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 'SM2.7ossl' command

$ man SM2.7ossl

EVP_PKEY-SM2(7ossl) OpenSSL EVP_PKEY-SM2(7ossl)

NAME

EVP_PKEY-SM2, EVP_KEYMGMT-SM2, SM2 - EVP_PKEY keytype support for the Chinese SM2 signature and encryption algorithms

DESCRIPTION

The SM2 algorithm was first defined by the Chinese national standard GM/T 0003-2012 and was later standardized by ISO as ISO/IEC 14888. SM2 is actually an elliptic curve based algorithm. The current implementation in OpenSSL supports both signature and encryption schemes via the EVP interface.

When doing the SM2 signature algorithm, it requires a distinguishing identifier to form the message prefix which is hashed before the real message is hashed.

Common SM2 parameters

SM2 uses the parameters defined in "Common EC parameters" in EVP_PKEY-EC(7). The following parameters are different:

"cofactor" (OSSL_PKEY_PARAM_EC_COFACTOR) <unsigned integer>

This parameter is ignored for SM2.

(OSSL_PKEY_PARAM_DEFAULT_DIGEST) <UTF8 string>

Getter that returns the default digest name. (Currently returns "SM3" as of OpenSSL 3.0).

NOTES

SM2 signatures can be generated by using the 'DigestSign' series of APIs, for instance, EVP_DigestSignInit(), EVP_DigestSignUpdate() and EVP_DigestSignFinal(). Ditto for the verification process by calling the 'DigestVerify' series of APIs.

Before computing an SM2 signature, an EVP_PKEY_CTX needs to be created, and an SM2 ID must be set for it, like this:

```
EVP_PKEY_CTX_set1_id(pctx, id, id_len);
```

Before calling the EVP_DigestSignInit() or EVP_DigestVerifyInit() functions, that EVP_PKEY_CTX should be assigned to the EVP_MD_CTX, like this:

```
EVP_MD_CTX_set_pkey_ctx(mctx, pctx);
```

There is normally no need to pass a pctx parameter to EVP_DigestSignInit() or EVP_DigestVerifyInit() in such a scenario.

SM2 can be tested with the openssl-speed(1) application since version 3.0. Currently, the only valid algorithm name is sm2.

Since version 3.0, SM2 keys can be generated and loaded only when the domain parameters specify the SM2 elliptic curve.

EXAMPLES

This example demonstrates the calling sequence for using an EVP_PKEY to

verify a message with the SM2 signature algorithm and the SM3 hash algorithm:

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

/* obtain an EVP_PKEY using whatever methods... */
mctx = EVP_MD_CTX_new();
pctx = EVP_PKEY_CTX_new(pkey, NULL);
EVP_PKEY_CTX_set1_id(pctx, id, id_len);
EVP_MD_CTX_set_pkey_ctx(mctx, pctx);
EVP_DigestVerifyInit(mctx, NULL, EVP_sm3(), NULL, pkey);
EVP_DigestVerifyUpdate(mctx, msg, msg_len);
EVP_DigestVerifyFinal(mctx, sig, sig_len)
```

SEE ALSO

EVP_PKEY_CTX_new(3), EVP_DigestSignInit(3), EVP_DigestVerifyInit(3), EVP_PKEY_CTX_set1_id(3), EVP_MD_CTX_set_pkey_ctx(3)

COPYRIGHT