Comparison of symmetric ciphers in OpenSSL
HM1-CNS Sapienza

Manuel Ivagnes 1698903

04/11/19

1 Introduction

For the comparison I have chosen first 2 broken and less powerful algorithms (DES, RC-2), supposing to have a lower execution time (discovered to be wrong during the tests), and then 2 still secure and more powerful algorithms (Blowfish, AES). This to see how and if increase the "security" loads on the computer performances.

With the same mind I decide to use ECB and CBC as operating modes. ECB is the simplest one and has access to a parallel implementation for both encryption and decryption, so this operating mode provides the best conditions to see the real differences in the execution time. CBC instead can parallelize only the decryption but not the encryption, this provides an higher speed-ratio and, considering that is more likely to have a decryption rather than an encryption, it can be accepted.

For all the tests I have used the MacOs terminal. To generate random passwords I have used the openssl command:

openssl rand -base64 *bytes*

Where -base64 is for the ASCII encode.

To take the time measures I have used the command "time" of the terminal (looking only at the user time for more precision) and repeated the test 3 times for each file and for each encryption and decryption. As input I have used at first a pdf file with size 35.5 MB and a then video file with size 100.7 MB.

2 Results test size 35.5 MB

1. Algorithm: DES

- Operating Mode 1: ECB
  - Encryption time average: 0.480s
  - Decryption time average: 0.490s
  - Speed-ratio: 0.480 / 0.490 = 0.97
- Operating Mode 2: CBC
  - Encryption time average: 0.518s
  - Decryption time average: 0.509s
  - Speed-ratio: 0.518 / 0.509 = 1.01

2. Cipher Algorithm: RC-2
   - Operating Mode 1: ECB
     - Encryption time average: 0.761s
     - Decryption time average: 0.418s
     - Speed-ratio: 0.761 / 0.418 = 1.82
   - Operating Mode 2: CBC
     - Encryption time average: 0.843s
     - Decryption time average: 0.428s
     - Speed-ratio: 0.843 / 0.428 = 1.96
3. Cipher Algorithm: Blowfish
   - Operating Mode 1: ECB
     - Encryption time average: 0.312s
     - Decryption time average: 0.295s
     - Speed-ratio: 0.312 / 0.295 = 1.05
   - Operating Mode 2: CBC
     - Encryption time average: 0.344s
     - Decryption time average: 0.300s
     - Speed-ratio: 0.344 / 0.300 = 1.14
4. Cipher Algorithm: AES-256
   - Operating Mode 1: ECB
     - Encryption time average: 0.025s
     - Decryption time average: 0.024s
     - Speed-ratio: 0.025 / 0.024 = 1.04

- Operating Mode 2: CBC
    - Encryption time average: 0.060s
    - Decryption time average: 0.024s
    - Speed-ratio: 0.060 / 0.024 = 2.5

3 Results test size 100.7 MB

1. Algorithm: DES
    - Operating Mode 1: ECB
        - Encryption time average: 1.344s
        - Decryption time average: 1.404s
        - Speed-ratio: 1.344 / 1.404 = 0.95
    - Operating Mode 2: CBC
        - Encryption time average: 1.368s
        - Decryption time average: 1.358s
        - Speed-ratio: 1.378 / 1.358 = 1.01
2. Cipher Algorithm: RC-2
    - Operating Mode 1: ECB
        - Encryption time average: 2.121s
        - Decryption time average: 1.193s
        - Speed-ratio: 2.121 / 1.193 = 1.77
    - Operating Mode 2: CBC
        - Encryption time average: 2.283s
        - Decryption time average: 1.172s
        - Speed-ratio: 2.283 / 1.172 = 1.94
3. Cipher Algorithm: Blowfish
    - Operating Mode 1: ECB
        - Encryption time average: 0.822s
        - Decryption time average: 0.831s
        - Speed-ratio: 0.822 / 0.831 = 0.98
    - Operating Mode 2: CBC
        - Encryption time average: 0.917s

    - Decryption time average: 0.848s
    - Speed-ratio: 0.917 / 0.848 = 1.08

4. Cipher Algorithm: AES-256
    - Operating Mode 1: ECB
        - Encryption time average: 0.061s
        - Decryption time average: 0.065s
        - Speed-ratio: 0.061 / 0.065 = 0.93
    - Operating Mode 2: CBC
        - Encryption time average: 0.146s
        - Decryption time average: 0.062s
        - Speed-ratio: 0.146 / 0.062 = 2.35

4 Conclusions

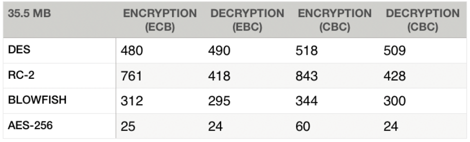
The first test, even running on a small file, already shows many differences between the ciphers. Surprisingly, the 2 less powerful algorithms (DES, RC-2) are also much slower, mostly regarding to the encryption time. Tthis result shows that the new algorithms are also more efficient and not only more powerful.

