



*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 'OSSL\_STORE\_INFO\_new\_PKEY.3ossl' command**

**\$ man OSSL\_STORE\_INFO\_new\_PKEY.3ossl**

OSSL\_STORE\_INFO(3ossl)      OpenSSL      OSSL\_STORE\_INFO(3ossl)

### NAME

OSSL\_STORE\_INFO, OSSL\_STORE\_INFO\_get\_type, OSSL\_STORE\_INFO\_get0\_NAME, OSSL\_STORE\_INFO\_get0\_NAME\_description, OSSL\_STORE\_INFO\_get0\_PARAMS, OSSL\_STORE\_INFO\_get0\_PUBKEY, OSSL\_STORE\_INFO\_get0\_PKEY, OSSL\_STORE\_INFO\_get0\_CERT, OSSL\_STORE\_INFO\_get0\_CRL, OSSL\_STORE\_INFO\_get1\_NAME, OSSL\_STORE\_INFO\_get1\_NAME\_description, OSSL\_STORE\_INFO\_get1\_PARAMS, OSSL\_STORE\_INFO\_get1\_PUBKEY, OSSL\_STORE\_INFO\_get1\_PKEY, OSSL\_STORE\_INFO\_get1\_CERT, OSSL\_STORE\_INFO\_get1\_CRL, OSSL\_STORE\_INFO\_type\_string, OSSL\_STORE\_INFO\_free, OSSL\_STORE\_INFO\_new\_NAME, OSSL\_STORE\_INFO\_set0\_NAME\_description, OSSL\_STORE\_INFO\_new\_PARAMS, OSSL\_STORE\_INFO\_new\_PUBKEY, OSSL\_STORE\_INFO\_new\_PKEY, OSSL\_STORE\_INFO\_new\_CERT, OSSL\_STORE\_INFO\_new\_CRL, OSSL\_STORE\_INFO\_new, OSSL\_STORE\_INFO\_get0\_data - Functions to manipulate OSSL\_STORE\_INFO objects

### SYNOPSIS

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

typedef struct ossl_store_info_st OSSL_STORE_INFO;

int OSSL_STORE_INFO_get_type(const OSSL_STORE_INFO *store_info);

const char *OSSL_STORE_INFO_get0_NAME(const OSSL_STORE_INFO *store_info);

char *OSSL_STORE_INFO_get1_NAME(const OSSL_STORE_INFO *store_info);

const char *OSSL_STORE_INFO_get0_NAME_description(const OSSL_STORE_INFO
```

```

        *store_info);

