Project1 408410019 資工三 徐怡娟

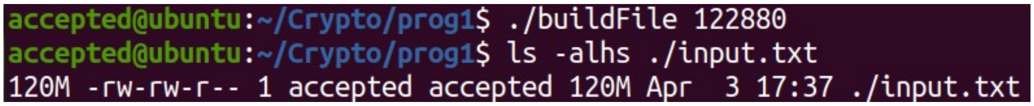
Information

- Github: https://github.com/cindy42833/Encryption-Decryption.git
- 使用 OpenSSL EVP 系列函數實做
- AES 為 AES-256
- IV 和 Key 為常數
- 可加密最大檔案為 128M
- 執行 buildFile.c 可以產生文字檔案
  - 執行方式： ./buildFile <fileSize>
  - 輸出檔案名稱為 input.txt
  - fileSize: 單位為 KB 的整數倍 e.g. ./buildFile 1024 產生 1MB 的文字檔案
- AES_CBC.c
  - 執行方式： ./AES_CBC <filename>
  - filename 為欲加密檔案, 輸出解密檔案名為 output_aes_256_cbc
- AES_CTRc
  - 執行方式： ./AES_CBC <filename>
  - filename 為欲加密檔案, 輸出解密檔案名為 output_aes_256_ctr
- ChaCha20.c
  - 執行方式： ./AES_CBC <filename>
  - filename 為欲加密檔案, 輸出解密檔案名為 output_ChaCha20
- AES_CBC.c
  - 執行方式： ./AES_CBC <filename> <mode>
  - filename 為欲加密檔案, 輸出解密檔案名為 output

