

Actividad 1 - Seguridad en Redes

Profesor: René Guerrero Torres

Ayudante: Iván Zuñiga

Alumno: Reinaldo Pacheco Parra

1. Se inicia la máquina Kali con la interfaz de red en modo NAT

2. Luego de iniciar, se configura el teclado en formato Latinoamericano

```
(kali㉿kali)-[~]  
$ setxkbmap -layout latam  
  
(kali㉿kali)-[~]  
$
```

3. Se crea un directorio de trabajo con el comando mkdir (make-directory) y cd (change-directory) para acceder a él

```
(kali㉿kali)-[~]  
$ mkdir crypto  
  
(kali㉿kali)-[~]  
$ cd crypto/  
  
(kali㉿kali)-[~/crypto]  
$
```

4. Se ejecuta el comando openssl para revisar los algoritmos soportados por openssl

```
(kali㉿kali)-[~/crypto]  
$ openssl help  
help:  
  
Standard commands  
asn1parse          ca                ciphers           cmp  
cms                crl               crl2pkcs7         dgst  
dhparam            dsa              dsaparam          ec  
ecparam            enc              engine            errstr  
fipsinstall        gendsa           genpkey            genrsa  
help               info             kdf               list  
mac                nseq             ocsf              passwd  
pkcs12             pkcs7            pkcs8             pkey  
pkeyparam          pkeyutl          prime             rand  
rehash             req              rsa               rsautl  
s_client           s_server         s_time            sess_id  
smime              speed            spkac             srp  
storeutl           ts               verify            version  
x509
```

```
Message Digest commands (see the `dgst' command for more details)  
blake2b512         blake2s256       md4               md5  
rmd160             sha1              sha224            sha256  
sha3-224            sha3-256          sha3-384          sha3-512  
sha384             sha512            sha512-224        sha512-256
```

A.- Generación de una llave privada

1. Se genera una llave privada de 2048 bits con el comando openssl genrsa -out private.key 2048

```
(kali㉿kali)-[~/cripto]
$ openssl genrsa -out private.key 2048

(kali㉿kali)-[~/cripto]
$ ls -l
total 4
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key
```

