

Lab 02- CS458

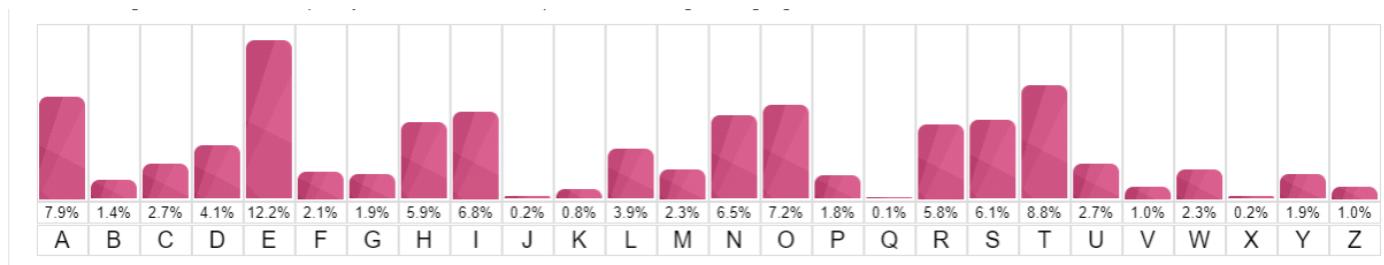
Task 1 Frequency Analysis

We are given a Cipher text and we must convert it into plain text using frequency analysis.

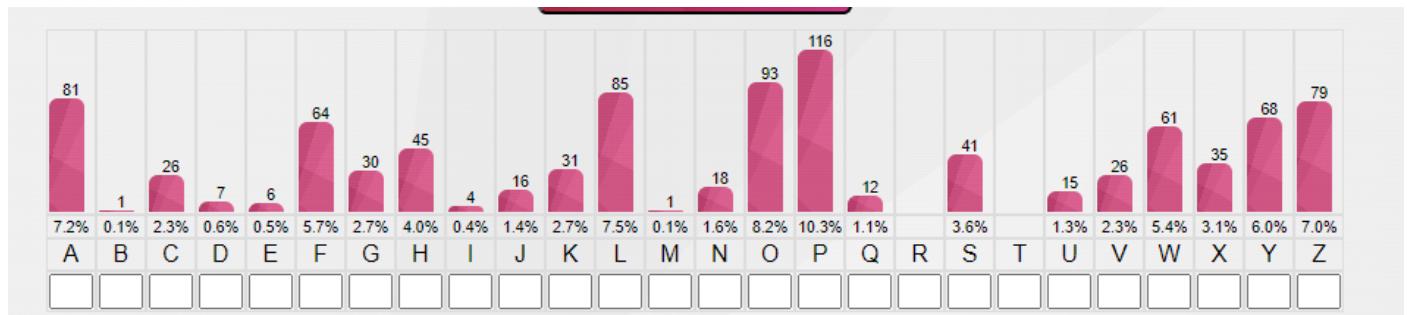
The given Ciphertext is as follows: -

hfcnkopw ahyplhp ya wznysgj hxzlvylv oxp qfwgs qyox lpq spdpgfncploa xznnplylv pdpjz
szj z wyvfwfka pskhzoyfl hfceylylv oxp oxfwj fu ylufwczoyfl zls hfcnkkozoyfl qyox
xzlsafl ajaopca zls afuoqzwp spayvl ya oxp ipj of akhhpaa za flp fu oxp fgspao hfcnkopw
ahyplhp spnzwcoploa yl oxp hxyhzvf zwpz oxp ha spnzwcoplo zo yyo xza z gflv xyaofwj fu
cppoylv oxya hxzggplvp oxwfkvx mkzgyoj pskhzoyfl ylaczgg hgzaawffc pldywflcploa zgflv
qyox ylopwlaxyn zls wpapzwhx fnnfwoklyoypa yl ylskaowj zls lzoylezg gzeffwzofwypa
yyo aoksploa qfwi qyox fkw uzhkgoj fl qfwgshgzaa wpapzwhx yl zwpza oxzo ylhgksp szoz
ahyplhp syaowyekops ajaopca ylufwczoyfl wpowypdzg hfcnkopw lpoqfwiylyl ylopggyvplo
ylufwczoyfl ajaopca zls zgvfwyoxca
oxp spnzwcoplo fuupwa ezhxpgfw fu ahyplhp czaopw fu ahyplhp nwfpupaayflzg czaopw zls nxs
spvwppa ngka vwskskzop hwoyuyhzopa zhhpgpwzops hfkwapa zls lflspvwpp aokspj nzwooycp
aoksploa hzl ozip pdplylv hgzaapa zls gflvsyaozlh aoksploa hzl pzwz czaopwa spvwppa
flgylp aoksploa wzop fkw opzhxylv za zcflv oxp epao zo oxp klydpwayoj zls fkw uzhkgoj
xzdp qfl lkcpwfka opzhxylv zqzwsa
oxp aphwpo aploplhp ya vffs bfe vkja

Given below are the frequencies of letters/alphabets A-Z. (Fig 1)



The table below shows the frequencies of the letters in the ciphertext. (Fig 2)



From the graphs above, we can infer the following: -

1. The frequency of P in Fig 2 is the highest and E in Fig 1 is the highest. So, P in ciphertext can be replaced by E, i.e., p->e
2. Similarly, the following letter with high frequency is O in Fig 2 and T in Fig 1. So, we can assume that O in the ciphertext is replaced by t, i.e., o->t
3. Most frequently used 2-letter words in the ciphertext(Fig 2) are :- ya, fu, za, yl, of, ha, zo, fl. Most frequently used 2 letter words in the English alphabet are: - an, at, as, he, be, in, is, it, on, or, to, of, do, go, no, so, my
4. Most frequently used double-letter words in the ciphertext (Fig 2) are: - a, f, g, h, n, o, p, u, y. Most frequently used doubled letter words in the English alphabet are: - s, e, t, f, l, m, o. So with this, we assume a->s

5. Frequently used 1 letter word in the ciphertext is z and frequently used letter words in the English alphabet are:- a, i. So we can replace z in ciphertext by either a or i, i.e, z->a, z->i. Hence, we conclude that za->as, zo->at
6. Frequently used 3-letter words in ciphertext are oxp, lpq, szj, zls, ipj, fip, yyo, xza, fkw, nxs, hzl, qfl, bfe, and frequently used 3-letter words in English alphabets are: - the, and, for, was, his, not, but, you, are, her. Hence, from this, we can conclude that oxp ->the and x ->h
7. Frequently used bigrams in Ciphertext are ao, yl, fl, ox, fw, lo, zo, sp, cp, op, zl, pa, pl
8. Frequently used trigrams in ciphertext are oxp, lhp, hyp, hfc, nko, azl, pao, cpl, loa, oyf