- mode 有三種
  - 0：AES-256-CBC
  - 1：AES-256-CTR
  - 2：ChaCha20

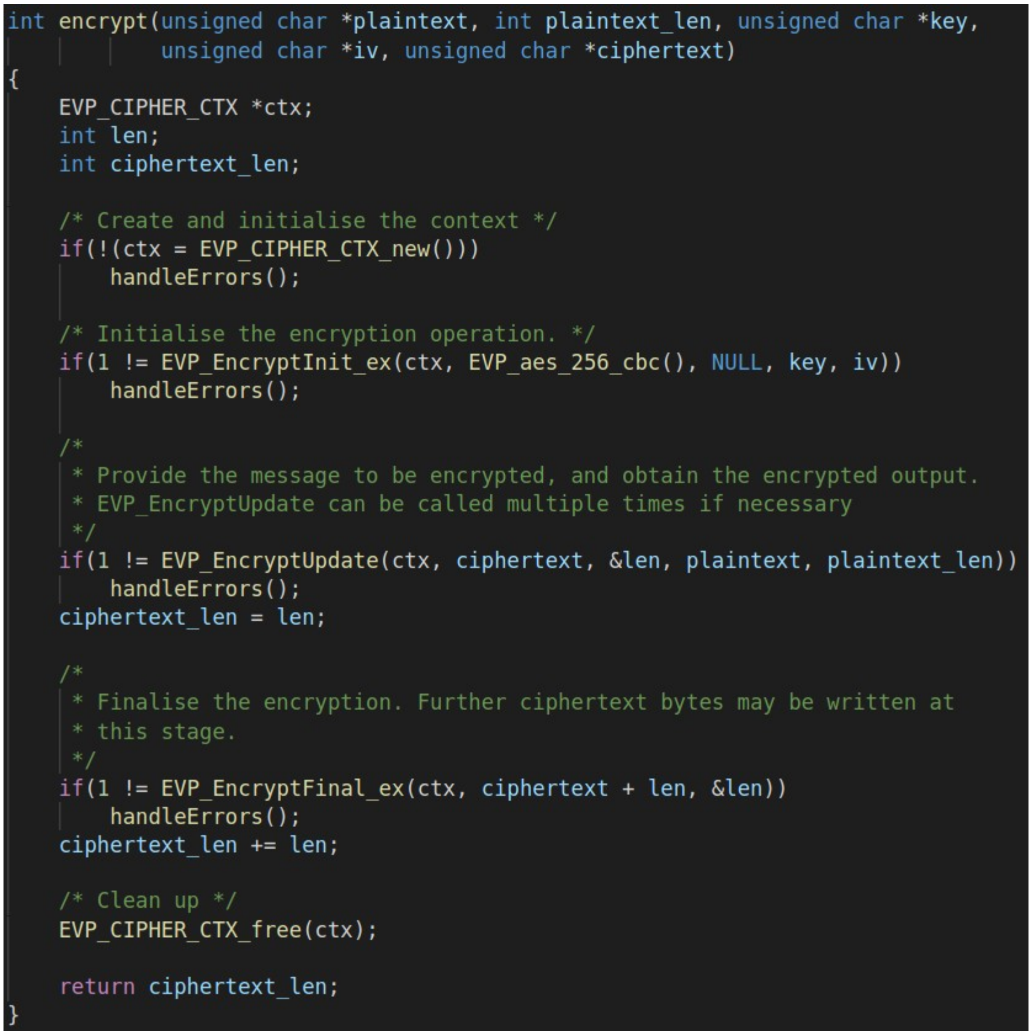
1. 程式碼和檔案大小

   a) 檔案大小 120M



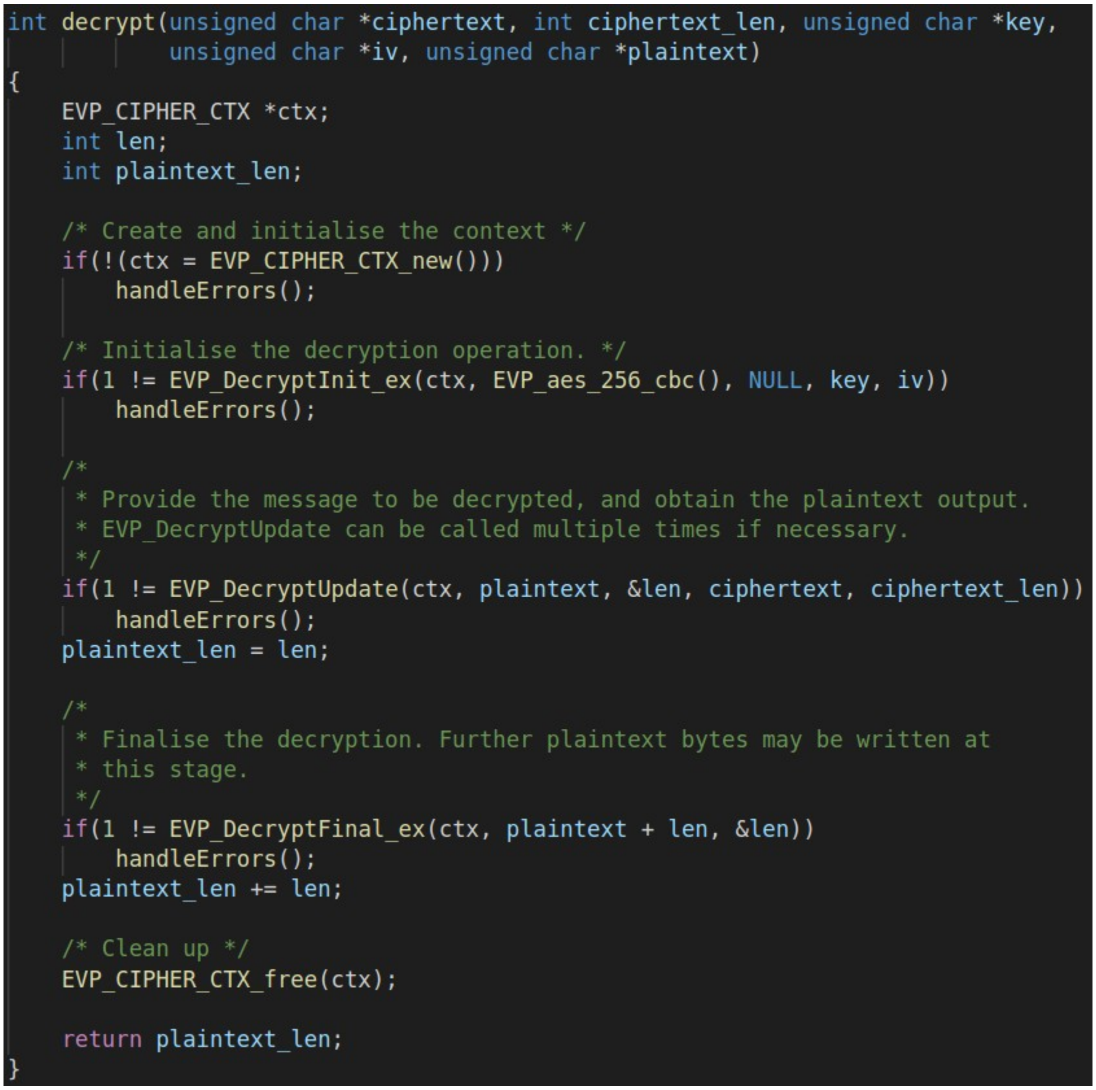
```
accepted@ubuntu:~/Crypto/prog1$ ./buildFile 122880
accepted@ubuntu:~/Crypto/prog1$ ls -alhs ./input.txt
120M -rw-rw-r-- 1 accepted accepted 120M Apr  3 17:37 ./input.txt
```

