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

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'X509_EXTENSION_create_by_NID.3ossl' command

$ man X509_EXTENSION_create_by_NID.3ossl

X509_EXTENSION_SET_OBJECT(3ossl) OpenSSL X509_EXTENSION_SET_OBJECT(3ossl)

NAME

X509_EXTENSION_set_object, X509_EXTENSION_set_critical, X509_EXTENSION_set_data, X509_EXTENSION_create_by_NID, X509_EXTENSION_create_by_OBJ, X509_EXTENSION_get_object, X509_EXTENSION_get_critical, X509_EXTENSION_get_data - extension utility functions

SYNOPSIS

```
int X509_EXTENSION_set_object(X509_EXTENSION *ex, const ASN1_OBJECT *obj);
int X509_EXTENSION_set_critical(X509_EXTENSION *ex, int crit);
int X509_EXTENSION_set_data(X509_EXTENSION *ex, ASN1_OCTET_STRING *data);
X509_EXTENSION *X509_EXTENSION_create_by_NID(X509_EXTENSION **ex,
                                             int nid, int crit,
                                             ASN1_OCTET_STRING *data);
X509_EXTENSION *X509_EXTENSION_create_by_OBJ(X509_EXTENSION **ex,
                                             const ASN1_OBJECT *obj, int crit,
                                             ASN1_OCTET_STRING *data);
ASN1_OBJECT *X509_EXTENSION_get_object(X509_EXTENSION *ex);
int X509_EXTENSION_get_critical(const X509_EXTENSION *ex);
ASN1_OCTET_STRING *X509_EXTENSION_get_data(X509_EXTENSION *ne);
```

DESCRIPTION

X509_EXTENSION_set_object() sets the extension type of ex to obj. The obj pointer is duplicated internally so obj should be freed up after

use.

X509_EXTENSION_set_critical() sets the criticality of ex to crit. If crit is zero the extension in non-critical otherwise it is critical.

X509_EXTENSION_set_data() sets the data in extension ex to data. The data pointer is duplicated internally.

X509_EXTENSION_create_by_NID() creates an extension of type nid, criticality crit using data data. The created extension is returned and written to *ex reusing or allocating a new extension if necessary so *ex should either be NULL or a valid X509_EXTENSION structure it must not be an uninitialised pointer.

X509_EXTENSION_create_by_OBJ() is identical to X509_EXTENSION_create_by_NID() except it creates and extension using obj instead of a NID.

X509_EXTENSION_get_object() returns the extension type of ex as an ASN1_OBJECT pointer. The returned pointer is an internal value which must not be freed up.

X509_EXTENSION_get_critical() returns the criticality of extension ex it returns 1 for critical and 0 for non-critical.

X509_EXTENSION_get_data() returns the data of extension ex. The returned pointer is an internal value which must not be freed up.

NOTES

These functions manipulate the contents of an extension directly. Most applications will want to parse or encode and add an extension: they should use the extension encode and decode functions instead such as X509_add1_ext_i2d() and X509_get_ext_d2i().

The data associated with an extension is the extension encoding in an ASN1_OCTET_STRING structure.

RETURN VALUES

X509_EXTENSION_set_object() X509_EXTENSION_set_critical() and X509_EXTENSION_set_data() return 1 for success and 0 for failure.

X509_EXTENSION_create_by_NID() and X509_EXTENSION_create_by_OBJ() return an X509_EXTENSION pointer or NULL if an error occurs.

X509_EXTENSION_get_object() returns an ASN1_OBJECT pointer.

X509_EXTENSION_get_critical() returns 0 for non-critical and 1 for critical.

X509_EXTENSION_get_data() returns an ASN1_OCTET_STRING pointer.

SEE ALSO

X509V3_get_d2i(3)

COPYRIGHT