NAME

RAND_bytes, RAND_pseudo_bytes - generate random data

SYNOPSIS

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

int RAND_bytes(unsigned char *buf, int num);
```

Deprecated:

```
#if OPENSSL_API_COMPAT < 0x10100000L
int RAND_pseudo_bytes(unsigned char *buf, int num);
#endif
```

DESCRIPTION

RAND_bytes() puts **num** cryptographically strong pseudo-random bytes into **buf**. An error occurs if the PRNG has not been seeded with enough randomness to ensure an unpredictable byte sequence.

RAND_pseudo_bytes() has been deprecated. Users should use *RAND_bytes()* instead. *RAND_pseudo_bytes()* puts **num** pseudo-random bytes into **buf**. Pseudo-random byte sequences generated by *RAND_pseudo_bytes()* will be unique if they are of sufficient length, but are not necessarily unpredictable. They can be used for non-cryptographic purposes and for certain purposes in cryptographic protocols, but usually not for key generation etc.

The contents of **buf** is mixed into the entropy pool before retrieving the new pseudo-random bytes unless disabled at compile time (see FAQ).

RETURN VALUES

RAND_bytes() returns 1 on success, 0 otherwise. The error code can be obtained by *ERR_get_error(3)*. *RAND_pseudo_bytes()* returns 1 if the bytes generated are cryptographically strong, 0 otherwise. Both functions return -1 if they are not supported by the current RAND method.

SEE ALSO

rand(3), *ERR_get_error(3)*, *RAND_add(3)*

COPYRIGHT