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 'RAND_get0_public.3ossl' command

$ man RAND_get0_public.3ossl

RAND_GET0_PRIMARY(3ossl) OpenSSL RAND_GET0_PRIMARY(3ossl)

NAME

RAND_get0_primary, RAND_get0_public, RAND_get0_private - get access to the global EVP_RAND_CTX instances

SYNOPSIS

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

EVP_RAND_CTX *RAND_get0_primary(OSSL_LIB_CTX *ctx);
EVP_RAND_CTX *RAND_get0_public(OSSL_LIB_CTX *ctx);
EVP_RAND_CTX *RAND_get0_private(OSSL_LIB_CTX *ctx);
```

DESCRIPTION

The default RAND API implementation (RAND_OpenSSL()) utilizes three shared DRBG instances which are accessed via the RAND API:

The public and private DRBG are thread-local instances, which are used by RAND_bytes() and RAND_priv_bytes(), respectively. The primary DRBG is a global instance, which is not intended to be used directly, but is used internally to reseed the other two instances.

These functions here provide access to the shared DRBG instances.

RETURN VALUES

RAND_get0_primary() returns a pointer to the primary DRBG instance for the given OSSL_LIB_CTX ctx.

RAND_get0_public() returns a pointer to the public DRBG instance for the given OSSL_LIB_CTX ctx.

RAND_get0_private() returns a pointer to the private DRBG instance for the given OSSL_LIB_CTX ctx.

In all the above cases the ctx parameter can be NULL in which case the default OSSL_LIB_CTX is used.

NOTES

It is not thread-safe to access the primary DRBG instance. The public and private DRBG instance can be accessed safely, because they are thread-local. Note however, that changes to these two instances apply only to the current thread.

For that reason it is recommended not to change the settings of these three instances directly. Instead, an application should change the default settings for new DRBG instances at initialization time, before creating additional threads.

During initialization, it is possible to change the reseed interval and reseed time interval. It is also possible to exchange the reseeding callbacks entirely.

To set the type of DRBG that will be instantiated, use the RAND_set_DRBG_type(3) call before accessing the random number generation infrastructure.

SEE ALSO

EVP_RAND(3), RAND_set_DRBG_type(3)

HISTORY

These functions were added in OpenSSL 3.0.

COPYRIGHT