NAME

BN_rand_ex, BN_rand, BN_priv_rand_ex, BN_priv_rand, BN_pseudo_rand, BN_rand_range_ex, BN_rand_range, BN_priv_rand_range_ex, BN_priv_rand_range, BN_pseudo_rand_range - generate pseudo-random number

SYNOPSIS

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

int BN_rand_ex(BIGNUM *rnd, int bits, int top, int bottom,
               unsigned int strength, BN_CTX *ctx);
int BN_rand(BIGNUM *rnd, int bits, int top, int bottom);

int BN_priv_rand_ex(BIGNUM *rnd, int bits, int top, int bottom,
                    unsigned int strength, BN_CTX *ctx);
int BN_priv_rand(BIGNUM *rnd, int bits, int top, int bottom);

int BN_rand_range_ex(BIGNUM *rnd, const BIGNUM *range, unsigned int strength,
                     BN_CTX *ctx);
int BN_rand_range(BIGNUM *rnd, const BIGNUM *range);

int BN_priv_rand_range_ex(BIGNUM *rnd, const BIGNUM *range, unsigned int strength,
                          BN_CTX *ctx);
int BN_priv_rand_range(BIGNUM *rnd, const BIGNUM *range);
```

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 BN_pseudo_rand(BIGNUM *rnd, int bits, int top, int bottom);
int BN_pseudo_rand_range(BIGNUM *rnd, const BIGNUM *range);
```

DESCRIPTION

BN_rand_ex() generates a cryptographically strong pseudo-random number of *bits* in length and security strength at least *strength* bits using the random number generator for the library context associated with *ctx*. The function stores the generated data in *rnd*. The parameter *ctx* may be NULL in which case the default library context is used. If *bits* is less than zero, or too small to accommodate the requirements specified by the *top* and *bottom* parameters, an error is returned. The *top* parameters specifies requirements on the most significant bit of the generated number. If it is **BN_RAND_TOP_ANY**, there is no constraint. If it is **BN_RAND_TOP_ONE**, the top bit must be one. If it is **BN_RAND_TOP_TWO**, the two most significant bits of the number will be set to 1, so that the product of two such random numbers will always have 2**bits* length. If *bottom* is

BN_RAND_BOTTOM_ODD, the number will be odd; if it is **BN_RAND_BOTTOM_ANY** it can be odd or even. If *bits* is 1 then *top* cannot also be **BN_RAND_TOP_TWO**.

BN_rand() is the same as **BN_rand_ex()** except that the default library context is always used.

BN_rand_range_ex() generates a cryptographically strong pseudo-random number *rnd*, of security strength at least *strength* bits, in the range 0 <= *rnd* < *range* using the random number generator for the library context associated with *ctx*. The parameter *ctx* may be NULL in which case the default library context is used.

BN_rand_range() is the same as **BN_rand_range_ex()** except that the default library context is always used.

BN_priv_rand_ex(), **BN_priv_rand()**, **BN_priv_rand_rand_ex()** and **BN_priv_rand_range()** have the same semantics as **BN_rand_ex()**, **BN_rand()**, **BN_rand_range_ex()** and **BN_rand_range()** respectively. They are intended to be used for generating values that should remain private, and mirror the same difference between **RAND_bytes**(3) and **RAND_priv_bytes**(3).

NOTES

Always check the error return value of these functions and do not take randomness for granted: an error occurs if the CSPRNG has not been seeded with enough randomness to ensure an unpredictable byte sequence.

RETURN VALUES

The functions return 1 on success, 0 on error. The error codes can be obtained by **ERR_get_error**(3).

SEE ALSO

ERR_get_error(3), **RAND_add**(3), **RAND_bytes**(3), **RAND_priv_bytes**(3), **RAND**(7), **EVP_RAND**(7)

HISTORY

- Starting with OpenSSL release 1.1.0, **BN_pseudo_rand()** has been identical to **BN_rand()** and **BN_pseudo_rand_range()** has been identical to **BN_rand_range()**. The **BN_pseudo_rand()** and **BN_pseudo_rand_range()** functions were deprecated in OpenSSL 3.0.

- The **BN_priv_rand()** and **BN_priv_rand_range()** functions were added in OpenSSL 1.1.1.

- The **BN_rand_ex()**, **BN_priv_rand_ex()**, **BN_rand_range_ex()** and **BN_priv_rand_range_ex()** functions were added in OpenSSL 3.0.

COPYRIGHT