Given below are screenshots of the output

```
[03/18/23]seed@VM:~$ cd Downloads
[03/18/23]seed@VM:~/Downloads$ tr 'oxpzayql' 'THEASIWN' < Ciphertext.txt > Result.txt
[03/18/23]seed@VM:~/Downloads$ cat Result.txt
hfcnkTEw ShIENhE IS wAnIsqj hHANvINV THE Wfwgs WITH NEW sEdEgfncENTS HAnnENINv EdEwj sAj A wIvfw
fkS EskhATIfN hfceININv THE THEfwj fu INufwcATIfN ANs hfcnkTATIfN WITH HANsSfN SjSTEcS ANs SfutW
AwE sESIVn IS THE iEj Tf SkhhESS AS fNE fu THE fgsEST hfcnkTEw ShIENhE sEnAwTcENTS IN THE hHIhAv
f AwEA THE hS sEnAwTcENT AT IIT HAS A gfNv HISTfwj fu cEETINv THIS hHAggENvE ThwfkvH mkAgITj Esk
hATIfN IN ScAgg hgASSwffc ENdIwfNcENTS AgfNv WITH INTEnNSHIn ANs wESEAwH fnnfwTkNITIES IN INsks
Twj ANs NATIfNAG gAefwATfwIES IIT STksENTS Wfwj WITH fkw uAhkgTj fN WfwgshgASS wESEAwH IN AwEAS
THAT INhgksE sATA ShIENhE sSITwIekTEs SjSTEcS INufwcATIfN wETwIEdAg hfcnkTEwNETWfwjINV INTeggIV
ENT INufwcATIfN SjSTEcS ANs AgvfwITHcS THE sEnAwTcENT fuuEwS eAhHEgfw fu ShIENhE cASTeW fu ShIEN
hE nwfuESSIfNAG cASTeW ANs nHs sEvwEES ngkS vwAskATE hEwTIuIhATES AhhEgEwATEs hfkwSES ANs NfNsEv
wEE STksj nAwTTIcE STksENTS hAN TAiE EdENINv hgASSES ANs gfNvsISTANhE STksENTS hAN EAwn cASTeWs
sEvwEES fNgINE STksENTS wATE fkw TEAhHINv AS AcfNv THE eEST AT THE kNIdEwSITj THE SEhwET SENTENh
E IS vffs bfe vkjs
[03/18/23]seed@VM:~/Downloads$
```

From this we can guess words like

SjSTEcS = Systems

Tf = to

ANs = and

STksENTS = students

wATE = rate

```
[03/18/23]seed@VM:~/Downloads$ tr 'oxpzayqlhjkSuWVf' 'THEASIWCYUDFRG0' < Ciphertext.txt > Result.txt
[03/18/23]seed@VM:~/Downloads$ cat Result.txt
C0cnUTEw SCIENCE IS wAnIDgY CHANvINV THE W0wgD WITH NEW DEdEg0ncENTS HAnnENINv EdEwY DAY A wIv0w
OUS EDUCATION CoceININv THE THE0wY OF INFOwcATION AND CocnUTATION WITH HANDSON SYSTEcS AND SOFTW
AwE DESIvN IS THE iEY TO SUCCESS AS ONE OF THE 0gDEST CocnUTEw SCIENCE DEnAwTcENTS IN THE CHICAv
0 AwEA THE CS DEnAwTcENT AT IIT HAS A g0Nv HIST0wY OF cEETINv THIS CHAggENvE Thw0UvH mUAgITY EDU
CATION IN ScAgg CgASSw00c ENdIw0NcENTS Ag0Nv WITH INTEnNSHIn AND wESEAwCH Onn0wTUNITIES IN INDUS
TwY AND NATIONAg gAe0wAT0wIES IIT STUDENTS W0wi WITH OUw FACUgTY ON W0wgDCgASS wESEAwCH IN AwEAS
THAT INCgUDE DATA SCIENCE DSITwIeUTED SYSTEcS INFOwcATION wETwIEdAg CoCnUTEwNETW0wiINV INTeggIV
ENT INFOwcATION SYSTEcS AND Agv0wITHcS THE DEnAwTcENT OFFEwS eACHEg0w OF SCIENCE cASTeW OF SCIEN
CE nw0FESSIOnAg cASTeW AND nHD DEvwEES ngUS vwADUATE CEwTIFICATES ACCEgEwATED COUwSES AND NONDEv
wEE STUDY nAwTTIcE STUDENTS CAN TAiE EdENINv CgASSES AND g0NvDISTANCE STUDENTS CAN EAwn cASTeWs
DEvwEES ONgINE STUDENTS wATE OUw TEACHINv AS Ac0Nv THE eEST AT THE UNIdEwSITY THE SECwET SENTENC
E IS vOOD b0e vUYS
[03/18/23]seed@VM:~/Downloads$
```

Words like the following can be guessed.

