NAME

i2d_ECPrivateKey, d2i_ECPrivate_key - Encode and decode functions for saving and reading EC_KEY structures

SYNOPSIS

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

EC_KEY *d2i_ECPrivateKey(EC_KEY **key, const unsigned char **in, long len);
int i2d_ECPrivateKey(EC_KEY *key, unsigned char **out);

unsigned int EC_KEY_get_enc_flags(const EC_KEY *key);
void EC_KEY_set_enc_flags(EC_KEY *eckey, unsigned int flags);
```

DESCRIPTION

The ECPrivateKey encode and decode routines encode and parse an **EC_KEY** structure into a binary format (ASN.1 DER) and back again.

These functions are similar to the *d2i_X509()* functions, and you should refer to that page for a detailed description (see *d2i_X509(3)*).

The format of the external representation of the public key written by i2d_ECPrivateKey (such as whether it is stored in a compressed form or not) is described by the point_conversion_form. See *EC_GROUP_copy(3)* for a description of point_conversion_form.

When reading a private key encoded without an associated public key (e.g. if EC_PKEY_NO_PUBKEY has been used - see below), then d2i_ECPrivateKey generates the missing public key automatically. Private keys encoded without parameters (e.g. if EC_PKEY_NO_PARAMETERS has been used - see below) cannot be loaded using d2i_ECPrivateKey.

The functions EC_KEY_get_enc_flags and EC_KEY_set_enc_flags get and set the value of the encoding flags for the **key**. There are two encoding flags currently defined - EC_PKEY_NO_PARAMETERS and EC_PKEY_NO_PUBKEY. These flags define the behaviour of how the **key** is converted into ASN1 in a call to i2d_ECPrivateKey. If EC_PKEY_NO_PARAMETERS is set then the public parameters for the curve are not encoded along with the private key. If EC_PKEY_NO_PUBKEY is set then the public key is not encoded along with the private key.

RETURN VALUES

d2i_ECPrivateKey() returns a valid **EC_KEY** structure or **NULL** if an error occurs. The error code that can be obtained by *ERR_get_error(3)*.

i2d_ECPrivateKey() returns the number of bytes successfully encoded or a negative value if an error occurs. The error code can be obtained by *ERR_get_error(3)*.

EC_KEY_get_enc_flags returns the value of the current encoding flags for the EC_KEY.

SEE ALSO

crypto(3), *ec(3)*, *EC_GROUP_new(3)*, *EC_GROUP_copy(3)*, *EC_POINT_new(3)*, *EC_POINT_add(3)*, *EC_GFp_simple_method(3)*, *d2i_ECPKParameters(3)*, *d2i_ECPrivateKey(3)*