OpenSSL

Links

- Create X.509 Certificate
- Create a CRL
- Verify a Certificate Chain

Digest

Example for SHA1, using low-level message digest functions of *OpenSSL*.

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

SHA_CTX ctx;

SHA1_Init(&ctx);

while (...)
{
    (void) SHA1_Update(&ctx, data, len); /* FIXME: handle return value */
}

SHA1_Final(buf, &ctx);
```

The same as above, but based on EVP layer of *OpenSSL*.

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

EVP_MD_CTX ctx;

EVP_MD_CTX_init(&ctx);
EVP_DigestInit_ex(&ctx, EVP_sha1(), NULL);

while (...)
{
    (void) EVP_DigestUpdate(&ctx, data, len); /* FIXME: handle return value
*/
}

EVP_DigestFinal_ex(&ctx, buf, NULL);
EVP_MD_CTX_cleanup(&ctx);
```

Fix Me!: `EVP_MD_CTX_init()` is not available anymore, use `EVP_MD_CTX_new()` instead.

When using *OpenSSL* in FIPS mode then only EVP layer functions are available.

Cipher

Typically, when using low-level functions for AES-128 CFB encryption (for example), the source code looks like this:[1]

```
AES_KEY keysched;

(void) AES_set_encrypt_key(key, 16, &keysched);
(void) AES_cfb128_encrypt(plaintext, ciphertext, plainlen,
                          &keysched, iv, &len, AES_ENCRYPT);
```

But if using EVP layer functions then it looks slightly more complicated, but is usable also in *OpenSSL* FIPS mode.

```
EVP_CIPHER_CTX ctx;

EVP_CIPHER_CTX_init(&ctx);

if ((EVP_EncryptInit_ex(&ctx, EVP_aes_128_cfb(), NULL, key, iv) != 1) ||
    (EVP_EncryptUpdate(&ctx, ciphertext, &len, plaintext, plainlen) != 1) ||
    (EVP_EncryptFinal_ex(&ctx, ciphertext + len, &len) != 1))
{
    /* error */
} /* if */

EVP_CIPHER_CTX_cleanup(&ctx);
```

FIPS Mode

Setting *OpenSSL* into FIPS mode is simple.

```
if (FIPS_mode() == 0) /* check if FIPS mode is already enabled */
{
    OpenSSL_add_all_algorithms(); /* optional (note: calls
ENGINE_load_builtin_engines()) */

    if (FIPS_mode_set(1) == 1)
        log(LOG_NOTICE, "FIPS mode enabled\n"); /* success */
    else
        log(LOG_EMERG, "FIPS mode enable failed with error %lu\n",
            ERR_get_error());
} /* if */
```

[1]

Without any error handling.

From:
https://wiki.rho62.de/ - **rho62 Wiki**

Permanent link:
https://wiki.rho62.de/doku.php?id=programming:openssl

Last update: **2022/01/21 10:58**