2. Se visualiza el archivo private.key con el comando cat

```
(kali㉿kali)-[~/cripto]
$ cat private.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQCf0/90ae57qD88
hF5rSZVgUUGBH+llrH860LwlveLe8c14DTW9bHIKpZQicZZtbpYHCLEYwKQ0o0Y1
KAj93dGwj10htCVvlbo2Q7xA6DwnwG1TkP2ivIANxjoZHQnh3PuEq/9DAX+4Zm7d
Xuas/N3SLj23os4aqe/BqpmYzUnDAerPTrHc15SXwYC8Cq8Q5ni9WlsNp6oadtON
63J/O8XhzeNEtFHUUSeg93q+AjW0Y+dkkyzS/BZDkTedFNR69Y9Ls85yhJ+EiL1
qc83jIPDhXdfBSjkVKB8NKK4g1e0cg0c5yVwbiTaGTCbE7Do63iSS0PAf6S2KPeF
d2yz6y8xAgMBAECggEAJWAoYjMehbTd2ttSkDDU0YKwfbDWfgQVea3T/dh2rC+s
HrmqGkq4rYQ0jMx2ws8ETw9JiIN5FBVS/t5hNKBZZIW/R7ZTUuckPGMnbofyA0g
HVvV9sUxi/uXXQWVKMgqWkd1iVgZo8fyf5qgg44bmndTWsJWyNn0aiBREInkG4s
e5XJA2b4Hk83u90Yg3tWomvS0fiuwm7BSUGdU+TNbTBQWns0/Btnq8wQ/dLAwmNo
HIR4Inme/saOGroU1aQA1mwrKg0F/4YXX+yYa13NhHrWABzyATN2R09UFw01FaRr
WItf5ji8zHHKdKxxfv0ZpVn3Q3Ve/s0M0KKFFejtJQKBgQDgWmTGAj75BSetraNa
jNgK5I1Bp9jkjtfG4Hel1Z1rqjz4bdKeV+89dXyKSoqusjuAG+Zl+4N+NKpIgBD0
EXTAC+W4nf8L9S2NyNBDH7+5bDv2hYxUEQJDApuZXWh4IJdVfIAAKcxkmwhOMP5t
16EXCcoUZVlFuJjXfFnthni7pQKBgQC2X4uDaNe/hwCz/4cIM+WRwpgRkBOwKqia
dKTck7v4sP/gZg/5x7d5YebHSEBgHTMTGnLEY1tto10yhBgabH7HegBv93cWaTiF
hft8709LY+v7dpeNU3uqhHAJzGz1K/lcXpksva65sxxuQP48qroXIHqpHccHsncy
wBbZgmC/nQKBgE8w+RM/pbTGkgvnF5sVzgiCRi4frhp3JVJWSUV7Enb3f0/9i/Oi
fGsINm6mAy/I7+G3d2sWDYMV2eL/cvyD0N8dE/hHpMg2wd6jPIgFA+rSfpP07eSa
oF7AUQ8AfqiwcFqXkMMWHfSBGLxo5NYm39C+/tqWL53BnAmH01F5PtKVAoGBAJ3w
GK5xOYbQr7cycAqWhxW4ZGGxWJuUffFNVIkCGCPDhZUCtnG0mEIymt2yoK2cHANL
QGD7es50R60t6gHYO+McsQxGT4fFEAHyil0kEqpH4rIffhgSWGLP1wdLVmbs8kMg
WZ75ljuT8EYpWwPHBunIYvHaq+bqk3gMCr3Lv2WxAoGAVQN71y7s4lEvPwbOr1nt
LBAzHi49G8EBVswlpYDLCS2b+PxRitPwVj90zh+iRBqgmYMJCK4j90bzTDTWAeNa
8oaNX/WHKxrr0SH9M1utyVrtrkA8a0/qJKC6eCxpqt050RaZfphjbX40o5dzY1QZ
BRA0b9SWokLRvTlomJr07mM=
-----END PRIVATE KEY-----
```

3. Se crea un archivo de texto con un mensaje con el editor nano

```
(kali㉿kali)-[~/cripto]
$ nano archivo.txt
```

```
GNU nano 8.1                                archivo.txt
Este es un mensaje secreto ...
```

B.- Cifrado con algoritmo DES

1. Se realiza el cifrado del archivo usando la llave privada creada con el algoritmo DES con el comando

`openssl enc -in archivo.txt -out archivo.enc -e -des-cbc -k private.key -pbkdf2`

```
(kali@kali)-[~/cripto]
$ openssl enc -in archivo.txt -out archivo.enc -e -des-cbc -k private.key -pbkdf2

(kali@kali)-[~/cripto]
$
```

2. Se visualiza el contenido del archivo cifrado con el comando `cat`

```
(kali@kali)-[~/cripto]
$ cat archivo.enc
Salted__333,333Y{K33333v3
r3343(`<RnW33,33qw

(kali@kali)-[~/cripto]
$
```

3. Se descifra el archivo cifrado con el comando

`openssl enc -in archivo.enc -out archivo2.txt -d -des-cbc -k private.key -pbkdf2`

```
(kali@kali)-[~/cripto]
$ openssl enc -in archivo.enc -out archivo2.txt -d -des-cbc -k private.key -pbkdf2

(kali@kali)-[~/cripto]
$ ls -l
total 16
-rw-rw-r-- 1 kali kali 32 Oct 14 19:09 archivo2.txt
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw-rw-r-- 1 kali kali 1704 Oct 14 18:55 private.key
```

4. Se visualiza el contenido del archivo descifrado

```
(kali@kali)-[~/cripto]
$ cat archivo2.txt
Este es un mensaje secreto ...
```

