



*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 'RSA\_generate\_multi\_prime\_key.3ossl'***

***\$ man RSA\_generate\_multi\_prime\_key.3ossl***

RSA\_GENERATE\_KEY(3ossl)      OpenSSL      RSA\_GENERATE\_KEY(3ossl)

**NAME**

EVP\_RSA\_gen, RSA\_generate\_key\_ex, RSA\_generate\_key,  
RSA\_generate\_multi\_prime\_key - generate RSA key pair

**SYNOPSIS**

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

```
EVP_PKEY *EVP_RSA_gen(unsigned int bits);
```

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_generate_key_ex(RSA *rsa, int bits, BIGNUM *e, BN_GENCB *cb);
```

```
int RSA_generate_multi_prime_key(RSA *rsa, int bits, int primes, BIGNUM *e, BN_GENCB *cb);
```

The following function has been deprecated since OpenSSL 0.9.8, and can be hidden entirely by defining OPENSSL\_API\_COMPAT with a suitable version value, see openssl\_user\_macros(7):

```
RSA *RSA_generate_key(int bits, unsigned long e,  
                      void (*callback)(int, int, void *), void *cb_arg);
```

## DESCRIPTION

EVP\_RSA\_gen() generates a new RSA key pair with modulus size bits.

All of the functions described below are deprecated. Applications should instead use EVP\_RSA\_gen(), EVP\_PKEY\_Q\_keygen(3), or EVP\_PKEY\_keygen\_init(3) and EVP\_PKEY\_keygen(3).

RSA\_generate\_key\_ex() generates a 2-prime RSA key pair and stores it in the RSA structure provided in rsa.

RSA\_generate\_multi\_prime\_key() generates a multi-prime RSA key pair and stores it in the RSA structure provided in rsa. The number of primes is given by the primes parameter. If the automatic seeding or reseeding of the OpenSSL CSPRNG fails due to external circumstances (see RAND(7)), the operation will fail.

The modulus size will be of length bits, the number of primes to form the modulus will be primes, and the public exponent will be e. Key sizes with num < 1024 should be considered insecure. The exponent is an odd number, typically 3, 17 or 65537.

In order to maintain adequate security level, the maximum number of permitted primes depends on modulus bit length:

<1024 | >=1024 | >=4096 | >=8192

-----+-----+-----+-----

A callback function may be used to provide feedback about the progress of the key generation. If `cb` is not `NULL`, it will be called as follows using the `BN_GENCB_call()` function described on the `BN_generate_prime(3)` page.

`RSA_generate_key()` is similar to `RSA_generate_key_ex()` but expects an old-style callback function; see `BN_generate_prime(3)` for information on the old-style callback.

? While a random prime number is generated, it is called as described in `BN_generate_prime(3)`.

? When the  $n$ -th randomly generated prime is rejected as not suitable for the key, `BN_GENCB_call(cb, 2, n)` is called.

? When a random  $p$  has been found with  $p-1$  relatively prime to  $e$ , it is called as `BN_GENCB_call(cb, 3, 0)`.

The process is then repeated for prime  $q$  and other primes (if any) with `BN_GENCB_call(cb, 3, i)` where  $i$  indicates the  $i$ -th prime.

## RETURN VALUES

`EVP_RSA_gen()` returns an `EVP_PKEY` or `NULL` on failure.

`RSA_generate_multi_prime_key()` returns 1 on success or 0 on error.

`RSA_generate_key_ex()` returns 1 on success or 0 on error. The error codes can be obtained by `ERR_get_error(3)`.

`RSA_generate_key()` returns a pointer to the `RSA` structure or `NULL` if the key generation fails.

## BUGS

BN\_GENCB\_call(cb, 2, x) is used with two different meanings.

## SEE ALSO

EVP\_PKEY\_Q\_keygen(3) BN\_generate\_prime(3), ERR\_get\_error(3),  
RAND\_bytes(3), RAND(7)

## HISTORY

EVP\_RSA\_gen() was added in OpenSSL 3.0. All other functions described here were deprecated in OpenSSL 3.0. For replacement see EVP\_PKEY-RSA(7).

## COPYRIGHT

Copyright 2000-2021 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            RSA\_GENERATE\_KEY(3openssl)