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

$ man EVP_KEM_fetch.3ossl

EVP_KEM_FREE(3ossl) OpenSSL EVP_KEM_FREE(3ossl)

NAME

EVP_KEM_fetch, EVP_KEM_free, EVP_KEM_up_ref, EVP_KEM_get0_name, EVP_KEM_is_a, EVP_KEM_get0_provider, EVP_KEM_do_all_provided, EVP_KEM_names_do_all, EVP_KEM_get0_description, EVP_KEM_gettable_ctx_params, EVP_KEM_settable_ctx_params - Functions to manage EVP_KEM algorithm objects

SYNOPSIS

```
#include <openssl/evp.h>
EVP_KEM *EVP_KEM_fetch(OSSL_LIB_CTX *ctx, const char *algorithm,
                       const char *properties);
void EVP_KEM_free(EVP_KEM *kem);
int EVP_KEM_up_ref(EVP_KEM *kem);
const char *EVP_KEM_get0_name(const EVP_KEM *kem);
int EVP_KEM_is_a(const EVP_KEM *kem, const char *name);
OSSL_PROVIDER *EVP_KEM_get0_provider(const EVP_KEM *kem);
void EVP_KEM_do_all_provided(OSSL_LIB_CTX *libctx,
                             void (*fn)(EVP_KEM *kem, void *arg), void *arg);
```

```
int EVP_KEM_names_do_all(const EVP_KEM *kem,
                         void (*fn)(const char *name, void *data), void *data);
const char *EVP_KEM_get0_description(const EVP_KEM *kem);
const OSSL_PARAM *EVP_KEM_gettable_ctx_params(const EVP_KEM *kem);
const OSSL_PARAM *EVP_KEM_settable_ctx_params(const EVP_KEM *kem);
```

DESCRIPTION

EVP_KEM_fetch() fetches the implementation for the given algorithm from any provider offering it, within the criteria given by the properties and in the scope of the given library context ctx (see OSSL_LIB_CTX(3)). The algorithm will be one offering functions for performing asymmetric kem related tasks such as key encapsulation and decapsulation. See "ALGORITHM FETCHING" in crypto(7) for further information.

The returned value must eventually be freed with EVP_KEM_free().

EVP_KEM_free() decrements the reference count for the EVP_KEM structure. Typically this structure will have been obtained from an earlier call to EVP_KEM_fetch(). If the reference count drops to 0 then the structure is freed.

EVP_KEM_up_ref() increments the reference count for an EVP_KEM structure.

EVP_KEM_is_a() returns 1 if kem is an implementation of an algorithm that's identifiable with name, otherwise 0.

EVP_KEM_get0_provider() returns the provider that kem was fetched from.

EVP_KEM_do_all_provided() traverses all EVP_KEMs implemented by all activated providers in the given library context libctx, and for each of the implementations, calls the given function fn with the implementation method and the given arg as argument.

EVP_KEM_get0_name() returns the algorithm name from the provided implementation for the given kem. Note that the kem may have multiple synonyms associated with it. In this case the first name from the algorithm definition is returned. Ownership of the returned string is retained by the kem object and should not be freed by the caller.

EVP_KEM_names_do_all() traverses all names for kem, and calls fn with

each name and data.

EVP_KEM_get0_description() returns a description of the kem, meant for display and human consumption. The description is at the discretion of the kem implementation.

EVP_KEM_gettable_ctx_params() and EVP_KEM_settable_ctx_params() return a constant OSSL_PARAM array that describes the names and types of key parameters that can be retrieved or set by a key encapsulation algorithm using EVP_PKEY_CTX_get_params(3) and EVP_PKEY_CTX_set_params(3).

RETURN VALUES

EVP_KEM_fetch() returns a pointer to an EVP_KEM for success or NULL for failure.

EVP_KEM_up_ref() returns 1 for success or 0 otherwise.

EVP_KEM_names_do_all() returns 1 if the callback was called for all names. A return value of 0 means that the callback was not called for any names.

EVP_KEM_gettable_ctx_params() and EVP_KEM_settable_ctx_params() return a constant OSSL_PARAM array or NULL on error.

SEE ALSO

"ALGORITHM FETCHING" in crypto(7), OSSL_PROVIDER(3)

HISTORY

The functions described here were added in OpenSSL 3.0.

COPYRIGHT