char *OSSL_STORE_INFO_get1_NAME_description(const OSSL_STORE_INFO *store_info);
EVP_PKEY *OSSL_STORE_INFO_get0_PARAMS(const OSSL_STORE_INFO *store_info);
EVP_PKEY *OSSL_STORE_INFO_get1_PARAMS(const OSSL_STORE_INFO *store_info);
EVP_PKEY *OSSL_STORE_INFO_get0_PUBKEY(const OSSL_STORE_INFO *info);
EVP_PKEY *OSSL_STORE_INFO_get1_PUBKEY(const OSSL_STORE_INFO *info);
EVP_PKEY *OSSL_STORE_INFO_get0_PKEY(const OSSL_STORE_INFO *store_info);
EVP_PKEY *OSSL_STORE_INFO_get1_PKEY(const OSSL_STORE_INFO *store_info);
X509 *OSSL_STORE_INFO_get0_CERT(const OSSL_STORE_INFO *store_info);
X509 *OSSL_STORE_INFO_get1_CERT(const OSSL_STORE_INFO *store_info);
X509_CRL *OSSL_STORE_INFO_get0_CRL(const OSSL_STORE_INFO *store_info);
X509_CRL *OSSL_STORE_INFO_get1_CRL(const OSSL_STORE_INFO *store_info);
const char *OSSL_STORE_INFO_type_string(int type);
void OSSL_STORE_INFO_free(OSSL_STORE_INFO *store_info);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_NAME(char *name);
int OSSL_STORE_INFO_set0_NAME_description(OSSL_STORE_INFO *info, char *desc);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_PARAMS(DSA *dsa_params);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_PUBKEY(EVP_PKEY *pubkey);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_PKEY(EVP_PKEY *pkey);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_CERT(X509 *x509);
OSSL_STORE_INFO *OSSL_STORE_INFO_new_CRL(X509_CRL *crl);
OSSL_STORE_INFO *OSSL_STORE_INFO_new(int type, void *data);
void *OSSL_STORE_INFO_get0_data(int type, const OSSL_STORE_INFO *info);

```

## DESCRIPTION

These functions are primarily useful for applications to retrieve supported objects from OSSL\_STORE\_INFO objects and for scheme specific loaders to create OSSL\_STORE\_INFO holders.

## Types

OSSL\_STORE\_INFO is an opaque type that's just an intermediary holder for the objects that have been retrieved by OSSL\_STORE\_load() and similar functions. Supported OpenSSL type object can be extracted using one of STORE\_INFO\_get0\_<TYPE>() where <TYPE> can be NAME, PARAMS, PKEY, CERT, or CRL. The life time of this extracted object is as long

as the life time of the OSSL\_STORE\_INFO it was extracted from, so care should be taken not to free the latter too early. As an alternative, STORE\_INFO\_get1\_<TYPE>() extracts a duplicate (or the same object with its reference count increased), which can be used after the containing OSSL\_STORE\_INFO has been freed. The object returned by STORE\_INFO\_get1\_<TYPE>() must be freed separately by the caller. See "SUPPORTED OBJECTS" for more information on the types that are supported.

## Functions

OSSL\_STORE\_INFO\_get\_type() takes a OSSL\_STORE\_INFO and returns the STORE type number for the object inside.

STORE\_INFO\_get\_type\_string() takes a STORE type number and returns a short string describing it.

OSSL\_STORE\_INFO\_get0\_NAME(), OSSL\_STORE\_INFO\_get0\_NAME\_description(), OSSL\_STORE\_INFO\_get0\_PARAMS(), OSSL\_STORE\_INFO\_get0\_PUBKEY(), OSSL\_STORE\_INFO\_get0\_PKEY(), OSSL\_STORE\_INFO\_get0\_CERT(), OSSL\_STORE\_INFO\_get0\_CRL() all take a OSSL\_STORE\_INFO and return the object it holds if the OSSL\_STORE\_INFO type (as returned by OSSL\_STORE\_INFO\_get\_type()) matches the function, otherwise NULL.

OSSL\_STORE\_INFO\_get1\_NAME(), OSSL\_STORE\_INFO\_get1\_NAME\_description(), OSSL\_STORE\_INFO\_get1\_PARAMS(), OSSL\_STORE\_INFO\_get1\_PUBKEY(), OSSL\_STORE\_INFO\_get1\_PKEY(), OSSL\_STORE\_INFO\_get1\_CERT() and OSSL\_STORE\_INFO\_get1\_CRL() all take a OSSL\_STORE\_INFO and return a duplicate the object it holds if the OSSL\_STORE\_INFO type (as returned by OSSL\_STORE\_INFO\_get\_type()) matches the function, otherwise NULL.

OSSL\_STORE\_INFO\_free() frees a OSSL\_STORE\_INFO and its contained type.

OSSL\_STORE\_INFO\_new\_NAME(), OSSL\_STORE\_INFO\_new\_PARAMS(), OSSL\_STORE\_INFO\_new\_PUBKEY(), OSSL\_STORE\_INFO\_new\_PKEY(), OSSL\_STORE\_INFO\_new\_CERT() and OSSL\_STORE\_INFO\_new\_CRL() create a OSSL\_STORE\_INFO object to hold the given input object. On success the input object is consumed.

Additionally, for OSSL\_STORE\_INFO\_NAME objects,

