

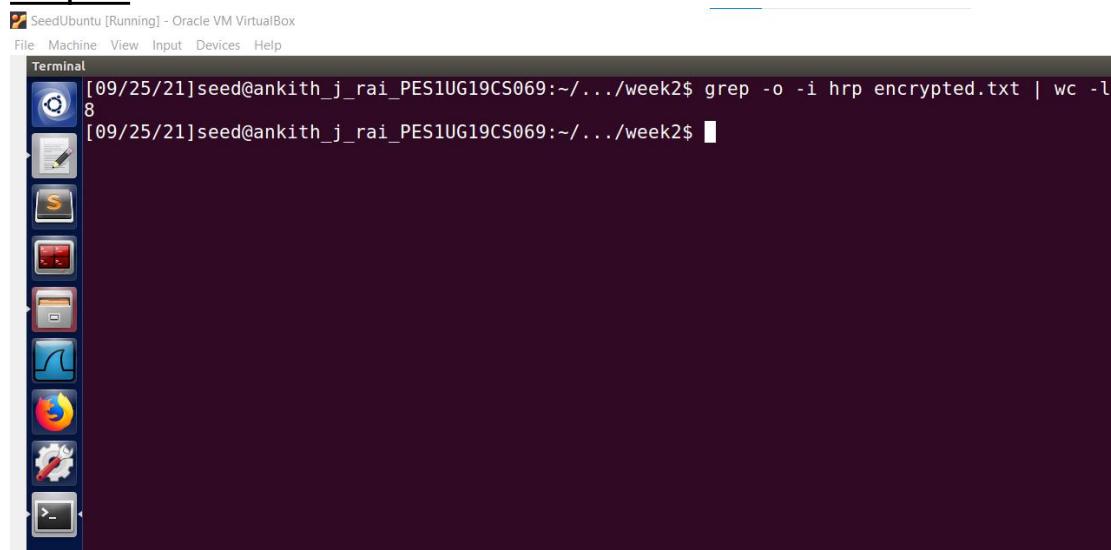
CRYPTOGRAPHY LAB- WEEK 2

Secret Key Encryption

NAME : Ankith J Rai
SRN : PES1UG19CS069
SEC : B

Task 1: Frequency Analysis Against Mono-alphabetic Substitution Cipher

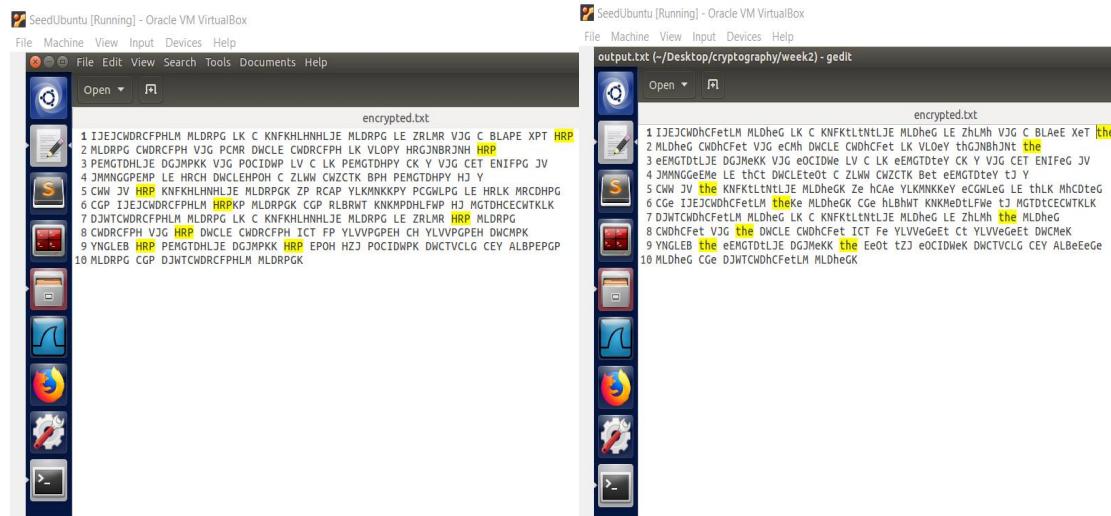
Step 1:



```
[09/25/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ grep -o -i hrp encrypted.txt | wc -l
8
[09/25/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

We can see that the number of instances of the word ‘HRP’ is 8 in the encrypted.txt file.

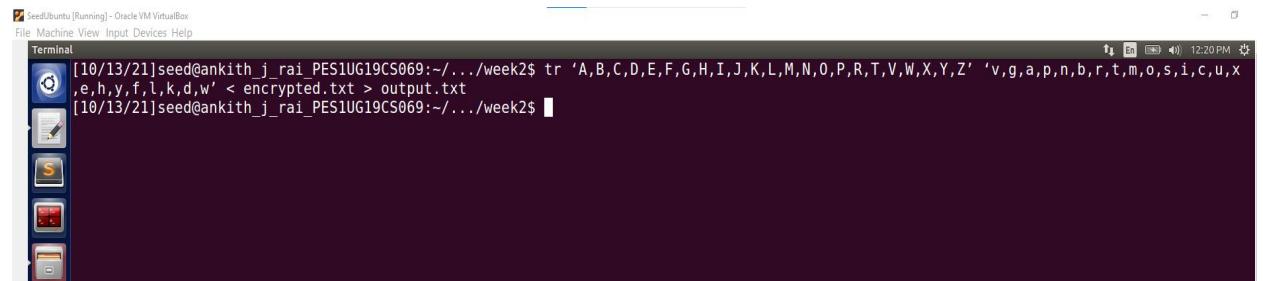
Step 2: As HRP is frequently occurring trigram hence HRP is replaced with the word ‘the’.



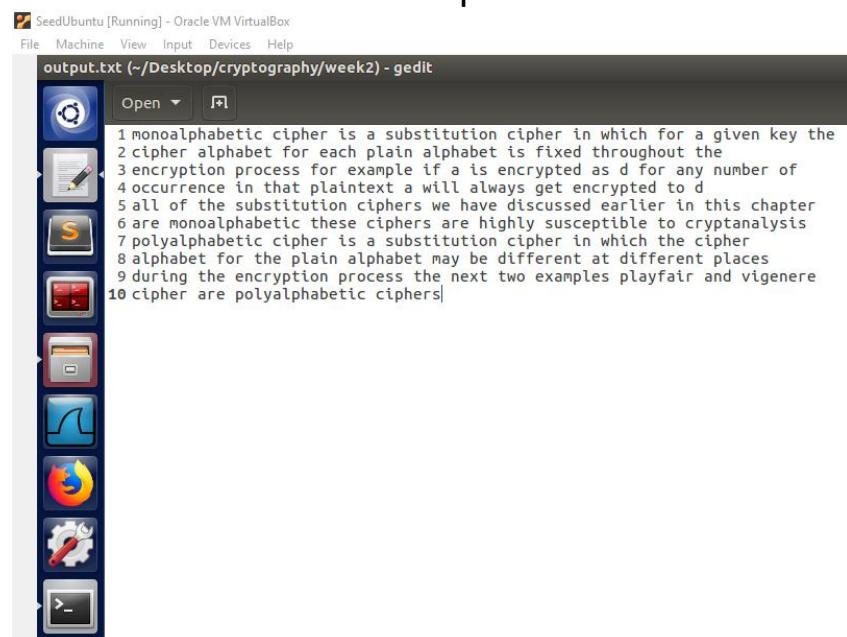
```
1 IJEJCWDRCFPHLM MLDRPG LK C KNFKHLHHNLJE MLDRPG LE ZRLMR VJG C BLAEP XPT HRP
2 MLDRPG CHDRCPH VJG PCMR DWCLE CHDRCPH LK VLQPY HRGNBRJNH HRP
3 PEMGTDHLJE DGJMPK VJG POCIDWP LV C LK PEMGTDHPY CK Y VJG CET ENIFPG JV
4 JMMNGGPEMP LE HRCH DNLCEPHOH C ZLWZ CNZCTK BPH PEMGTDHPY HJ Y
5 CMW JV HRP KNFKHLHHNLJE MLDRPGK ZP RCAP YLKMNPCKP PCGMLPG LE HRLK MRCDHGP
6 CGP IJEJCWDRCFPHLM HRPK MLDRPGK CGP RLBRWT KNKMPDHLFWP HJ MGTDHECWTKLK
7 DJWTCWDRCFPHLM MLDRPG LK C KNFKHLHHNLJE MLDRPG LE ZRLMR HRP MLDRPG
8 CHDRCPH VJG HRP DNLCE CHDRCPH ICT FP YLUVPGPEH CH YLUVPGPEH DWCMKP
9 YNGLEB HRP PEMGTDHLJE DGJMPK HRP EPQH HZJ POCIDWPK DWCTVCLG CEY ALBPEPGP
10 MLDRPG CGP DJWTCWDRCFPHLM MLDRPGK
```