C.- Cifrado con el algoritmo 3DES

1. Se realiza el cifrado del archivo usando la llave privada creada con el algoritmo 3DES usando el comando
openssl enc -in archivo.txt -out archivo2.enc -e -des-ede3-cbc -k private.key -pbkdf2

```
(kali@kali)-[~/cripto]
$ openssl enc -in archivo.txt -out archivo2.enc -e -des-ede3-cbc -k private.key -pbkdf2

(kali@kali)-[~/cripto]
$ ls -l
total 20
-rw-rw-r-- 1 kali kali 56 Oct 14 19:13 archivo2.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:09 archivo2.txt
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key
```

2. Se visualiza el contenido del archivo cifrado con el comando cat

```
(kali@kali)-[~/cripto]
$ cat archivo2.enc
Salted__7/???a?U?0?|??jMg?ck:R?V-Z?7?M?zC(???\?;]7?
```

3. Se descifra el archivo utilizando el comando
openssl enc -in archivo2.enc -out archivo2.txt -d -des-ede3-cbc -k private.key -pbkdf2

```
(kali@kali)-[~/cripto]
$ openssl enc -in archivo2.enc -out archivo2.txt -d -des-ede3-cbc -k private.key -pbkdf2
2

(kali@kali)-[~/cripto]
$ ls -l
total 20
-rw-rw-r-- 1 kali kali 56 Oct 14 19:13 archivo2.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:18 archivo2.txt
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key
```

4. Se visualiza el contenido del archivo con el comando cat

```
(kali@kali)-[~/cripto]
$ cat archivo2.txt
Este es un mensaje secreto ...
```

D.- Cifrado con el algoritmo AES

1. Se realiza el cifrado del documento usando la llave privada creada con el algoritmo AES, con el comando
openssl enc -in archivo.txt -out archivo3.enc -e -aes-256-cbc -k private.key -pbkdf2

```
(kali㉿kali)-[~/cripto]
$ ls -l
total 24
-rw-rw-r-- 1 kali kali 56 Oct 14 19:13 archivo2.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:18 archivo2.txt
-rw-rw-r-- 1 kali kali 64 Oct 14 19:28 archivo3.enc
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key

(kali㉿kali)-[~/cripto]
$
```

2. Se visualiza el contenido del archivo cifrado

```
(kali㉿kali)-[~/cripto]
$ more archivo3.enc
Salted__G***.***0*$***4`E♣F♣e♣ wo***W***Ал*.***9]♣I
```

3. Se descifra el archivo utilizando el comando
openssl enc -in archivo3.enc -out archivo2.txt -d -aes-256-cbc -k private.key -pbkdf2

```
(kali㉿kali)-[~/cripto]
$ openssl enc -in archivo3.enc -out archivo2.txt -d -aes-256-cbc -k private.key -pbkdf2

(kali㉿kali)-[~/cripto]
$ ls -l
total 24
-rw-rw-r-- 1 kali kali 56 Oct 14 19:13 archivo2.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:31 archivo2.txt
-rw-rw-r-- 1 kali kali 64 Oct 14 19:28 archivo3.enc
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key
```

4. Se visualiza el contenido del archivo con el comando cat

```
(kali㉿kali)-[~/cripto]
$ cat archivo2.txt
ESte es un mensaje secreto ...
```

E.- Cifrado con el algoritmo AES utilizando semilla

1. Se cifra el archivo utilizando el siguiente comando y se ingresa la contraseña

```
(kali㉿kali)-[~/cripto]
$ openssl enc -in archivo.txt -out archivo4.enc -e -aes-256-cbc -pbkdf2
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:

(kali㉿kali)-[~/cripto]
$
```

2. Se visualiza el contenido del archivo cifrado

```
(kali㉿kali)-[~/cripto]
$ more archivo4.enc
Salted__o♦♦yHJ♦x♦`♦/♦弥♦♦♦♦♦Qp+墙EH♦b♦X♦♦♦♦:♦♦V♦狸H♦f♦♦
```

