



*Full credit is given to the above companies including the OS that this PDF file was generated!*

***Rocky Enterprise Linux 9.2 Manual Pages on command 'i2d\_PKCS8PrivateKey\_nid\_fp.3ossl'***

***\$ man i2d\_PKCS8PrivateKey\_nid\_fp.3ossl***

D2I\_PKCS8PRIVATEKEY\_BIO(3ossl) OpenSSL D2I\_PKCS8PRIVATEKEY\_BIO(3ossl)

**NAME**

d2i\_PKCS8PrivateKey\_bio, d2i\_PKCS8PrivateKey\_fp,  
i2d\_PKCS8PrivateKey\_bio, i2d\_PKCS8PrivateKey\_fp,  
i2d\_PKCS8PrivateKey\_nid\_bio, i2d\_PKCS8PrivateKey\_nid\_fp - PKCS#8 format  
private key functions

**SYNOPSIS**

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

```
EVP_PKEY *d2i_PKCS8PrivateKey_bio(BIO *bp, EVP_PKEY **x, pem_password_cb *cb, void *u);
```

```
EVP_PKEY *d2i_PKCS8PrivateKey_fp(FILE *fp, EVP_PKEY **x, pem_password_cb *cb, void *u);
```

```
int i2d_PKCS8PrivateKey_bio(BIO *bp, const EVP_PKEY *x, const EVP_CIPHER *enc,  
char *kstr, int klen,  
pem_password_cb *cb, void *u);
```

```
int i2d_PKCS8PrivateKey_fp(FILE *fp, const EVP_PKEY *x, const EVP_CIPHER *enc,  
    char *kstr, int klen,  
    pem_password_cb *cb, void *u);
```

```
int i2d_PKCS8PrivateKey_nid_bio(BIO *bp, const EVP_PKEY *x, int nid,  
    char *kstr, int klen,  
    pem_password_cb *cb, void *u);
```

```
int i2d_PKCS8PrivateKey_nid_fp(FILE *fp, const EVP_PKEY *x, int nid,  
    char *kstr, int klen,  
    pem_password_cb *cb, void *u);
```

## DESCRIPTION

The PKCS#8 functions encode and decode private keys in PKCS#8 format using both PKCS#5 v1.5 and PKCS#5 v2.0 password based encryption algorithms.

Other than the use of DER as opposed to PEM these functions are identical to the corresponding PEM function as described in `PEM_read_PrivateKey(3)`.

## NOTES

These functions are currently the only way to store encrypted private keys using DER format.

Currently all the functions use BIOs or FILE pointers, there are no functions which work directly on memory: this can be readily worked around by converting the buffers to memory BIOs, see `BIO_s_mem(3)` for details.

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

`d2i_PKCS8PrivateKey_bio()` and `d2i_PKCS8PrivateKey_fp()` return a valid `EVP_PKEY` structure or `NULL` if an error occurred.

`i2d_PKCS8PrivateKey_bio()`, `i2d_PKCS8PrivateKey_fp()`,  
`i2d_PKCS8PrivateKey_nid_bio()` and `i2d_PKCS8PrivateKey_nid_fp()` return 1 on success or 0 on error.

## SEE ALSO

`PEM_read_PrivateKey(3)`, `passphrase-encoding(7)`

## COPYRIGHT

Copyright 2002-2018 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>.

3.0.7                    2023-07-13    `D2I_PKCS8PRIVATEKEY_BIO(3openssl)`