```
1 IJEJCWDRCFetLM MLdheG LK C KNFKLtLNLJE MLdheG LE ZhlMh VJG C BLAeE XeT the
2 MLdheG CHdHCfet VJG eChM DWCLe CHdHCfet LK VLQey thGJBhJnt the
3 eEMGTDtLJE DGjMekk VJG eOCIDwE LV C LK eEMGTDtey CK Y VJG CET ENIFeG JV
4 JMMNGGeMe LE thtC DNCLetot C ZLWZ CNZCTK Bet eEMGTDtey tJ Y
5 CMW JV the KNFKLtLNLJE MLdheG ZChcA YLKMNPKey eCGNleg LE thLk MhCdteG
6 CGe IJEJCWDRCfetLM theke MLdheG CGe lhbhIT KNKMedtLFw tJ MGTDtCEWTKLK
7 DJWTCWDRCfetLM MLdheG LK C KNFKLtLNLJE MLdheG LE ZhlMh the MLdheG
8 CHdHCfet VJG the DWCLe CHdHCfet ICT Fe YLVeGeEt ct YLVeGeEt DWCMek
9 YNGLEB the eEMGTDtLJE DGjMekk the Eeo tZJ eOCIDwE DWCTVCLG CEY ALBeEeG
10 MLdheG CGe DJWTCWDRCfetLM MLdheG
```

After Frequency analysis we get the mapping of the cipher character with that of the plain text character.

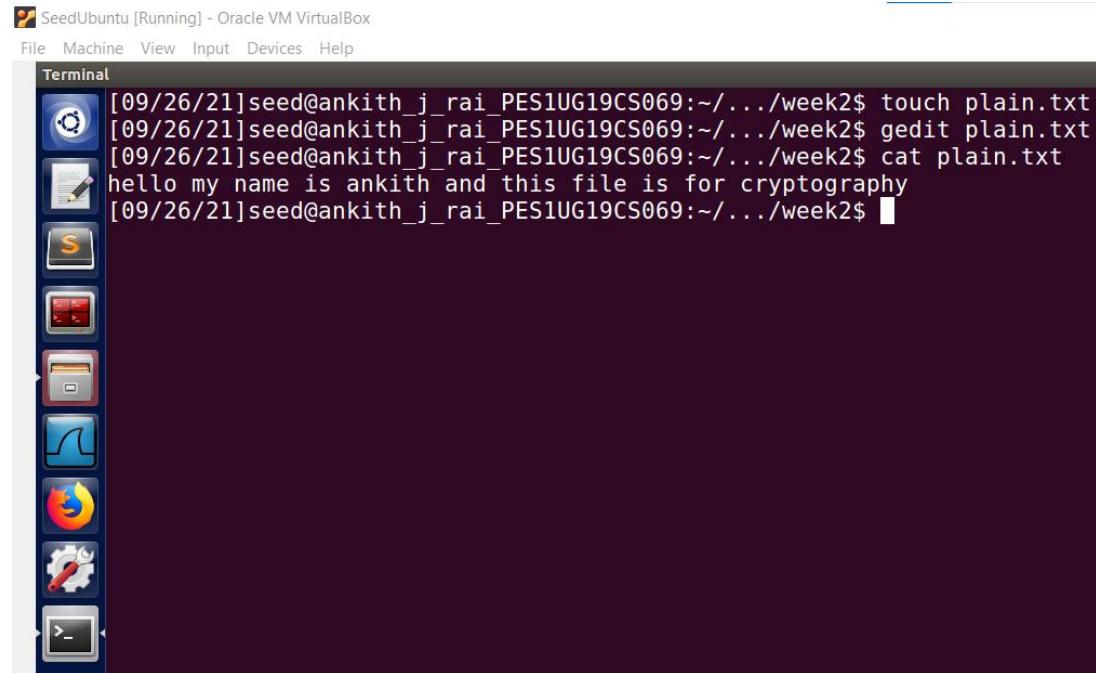


After replacing the cipher character that of the plain text character we can see the plain text which was encrypted below.



Task 2: Encryption using different ciphers and modes.

Step 1: First we create a plain.txt file with our own input.



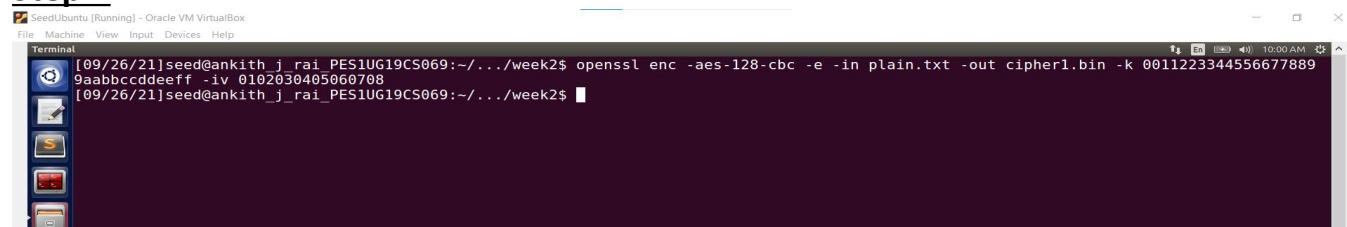
The screenshot shows a terminal window titled "Terminal" with the following command history:

```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ touch plain.txt
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit plain.txt
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ cat plain.txt
hello my name is ankith and this file is for cryptography
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

The desktop environment includes a dock with icons for the terminal, file manager, browser, and system settings.

For cipher type = aes-128-cbc

Step2:

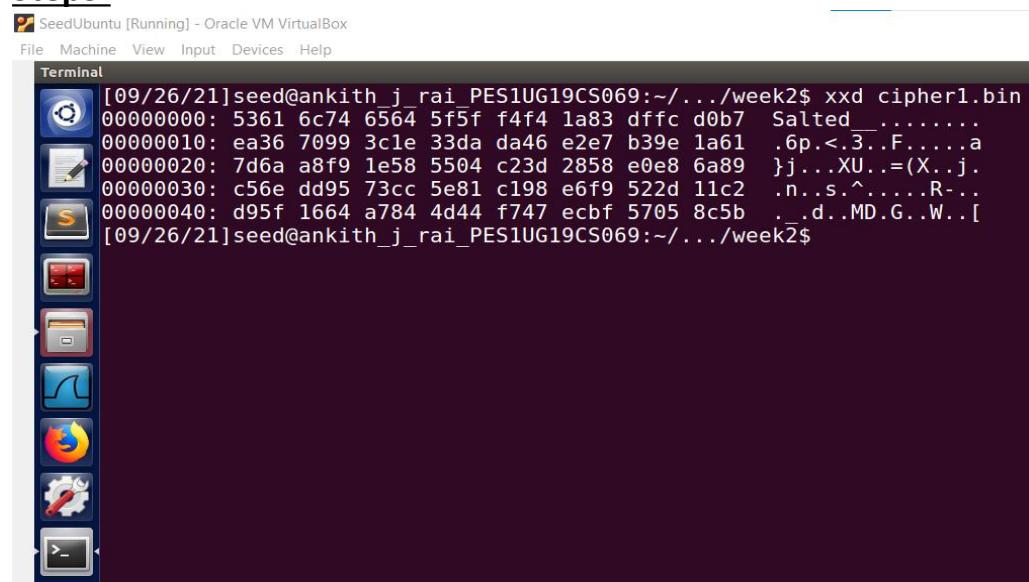


The screenshot shows a terminal window titled "Terminal" with the following command history:

```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher1.bin -k 0011223344556677889
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

The desktop environment includes a dock with icons for the terminal, file manager, browser, and system settings.

Step3:



The screenshot shows a terminal window titled "Terminal" with the following command history:

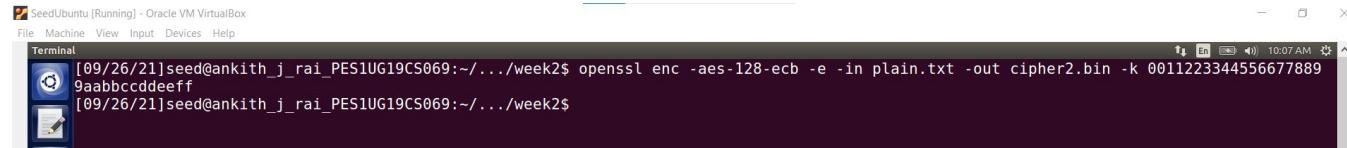
```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher1.bin
00000000: 5361 6c74 6564 5f5f f4f4 1a83 dffc d0b7 Salted.....
00000010: ea36 7099 3c1e 33da da46 e2e7 b39e 1a61 .6p.<.3..F....a
00000020: 7d6a a8f9 1e58 5504 c23d 2858 e0e8 6a89 }j...XU..=(X..j.
00000030: c56e dd95 73cc 5e81 c198 e6f9 522d 11c2 .n..s.^....R...
00000040: d95f 1664 a784 4d44 f747 ecbf 5705 8c5b ..d..MD.G..W..[
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

The desktop environment includes a dock with icons for the terminal, file manager, browser, and system settings.

The above screenshot is the cipher text of plain.txt file content for cipher type = aes128-cbc.