vUYS = guys

VOOD =good

UNIdEwSITY = university

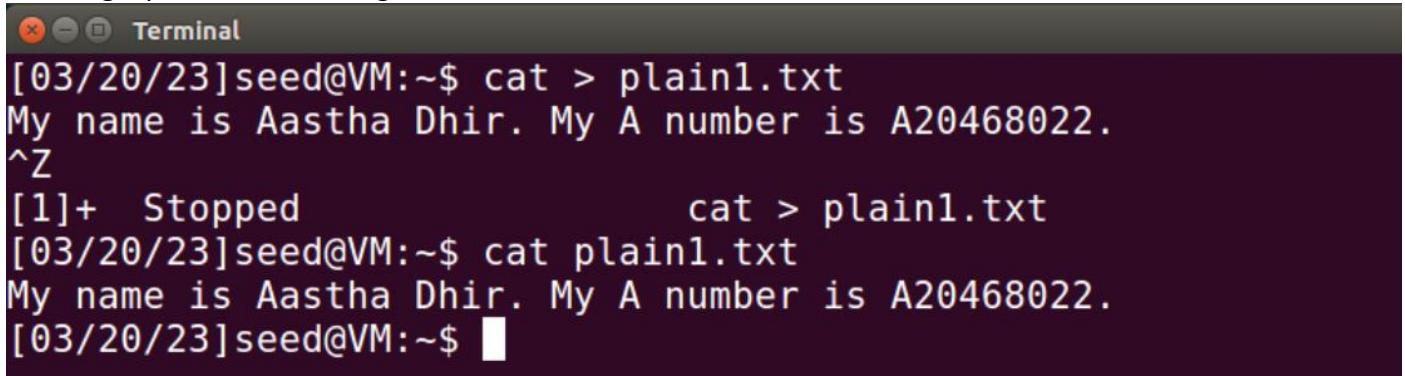
INFOwcATION = Information

Here is the final decrypted text after we have guessed all the words of the given ciphertext.

```
[03/18/23]seed@VM:~/Downloads$ tr 'oxpzayqlhjkswvfcngeidbm' 'THEASIWNCYUDFRGOMPLBKVJQ' < Cipher
text.txt > Result.txt
[03/18/23]seed@VM:~/Downloads$ cat Result.txt
COMPUTER SCIENCE IS RAPIDLY CHANGING THE WORLD WITH NEW DEVELOPMENTS HAPPENING EVERY DAY A RIGOROUS EDUCATION COMBINING THE THEORY OF INFORMATION AND COMPUTATION WITH HANDSON SYSTEMS AND SOFTWARE DESIGN IS THE KEY TO SUCCESS AS ONE OF THE OLDEST COMPUTER SCIENCE DEPARTMENTS IN THE CHICAGO AREA THE CS DEPARTMENT AT IIT HAS A LONG HISTORY OF MEETING THIS CHALLENGE THROUGH QUALITY EDUCATION IN SMALL CLASSROOM ENVIRONMENTS ALONG WITH INTERNSHIP AND RESEARCH OPPORTUNITIES IN INDUSTRY AND NATIONAL LABORATORIES IIT STUDENTS WORK WITH OUR FACULTY ON WORLDCLASS RESEARCH IN AREAS THAT INCLUDE DATA SCIENCE DISTRIBUTED SYSTEMS INFORMATION RETRIEVAL COMPUTER NETWORKING INTELLIGENT INFORMATION SYSTEMS AND ALGORITHMS THE DEPARTMENT OFFERS BACHELOR OF SCIENCE MASTER OF SCIENCE PROFESSIONAL MASTER AND PHD DEGREES PLUS GRADUATE CERTIFICATES ACCELERATED COURSES AND NONDEGREE STUDY PARTTIME STUDENTS CAN TAKE EVENING CLASSES AND LONGDISTANCE STUDENTS CAN EARN MASTERS DEGREES ONLINE STUDENTS RATE OUR TEACHING AS AMONG THE BEST AT THE UNIVERSITY THE SECRET SENTENCE IS GOOD JOB GUYS
[03/18/23]seed@VM:~/Downloads$
```

Task 2: Encryption using different Ciphers and Modes

Creating a plain text file using the cat command



```
[03/20/23]seed@VM:~$ cat > plain1.txt
My name is Aastha Dhir. My A number is A20468022.
^Z
[1]+ Stopped                  cat > plain1.txt
[03/20/23]seed@VM:~$ cat plain1.txt
My name is Aastha Dhir. My A number is A20468022.
[03/20/23]seed@VM:~$
```

For this task, I am using different ciphers and Modes and performing encryption using them.

1. AES-128-CFB - It has a block size of 128 bits.

```
[03/20/23]seed@VM:~$ openssl enc -aes-128-cfb -e -in plain1.txt -out cipher.bin -k 00010203040506070809
06070809aabcccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 5361 6c74 6564 5f5f 94d1 a711 d157 f907  Salted.....W..
00000010: 3c77 cb00 3282 0087 82a0 8bbe fb7d 9cdb <w..2.....}..
00000020: 777b 9df6 9a4d 0f2a b5d3 6896 7131 94b3 w{...M.*..h.q1..
00000030: 28fd eecf aa15 d61c fc93 82f8 bd1d b8c5 (.....
00000040: cdbb .. .
[03/20/23]seed@VM:~$
```

2. AES-256-ECB - In this no initialization vector is required and block size will be 256 bits.

```
[03/20/23]seed@VM:~$ openssl enc -aes-256-ecb -e -in plain1.txt -out cipher.bin -k 00010203040506070809aabcccddeeff
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 5361 6c74 6564 5f5f c8c6 d3dc bb4b 3cbe  Salted.....K<.
00000010: b4b5 9e3e d66e 4e7c 4bdf eef5 603a 57ad ...>.nN|K...`:W.
00000020: 8848 78fa c15c 5c4f 4f73 5193 3d67 1be5 .Hx..\00sQ.=g..
00000030: 2288 f7c1 1d7c 53e5 2078 a940 ab81 a6f6 "....|S. x.@....
00000040: ebf0 6597 fe69 ba29 3c8e 25e1 0a18 2c94 ..e..i.)<.%...,.
[03/20/23]seed@VM:~$
```

3. AES-128-CBC – In this the cipher text that is generated is 128 bits and this is block encryption with a block size of 128 bits.

```
[03/20/23]seed@VM:~$ openssl enc -aes-128-cbc -e -in plain1.txt -out cipher.bin -k 00010203040506070809
506070809aabccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 5361 6c74 6564 5f5f f732 f148 6d35 e0fa Salted_2.Hm5..
00000010: 5c89 5627 c8ce f14e 8f0d c6f7 c80a ce54 \.V'...N.....T
00000020: 861d b47d f8a7 7b4c d07c 1f8f 0e5d cd31 ...}..{L.|...].1
00000030: 940b 9aad eaca 61d4 e7d1 473d 24f5 6afb .....a...G=$.j.
00000040: 74a5 a954 b7d5 397b 6c51 b8e5 983b 815e t..T..9{lQ...;.^
[03/20/23]seed@VM:~$ █
```

4. DES-CBC – it will generate cipher text of 64 bits.

```
[03/20/23]seed@VM:~$ openssl enc -des-cbc -e -in plain1.txt -out cipher.bin -K 011223344556677
-iv 010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 2156 3182 37cb 4cb6 c7ed 9c6f 4cd2 b3a7 !V1.7.L...oL...
00000010: 05d7 6ec6 47a7 36c4 f155 b175 0893 9b48 ..n.G.6..U.u...H
00000020: 3fe2 917d 396b 472d be10 05ba f2f0 09b2 ?..}9kG-.....
00000030: 4039 08f1 f7f5 deaa @9.....
[03/20/23]seed@VM:~$ █
```

5. DES-CFB = In this if there is no plaintext, cipher text is not generated

```
[03/20/23]seed@VM:~$ openssl enc -des-cfb -e -in plain1.txt -out cipher.bin -K 011223344556677
-iv 010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 4b26 90a8 cb0a 4638 8b3e 8808 a310 98bf K&....F8.>.....
00000010: 4314 5c65 d1ac 8964 6a4e a0e9 5747 cc5f C.\e...djN..WG.-
00000020: 9b44 5411 23a7 fe0e 0eee 28ae c7b0 a6ce .DT.#....(.....
00000030: da30 .0
[03/20/23]seed@VM:~$ █
```

6. BF-CBC= It is a block cipher encryption technique with block size of 64 bits

```
[03/20/23]seed@VM:~$ openssl enc -bf-cbc -e -in plain1.txt -out cipher.bin -K 00010203040506070
809aabccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: 9ed9 a750 e987 f959 ac20 5c0e 6c8e abe9 ...P...Y. \.l...
00000010: 136f 18a9 722c 5844 8c94 5e77 15bf f016 .o...r,XD..^w....
00000020: b98d b3e2 335e 0f5f 4906 371e 2153 d116 ....3^. I.7.!S..
00000030: f80f 4c29 b644 fe6b ..L).D.K
[03/20/23]seed@VM:~$ █
```