3. Se descifra el archivo con el siguiente comando
openssl enc -in archivo4.enc -out archivo2.txt -d -aes-256-cbc -pbkdf2
y luego se ingresa la contraseña establecida en el cifrado

```
(kali㉿kali)-[~/cripto]
$ openssl enc -in archivo4.enc -out archivo2.txt -d -aes-256-cbc -pbkdf2
enter AES-256-CBC decryption password:

(kali㉿kali)-[~/cripto]
$ ls -l
total 28
-rw-rw-r-- 1 kali kali 56 Oct 14 19:13 archivo2.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:41 archivo2.txt
-rw-rw-r-- 1 kali kali 64 Oct 14 19:28 archivo3.enc
-rw-rw-r-- 1 kali kali 64 Oct 14 19:35 archivo4.enc
-rw-rw-r-- 1 kali kali 56 Oct 14 19:06 archivo.enc
-rw-rw-r-- 1 kali kali 32 Oct 14 19:03 archivo.txt
-rw----- 1 kali kali 1704 Oct 14 18:55 private.key
```

4. Se visualiza el contenido del archivo descifrado

```
(kali㉿kali)-[~/cripto]
$ cat archivo2.txt
EStE es un mensaje secreto ...
```

F.- Cifrado DES utilizando recursos online

1. Se utiliza el siguiente recurso para cifrar un texto
<http://des.online-domain-tools.com/>

DES – Symmetric Ciphers Online

Input type: Text

Input text:
(plain) Este es un texto cifrado con AES

☒ Plaintext ☐ Hex Autodetect: **ON** | OFF

Function: DES



Mode: CBC (cipher block chaining)

Key:
(plain) 1234

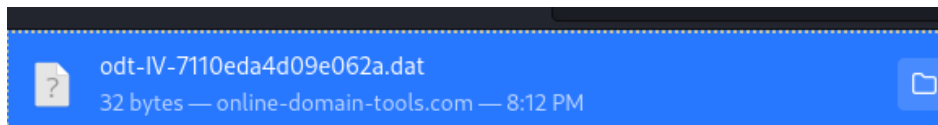
☒ Plaintext ☐ Hex

Init. vector: 71 10 ed a4 d0 9e 06 2a

> Encrypt! > Decrypt!

2. Se descarga el archivo en formato binario a Kali



3. Se carga el archivo descargado y se realiza el descifrado agregando el archivo

DES – Symmetric Ciphers Online

Input type: File

File: C:\fakepath\odt-IV-7110eda4d09e062a.dat Browse

Function: DES

Mode: CBC (cipher block chaining)

Key: 1234
(plain)

☒ Plaintext ☐ Hex

Init. vector: 71 10 ed a4 d0 9e 06 2a

> Encrypt! > Decrypt! ▶ 🔗

4. Se compara el resultado del texto descifrado con el texto inicial

Initialization vector:

7110eda4d09e062a (128 bits)

Decrypted text:

00000000	45 73 74 65 20 65 73 20 75 6e 20 74 65 78 74 6f	E s t e e s u n t e x t o
00000010	20 63 69 66 72 61 64 6f 20 63 6f 6e 20 41 45 53	

[\[Download as a binary file\] \[?\]](#) Inactive

5. Se visualiza el archivo cifrado en Kali

```
(kali㉿kali)-[~/cripto]
└─$ cat "odt-IV-7110eda4d09e062a.dat"
♦/♦β♦♦♦♦r♦♦Y
      CU♦♦♦♦♦7
                ]T♦♦
```

6. Se cifra el siguiente mensaje utilizando el algoritmo AES y se agrega una key

Input type: Text

Input text:
(plain) Lo que hace mas importante a tu rosa, es el tiempo que le has dedicado

☒ Plaintext ☐ Hex Autodetect: **ON** | **OFF**

Function: AES



Mode: CBC (cipher block chaining)

Key:
(plain) password

☒ Plaintext ☐ Hex

Init. vector: 5b aa 61 e4 c9 b9 3f 3f 06 82 25 0b 6c f8 33 1b

> Encrypt! > Decrypt!