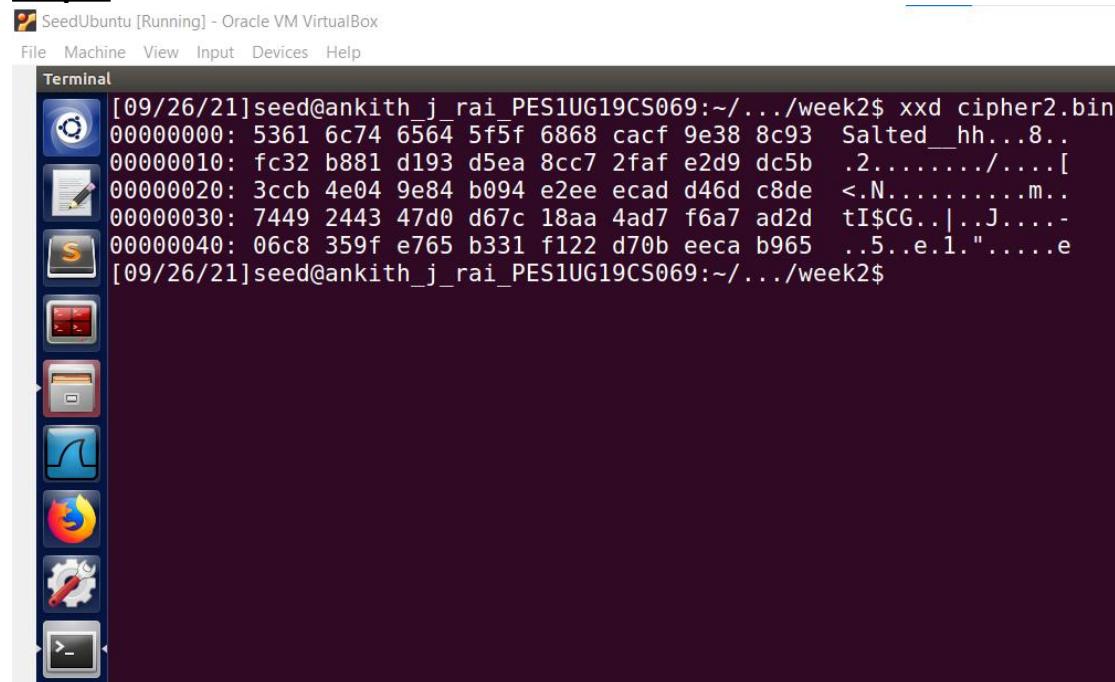
For cipher type = aes-128-ecb

Step2:



```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in plain.txt -out cipher2.bin -k 0011223344556677889
9aabcccddeeff
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Step3:

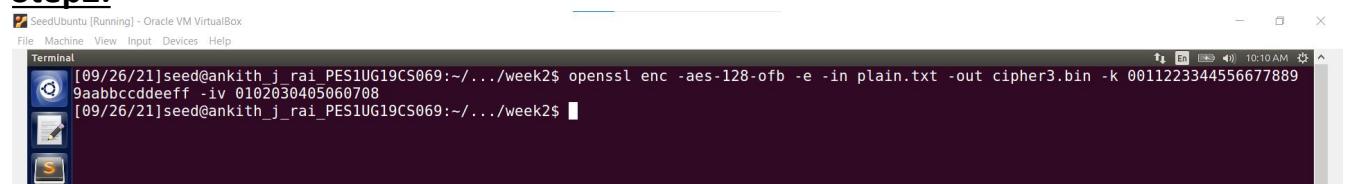


```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher2.bin
00000000: 5361 6c74 6564 5f5f 6868 cacf 9e38 8c93 Salted _hh...8..
00000010: fc32 b881 d193 d5ea 8cc7 2faf e2d9 dc5b .2...../....[_
00000020: 3ccb 4e04 9e84 b094 e2ee ecad d46d c8de <.N.....m..
00000030: 7449 2443 47d0 d67c 18aa 4ad7 f6a7 ad2d tI$CG..|..J....-
00000040: 06c8 359f e765 b331 f122 d70b eeca b965 ..5..e.l."....e
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

The above screenshot is the cipher text of plain.txt file content for cipher type = aes128-ecb

For cipher type = aes-128-ofb

Step2:



```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -e -in plain.txt -out cipher3.bin -k 0011223344556677889
9aabcccddeeff -iv 0102030405060708
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Step3:

The screenshot shows a terminal window titled 'Terminal' with the following command and output:

```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher3.bin
00000000: 5361 6c74 6564 5f5f 8efe 51d2 47eb 9f44 Salted_..Q.G..D
00000010: d664 0f4b 76a7 2902 8676 ad93 51f7 b408 .d.Kv.)..v..Q...
00000020: ad38 d33a d428 2cff e28a 506d 3abf afe8 .8.:.(,...Pm:...
00000030: d5a8 cc15 ecdf 5831 4419 dfbe 841f 2f5d .....X1D....]
00000040: bb09 590f d26f 9536 07c2 ..Y..o.6..
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

The above screenshot is the cipher text of plain.txt file content for cipher type = aes128-ofb

For cipher type = rc4

Step2:

The screenshot shows a terminal window titled 'Terminal' with the following command and output:

```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -e -in plain.txt -out cipher4.bin -k 00112233445566778899aabbcdd
deeff
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Step3:

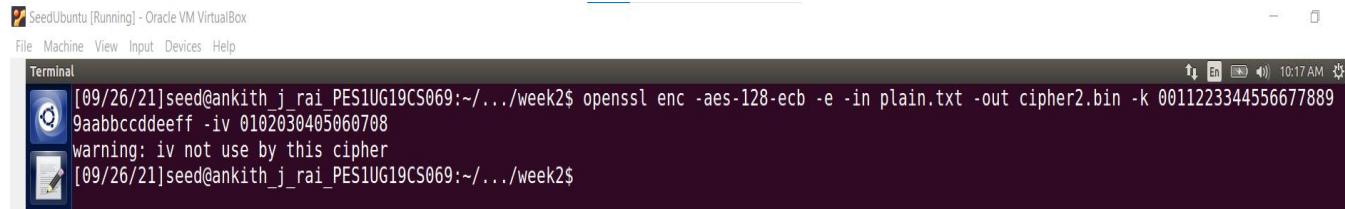
The screenshot shows a terminal window titled 'Terminal' with the following command and output:

```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher4.bin
00000000: 5361 6c74 6564 5f5f 6c8c 0705 3ff4 3f71 Salted_l...?.?q
00000010: 8a71 ea7e 5dd6 54a5 3876 7f2c 9e8a fe9a .q.~].T.8v.,....
00000020: de1e 5803 57e5 5bb1 6464 afde 3772 8257 ..X.W.[.dd..7r.W
00000030: 12bb 6590 af67 d80f bfcl 32d9 e577 d1f3 ..e..g....2..w..
00000040: eee5 edff d104 2c39 32ea .....92.
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

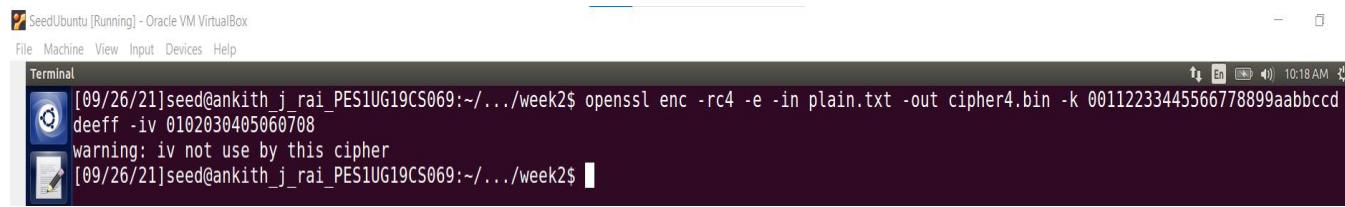
The above screenshot is the cipher text of plain.txt file content for cipher type = rc4

Q1. Which out of the above ciphers and modes do not require an initialization vector?

Ans) From doing the above task it was noted that cipher type aes-128-ecb and rc4 do not require a initialization vector.



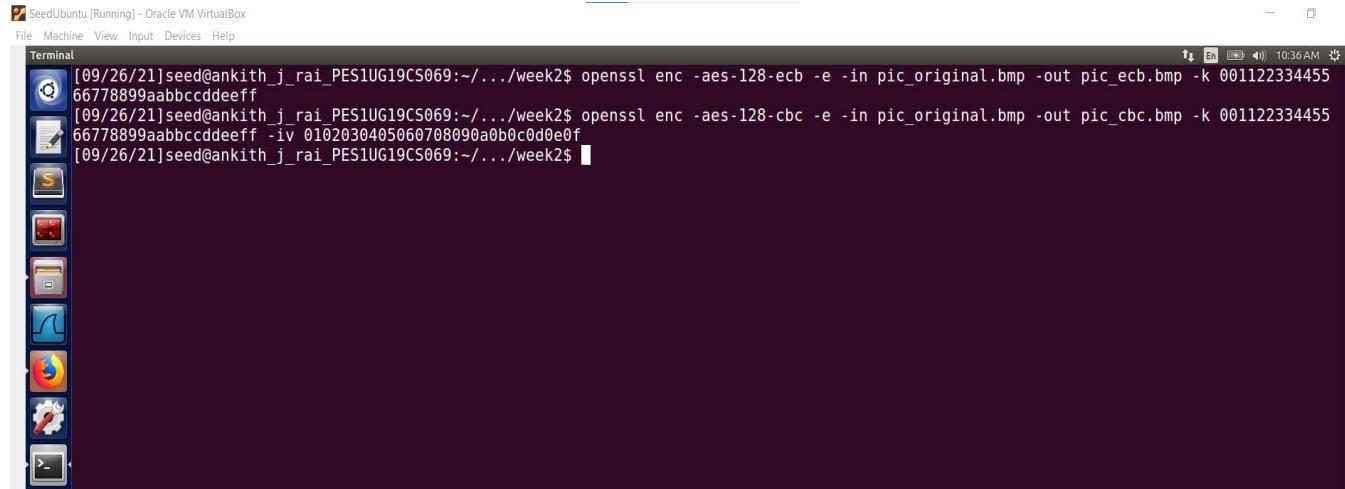
```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in plain.txt -out cipher2.bin -k 00112233445566778899aabbccddeeff -iv 0102030405060708090a0b0c0d0e0f
warning: iv not use by this cipher
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```



```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -e -in plain.txt -out cipher4.bin -k 00112233445566778899aabbccddeeff -iv 0102030405060708090a0b0c0d0e0f
warning: iv not use by this cipher
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Task 3: Encryption Mode ECB vs CBC

Step1:



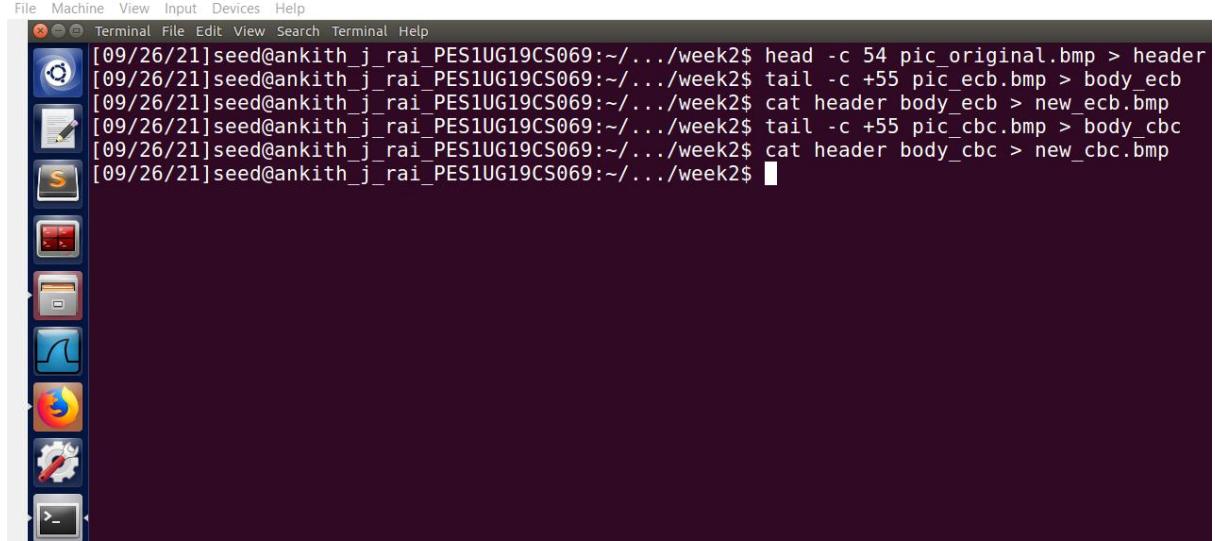
```
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in pic_original.bmp -out pic_ecb.bmp -k 00112233445566778899aabbccddeeff
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in pic_original.bmp -out pic_cbc.bmp -k 00112233445566778899aabbccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Hence from above commands pic_ecb.bmp and pic_cbc.bmp is generated.

Step2:

SeedUbuntu [Running] - Oracle VM VirtualBox

```
Terminal File Edit View Search Terminal Help
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ head -c 54 pic_original.bmp > header
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ tail -c +55 pic_ecb.bmp > body_ecb
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ cat header body_ecb > new_ecb.bmp
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ tail -c +55 pic_cbc.bmp > body_cbc
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ cat header body_cbc > new_cbc.bmp
[09/26/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

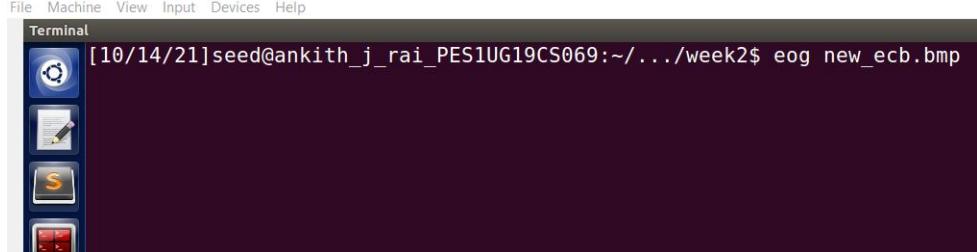


Step3:

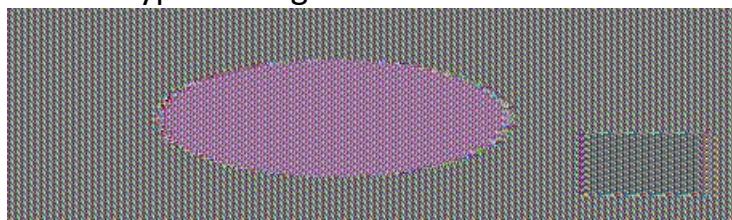
For ecb encryption:

SeedUbuntu [Running] - Oracle VM VirtualBox

```
Terminal File Edit View Input Devices Help
[10/14/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ eog new_ecb.bmp
```



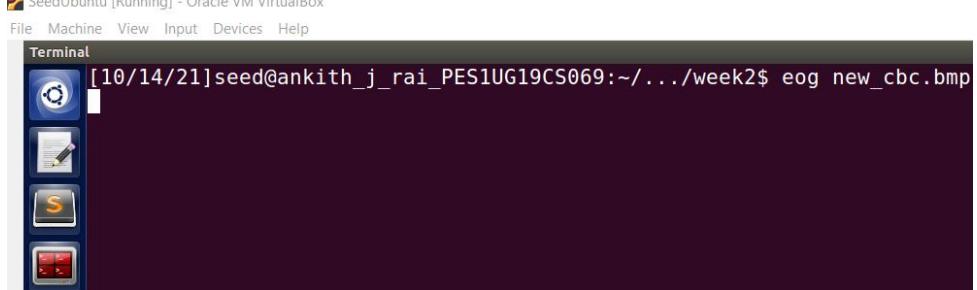
The encrypted image is:



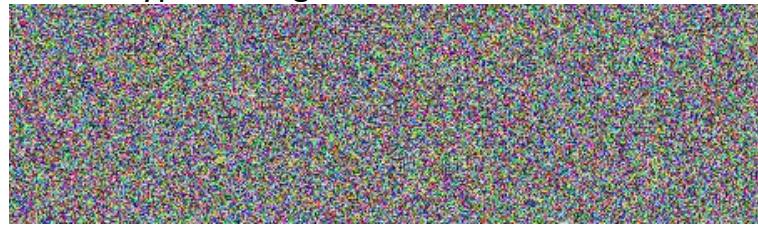
For cbc encryption:

SeedUbuntu [Running] - Oracle VM VirtualBox

```
Terminal File Edit View Input Devices Help
[10/14/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ eog new_cbc.bmp
```



The encrypted image is:



Q) Which out of the two is better and why?

Ans) From the above two encrypted images we can see that the cbc encrypted image is much more protected(i.e better) than a ecb encrypted image.

Task 4: Padding

Step1: Creating three files of size 5,10,15 bytes.

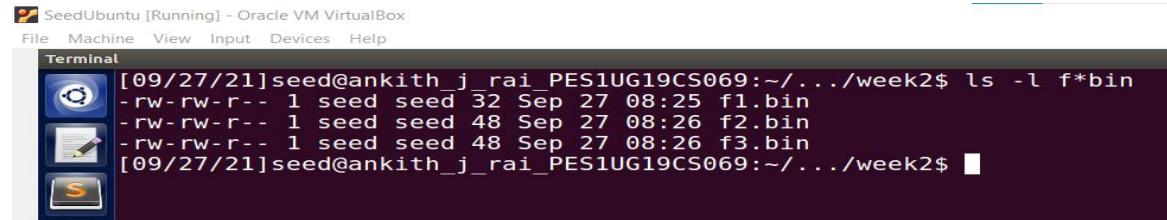
```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ echo -n "12345">f1.txt
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ echo -n "1234567890">f2.txt
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ echo -n "1234567890abcdef">f3.txt
-rw-rw-r-- 1 seed seed 11 Sep 27 08:17 f1.txt
-rw-rw-r-- 1 seed seed 16 Sep 27 08:17 f2.txt
-rw-rw-r-- 1 seed seed 22 Sep 27 08:18 f3.txt
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l f*.txt
```

Let file file f1,f2,f3 be files of size 5 bytes,10 bytes and 15 bytes respectively.

A) Step2: Using cbc mode

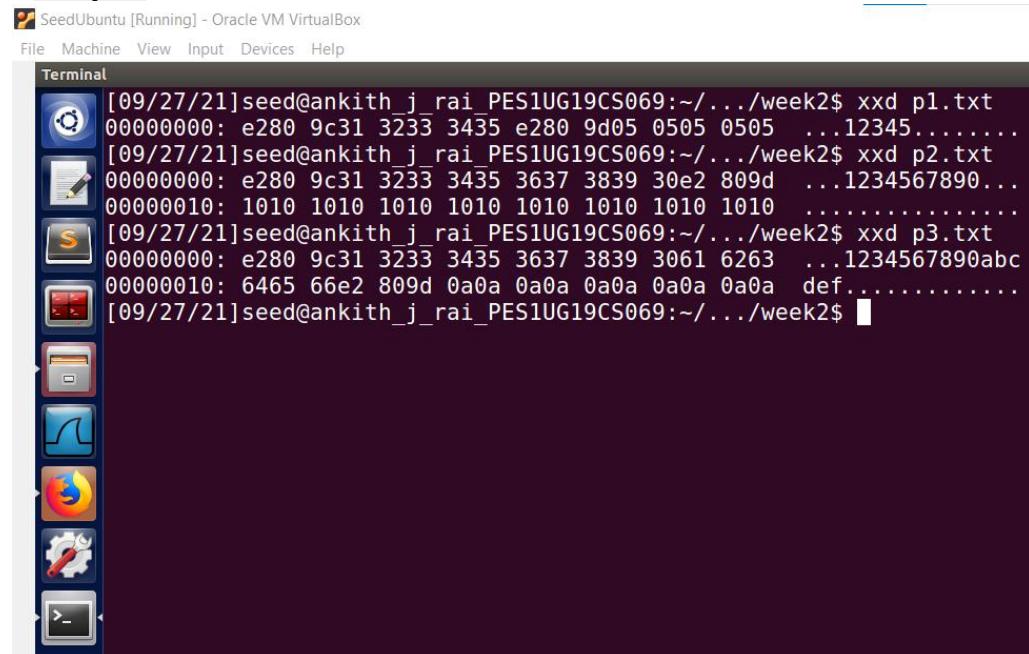
```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in f1.txt -out f1.bin -k 001122334455667788899aabcc
ddefeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in f2.txt -out f2.bin -k 001122334455667788899aabcc
ddefeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in f3.txt -out f3.bin -k 001122334455667788899aabcc
ddefeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ $openssl enc -aes-128-cbc -d -in f1.bin -out p1.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
No command 'enc' found did you mean:
 Command 'ecc' from package 'ecerc dev' (universe)
 Command 'inc' from package 'mailutils-mh' (universe)
 Command 'inc' from package 'nmh' (universe)
 Command 'nc' from package 'netcat-traditional' (universe)
 Command 'nc' from package 'netcat-openbsd' (main)
 Command 'tnc' from package 'tendra' (universe)
 Command 'env' from package 'coreutils' (main)
 Command 'enca' from package 'enca' (universe)
 Command 'znc' from package 'znc' (universe)
 Command 'eenc' from package 'refdb-clients' (universe)
 Command 'ent' from package 'ent' (universe)
enc: command not found
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -d -in f1.bin -out p1.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -d -in f2.bin -out p2.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -d -in f3.bin -out p3.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

In this step we have encrypted using cbc mode and decrypted without using the padding.



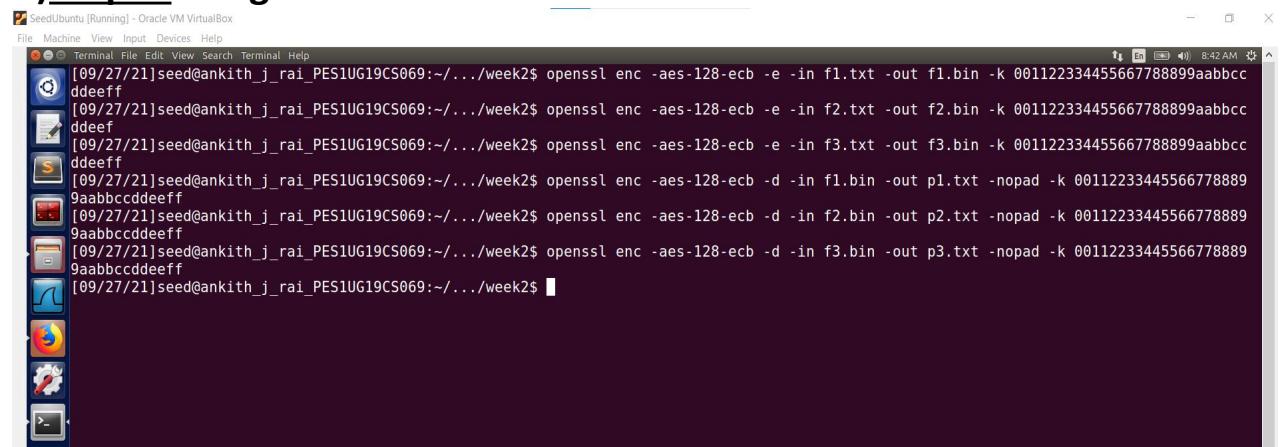
```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l f*bin
-rw-rw-r-- 1 seed seed 32 Sep 27 08:25 f1.bin
-rw-rw-r-- 1 seed seed 48 Sep 27 08:26 f2.bin
-rw-rw-r-- 1 seed seed 48 Sep 27 08:26 f3.bin
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Step3:

