NAME

BN_mod_mul_montgomery, BN_MONT_CTX_new, BN_MONT_CTX_free, BN_MONT_CTX_set, BN_MONT_CTX_copy, BN_from_montgomery, BN_to_montgomery - Montgomery multiplication

SYNOPSIS

    #include <openssl/bn.h>

    BN_MONT_CTX *BN_MONT_CTX_new(void);
    void BN_MONT_CTX_free(BN_MONT_CTX *mont);

    int BN_MONT_CTX_set(BN_MONT_CTX *mont, const BIGNUM *m, BN_CTX *ctx);
    BN_MONT_CTX *BN_MONT_CTX_copy(BN_MONT_CTX *to, BN_MONT_CTX *from);

    int BN_mod_mul_montgomery(BIGNUM *r, BIGNUM *a, BIGNUM *b,
                BN_MONT_CTX *mont, BN_CTX *ctx);

    int BN_from_montgomery(BIGNUM *r, BIGNUM *a, BN_MONT_CTX *mont,
                BN_CTX *ctx);

    int BN_to_montgomery(BIGNUM *r, BIGNUM *a, BN_MONT_CTX *mont,
                BN_CTX *ctx);

DESCRIPTION

These functions implement Montgomery multiplication. They are used automatically when **BN_mod_exp**(3) is called with suitable input, but they may be useful when several operations are to be performed using the same modulus.

BN_MONT_CTX_new() allocates and initializes a **BN_MONT_CTX** structure.

BN_MONT_CTX_set() sets up the *mont* structure from the modulus *m* by precomputing its inverse and a value R.

BN_MONT_CTX_copy() copies the **BN_MONT_CTX** *from* to *to*.

BN_MONT_CTX_free() frees the components of the **BN_MONT_CTX**, and, if it was created by **BN_MONT_CTX_new()**, also the structure itself. If **mont** is NULL, nothing is done.

BN_mod_mul_montgomery() computes Mont(*a*,*b*):=$a*b*R^{-1}$ and places the result in *r*.

BN_from_montgomery() performs the Montgomery reduction $r = a*R^{-1}$.

BN_to_montgomery() computes Mont(a,R^2), i.e. a*R. Note that a must be nonnegative and smaller than the modulus.

For all functions, *ctx* is a previously allocated **BN_CTX** used for temporary variables.

RETURN VALUES

    BN_MONT_CTX_new() returns the newly allocated **BN_MONT_CTX**, and NULL on error.

    BN_MONT_CTX_free() has no return value.

    For the other functions, 1 is returned for success, 0 on error. The error codes can be obtained by **ERR_get_error**(3).

WARNINGS

    The inputs must be reduced modulo **m**, otherwise the result will be outside the expected range.

SEE ALSO

    ERR_get_error(3), **BN_add**(3), **BN_CTX_new**(3)

HISTORY

    BN_MONT_CTX_init() was removed in OpenSSL 1.1.0

COPYRIGHT