7. BF-CFB= It is a block encryption cipher.

```
[03/20/23]seed@VM:~$ openssl enc -bf-cfb -e -in plain1.txt -out cipher.bin -K 00010203040506070
809aabccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: ee00 df03 c150 1b98 c951 4f43 3c19 795a .....P...Q0C<.yZ
00000010: dc27 baeb 395e f5bd dfa5 9bf6 9304 47ce .'..9^.....G.
00000020: 1002 6dcf 88d0 42de 2a78 f5a8 7858 2931 ..m...B.*x..xX)1
00000030: e1b5 ..
[03/20/23]seed@VM:~$
```

8. BF-OFB

```
[03/20/23]seed@VM:~$ openssl enc -bf-ofb -e -in plain1.txt -out cipher.bin -K 00010203040506070
809aabccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: ee00 df03 c150 1b98 7191 bec9 7610 2391 .....P..q...v.#.
00000010: cbac 2bbb dcfc 4fa8 b4ef f6d4 8275 7baf ..+...0.....u{.
00000020: 3e25 1e53 dab2 e88a 45ed fe8a 08a2 b3c3 >%..S....E.....
00000030: 7ec7 ~.
[03/20/23]seed@VM:~$
```

Task 3: Encryption Mode – ECB V/s CBC

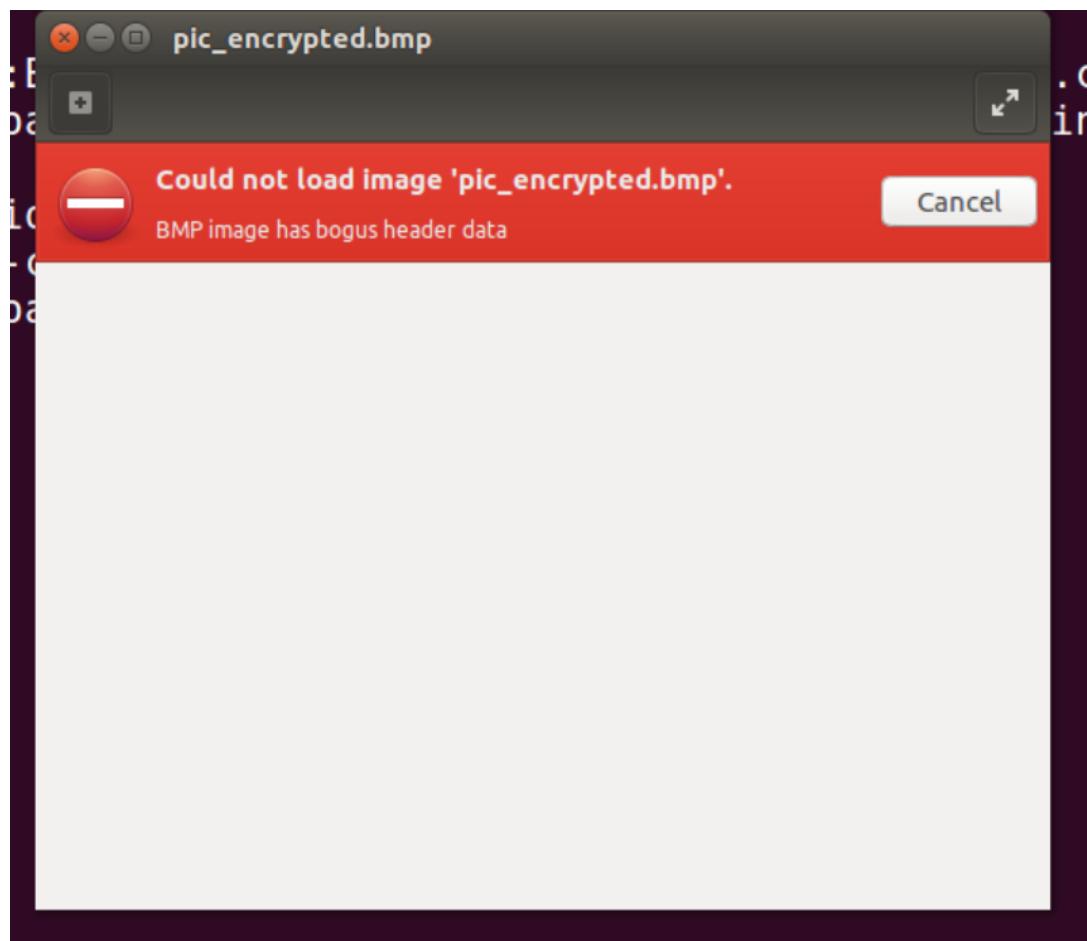
The following picture is used to demonstrate Task 3 in this assignment.



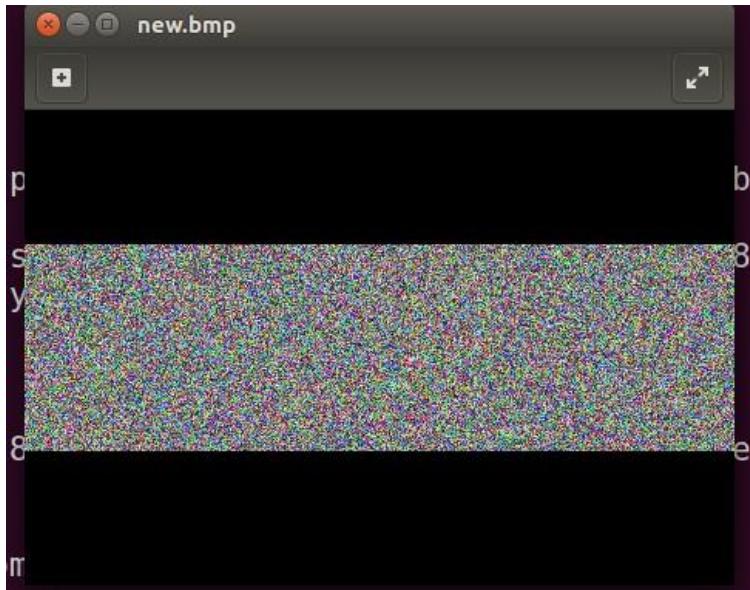
1. CBC

Encrypting picture with AES-128-CBC cipher

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -in pic_original.bmp -out pic_encrypted.bmp
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
[03/22/23]seed@VM:~/Downloads$ eog pic_encrypted.bmp
```