7. Se visualiza el código hexadecimal generado con el texto cifrado

Encrypted text:

00000000	3d f4 d5 e0 00 32 4a 35 88 e5 87 cc 2f e8 c6 34	= ô Õ à . 2 J 5 å . Ì / è Æ 4
00000010	99 78 91 2c 8d 54 cd 14 33 26 be 78 d3 4a ae 04	. x , T Í . 3 & ¼ x Ó J © .
00000020	21 6f c4 60 d5 a0 4e 80 66 b8 e8 d4 5a ad 17 3e	! o Ä ` Õ N . f , è Ô Z . . >
00000030	ff 52 a8 0d 8a ec 50 48 69 8d c9 a9 5e 9e 21 86	ÿ R ~ . . ì P H i É © ^ . ! .
00000040	38 09 37 1f 7b aa 41 90 79 8a 35 8c e7 4b 55 94	8 . 7 . { º A y . 5 . ç K U .

[\[Download as a binary file\] \[?\]](#) Inactive

Checkout ?

8. Se copia el código hexadecimal y se agrega en el input seleccionando que el texto es un hexadecimal, se agrega la key con la que se cifró el texto y se procede a seleccionar descifrar.

AES – Symmetric Ciphers Online

Input type: Text

Input text:
(hex)

e8	d4	5a	ad	17	5e				
ff	52	a8	0d	8a	ec	50	48	69	8d
c9	a9	5e	9e	21	86				
38	09	37	1f	7b	8a	41	90	79	8a
35	8c	e7	4b	55	94				

☐ Plaintext ☒ Hex Autodetect: ON | OFF

Function: AES

Mode: CBC (cipher block chaining)

Key:
(plain) password

☒ Plaintext ☐ Hex

Init. vector: 5b aa 61 e4 c9 b9 3f 3f 06 82 25 0b 6c f8 33 1b

> Encrypt! > Decrypt! ▶ 🔗

9. Se comprueba el texto obtenido y se compara con el texto ingresado inicialmente.

Initialization vector:
5baa61e4c9b93f3f0682250b6cf8331b (256 bits)

Decrypted text:

00000000	4c 6f 20 71 75 65 20 68 61 63 65 20 6d 61 73 20	L o q u e h a c e m a s
00000010	69 6d 70 6f 72 74 61 6e 74 65 20 61 20 74 75 20	i m p o r t a n t e a t u
00000020	72 6f 73 61 2c 65 73 20 65 6c 20 74 69 65 6d 70	r o s a , e s e l t i e m p
00000030	6f 20 71 75 65 20 6c 65 20 68 61 73 20 64 65 64	o q u e l e h a s d e d
00000040	69 63 61 64 6f 00 00 00 00 00 00 00 00 00 00 00	i c a d o

[\[Download as a binary file\] \[?\]](#) Inactive

G.- Cifrado con RC4

1.- Se utiliza la siguiente aplicación para cifrar un texto con el algoritmo RC4

<https://www.browserling.com/tools/rc4-encrypt>

Este es un texto que será cifrado con el algoritmo RC4

Password:

password

RC4 Encrypt!

Copy to clipboard

2.- Se agrega un texto, se escribe una contraseña y se selecciona la opción "RC4 Encrypt!" para cifrar el texto

U2FsdGVkX1+ca2SLk61N/aJv2423/WW6oePLvn+7j5tS6izDdqBLE
/oK83y1VtTKkw9hD+Z6x864fTsAv8Ixy2gZR9J4QAo=

Password:

password

RC4 Encrypt!

Copy to clipboard

[\(undo\)](#)

3. Luego, se descifra el texto utilizando la siguiente aplicación:

<https://www.browserling.com/tools/rc4-decryptc>

Se agrega el texto cifrado y se escribe la contraseña con la que se cifró el texto

```
U2FsdGVkX1+ca2SLk61N/aJv2423/Ww6oePLvn+7j5tS6izDdqbLE  
/oK83y1VtTKkw9hD+Z6x864fTsAv8Ixy2gZR9J4QAO=
```

Password:

RC4 Decrypt!

Copy to clipboard

Want to RC4-encrypt text?

