NAME

OSSL_ENCODER, OSSL_ENCODER_fetch, OSSL_ENCODER_up_ref, OSSL_ENCODER_free, OSSL_ENCODER_get0_provider, OSSL_ENCODER_get0_properties, OSSL_ENCODER_is_a, OSSL_ENCODER_get0_name, OSSL_ENCODER_get0_description, OSSL_ENCODER_do_all_provided, OSSL_ENCODER_names_do_all, OSSL_ENCODER_gettable_params, OSSL_ENCODER_get_params - Encoder method routines

SYNOPSIS

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

typedef struct ossl_encoder_st OSSL_ENCODER;

OSSL_ENCODER *OSSL_ENCODER_fetch(OSSL_LIB_CTX *ctx, const char *name,
                                 const char *properties);
int OSSL_ENCODER_up_ref(OSSL_ENCODER *encoder);
void OSSL_ENCODER_free(OSSL_ENCODER *encoder);
const OSSL_PROVIDER *OSSL_ENCODER_get0_provider(const OSSL_ENCODER *encoder);
const char *OSSL_ENCODER_get0_properties(const OSSL_ENCODER *encoder);
int OSSL_ENCODER_is_a(const OSSL_ENCODER *encoder, const char *name);
const char *OSSL_ENCODER_get0_name(const OSSL_ENCODER *encoder);
const char *OSSL_ENCODER_get0_description(const OSSL_ENCODER *encoder);
void OSSL_ENCODER_do_all_provided(OSSL_LIB_CTX *libctx,
                                  void (*fn)(OSSL_ENCODER *encoder, void *arg),
                                  void *arg);
int OSSL_ENCODER_names_do_all(const OSSL_ENCODER *encoder,
                              void (*fn)(const char *name, void *data),
                              void *data);
const OSSL_PARAM *OSSL_ENCODER_gettable_params(OSSL_ENCODER *encoder);
int OSSL_ENCODER_get_params(OSSL_ENCODER_CTX *ctx, const OSSL_PARAM params[]);
```

DESCRIPTION

OSSL_ENCODER is a method for encoders, which know how to encode an object of some kind to a encoded form, such as PEM, DER, or even human readable text.

OSSL_ENCODER_fetch() looks for an algorithm within the provider that has been loaded into the **OSSL_LIB_CTX** given by *ctx*, having the name given by *name* and the properties given by *properties*. The *name* determines what type of object the fetched encoder method is expected to be able to encode, and the properties are used to determine the expected output type. For known properties and the values they may have, please have a look in "Names and properties" in **provider-encoder**(7).

OSSL_ENCODER_up_ref() increments the reference count for the given *encoder*.

OSSL_ENCODER_free() decrements the reference count for the given *encoder*, and when the count reaches zero, frees it.

OSSL_ENCODER_get0_provider() returns the provider of the given *encoder*.

OSSL_ENCODER_get0_properties() returns the property definition associated with the given *encoder*.

OSSL_ENCODER_is_a() checks if *encoder* is an implementation of an algorithm that's identifiable with *name*.

OSSL_ENCODER_get0_name() returns the name used to fetch the given *encoder*.

OSSL_ENCODER_get0_description() returns a description of the *loader*, meant for display and human consumption. The description is at the discretion of the *loader* implementation.

OSSL_ENCODER_names_do_all() traverses all names for the given *encoder*, and calls *fn* with each name and *data* as arguments.

OSSL_ENCODER_do_all_provided() traverses all encoder implementations by all activated providers in the library context *libctx*, and for each of the implementations, calls *fn* with the implementation method and *arg* as arguments.

OSSL_ENCODER_gettable_params() returns an **OSSL_PARAM**(3) array of parameter descriptors.

OSSL_ENCODER_get_params() attempts to get parameters specified with an **OSSL_PARAM**(3) array *params*. Parameters that the implementation doesn't recognise should be ignored.

RETURN VALUES

OSSL_ENCODER_fetch() returns a pointer to the key management implementation represented by an OSSL_ENCODER object, or NULL on error.

OSSL_ENCODER_up_ref() returns 1 on success, or 0 on error.

OSSL_ENCODER_free() doesn't return any value.

OSSL_ENCODER_get0_provider() returns a pointer to a provider object, or NULL on error.

OSSL_ENCODER_get0_properties() returns a pointer to a property definition string, or NULL on

error.

OSSL_ENCODER_is_a() returns 1 of *encoder* was identifiable, otherwise 0.

OSSL_ENCODER_get0_name() returns the algorithm name from the provided implementation for the given *encoder*. Note that the *encoder* 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 *encoder* object and should not be freed by the caller.

OSSL_ENCODER_get0_description() returns a pointer to a description, or NULL if there isn't one.

OSSL_ENCODER_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.

SEE ALSO

provider(7), **OSSL_ENCODER_CTX**(3), **OSSL_ENCODER_to_bio**(3), **OSSL_ENCODER_CTX_new_for_pkey**(3), **OSSL_LIB_CTX**(3)

HISTORY

The functions described here were added in OpenSSL 3.0.

COPYRIGHT

Copyright 2019-2023 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy in the file LICENSE in the source distribution or at <https://www.openssl.org/source/license.html>.