```
[03/22/23]seed@VM:~/Downloads$ head -c 54 pic_original.bmp > header
[03/22/23]seed@VM:~/Downloads$ tail -c +55 pic_encrypted.bmp > body
[03/22/23]seed@VM:~/Downloads$ cat header body > new.bmp
[03/22/23]seed@VM:~/Downloads$ eog pic_encrypted.bmp
[03/22/23]seed@VM:~/Downloads$ █
```

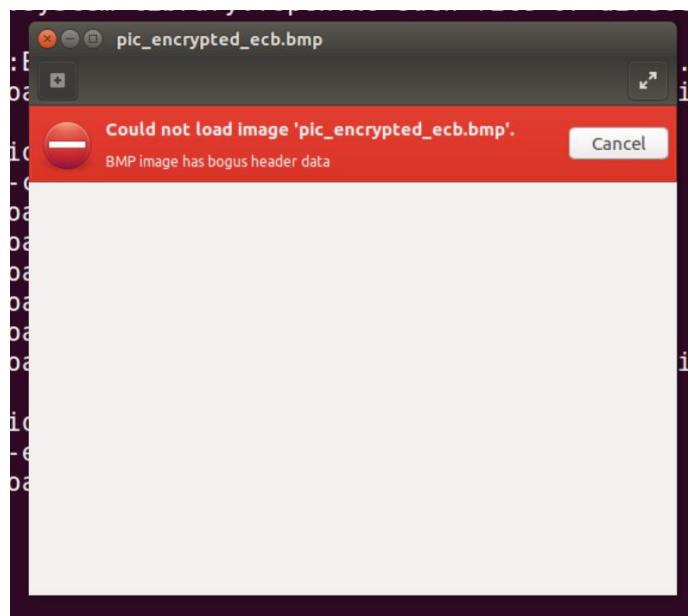


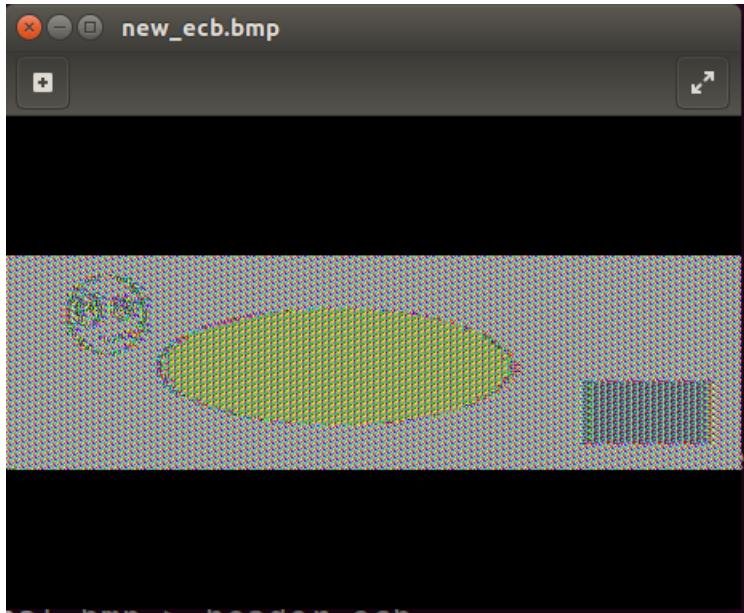
We can see that the image is completely encrypted, and we cannot identify the original image from this. This image is not at all clear.

2. ECB

Encrypting the picture with AES-128-ECB cipher

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ecb -in pic_original.bmp -out pic_encrypt  
ed_ecb.bmp  
enter aes-128-ecb encryption password:  
Verifying - enter aes-128-ecb encryption password:  
[03/22/23]seed@VM:~/Downloads$ █
```





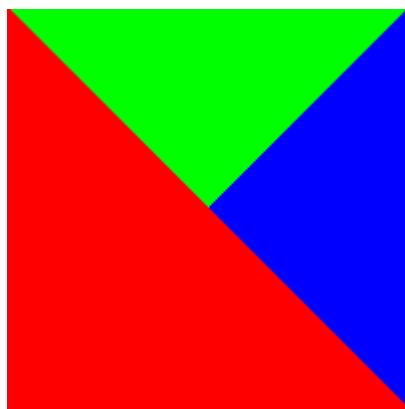
In encrypted image mode, we can still see the outline of the original image, whereas, in cipher mode (ECB), each block is encrypted separately, which results in different outputs. With CBC, the location of the block is also taken into consideration, so it is better than ECB.

Task 2.3.1

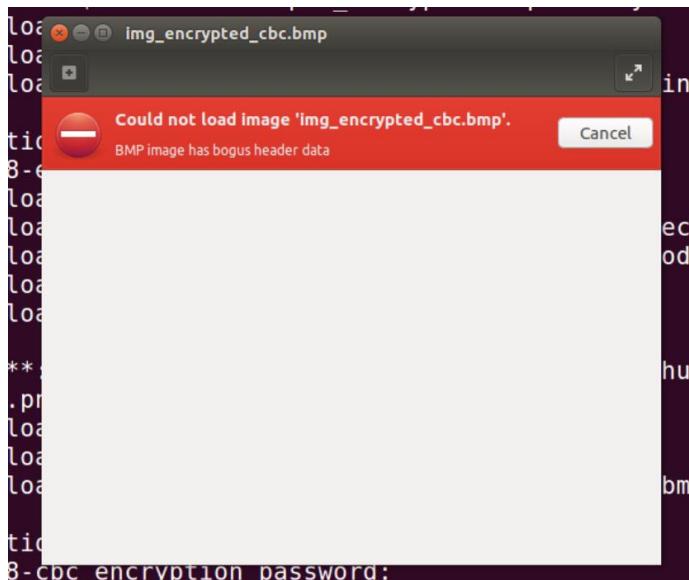
Select a picture of your choice, repeat the experiment above, and report your observations.

