL_LIB_CTX associated with an ASN1_VALUE if any. A pointer to an OSSL_LIB_CTX should be stored in *exarg if such a value exists.

ASN1_OP_GET0_PROPQ

Invoked in order to obtain the property query string associated with an ASN1_VALUE if any. A pointer to the property query string should be stored in *exarg if such a value exists.

An ASN1_PRINT_ARG object is used during processing of ASN1_OP_PRINT_PRE and ASN1_OP_PRINT_POST callback operations. It contains the following information.

out The BIO being used to print the data out.

ndef_bio

The current number of indent spaces that should be used for printing this data.

pctx

The context for the ASN1_PCTX operation.

An ASN1_STREAM_ARG object is used during processing of ASN1_OP_STREAM_PRE, ASN1_OP_STREAM_POST, ASN1_OP_DETACHED_PRE and ASN1_OP_DETACHED_POST callback operations. It contains the following information.

out The BIO to stream through

ndef_bio

The BIO with filters appended

boundary

The streaming I/O boundary.

RETURN VALUES

The callbacks return 0 on error and a positive value on success. Some operations require specific positive success values as noted above.

SEE ALSO

ASN1_item_new_ex(3)

HISTORY

The ASN1_aux_const_cb() callback and the ASN1_OP_GET0_LIBCTX and ASN1_OP_GET0_PROPQ operation types were added in OpenSSL 3.0.

COPYRIGHT