OSSL\_STORE\_INFO\_set0\_NAME\_description() can be used to add an extra

description. This description is meant to be human readable and should be used for information printout.

`OSSL_STORE_INFO_new()` creates a `OSSL_STORE_INFO` with an arbitrary type number and data structure. It's the responsibility of the caller to define type numbers other than the ones defined by `<openssl/store.h>`, and to handle freeing the associated data structure on their own.

Using type numbers that are defined by `<openssl/store.h>` may cause undefined behaviours, including crashes.

`OSSL_STORE_INFO_get0_data()` returns the data pointer that was passed to `OSSL_STORE_INFO_new()` if type matches the type number in `info`.

`OSSL_STORE_INFO_new()` and `OSSL_STORE_INFO_get0_data()` may be useful for applications that define their own STORE data, but must be used with care.

## SUPPORTED OBJECTS

Currently supported object types are:

### `OSSL_STORE_INFO_NAME`

A name is exactly that, a name. It's like a name in a directory, but formatted as a complete URI. For example, the path in URI `"file:/foo/bar/"` could include a file named `"cookie.pem"`, and in that case, the returned `OSSL_STORE_INFO_NAME` object would have the URI `"file:/foo/bar/cookie.pem"`, which can be used by the application to get the objects in that file. This can be applied to all schemes that can somehow support a listing of object URIs. For `"file:"` URIs that are used without the explicit scheme, the returned name will be the path of each object, so if `"/foo/bar"` was given and that path has the file `"cookie.pem"`, the name `"/foo/bar/cookie.pem"` will be returned.

The returned URI is considered canonical and must be unique and permanent for the storage where the object (or collection of objects) resides. Each loader is responsible for ensuring that it only returns canonical URIs. However, it's possible that certain schemes allow an object (or collection thereof) to be reached with alternative URIs; just because one URI is canonical doesn't mean

that other variants can't be used.

At the discretion of the loader that was used to get these names,  
an extra description may be attached as well.

#### OSSL\_STORE\_INFO\_PARAMS

Key parameters.

#### OSSL\_STORE\_INFO\_PKEY

A keypair or just a private key (possibly with key parameters).

#### OSSL\_STORE\_INFO\_PUBKEY

A public key (possibly with key parameters).

#### OSSL\_STORE\_INFO\_CERT

An X.509 certificate.

#### OSSL\_STORE\_INFO\_CRL

A X.509 certificate revocation list.

#### RETURN VALUES

OSSL\_STORE\_INFO\_get\_type() returns the STORE type number of the given  
OSSL\_STORE\_INFO. There is no error value.

OSSL\_STORE\_INFO\_get0\_NAME(), OSSL\_STORE\_INFO\_get0\_NAME\_description(),  
OSSL\_STORE\_INFO\_get0\_PARAMS(), OSSL\_STORE\_INFO\_get0\_PKEY(),  
OSSL\_STORE\_INFO\_get0\_CERT() and OSSL\_STORE\_INFO\_get0\_CRL() all return a  
pointer to the OpenSSL object on success, NULL otherwise.

OSSL\_STORE\_INFO\_get1\_NAME(), OSSL\_STORE\_INFO\_get1\_NAME\_description(),  
OSSL\_STORE\_INFO\_get1\_PARAMS(), OSSL\_STORE\_INFO\_get1\_PKEY(),  
OSSL\_STORE\_INFO\_get1\_CERT() and OSSL\_STORE\_INFO\_get1\_CRL() all return a  
pointer to a duplicate of the OpenSSL object on success, NULL  
otherwise.

OSSL\_STORE\_INFO\_type\_string() returns a string on success, or NULL on  
failure.

OSSL\_STORE\_INFO\_new\_NAME(), OSSL\_STORE\_INFO\_new\_PARAMS(),  
OSSL\_STORE\_INFO\_new\_PKEY(), OSSL\_STORE\_INFO\_new\_CERT() and  
OSSL\_STORE\_INFO\_new\_CRL() return a OSSL\_STORE\_INFO pointer on success,  
or NULL on failure.

OSSL\_STORE\_INFO\_set0\_NAME\_description() returns 1 on success, or 0 on  
failure.

## SEE ALSO

`ossl_store(7)`, `OSSL_STORE_open(3)`, `OSSL_STORE_register_loader(3)`

## HISTORY

The `OSSL_STORE` API was added in OpenSSL 1.1.1.

The `OSSL_STORE_INFO_PUBKEY` object type was added in OpenSSL 3.0.

## COPYRIGHT

Copyright 2016-2021 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>](https://www.openssl.org/source/license.html).

3.0.7                    2023-07-13            `OSSL_STORE_INFO(3ossl)`