   b) 程式碼 – Encryption



```c
int encrypt(unsigned char *plaintext, int plaintext_len, unsigned char *key,
            unsigned char *iv, unsigned char *ciphertext)
{
    EVP_CIPHER_CTX *ctx;
    int len;
    int ciphertext_len;

    /* Create and initialise the context */
    if(!(ctx = EVP_CIPHER_CTX_new()))
        handleErrors();

    /* Initialise the encryption operation. */
    if(1 != EVP_EncryptInit_ex(ctx, EVP_aes_256_cbc(), NULL, key, iv))
        handleErrors();

    /*
     * Provide the message to be encrypted, and obtain the encrypted output.
     * EVP_EncryptUpdate can be called multiple times if necessary
     */
    if(1 != EVP_EncryptUpdate(ctx, ciphertext, &len, plaintext, plaintext_len))
        handleErrors();
    ciphertext_len = len;

    /*
     * Finalise the encryption. Further ciphertext bytes may be written at
     * this stage.
     */
    if(1 != EVP_EncryptFinal_ex(ctx, ciphertext + len, &len))
        handleErrors();
    ciphertext_len += len;

    /* Clean up */
    EVP_CIPHER_CTX_free(ctx);

    return ciphertext_len;
}
```

c) 程式碼 – Decryption



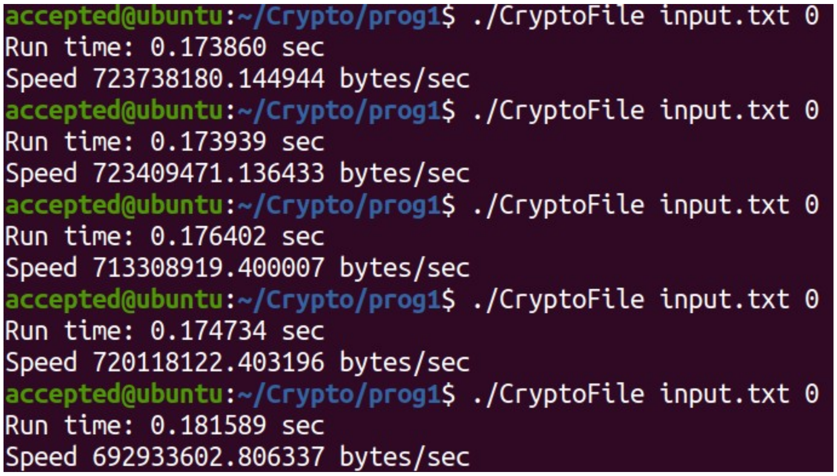
```c
int decrypt(unsigned char *ciphertext, int ciphertext_len, unsigned char *key,
            unsigned char *iv, unsigned char *plaintext)
{
    EVP_CIPHER_CTX *ctx;
    int len;
    int plaintext_len;

    /* Create and initialise the context */
    if(!(ctx = EVP_CIPHER_CTX_new()))
        handleErrors();

    /* Initialise the decryption operation. */
    if(1 != EVP_DecryptInit_ex(ctx, EVP_aes_256_cbc(), NULL, key, iv))
        handleErrors();

    /*
     * Provide the message to be decrypted, and obtain the plaintext output.
     * EVP_DecryptUpdate can be called multiple times if necessary.
     */
    if(1 != EVP_DecryptUpdate(ctx, plaintext, &len, ciphertext, ciphertext_len))
        handleErrors();
    plaintext_len = len;

    /*
     * Finalise the decryption. Further plaintext bytes may be written at
     * this stage.
     */
    if(1 != EVP_DecryptFinal_ex(ctx, plaintext + len, &len))
        handleErrors();
    plaintext_len += len;

    /* Clean up */
    EVP_CIPHER_CTX_free(ctx);

    return plaintext_len;
}
```

2. 執行以上三種加密方式的速度 (每格的值為執行五次做平均)

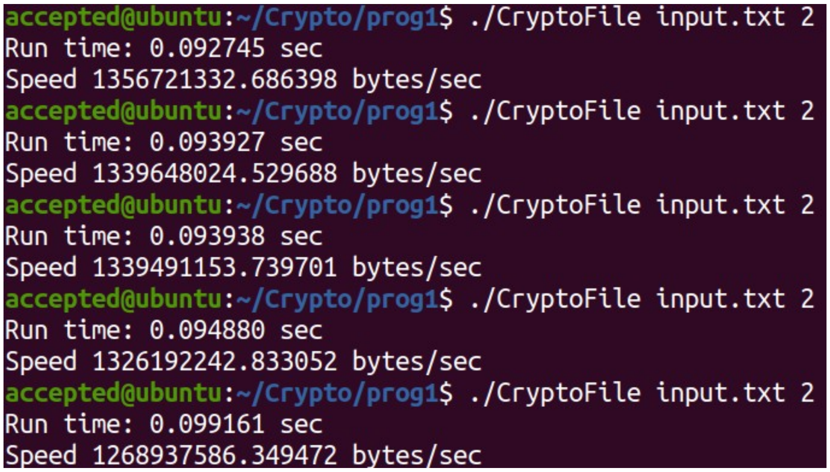
a) 執行結果比較 (僅計算加密函數執行的時間)

加密方法	AES-256-CBC	AES-256-CTR	ChaCha20
執行時間 (s)	0.17610	0.08211	0.09493
Gbytes/sec	0.67	1.43	1.24