1. For CBC

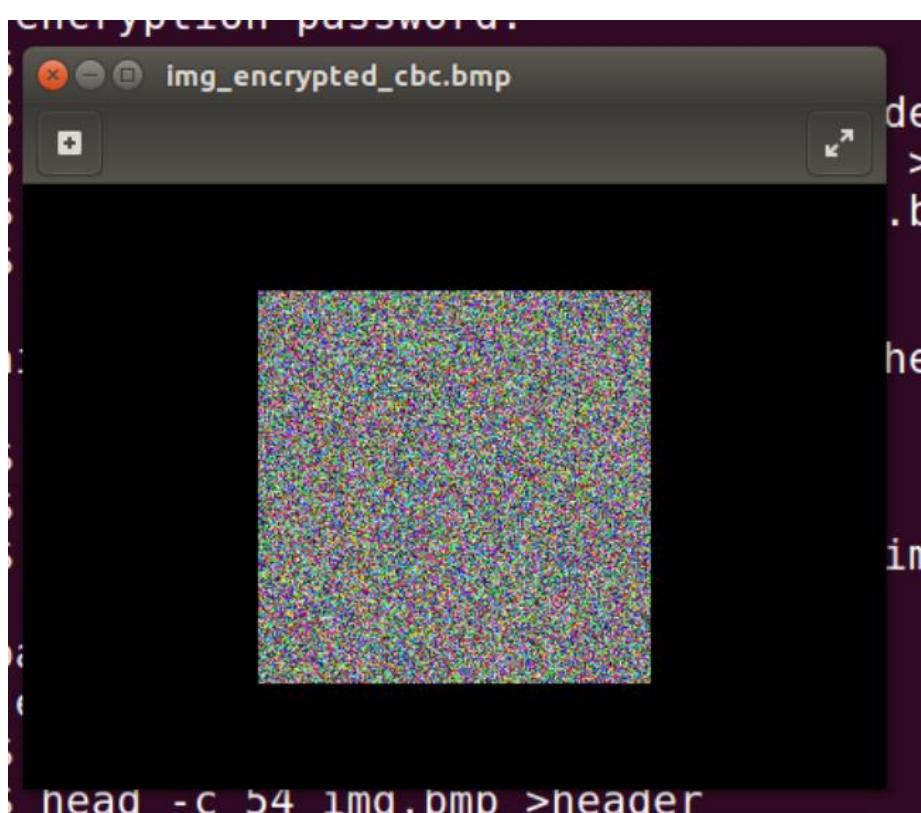
The picture that is being encrypted is as follows: -



```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in img.bmp -out img_encrypted_cbc.bmp
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
[03/22/23]seed@VM:~/Downloads$
```



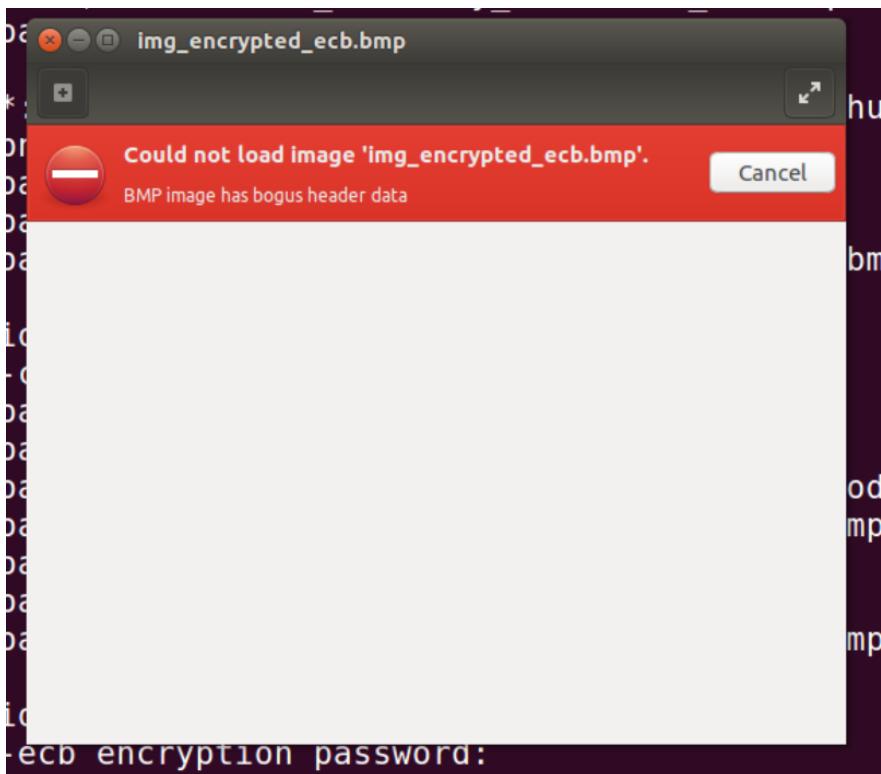
```
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_cbc.bmp  
[03/22/23]seed@VM:~/Downloads$ head -c 54 img.bmp >header  
[03/22/23]seed@VM:~/Downloads$ tail -c +55 img_encrypted_cbc.bmp > body  
[03/22/23]seed@VM:~/Downloads$ cat header body > img_encrypted_cbc.bmp  
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_cbc.bmp
```



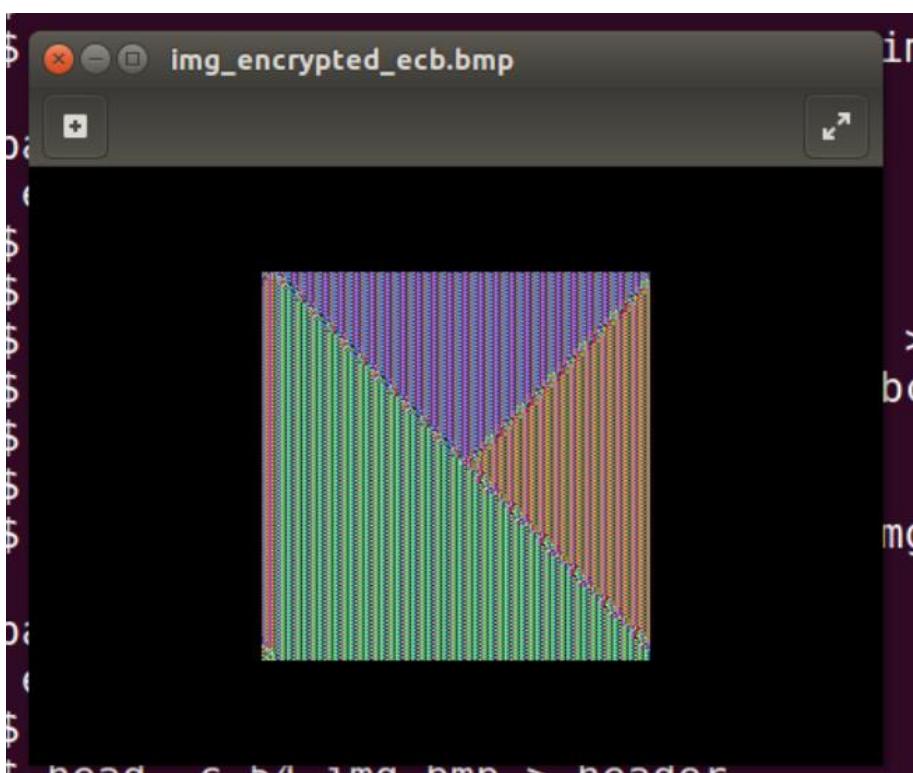
2. ECB

We are using the same picture for ECB as well.

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ecb -e -in img.bmp -out img_encrypted_ecb.bmp
enter aes-128-ecb encryption password:
Verifying - enter aes-128-ecb encryption password:
[03/22/23]seed@VM:~/Downloads$
```



```
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_ecb.bmp
[03/22/23]seed@VM:~/Downloads$ head -c 54 img.bmp > header
[03/22/23]seed@VM:~/Downloads$ tail -c +55 img_encrypted_ecb.bmp > body
[03/22/23]seed@VM:~/Downloads$ cat header body > img_encrypted_ecb.bmp
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_ecb.bmp
```



Task 4: Padding

1. Creating files of 5, 10, and 16 bytes respectively.

