



*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'BN\_new.3ossl'***

***\$ man BN\_new.3ossl***

BN\_NEW(3ossl)                      OpenSSL                      BN\_NEW(3ossl)

#### NAME

BN\_new, BN\_secure\_new, BN\_clear, BN\_free, BN\_clear\_free - allocate and free BIGNUMs

#### SYNOPSIS

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

```
BIGNUM *BN_new(void);
```

```
BIGNUM *BN_secure_new(void);
```

```
void BN_clear(BIGNUM *a);
```

```
void BN_free(BIGNUM *a);
```

```
void BN_clear_free(BIGNUM *a);
```

## DESCRIPTION

`BN_new()` allocates and initializes a `BIGNUM` structure. `BN_secure_new()` does the same except that the secure heap `OPENSSL_secure_malloc(3)` is used to store the value.

`BN_clear()` is used to destroy sensitive data such as keys when they are no longer needed. It erases the memory used by `a` and sets it to the value 0. If `a` is `NULL`, nothing is done.

`BN_free()` frees the components of the `BIGNUM`, and if it was created by `BN_new()`, also the structure itself. `BN_clear_free()` additionally overwrites the data before the memory is returned to the system. If `a` is `NULL`, nothing is done.

## RETURN VALUES

`BN_new()` and `BN_secure_new()` return a pointer to the `BIGNUM` initialised to the value 0. If the allocation fails, they return `NULL` and set an error code that can be obtained by `ERR_get_error(3)`.

`BN_clear()`, `BN_free()` and `BN_clear_free()` have no return values.

## SEE ALSO

`ERR_get_error(3)`, `OPENSSL_secure_malloc(3)`

## HISTORY

`BN_init()` was removed in OpenSSL 1.1.0; use `BN_new()` instead.

## COPYRIGHT

Copyright 2000-2017 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy

in the file LICENSE in the source distribution or at  
<<https://www.openssl.org/source/license.html>>.

3.0.7

2023-07-13

BN\_NEW(3ossl)