Increasing the size of the file, the ratio does not change a lot, the only difference is obviously in the execution time.

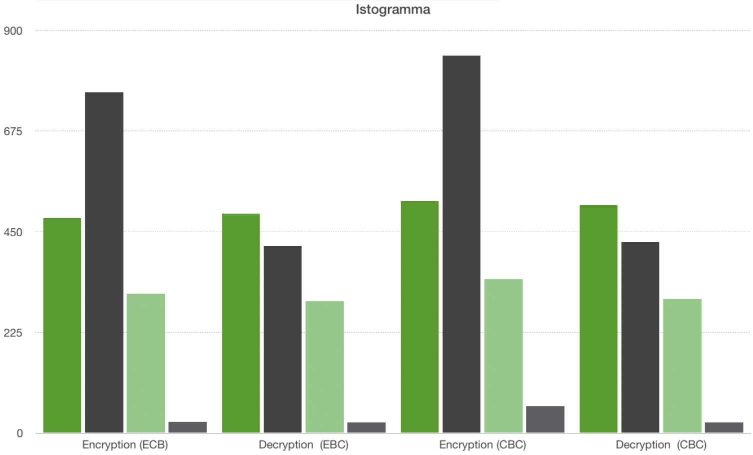
As supposed all the algorithms increase a bit the ratio value in CBC mode, given the fact that in this case is not possible to parallelize the process. Again, given the fact that is more likely to have a decryption rather than an encryption, and CBC conceals plaintext patterns, this seems to be a better option. Mostly in the case of Rijndael where the ratio gets over 2.

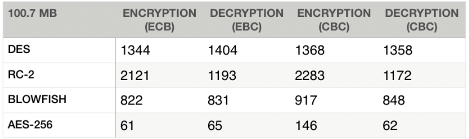
In general the speed-ratio is around 1 for most of the algorithms except for RC-2, which has a too long encryption time, and of course Rijndael in the CBC mode.

5 graphical representations

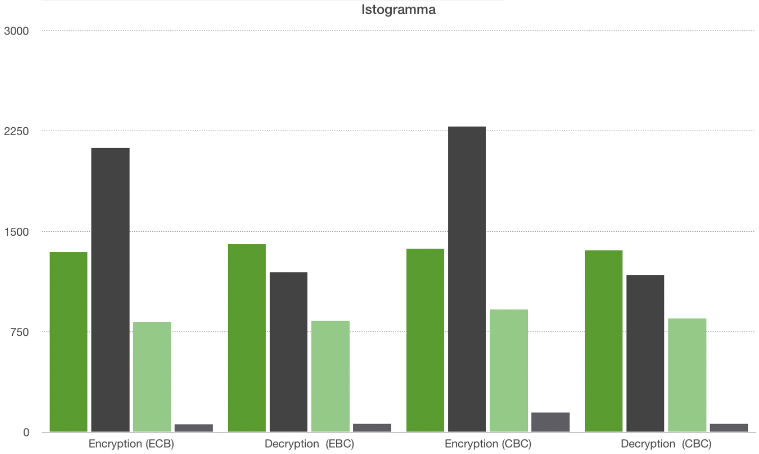


35.5 MB	ENCRYPTION (ECB)	DECRYPTION (EBC)	ENCRYPTION (CBC)	DECRYPTION (CBC)
DES	480	490	518	509
RC-2	761	418	843	428
BLOWFISH	312	295	344	300
AES-256	25	24	60	24





100.7 MB	ENCRYPTION (ECB)	DECRYPTION (EBC)	ENCRYPTION (CBC)	DECRYPTION (CBC)
DES	1344	1404	1368	1358
RC-2	2121	1193	2283	1172
BLOWFISH	822	831	917	848
AES-256	61	65	146	62


Istogramma
Encryption (ECB)
Decryption (EBC)
Encryption (CBC)
Decryption (CBC)