```
[03/22/23]seed@VM:~/Downloads$ echo -n "12345" > f1.txt
[03/22/23]seed@VM:~/Downloads$ echo -n "1234567890" > f2.txt
[03/22/23]seed@VM:~/Downloads$ echo -n "1234567890123456" > f3.txt
[03/22/23]seed@VM:~/Downloads$
```

2. Now, let's encrypt the files.

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in f1.txt -out f1_encrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in f2.txt -out f2_encrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in f3.txt -out f3_encrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$
```

```
[03/22/23]seed@VM:~/Downloads$ ls -l
total 84
-rw-rw-r-- 1 seed seed 8378 Mar 22 12:25 body
-rw-rw-r-- 1 seed seed 1075 Mar 11 14:57 Ciphertext.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:44 f1_encrypt.txt
-rw-rw-r-- 1 seed seed 5 Mar 22 13:34 f1.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:45 f2_encrypt.txt
-rw-rw-r-- 1 seed seed 10 Mar 22 13:34 f2.txt
-rw-rw-r-- 1 seed seed 32 Mar 22 13:45 f3_encrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:35 f3.txt
-rw-rw-r-- 1 seed seed 54 Mar 22 12:24 header
-rw-rw-r-- 1 seed seed 8413 Mar 22 11:23 panda.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 11:48 panda_encrypted_cbc.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 12:26 panda_encrypted_ecb.bmp
-rw-rw-r-- 1 seed seed 1075 Mar 18 21:04 Result.txt
[03/22/23]seed@VM:~/Downloads$
```

After encrypting the files f1, f2, and f3 we can see that their sizes have been changed. The files which were earlier of 5(f1.txt), 10(f2.txt), and 16(f3.txt) bytes have now become 16(f1_encrypt.txt), 16(f2_encrypt.txt), and 32(f3_encrypt.txt) bytes respectively.

3. Decryption with -nopad option

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -nopad -in f1_encrypt.txt -out f1_decrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -nopad -in f2_encrypt.txt -out f2_decrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -nopad -in f3_encrypt.txt -out f3_decrypt.txt -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
```

```
[03/22/23]seed@VM:~/Downloads$ ls -l
total 96
-rw-rw-r-- 1 seed seed 8378 Mar 22 12:25 body
-rw-rw-r-- 1 seed seed 1075 Mar 11 14:57 Ciphertext.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:56 f1_decrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:44 f1_encrypt.txt
-rw-rw-r-- 1 seed seed 5 Mar 22 13:34 f1.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:57 f2_decrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:45 f2_encrypt.txt
-rw-rw-r-- 1 seed seed 10 Mar 22 13:34 f2.txt
-rw-rw-r-- 1 seed seed 32 Mar 22 13:57 f3_decrypt.txt
-rw-rw-r-- 1 seed seed 32 Mar 22 13:45 f3_encrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:35 f3.txt
-rw-rw-r-- 1 seed seed 54 Mar 22 12:24 header
-rw-rw-r-- 1 seed seed 8413 Mar 22 11:23 panda.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 11:48 panda_encrypted_cbc.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 12:26 panda_encrypted_ecb.bmp
-rw-rw-r-- 1 seed seed 1075 Mar 18 21:04 Result.txt
[03/22/23]seed@VM:~/Downloads$
```

Here, the padding is retained even after decrypting the files f1, f2, and f3. The size of the decrypting files is the same as that of the encrypted files.

```
[03/22/23]seed@VM:~/Downloads$ xxd f1_decrypt.txt
00000000: 3132 3334 350b 0b0b 0b0b 0b0b 0b0b 0b0b 12345.....
[03/22/23]seed@VM:~/Downloads$ xxd f2_decrypt.txt
00000000: 3132 3334 3536 3738 3930 0606 0606 0606 1234567890.....
[03/22/23]seed@VM:~/Downloads$ xxd f3_decrypt.txt
00000000: 3132 3334 3536 3738 3930 3132 3334 3536 1234567890123456
00000010: 1010 1010 1010 1010 1010 1010 1010 1010 ..... .
[03/22/23]seed@VM:~/Downloads$
```

We can view the padding of the files using the hexdump command.

For f1

```
[03/22/23]seed@VM:~/Downloads$ hexdump -c f1.txt
00000000 1 2 3 4 5
00000005
[03/22/23]seed@VM:~/Downloads$ hexdump -c f1_encrypt.txt
00000000 0 215 W 0 ^ 0 003 > w 0 030 - 201 0 $
00000010
[03/22/23]seed@VM:~/Downloads$ hexdump -c f1_decrypt.txt
00000000 1 2 3 4 5 \v \v
00000010
[03/22/23]seed@VM:~/Downloads$
```

For f2

```
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2.txt
00000000 1 2 3 4 5 6 7 8 9 0
0000000a
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2_encrypt.txt
00000000 0 / 0 ~ 0 225 | 224 + 0 0 237 0 0 0
00000010
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2_decrypt.txt
00000000 1 2 3 4 5 6 7 8 9 0 006 006 006 006 006 006
00000010
[03/22/23]seed@VM:~/Downloads$
```

For f3

```
[03/22/23]seed@VM:~/Downloads$ hexdump -c f3.txt
0000000 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
0000010
[03/22/23]seed@VM:~/Downloads$ hexdump -c f3_encrypt.txt
0000000 0 : \v 0 0 0 221 236 0 0 k 023 177 020 203
0000010 > 3 ( H . * 0 b ' q 0 0 h 0 4 0
0000020
[03/22/23]seed@VM:~/Downloads$ hexdump -c f3_decrypt.txt
0000000 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
0000010 020 020 020 020 020 020 020 020 020 020 020 020 020 020 020
0000020
[03/22/23]seed@VM:~/Downloads$
```

Here we can see the extra data added for all 3 files. For f1, it's \v, for f2 it is 006 and finally, for f3 it is 020 respectively.

Task 5: Error Propagation- Corrupted Cipher Text

1. Creating a file 1000 bytes long

```
[03/22/23]seed@VM:~/Downloads$ cat plain2.txt
Once upon a time, long, long ago a king and queen ruled over a distant land. The queen was kind and lovely and all the people of the realm adored her. The only sadness in the queen's life was that she wished for a child but did not have one.

One winter day, the queen was doing needle work while gazing out her ebony window at the new fallen snow. A bird flew by the window startling the queen and she pricked her finger. A single drop of blood fell on the snow outside her window. As she looked at the blood on the snow she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow, lips as red as blood, and hair as black as ebony."