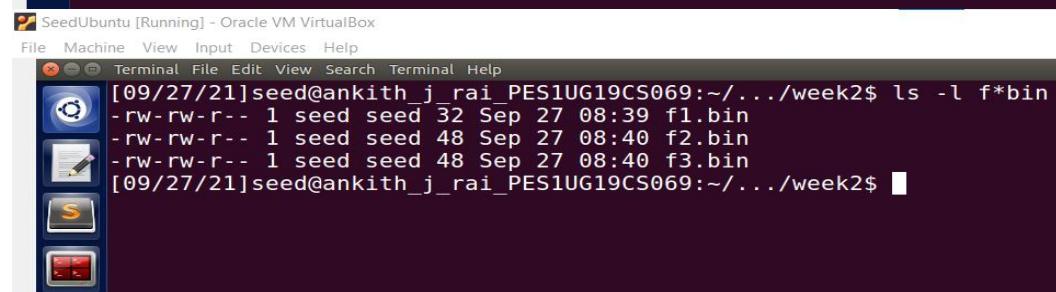


```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p1.txt
00000000: e280 9c31 3233 3435 e280 9d05 0505 0505 ...12345.....
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p2.txt
00000000: e280 9c31 3233 3435 3637 3839 30e2 809d ...1234567890...
00000010: 1010 1010 1010 1010 1010 1010 1010 ...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p3.txt
00000000: e280 9c31 3233 3435 3637 3839 3061 6263 ...1234567890abc
00000010: 6465 66e2 809d 0a0a 0a0a 0a0a 0a0a 0a0a def....
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

B) Step2: Using ecb mode



```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in f1.txt -out f1.bin -k 001122334455667788899aabccc
ddeeef
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in f2.txt -out f2.bin -k 001122334455667788899aabccc
ddeeef
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in f3.txt -out f3.bin -k 001122334455667788899aabccc
ddeeef
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -d -in f1.bin -out p1.txt -nopad -k 001122334455667788899aabccc
9aabcccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -d -in f2.bin -out p2.txt -nopad -k 001122334455667788899aabccc
9aabcccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -d -in f3.bin -out p3.txt -nopad -k 001122334455667788899aabccc
9aabcccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```



```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l f*bin
-rw-rw-r-- 1 seed seed 32 Sep 27 08:39 f1.bin
-rw-rw-r-- 1 seed seed 48 Sep 27 08:40 f2.bin
-rw-rw-r-- 1 seed seed 48 Sep 27 08:40 f3.bin
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

Step3:

SeedUbuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminal

```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p1.txt
00000000: e280 9c31 3233 3435 e280 9d05 0505 0505 ...12345.....
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p2.txt
00000000: e280 9c31 3233 3435 3637 3839 30e2 809d ...1234567890...
00000010: 1010 1010 1010 1010 1010 1010 1010 1010 ..... .
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p3.txt
00000000: e280 9c31 3233 3435 3637 3839 3061 6263 ...1234567890abc
00000010: 6465 66e2 809d 0a0a 0a0a 0a0a 0a0a 0a0a def..... .
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ █
```

C)Step2: Using ofb mode

SeedUbuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminal

```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -e -in f1.txt -out f1.bin -k 00112233445566778899aabcc
ddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -e -in f2.txt -out f2.bin -k 00112233445566778899aabcc
ddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -e -in f3.txt -out f3.bin -k 00112233445566778899aabcc
ddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -d -in f1.bin -out p1.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -d -in f2.bin -out p2.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -d -in f3.bin -out p3.txt -nopad -k 001122334455667788899aabcc
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

SeedUbuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminal

```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l f*bin
-rw-rw-r-- 1 seed seed 27 Sep 27 09:05 f1.bin
-rw-rw-r-- 1 seed seed 32 Sep 27 09:05 f2.bin
-rw-rw-r-- 1 seed seed 38 Sep 27 09:05 f3.bin
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ █
```

Step3:

SeedUbuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminal

```
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p1.txt
00000000: e280 9c31 3233 3435 e280 9d ...12345...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p2.txt
00000000: e280 9c31 3233 3435 3637 3839 30e2 809d ...1234567890...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p3.txt
00000000: e280 9c31 3233 3435 3637 3839 3061 6263 ...1234567890abc
00000010: 6465 66e2 809d def...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ █
```

D)Step2: Using rc4 mode

```

[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -e -in f1.txt -out f1.bin -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -e -in f2.txt -out f2.bin -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -e -in f3.txt -out f3.bin -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -d -in f1.bin -out p1.txt -nopad -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -d -in f2.bin -out p2.txt -nopad -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -rc4 -d -in f3.bin -out p3.txt -nopad -k 001122334455667788899aabccddeeff
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l f*bin
-rw-rw-r-- 1 seed seed 27 Sep 27 09:09 f1.bin
-rw-rw-r-- 1 seed seed 32 Sep 27 09:09 f2.bin
-rw-rw-r-- 1 seed seed 38 Sep 27 09:09 f3.bin
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$

```

Step3:

```

[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p1.txt
00000000: e280 9c31 3233 3435 e280 9d ...12345...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p2.txt
00000000: e280 9c31 3233 3435 3637 3839 30e2 809d ...1234567890...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd p3.txt
00000000: e280 9c31 3233 3435 3637 3839 3061 6263 ...1234567890abc
00000010: 6465 66e2 809d def...
[09/27/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$

```

Task 5: Error propagation – corrupted Ciphertext

Step1: Creating a file plain.txt with size more than 1000 bytes.

```

[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ touch plain.txt
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit plain.txt
^Z
[1]+ Stopped gedit plain.txt
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ls -l plain.txt
-rw-rw-r-- 1 seed seed 1510 Oct 15 08:02 plain.txt
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$

```

A) For ecb mode

Step2:

SeedUbuntu [Running] - Oracle VM VirtualBox
 File Machine View Input Devices Help

```
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -e -in plain.txt -out cipher1.bin -k 0011223344556677889
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ghex cipher1.bin
```

Before changing the bit:

cipher1.bin - GHex

0000000000	53 61 6C 74 65 64 5F 5F 6B 85 38 21 01 FB 4F 33	Salted_k.8!..03
000000010	24 3A FA 8C 95 6A 6E 0D 24 66 AB B0 88 3D 45 BB	\$:..jn.\$f...=E.
000000020	7D F7 83 77 1B 04 5C 2E 3A DD 24 D2 7B 3E D6 77	.w..\:\$.{>.w
000000030	07 92 B6 9F 04 70 F6 80 AD 6D C6 C9 59 9C 73 DAp...m..Y.s.
000000040	20 B5 FC 3C A3 6F 40 09 34 AF FC 35 49 06 F8 8A	..<.o@.4..5I...
000000050	83 9C 91 F7 2B B4 0F 65 8D A2 04 27 70 A1 F3 B3+..e...'p...
000000060	45 9D 09 9A B9 D0 14 49 0A 2F 7B 9C F6 74 64 4CEI./{..tdL
000000070	CD F8 C1 7D FD 48 09 7F B0 70 97 4E 8B 38 37 D2	..}.H...p.N.87.
000000080	20 1E D8 AC 7D 82 11 8B 0D AA 66 B2 AD 5B 54 50	...}....f...[TP
000000090	BC 90 05 B2 0A 97 F7 6F 8B FF B4 FA D7 8F 09 7Eo.....~