Use the [RC4-encrypt tool!](#)

4. Se presiona en RC4 Decrypt! para descifrar el texto y obtener el texto original

```
Este es un texto que será cifrado con el algoritmo RC4
```

Password:

RC4 Decrypt!

Copy to clipboard

[\(undo\)](#)

H.- Medición de Rendimiento

1. Se realiza la medición del tiempo de cifrado para los algoritmos
- DES, 3DES, AES-128
con el comando:
openssl speed "algoritmo"

Algoritmo DES

Tamaño de bloque	Cantidad de bloques cifrados
16	14.750.239
64	3.783.137
256	992.735
1024	253.622
8192	31.756

```
(kali㉿kali)-[~/cripto] # openssl speed "algoritmo"
$ openssl speed des
Doing des-cbc ops for 3s on 16 size blocks: 14750239 des-cbc ops in 2.91s
Doing des-cbc ops for 3s on 64 size blocks: 3783137 des-cbc ops in 2.78s
Doing des-cbc ops for 3s on 256 size blocks: 992735 des-cbc ops in 2.94s
Doing des-cbc ops for 3s on 1024 size blocks: 253622 des-cbc ops in 2.91s
Doing des-cbc ops for 3s on 8192 size blocks: 31756 des-cbc ops in 2.93s
Doing des-cbc ops for 3s on 16384 size blocks: 15841 des-cbc ops in 2.93s
Doing des-ede3 ops for 3s on 16 size blocks: 5367394 des-ede3 ops in 2.87s
Doing des-ede3 ops for 3s on 64 size blocks: 1326448 des-ede3 ops in 2.85s
Doing des-ede3 ops for 3s on 256 size blocks: 358157 des-ede3 ops in 2.89s
Doing des-ede3 ops for 3s on 1024 size blocks: 93538 des-ede3 ops in 2.84s
Doing des-ede3 ops for 3s on 8192 size blocks: 10761 des-ede3 ops in 2.79s
Doing des-ede3 ops for 3s on 16384 size blocks: 5978 des-ede3 ops in 2.92s
```

Algoritmo 3DES

Tamaño de bloque	Cantidad de bloques cifrados
16	5.924.379
64	1.528.190
256	385.583
1024	11.999
8192	5.954

```
(kali㉿kali)-[~/cripto]
$ openssl speed des-ede3
Doing des-ede3 ops for 3s on 16 size blocks: 5924379 des-ede3 ops in 2.90s
Doing des-ede3 ops for 3s on 64 size blocks: 1528190 des-ede3 ops in 2.93s
Doing des-ede3 ops for 3s on 256 size blocks: 385583 des-ede3 ops in 2.93s
Doing des-ede3 ops for 3s on 1024 size blocks: 96845 des-ede3 ops in 2.92s
Doing des-ede3 ops for 3s on 8192 size blocks: 11999 des-ede3 ops in 2.93s
Doing des-ede3 ops for 3s on 16384 size blocks: 5954 des-ede3 ops in 2.91s
```

Algoritmo AES-128

Tamaño de bloque	Cantidad de bloques cifrados
16	196.744.726
64	68.603.597
256	17.525.472
1024	4.379.300
8192	531.599

```
(kali㉿kali)-[~/cripto]
└─$ openssl speed aes-128-cbc
Doing aes-128-cbc ops for 3s on 16 size blocks: 196744726 aes-128-cbc ops in 2.91s
Doing aes-128-cbc ops for 3s on 64 size blocks: 68603597 aes-128-cbc ops in 2.92s
Doing aes-128-cbc ops for 3s on 256 size blocks: 17525472 aes-128-cbc ops in 2.93s
Doing aes-128-cbc ops for 3s on 1024 size blocks: 4379300 aes-128-cbc ops in 2.92s
Doing aes-128-cbc ops for 3s on 8192 size blocks: 531599 aes-128-cbc ops in 2.90s
Doing aes-128-cbc ops for 3s on 16384 size blocks: 261532 aes-128-cbc ops in 2.89s
```