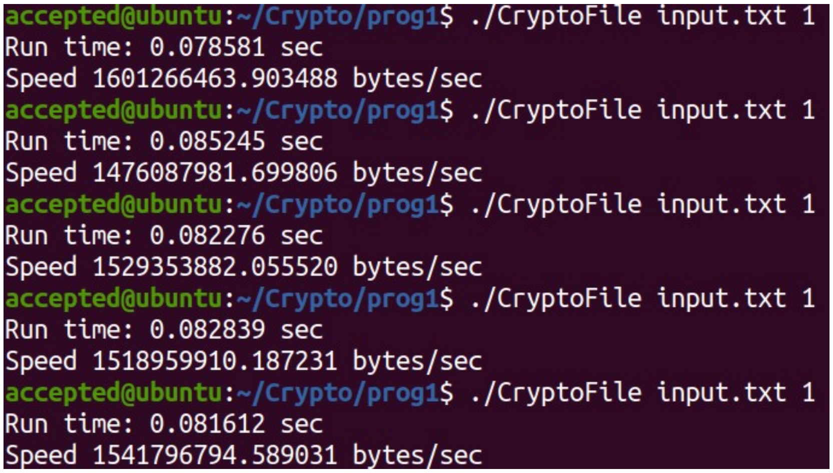
b) 執行畫面



```
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.173860 sec
Speed 723738180.144944 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.173939 sec
Speed 723409471.136433 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.176402 sec
Speed 713308919.400007 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.174734 sec
Speed 720118122.403196 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.181589 sec
Speed 692933602.806337 bytes/sec
```

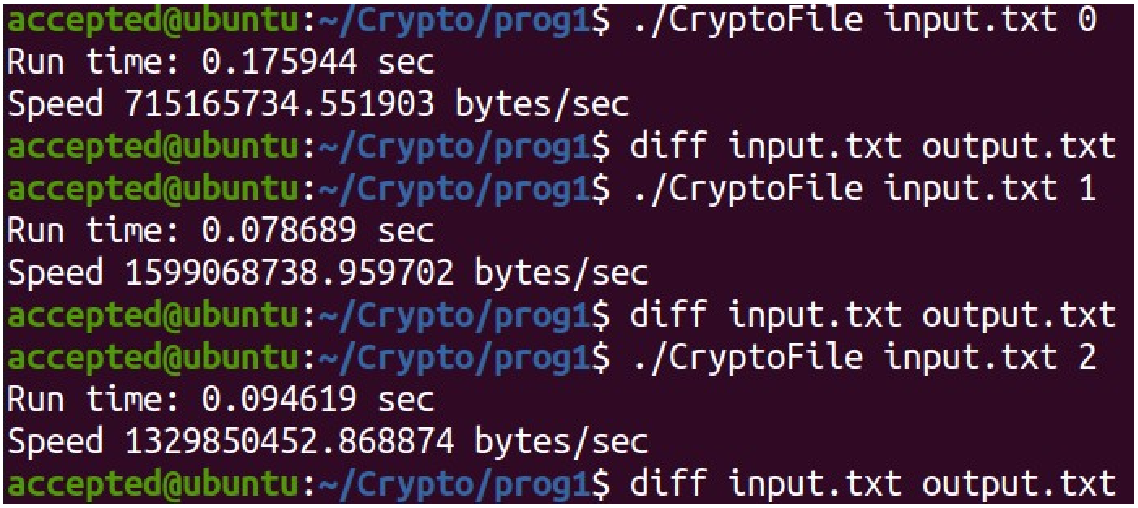


```
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.092745 sec
Speed 1356721332.686398 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.093927 sec
Speed 1339648024.529688 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.093938 sec
Speed 1339491153.739701 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.094880 sec
Speed 1326192242.833052 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.099161 sec
Speed 1268937586.349472 bytes/sec
```



```
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.078581 sec
Speed 1601266463.903488 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.085245 sec
Speed 1476087981.699806 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.082276 sec
Speed 1529353882.055520 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.082839 sec
Speed 1518959910.187231 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.081612 sec
Speed 1541796794.589031 bytes/sec
```

3. 比較解密後的檔案與原始檔



```
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 0
Run time: 0.175944 sec
Speed 715165734.551903 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ diff input.txt output.txt
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 1
Run time: 0.078689 sec
Speed 1599068738.959702 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ diff input.txt output.txt
accepted@ubuntu:~/Crypto/prog1$ ./CryptoFile input.txt 2
Run time: 0.094619 sec
Speed 1329850452.868874 bytes/sec
accepted@ubuntu:~/Crypto/prog1$ diff input.txt output.txt
```