Signed 8 bit: 125 Signed 32 bit: 2005137277 Hexadecimal: 0D
 Unsigned 8 bit: 125 Unsigned 32 bit: 2005137277 Octal: 015
 Signed 16 bit: -2179 Signed 64 bit: 2005137277 Binary: 0111
 Unsigned 16 bit: 63357 Unsigned 64 bit: 2005137277 Stream Length: 4 - +
 Float 32 bit: 5.353207e+33 Float 64 bit: 2.253347e-85
 Show little endian decoding Show unsigned and float as hexadecimal
 Offset: 0x20

After changing the bit:

cipher1.bin - GHex

0000000000	53 61 6C 74 65 64 5F 5F 6B 85 38 21 01 FB 4F 33	Salted_k.8!..03
000000010	24 3A FA 8C 95 6A 6E 0D 24 66 AB B0 88 3D 45 BB	\$:..jn.\$f...=E.
000000020	8D F7 83 77 1B 04 5C 2E 3A DD 24 D2 7B 3E D6 77	.w..\:\$.{>.w
000000030	07 92 B6 9F 04 70 F6 80 AD 6D C6 C9 59 9C 73 DAp...m..Y.s.
000000040	20 B5 FC 3C A3 6F 40 09 34 AF FC 35 49 06 F8 8A	..<.o@.4..5I...
000000050	83 9C 91 F7 2B B4 0F 65 8D A2 04 27 70 A1 F3 B3+..e...'p...
000000060	45 9D 09 9A B9 D0 14 49 0A 2F 7B 9C F6 74 64 4CEI./{..tdL
000000070	CD F8 C1 7D FD 48 09 7F B0 70 97 4E 8B 38 37 D2	..}.H...p.N.87.
000000080	20 1E D8 AC 7D 82 11 8B 0D AA 66 B2 AD 5B 54 50	...}....f...[TP
000000090	BC 90 05 B2 0A 97 F7 6F 8B FF B4 FA D7 8F 09 7Eo.....~

Signed 8 bit: -115 Signed 32 bit: 2005137293 Hexadecimal: 0D
 Unsigned 8 bit: 141 Unsigned 32 bit: 2005137293 Octal: 015
 Signed 16 bit: -2163 Signed 64 bit: 2005137293 Binary: 1000
 Unsigned 16 bit: 63373 Unsigned 64 bit: 2005137293 Stream Length: 4 - +
 Float 32 bit: 5.353217e+33 Float 64 bit: 2.253347e-85
 Show little endian decoding Show unsigned and float as hexadecimal
 Offset: 0x20

Step3:

SeedUbuntu [Running] - Oracle VM VirtualBox
 File Machine View Input Devices Help

```
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ecb -d -in cipher1.bin -out page1.txt -nopad -k 0011223344556677889
66778899aabccddeff
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit page1.txt
Failed to register: GDBus.Error:org.freedesktop.DBus.Error.NoReply: Message recipient disconnected from message bus without replying
[1]+ Killed gedit plain.txt
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit page1.txt
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit page1.txt
```

```

SeedUbuntu [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
page1.txt (-/Desktop/cryptography/week2) - gedit
Open ▾ F Save
1 Born to Tamil Br...ms...rōōāhN...aman was a precocious child, completing his secondary and higher secondary education from St Aloysius' Anglo-Indian High School at the ages of 11 and 13, respectively. He topped the bachelor's degree examination of the University of Madras with honours in physics from Presidency College at age 16. His first research paper, on diffraction of light, was published in 1906 while he was still a graduate student. The next year he obtained a master's degree. He joined the Indian Finance Service in Calcutta as Assistant Accountant General where he made his major contributions in acoustics and optics.
2 In 1917, he was appointed as the first Palit Professor of Physics by Ashutosh Mukherjee at the Rajabazar Science College under the University of Calcutta. On his first trip to Europe, seeing the Mediterranean Sea motivated him to identify the prevailing explanation for the blue colour of the sea at the time, namely the reflected Rayleigh-scattered light from the sky, as being incorrect. He founded the Indian Journal of Physics in 1926. He moved to Bangalore in 1933 to become the first Indian director of the Indian Institute of Science. He founded the Indian Academy of Sciences the same year. He established the Raman Research Institute in 1948 where he worked to his last days.
3
4
5
6
7
8
9
10
11
12
13

```

On decrypting Cipher1.bin we can see that the decrypted file has been corrupted.

B) For cbc mode

Step2:

```

[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher1.bin -k 0011223344556677889
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ghex cipher1.bin

```

Before changing the bit:

00000000	53	61	6C	74	65	64	5F	5F	8E	C3	8F	0B	18	34	4C	A9
00000010	95	F7	98	83	91	D5	F0	EC	AA	C1	3A	C0	54	4D	F6	C6
00000020	8B	EB	18	DE	41	3A	F1	00	B0	48	BA	FF	3E	49	CF	C3
00000030	2D	29	BF	BD	CB	F4	1B	05	E7	4D	A7	E6	78	AF	D2	7B
00000040	80	27	5B	9D	CC	8B	2E	5C	9D	05	C5	A7	B0	25	34	FB
00000050	09	C9	54	93	AA	8F	B6	B4	D8	B4	63	98	03	86	16	98
00000060	D4	75	92	85	0B	9D	24	33	FC	F7	00	50	63	6B	49	CF
00000070	00	31	C3	6D	84	B0	08	8A	8C	6B	FF	DD	C3	34	53	CC
00000080	88	9A	EB	13	2C	B3	81	3F	D1	7B	0A	70	DE	34	30	D1
00000090	30	D7	0C	C8	45	34	A8	38	43	3D	38	9C	E9	F0	1D	26

Signed 8 bit: 29 Unsigned 8 bit: 29 Signed 16 bit: 10525 Unsigned 16 bit: 10525 Float 32 bit: -9.334014e-02 Show little endian decoding
 Signed 32 bit: -1111545571 Unsigned 32 bit: 3183421725 Signed 64 bit: 3183421725 Unsigned 64 bit: 3183421725 Float 64 bit: 4.700053e-284
 Hexadecimal: 1D Octal: 035 Binary: 00011101 Stream Length: 8 Show unsigned and float as hexadecimal
 Offset: 0x30

After changing the bit:

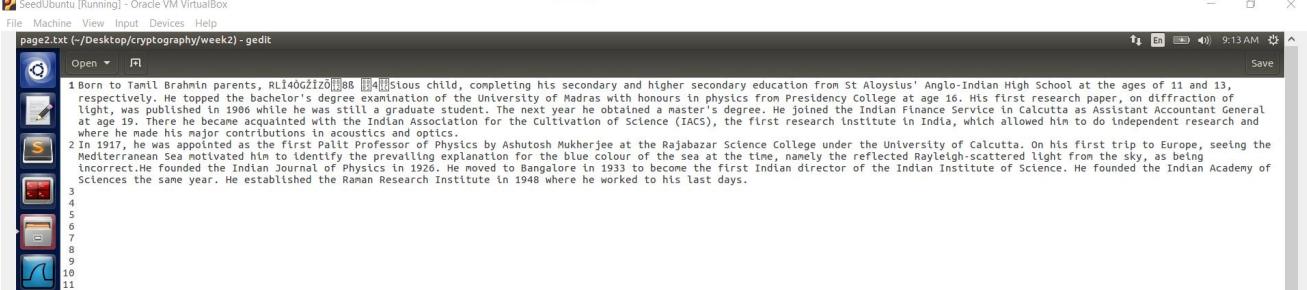
00000000	53	61	6C	74	65	64	5F	5F	8E	C3	8F	0B	18	34	4C	A9
00000010	95	F7	98	83	91	D5	F0	EC	AA	C1	3A	C0	54	4D	F6	C6
00000020	8B	EB	18	DE	41	3A	F1	00	B0	48	BA	FF	3E	49	CF	C3
00000030	2D	29	BF	BD	CB	F4	1B	05	E7	4D	A7	E6	78	AF	D2	7B
00000040	80	27	5B	9D	CC	8B	2E	5C	9D	05	C5	A7	B0	25	34	FB
00000050	09	C9	54	93	AA	8F	B6	B4	D8	B4	63	98	03	86	16	98
00000060	D4	75	92	85	0B	9D	24	33	FC	F7	00	50	63	6B	49	CF
00000070	00	31	C3	6D	84	B0	08	8A	8C	6B	FF	DD	C3	34	53	CC
00000080	88	9A	EB	13	2C	B3	81	3F	D1	7B	0A	70	DE	34	30	D1
00000090	30	D7	0C	C8	45	34	A8	38	43	3D	38	9C	E9	F0	1D	26

Signed 8 bit: 45 Unsigned 8 bit: 45 Signed 16 bit: 10541 Unsigned 16 bit: 10541 Float 32 bit: -9.334026e-02 Show little endian decoding
 Signed 32 bit: -1111545555 Unsigned 32 bit: 3183421741 Signed 64 bit: 3183421741 Unsigned 64 bit: 3183421741 Float 64 bit: 4.700053e-284
 Hexadecimal: 2D Octal: 055 Binary: 00101101 Stream Length: 8 Show unsigned and float as hexadecimal
 Offset: 0x30

Step3:



```
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -d -in cipher1.bin -out page2.txt -nopad -k 0011223344556677889
66778899aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit page2.txt
```

