NAME

RSA_private_encrypt, RSA_public_decrypt - low-level signature operations

SYNOPSIS

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

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):

```
int RSA_private_encrypt(int flen, unsigned char *from,
                        unsigned char *to, RSA *rsa, int padding);

int RSA_public_decrypt(int flen, unsigned char *from,
                       unsigned char *to, RSA *rsa, int padding);
```

DESCRIPTION

Both of the functions described on this page are deprecated. Applications should instead use **EVP_PKEY_sign_init_ex**(3), **EVP_PKEY_sign**(3), **EVP_PKEY_verify_recover_init**(3), and **EVP_PKEY_verify_recover**(3).

These functions handle RSA signatures at a low-level.

RSA_private_encrypt() signs the **flen** bytes at **from** (usually a message digest with an algorithm identifier) using the private key **rsa** and stores the signature in **to**. **to** must point to **RSA_size(rsa)** bytes of memory.

padding denotes one of the following modes:

RSA_PKCS1_PADDING
: PKCS #1 v1.5 padding. This function does not handle the **algorithmIdentifier** specified in PKCS #1. When generating or verifying PKCS #1 signatures, **RSA_sign**(3) and **RSA_verify**(3) should be used.

RSA_NO_PADDING
: Raw RSA signature. This mode should *only* be used to implement cryptographically sound padding modes in the application code. Signing user data directly with RSA is insecure.

RSA_public_decrypt() recovers the message digest from the **flen** bytes long signature at **from** using the signer's public key **rsa**. **to** must point to a memory section large enough to hold the message digest (which is smaller than **RSA_size(rsa) - 11**). **padding** is the padding mode that was used to sign the data.

RETURN VALUES

RSA_private_encrypt() returns the size of the signature (i.e., RSA_size(rsa)). **RSA_public_decrypt()** returns the size of the recovered message digest.

On error, -1 is returned; the error codes can be obtained by **ERR_get_error**(3).

SEE ALSO

ERR_get_error(3), **RSA_sign**(3), **RSA_verify**(3), **EVP_PKEY_sign**(3), **EVP_PKEY_verify_recover**(3)

HISTORY

Both of these functions were deprecated in OpenSSL 3.0.

COPYRIGHT