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 'PEM_read_PKCS8_PRIV_KEY_INFO.3ossl' comman

$ man PEM_read_PKCS8_PRIV_KEY_INFO.3ossl

PEM_READ_CMS(3ossl) OpenSSL PEM_READ_CMS(3ossl)

NAME

DECLARE_PEM_rw, PEM_read_CMS, PEM_read_bio_CMS, PEM_write_CMS,

PEM_write_bio_CMS, PEM_write_DHxparams, PEM_write_bio_DHxparams,

PEM_read_ECPKParameters, PEM_read_bio_ECPKParameters,

PEM_write_ECPKParameters, PEM_write_bio_ECPKParameters,

PEM_read_ECPrivateKey, PEM_write_ECPrivateKey,

PEM_write_bio_ECPrivateKey, PEM_read_EC_PUBKEY, PEM_read_bio_EC_PUBKEY,

PEM_write_EC_PUBKEY, PEM_write_bio_EC_PUBKEY,

PEM_read_NETSCAPE_CERT_SEQUENCE, PEM_read_bio_NETSCAPE_CERT_SEQUENCE,

PEM_write_NETSCAPE_CERT_SEQUENCE, PEM_write_bio_NETSCAPE_CERT_SEQUENCE,

PEM_read_PKCS8, PEM_read_bio_PKCS8, PEM_write_PKCS8,

PEM_write_bio_PKCS8, PEM_write_PKCS8_PRIV_KEY_INFO,

PEM_read_bio_PKCS8_PRIV_KEY_INFO, PEM_read_PKCS8_PRIV_KEY_INFO,

PEM_write_bio_PKCS8_PRIV_KEY_INFO, PEM_read_SSL_SESSION,

PEM_read_bio_SSL_SESSION, PEM_write_SSL_SESSION,

PEM_write_bio_SSL_SESSION, PEM_read_X509_PUBKEY,

PEM_read_bio_X509_PUBKEY, PEM_write_X509_PUBKEY,

PEM_write_bio_X509_PUBKEY - PEM object encoding routines

SYNOPSIS

#include <openssl/pem.h>

```
DECLARE_PEM_rw(name, TYPE)

TYPE *PEM_read_TYPE(FILE *fp, TYPE **a, pem_password_cb *cb, void *u);
TYPE *PEM_read_bio_TYPE(BIO *bp, TYPE **a, pem_password_cb *cb, void *u);
int PEM_write_TYPE(FILE *fp, const TYPE *a);
int PEM_write_bio_TYPE(BIO *bp, const TYPE *a);
```

The following functions have been deprecated since OpenSSL 3.0, and can
be hidden entirely by defining OPENSSL_API_COMPAT with a suitable
version value, see openssl_user_macros(7):

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

int PEM_write_DHxparams(FILE *out, const DH *dh);
int PEM_write_bio_DHxparams(BIO *out, const DH *dh);
EC_GROUP *PEM_read_ECPKParameters(FILE *fp, EC_GROUP **x, pem_password_cb *cb, void *u);
EC_GROUP *PEM_read_bio_ECPKParameters(BIO *bp, EC_GROUP **x, pem_password_cb *cb, void *u);
int PEM_write_ECPKParameters(FILE *out, const EC_GROUP *x);
int PEM_write_bio_ECPKParameters(BIO *out, const EC_GROUP *x),

EC_KEY *PEM_read_EC_PUBKEY(FILE *fp, EC_KEY **x, pem_password_cb *cb, void *u);
EC_KEY *PEM_read_bio_EC_PUBKEY(BIO *bp, EC_KEY **x, pem_password_cb *cb, void *u);
int PEM_write_EC_PUBKEY(FILE *out, const EC_KEY *x);
int PEM_write_bio_EC_PUBKEY(BIO *out, const EC_KEY *x);

EC_KEY *PEM_read_ECPrivateKey(FILE *out, EC_KEY **x, pem_password_cb *cb, void *u);
EC_KEY *PEM_read_bio_ECPrivateKey(BIO *out, EC_KEY **x, pem_password_cb *cb, void *u);
int PEM_write_ECPrivateKey(FILE *out, const EC_KEY *x, const EVP_CIPHER *enc,
                           const unsigned char *kstr, int klen,
                           pem_password_cb *cb, void *u);
int PEM_write_bio_ECPrivateKey(BIO *out, const EC_KEY *x, const EVP_CIPHER *enc,
                               const unsigned char *kstr, int klen,
```

pem_password_cb *cb, void *u);

DESCRIPTION

All of the functions described on this page are deprecated.
Applications should use OSSL_ENCODER_to_bio() and
OSSL_DECODER_from_bio() instead.

In the description below, TYPE is used as a placeholder for any of the
OpenSSL datatypes, such as X509. The macro DECLARE_PEM_rw expands to
the set of declarations shown in the next four lines of the synopsis.

These routines convert between local instances of ASN1 datatypes and
the PEM encoding. For more information on the templates, see
ASN1_ITEM(3). For more information on the lower-level routines used by
the functions here, see PEM_read(3).

PEM_read_TYPE() reads a PEM-encoded object of TYPE from the file fp and
returns it. The cb and u parameters are as described in
pem_password_cb(3).

PEM_read_bio_TYPE() is similar to PEM_read_TYPE() but reads from the
BIO bp.

PEM_write_TYPE() writes the PEM encoding of the object a to the file
fp.

PEM_write_bio_TYPE() similarly writes to the BIO bp.

NOTES

These functions make no assumption regarding the pass phrase received
from the password callback. It will simply be treated as a byte
sequence.

RETURN VALUES

PEM_read_TYPE() and PEM_read_bio_TYPE() return a pointer to an allocated object, which should be released by calling TYPE_free(), or NULL on error.

PEM_write_TYPE() and PEM_write_bio_TYPE() return the number of bytes written or zero on error.

SEE ALSO

PEM_read(3), passphrase-encoding(7)

HISTORY

The functions PEM_write_DHxparams(), PEM_write_bio_DHxparams(), PEM_read_ECPKParameters(), PEM_read_bio_ECPKParameters(), PEM_write_ECPKParameters(), PEM_write_bio_ECPKParameters(), PEM_read_EC_PUBKEY(), PEM_read_bio_EC_PUBKEY(), PEM_write_EC_PUBKEY(), PEM_write_bio_EC_PUBKEY(), PEM_read_ECPrivateKey(), PEM_read_bio_ECPrivateKey(), PEM_write_ECPrivateKey() and PEM_write_bio_ECPrivateKey() were deprecated in OpenSSL 3.0.

COPYRIGHT