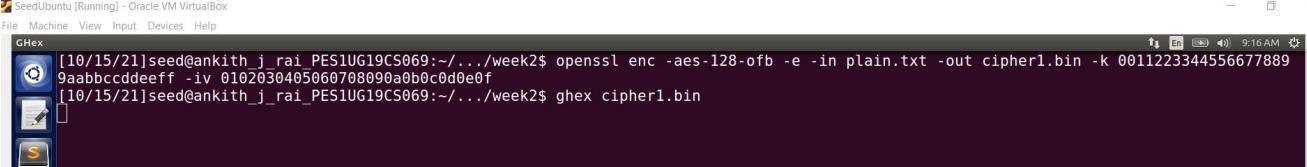


```
page2.txt (-/Desktop/cryptography/week2)-gedit
File Machine View Input Devices Help
Open ▾ | Save
1 Born to Tamil Brahmin parents, RLi40GZiZ0[...]88 [14]Stious child, completing his secondary and higher secondary education from St Aloysius' Anglo-Indian High School at the ages of 11 and 13, respectively. He topped the bachelor's degree examination of the University of Madras with honours in physics from Presidency College at age 16. His first research paper, on diffraction of light, was published in 1906 while he was still a graduate student. The next year he obtained a master's degree. He joined the Indian Finance Service in Calcutta as Assistant Accountant General at age 19. There he became acquainted with the Indian Association for the Cultivation of Science (IACS), the first research institute in India, which allowed him to do independent research and where he made his major contributions in acoustics and optics.
2 In 1919, he accepted a post as the first professor of Physics by Ashutosh Mukherjee at the Rajabazar Science College under the University of Calcutta. On his first trip to Europe, seeing the Mediterranean Sea motivated him to identify the prevailing explanation for the blue colour of the sea at the time, namely the reflected Rayleigh-scattered light from the sky, as being incorrect. He founded the Indian Journal of Physics in 1926. He moved to Bangalore in 1933 to become the first Indian director of the Indian Institute of Science. He founded the Indian Academy of Sciences the same year. He established the Raman Research Institute in 1948 where he worked to his last days.
3
4
5
6
7
8
9
10
11
12
```

On decrypting Cipher1.bin we can see that the decrypted file has been corrupted.

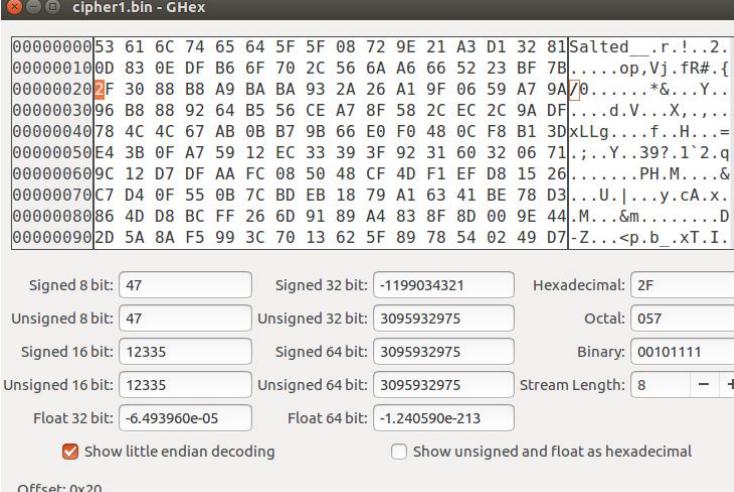
C) For ofb mode

Step2:



```
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -e -in plain.txt -out cipher1.bin -k 0011223344556677889
9aabcccddeeff -iv 0102030405060708090a0b0c0d0e0f
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ghex cipher1.bin
```

Before changing the bit:



The screenshot shows the GHex hex editor interface. The main pane displays the raw binary data of cipher1.bin, which starts with the string "Salted__r...". Below the main pane are several conversion and search controls:

- Signed 8 bit: 47
- Signed 32 bit: -1199034321
- Hexadecimal: 2F
- Unsigned 8 bit: 47
- Unsigned 32 bit: 3095932975
- Octal: 057
- Signed 16 bit: 12335
- Signed 64 bit: 3095932975
- Binary: 00101111
- Unsigned 16 bit: 12335
- Unsigned 64 bit: 3095932975
- Stream Length: 8
- Float 32 bit: -6.493960e-05
- Float 64 bit: -1.240590e-213
- Show little endian decoding (checkbox checked)
- Show unsigned and float as hexadecimal (checkbox unchecked)
- Offset: 0x20

After changing the bit:

Signed 8 bit: 63 Signed 32 bit: -1199034305 Hexadecimal: 3F
Unsigned 8 bit: 63 Unsigned 32 bit: 3095932991 Octal: 077
Signed 16 bit: 12351 Signed 64 bit: 3095932991 Binary: 00111111
Unsigned 16 bit: 12351 Unsigned 64 bit: 3095932991 Stream Length: 8 - +
Float 32 bit: -6.493972e-05 Float 64 bit: -1.240590e-213
 Show little endian decoding Show unsigned and float as hexadecimal
Offset: 0x20

Step3:

```
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-ofb -d -in cipher1.bin -out page3.txt -nopad -k 00112233445566778899  
[10/15/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gedit page3.txt
```

```
page3.txt (~/Desktop/cryptography/week2) - gedit  
Open ▾ Save  
1 Born to Tamil Brqhmin parents, Raman was a precocious child, completing his secondary and higher secondary education from St Aloysius' Anglo-Indian High School at the ages of 11 and 13, respectively. He topped the bachelor's degree examination of the University of Madras with honours in physics from Presidency College at age 16. His first research paper, on diffraction of light, was published in 1906 while he was still a graduate student. The next year he obtained a master's degree. He joined the Indian Finance Service in Calcutta as Assistant Accountant General at age 19. There he became acquainted with the Indian Association for the Cultivation of Science (IACS), the first research institute in India, which allowed him to do independent research and where he made his major contributions in acoustics and optics.  
2 In 1917, he was appointed as the first Palit Professor of Physics by Ashutosh Mukherjee at the Rajabazar Science College under the University of Calcutta. On his first trip to Europe, seeing the Mediterranean Sea motivated him to identify the prevailing explanation for the blue colour of the sea at the time, namely the reflected Rayleigh-scattered light from the sky, as being incorrect. He founded the Indian Journal of Physics in 1926. He moved to Bangalore in 1933 to become the first Indian director of the Indian Institute of Science. He founded the Indian Academy of Sciences the same year. He established the Raman Research Institute in 1948 where he worked to his last days.
```

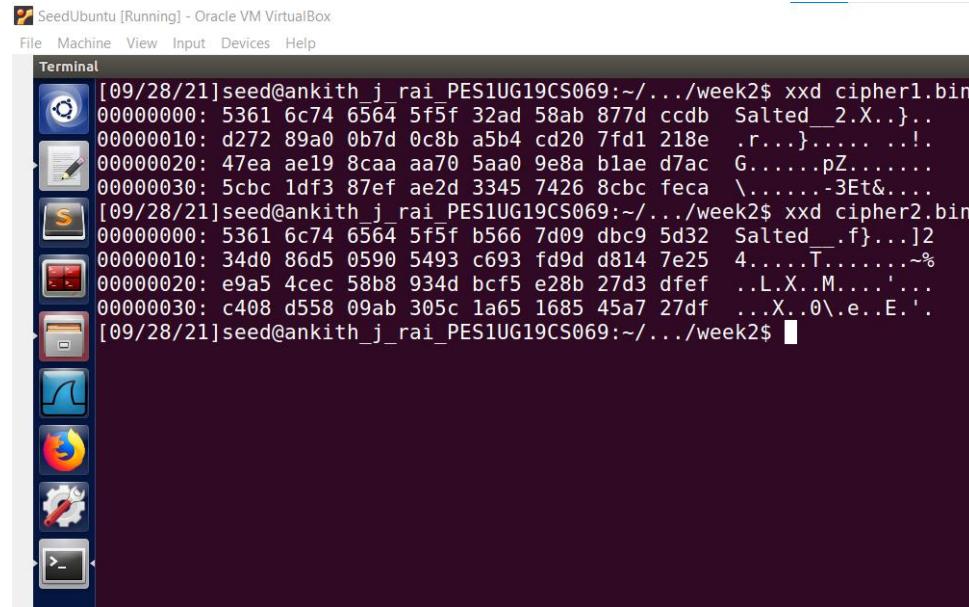
On decrypting Cipher1.bin we can see that the decrypted file has been corrupted as we can see that in the word 'Brahmin' in plain text has been changed to 'Brqhmin' in the page3.txt in decrypted file.

Task 6: Initialization Vector

Step 1: Encrypting the same plain text using different initial vector

```
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher1.bin -k 00112233445566778899  
9aabbbccddeeff -iv 0102030405060708090a0b0c0d0e0f  
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher2.bin -k 00112233445566778899  
9aabbbccddeeff -iv 102030405060708090a0b0c0d0e0f0  
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

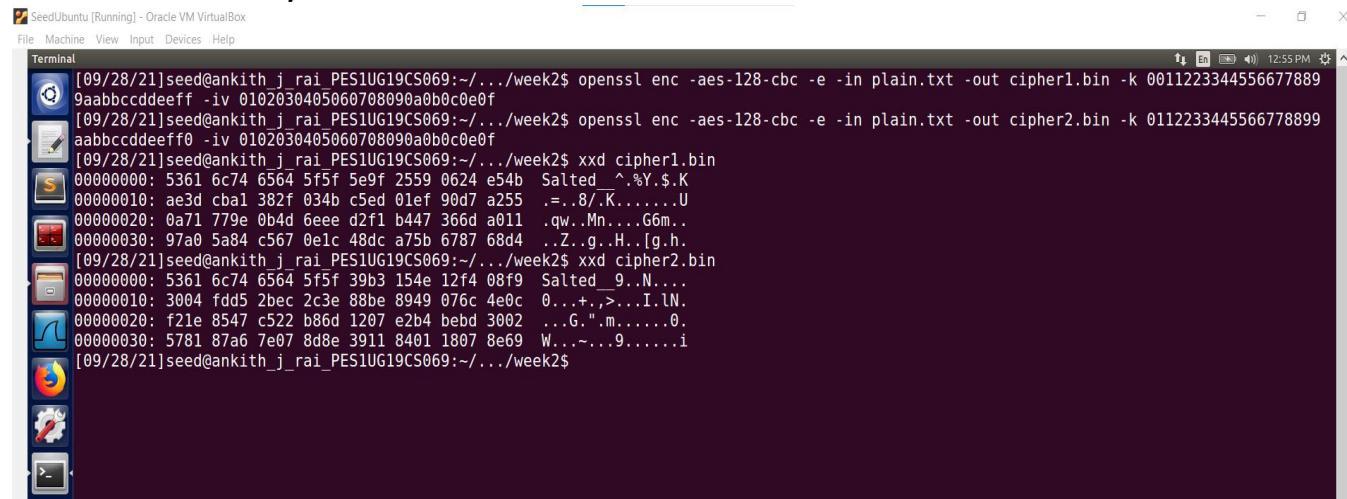
Now we have cipher1.bin which was formed using initial vector **0102030405060708090a0b0c0e0f** and we have cipher2.bin which was formed using initial vector **102030405060708090a0b0c0d0e0f0**.



