ASN1_STRING_length(3SSL)

NAME

ASN1_STRING_dup, ASN1_STRING_cmp, ASN1_STRING_set, ASN1_STRING_length, ASN1_STRING_length_set, ASN1_STRING_type, ASN1_STRING_data - ASN1_STRING utility functions

SYNOPSIS

```
#include <openssl/asn1.h>
int ASN1_STRING_length(ASN1_STRING *x);
unsigned char * ASN1_STRING_data(ASN1_STRING *x);
ASN1_STRING * ASN1_STRING_dup(ASN1_STRING *a);
int ASN1_STRING_cmp(ASN1_STRING *a, ASN1_STRING *b);
int ASN1_STRING_set(ASN1_STRING *str, const void *data, int len);
int ASN1_STRING_type(ASN1_STRING *x);
int ASN1_STRING_to_UTF8(unsigned char **out, ASN1_STRING *in);
```

DESCRIPTION

These functions allow an **ASN1_STRING** structure to be manipulated.

ASN1_STRING_length() returns the length of the content of **x**.

ASN1_STRING_data() returns an internal pointer to the data of **x**. Since this is an internal pointer it should **not** be freed or modified in any way.

ASN1_STRING_dup() returns a copy of the structure **a**.

ASN1_STRING_cmp() compares **a** and **b** returning 0 if the two are identical. The string types and content are compared.

ASN1_STRING_set() sets the data of string **str** to the buffer **data** or length **len**. The supplied data is copied. If **len** is -1 then the length is determined by strlen(data).

ASN1_STRING_type() returns the type of **x**, using standard constants such as **V_ASN1_OCTET_STRING**.

ASN1_STRING_to_UTF8() converts the string **in** to UTF8 format, the converted data is allocated in a buffer in ***out**. The length of **out** is returned or a negative error code. The buffer ***out** should be free using OPENSSL_free().

NOTES

Almost all ASN1 types in OpenSSL are represented as an **ASN1_STRING** structure. Other types such as **ASN1_OCTET_STRING** are simply typedefed to **ASN1_STRING** and the functions call the **ASN1_STRING** equivalents. **ASN1_STRING** is also used for some **CHOICE** types which consist entirely of primitive string types such as **DirectoryString** and **Time**.

These functions should **not** be used to examine or modify **ASN1_INTEGER** or **ASN1_ENUMERATED** types: the relevant **INTEGER** or **ENUMERATED** utility functions should be used instead.

In general it cannot be assumed that the data returned by ASN1_STRING_data() is null terminated or does not contain embedded nulls. The actual format of the data will depend on the actual string type itself: for example for and IA5String the data will be ASCII, for a BMPString two bytes per character in big endian format, UTF8String will be in UTF8 format.

Similar care should be take to ensure the data is in the correct format when calling ASN1_STRING_set().

RETURN VALUES

SEE ALSO

ERR_get_error(3)

HISTORY