



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

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'DH_check_params.3oss1' command

\$ man DH_check_params.3oss1

DH_GENERATE_PARAMETERS(3oss1) OpenSSL DH_GENERATE_PARAMETERS(3oss1)

NAME

DH_generate_parameters_ex, DH_generate_parameters, DH_check, DH_check_params, DH_check_ex, DH_check_params_ex, DH_check_pub_key_ex - generate and check Diffie-Hellman parameters

SYNOPSIS

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

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 DH_generate_parameters_ex(DH *dh, int prime_len, int generator, BN_GENCB *cb);
```

```
int DH_check(DH *dh, int *codes);
```

```
int DH_check_params(DH *dh, int *codes);
```

```
int DH_check_ex(const DH *dh);
```

```
int DH_check_params_ex(const DH *dh);
```

```
int DH_check_pub_key_ex(const DH *dh, const BIGNUM *pub_key);
```

The following functions have 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):

```
DH *DH_generate_parameters(int prime_len, int generator,  
                          void (*callback)(int, int, void *), void *cb_arg);
```

DESCRIPTION

All of the functions described on this page are deprecated.

Applications should instead use `EVP_PKEY_check(3)`,
`EVP_PKEY_public_check(3)`, `EVP_PKEY_private_check(3)` and
`EVP_PKEY_param_check(3)`.

`DH_generate_parameters_ex()` generates Diffie-Hellman parameters that can be shared among a group of users, and stores them in the provided DH structure. The pseudo-random number generator must be seeded before calling it. The parameters generated by `DH_generate_parameters_ex()` should not be used in signature schemes.

`prime_len` is the length in bits of the safe prime to be generated.

`generator` is a small number > 1 , typically 2 or 5.

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 described in `BN_generate_prime(3)` while a random prime number is generated, and when a prime has been found, `BN_GENCB_call(cb, 3, 0)` is called. See `BN_generate_prime_ex(3)` for information on the `BN_GENCB_call()` function.

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

`DH_check_params()` confirms that the `p` and `g` are likely enough to be valid. This is a lightweight check, if a more thorough check is needed, use `DH_check()`. The value of `*codes` is updated with any problems found. If `*codes` is zero then no problems were found, otherwise the following bits may be set:

`DH_CHECK_P_NOT_PRIME`

The parameter `p` has been determined to not being an odd prime.

Note that the lack of this bit doesn't guarantee that `p` is a prime.

`DH_NOT_SUITABLE_GENERATOR`

The generator `g` is not suitable. Note that the lack of this bit doesn't guarantee that `g` is suitable, unless `p` is known to be a strong prime.

`DH_MODULUS_TOO_SMALL`

The modulus is too small.

DH_MODULUS_TOO_LARGE

The modulus is too large.

DH_check() confirms that the Diffie-Hellman parameters dh are valid.

The value of *codes is updated with any problems found. If *codes is zero then no problems were found, otherwise the following bits may be

set:

DH_CHECK_P_NOT_PRIME

The parameter p is not prime.

DH_CHECK_P_NOT_SAFE_PRIME

The parameter p is not a safe prime and no q value is present.

DH_UNABLE_TO_CHECK_GENERATOR

The generator g cannot be checked for suitability.

DH_NOT_SUITABLE_GENERATOR

The generator g is not suitable.

DH_CHECK_Q_NOT_PRIME

The parameter q is not prime.

DH_CHECK_INVALID_Q_VALUE

The parameter q is invalid.

DH_CHECK_INVALID_J_VALUE

The parameter j is invalid.

DH_check_ex(), DH_check_params() and DH_check_pub_key_ex() are similar to DH_check() and DH_check_params() respectively, but the error reasons are added to the thread's error queue instead of provided as return values from the function.

RETURN VALUES

DH_generate_parameters_ex(), DH_check() and DH_check_params() return 1 if the check could be performed, 0 otherwise.

DH_generate_parameters() returns a pointer to the DH structure or NULL if the parameter generation fails.

DH_check_ex(), DH_check_params() and DH_check_pub_key_ex() return 1 if the check is successful, 0 for failed.

The error codes can be obtained by ERR_get_error(3).

DH_new(3), ERR_get_error(3), RAND_bytes(3), DH_free(3)

HISTORY

All of these functions were deprecated in OpenSSL 3.0.

DH_generate_parameters() was deprecated in OpenSSL 0.9.8; use

DH_generate_parameters_ex() instead.

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 DH_GENERATE_PARAMETERS(3openssl)