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 'EVP_KDF-X963.7ossl' command

$ man EVP_KDF-X963.7ossl

EVP_KDF-X963(7ossl) OpenSSL EVP_KDF-X963(7ossl)

NAME

EVP_KDF-X963 - The X9.63-2001 EVP_KDF implementation

DESCRIPTION

The EVP_KDF-X963 algorithm implements the key derivation function (X963KDF). X963KDF is used by Cryptographic Message Syntax (CMS) for EC KeyAgreement, to derive a key using input such as a shared secret key and shared info.

Identity

"X963KDF" is the name for this implementation; it can be used with the EVP_KDF_fetch() function.

Supported parameters

The supported parameters are:

"properties" (OSSL_KDF_PARAM_PROPERTIES) <UTF8 string>

"digest" (OSSL_KDF_PARAM_DIGEST) <UTF8 string>

These parameters work as described in "PARAMETERS" in EVP_KDF(3).

"key" (OSSL_KDF_PARAM_KEY) <octet string>

The shared secret used for key derivation. This parameter sets the secret.

"info" (OSSL_KDF_PARAM_INFO) <octet string>

This parameter specifies an optional value for shared info.

NOTES

X963KDF is very similar to the SSKDF that uses a digest as the auxiliary function, X963KDF appends the counter to the secret, whereas SSKDF prepends the counter.

A context for X963KDF can be obtained by calling:

```
EVP_KDF *kdf = EVP_KDF_fetch(NULL, "X963KDF", NULL);
EVP_KDF_CTX *kctx = EVP_KDF_CTX_new(kdf);
```

The output length of an X963KDF is specified via the keylen parameter to the EVP_KDF_derive(3) function.

EXAMPLES

This example derives 10 bytes, with the secret key "secret" and sharedinfo value "label":

```
EVP_KDF *kdf;
EVP_KDF_CTX *kctx;
unsigned char out[10];
OSSL_PARAM params[4], *p = params;

kdf = EVP_KDF_fetch(NULL, "X963KDF", NULL);
kctx = EVP_KDF_CTX_new(kdf);
EVP_KDF_free(kdf);

*p++ = OSSL_PARAM_construct_utf8_string(OSSL_KDF_PARAM_DIGEST,
```

```
                                    SN_sha256, strlen(SN_sha256));
    *p++ = OSSL_PARAM_construct_octet_string(OSSL_KDF_PARAM_SECRET,
                                    "secret", (size_t)6);
    *p++ = OSSL_PARAM_construct_octet_string(OSSL_KDF_PARAM_INFO,
                                    "label", (size_t)5);
    *p = OSSL_PARAM_construct_end();
    if (EVP_KDF_derive(kctx, out, sizeof(out), params) <= 0) {
        error("EVP_KDF_derive");
    }

    EVP_KDF_CTX_free(kctx);
```

CONFORMING TO

"SEC 1: Elliptic Curve Cryptography"

SEE ALSO

EVP_KDF(3), EVP_KDF_CTX_new(3), EVP_KDF_CTX_free(3),
EVP_KDF_CTX_set_params(3), EVP_KDF_CTX_get_kdf_size(3),
EVP_KDF_derive(3), "PARAMETERS" in EVP_KDF(3)

HISTORY

This functionality was added to OpenSSL 3.0.

COPYRIGHT