```
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher1.bin
00000000: 5361 6c74 6564 5f5f 32ad 58ab 877d ccdb Salted_2.X...}.
00000010: d272 89a0 0b7d 0c8b a5b4 cd20 7fd1 218e .r...}....!.
00000020: 47ea ae19 8caa aa70 5aa0 9e8a b1ae d7ac G.....pZ.....}.
00000030: 5cbc 1df3 87ef ae2d 3345 7426 8cbc fec4 \.....-3Et&....}.
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher2.bin
00000000: 5361 6c74 6564 5f5f b566 7d09 dbc9 5d32 Salted_.f}...].2
00000010: 34d0 86d5 0590 5493 c693 fd9d d814 7e25 4.....T.....~%.
00000020: e9a5 4cec 58b8 934d bcf5 e28b 27d3 dfef ..L..X..M.....'.
00000030: c408 d558 09ab 305c 1a65 1685 45a7 27df ...X..0\.e..E.'.
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

We can see that by changing the initial vector by a small amount a completely new cipher text is generated.

Step 2: Encrypting the same plain text using same initial vector but different keys.

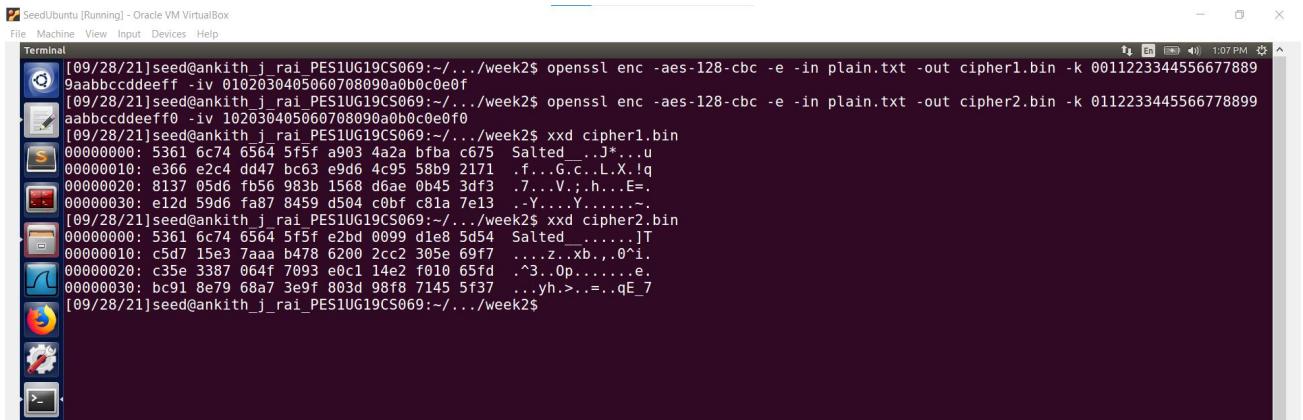


```
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher1.bin -k 00112233445566778899aabbccddeeff
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher2.bin -k 0112233445566778899aabcccddeeff0
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher1.bin
00000000: 5361 6c74 6564 5f5f 5e9f 2559 0624 e54b Salted_^.Y.$.K
00000010: ae3d cb1 382f 034b c5ed 01ef 90d7 a255 .=./K.....U
00000020: 0a71 779e 0b4d 6eee d2f1 b447 366d a011 .qw..Mn...G6m..
00000030: 97a0 5a84 c567 0e1c 48dc a75b 6787 6844 ..Z..g..H..[g..h.
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher2.bin
00000000: 5361 6c74 6564 5f5f 39b3 154e 12f4 08f9 Salted_9..N...
00000010: 3004 fdd5 2bec 2c3e 88be 8949 076c 4e0c 0...+,>..I.I.N.
00000020: f21e 8547 c522 b86d 1207 ezb4 bebd 3002 ...G.".m.....0.
00000030: 5781 87a6 7e07 8d8e 3911 8401 1807 8e69 W...~..9.....i
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

From the above screenshot we can see that cipher1.bin has been formed using key1 **00112233445566778899aabbccddeeff** and we have cipher2.bin has been formed using key2 **0112233445566778899aabcccddeeff0**.

We can see from the output in cipher1.bin and cipher2.bin that the cipher text in both of them have changed drastically even though there was only a slight change in the keys.

Step 3: Encrypting the same plain text using different initial vector and different keys.



```
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher1.bin -k 00112233445566778899aabcccddeeff -iv 0102030405060708090a0b0c0e0f
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ openssl enc -aes-128-cbc -e -in plain.txt -out cipher2.bin -k 0112233445566778899aabcccddeeff0 -iv 102030405060708090a0b0c0e0f0
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher1.bin
00000000: 5361 6c74 6564 5f5f a903 4aa2 b7ba c675 Salted__J*..u
00000010: e366 e2c3 dd47 bc63 e9d6 4c95 58b9 2171 .f..G.c..L.X.!q
00000020: 8137 05d6 fb56 983b 1568 d6ae 0b45 3df3 .7..V.;h..E=.
00000030: e12d 59d6 fa87 8459 d504 c0bf c81a 7e13 .-Y....Y.....-
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ xxd cipher2.bin
00000000: 5361 6c74 6564 5f5f e2bd 0099 d1e8 5d54 Salted_____.)T
00000010: c5d7 15e3 7aaa b478 6200 2cc2 305e 69f7 ...z.xb.,0'i.
00000020: c35e 3387 064f 7093 e0c1 14e2 f010 65fd .^3..0p.....e.
00000030: bc91 8e79 68a7 3e9f 803d 98f8 7145 5f37 ..yh.>...=..qE_7
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$
```

From the above screenshot we can see that cipher1.bin has been formed using (**key1 00112233445566778899aabcccddeeff and initial vector 0102030405060708090a0b0c0e0f**) and we have cipher2.bin has been formed using (**key2 0112233445566778899aabcccddeeff0 and initial vector 0102030405060708090a0b0c0e0f0**).

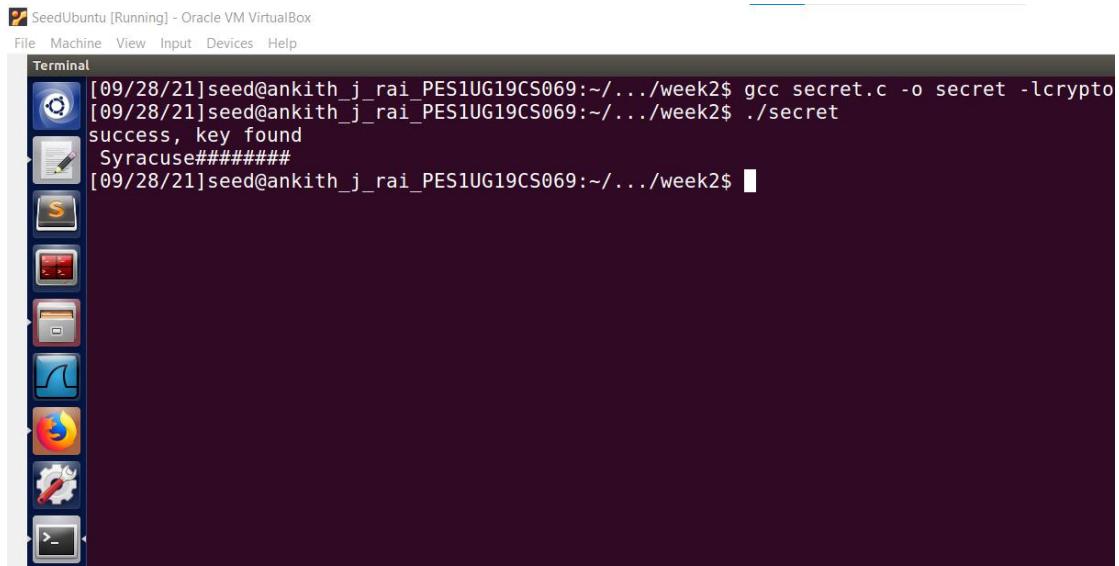
From the above screenshot we can also see that the the cipher text in both of files have changed drastically even though there was only a slight change in the keys and the initial vector.

Task 7: Programming Using the crypto Library

In this task we are going to find the key from a word list given the plain text,initial vector and cypher text.

Here we first create a secret.c file which contains the code to find the key and words.txt which is English words list.

Secret.c code is designed in such a way that the plain text is encrypted using the initial vector and every word in the word.txt(as key).This encrypted text is compared with the cipher text and if there is match of these two the word which is used to encrypt the plain text is returned as key.



The screenshot shows a terminal window titled "Terminal" running on a virtual machine named "SeedUbuntu [Running] - Oracle VM VirtualBox". The terminal window has a dark background and contains the following text:

```
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ gcc secret.c -o secret -lcrypto  
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ ./secret  
success, key found  
Syracuse#####  
[09/28/21]seed@ankith_j_rai_PES1UG19CS069:~/.../week2$ █
```

The terminal window is part of a desktop environment, as evidenced by the docked application icons on the left side of the screen.

On running `secret.c` we find that the key has been found and the key is **Syracuse#####**. The 8 # are present because the key is 16 bytes(I.e 128 bits) in length.