Soon after that, the kind queen got her wish when she gave birth to a baby girl who had skin white as snow, lips red as blood, and hair black as ebony. They named the baby princess Snow White, but sadly, the queen died after giving birth to Snow White.

Soon after, the king married a new woman who was beautiful, but as well proud and cruel. She had studied dark magic and owned a magic mirror, of which she would daily ask,
```

Mirror, mirror on the wall, who's the fairest of them all?

Each time this question was asked, the mirror would give the same answer, "Thou, O Queen, art the fairest of all." This pleased the queen greatly as she knew that her magical mirror cou

Since the file is too long, we can use the truncate command to reduce the size to 1000 bytes.

```
[03/22/23]seed@VM:~/Downloads$ truncate -s 1K plain2.txt
[03/22/23]seed@VM:~/Downloads$ ls -l
total 100
-rw-rw-r-- 1 seed seed 8378 Mar 22 12:25 body
-rw-rw-r-- 1 seed seed 1075 Mar 11 14:57 Ciphertext.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:56 f1_decrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:44 f1_encrypt.txt
-rw-rw-r-- 1 seed seed 5 Mar 22 13:34 f1.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:57 f2_decrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:45 f2_encrypt.txt
-rw-rw-r-- 1 seed seed 10 Mar 22 13:34 f2.txt
-rw-rw-r-- 1 seed seed 32 Mar 22 13:57 f3_decrypt.txt
-rw-rw-r-- 1 seed seed 32 Mar 22 13:45 f3_encrypt.txt
-rw-rw-r-- 1 seed seed 16 Mar 22 13:35 f3.txt
-rw-rw-r-- 1 seed seed 54 Mar 22 12:24 header
-rw-rw-r-- 1 seed seed 8413 Mar 22 11:23 panda.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 11:48 panda_encrypted_cbc.bmp
-rw-rw-r-- 1 seed seed 8432 Mar 22 12:26 panda_encrypted_ecb.bmp
-rw-rw-r-- 1 seed seed 1024 Mar 22 17:17 plain2.txt
-rw-rw-r-- 1 seed seed 1075 Mar 18 21:04 Result.txt
[03/22/23]seed@VM:~/Downloads$
```

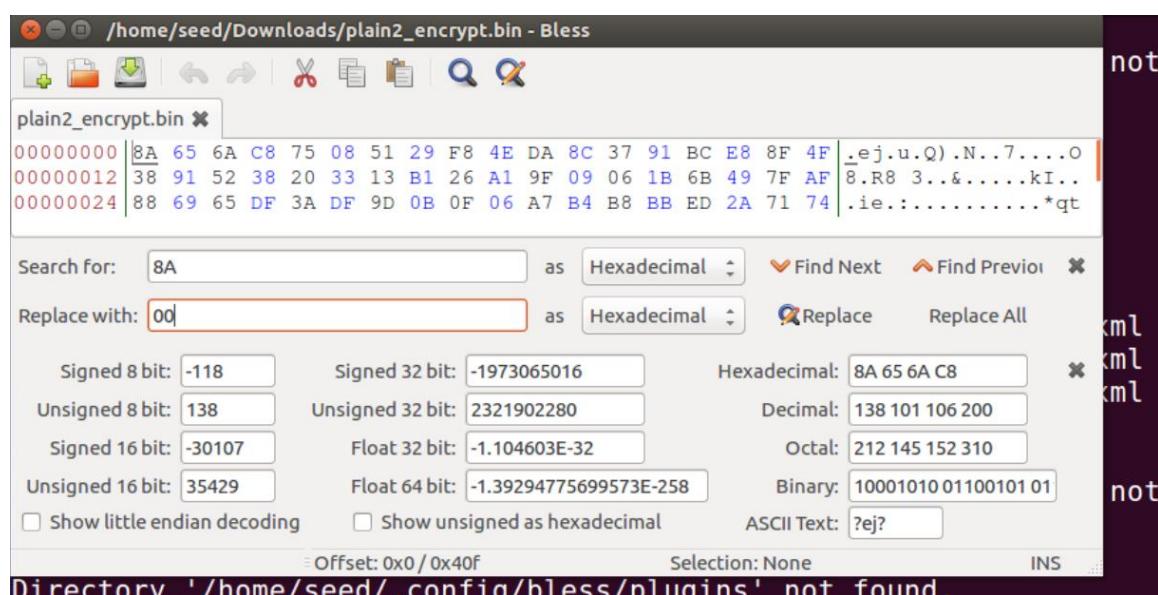
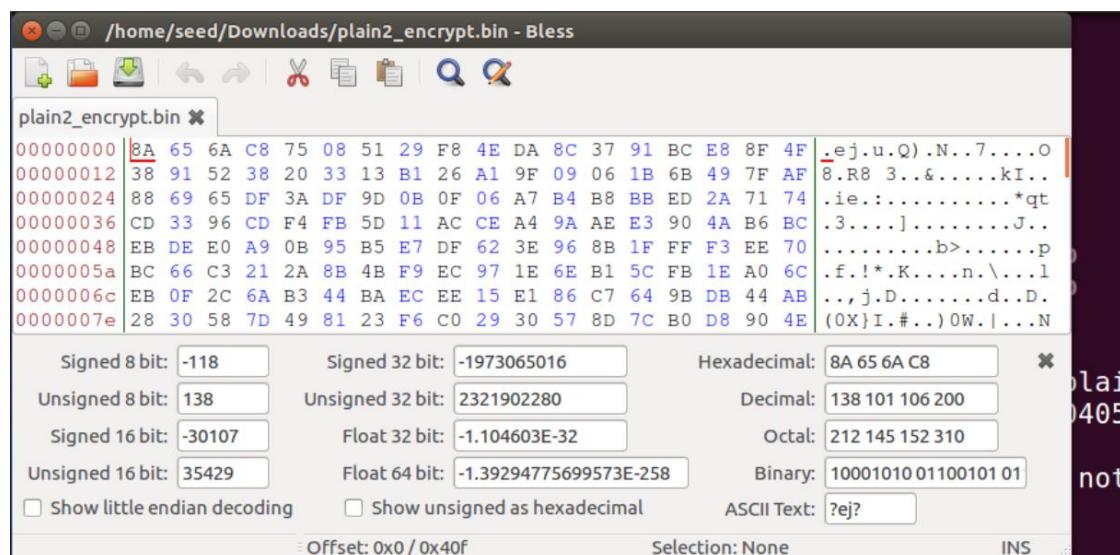
How much information can you recover by decrypting the corrupted file, if the encryption mode is ECB, CBC, CFB, or OFB, respectively?

Answer- Decryption of a corrupted file can depend on several factors, including the encryption mode used, the extent and nature of the corruption, and the availability of any additional information or context. If you're using encryption in ECB mode, it's easier to get some information out of a damaged file than if you use a different mode. This is because each block of plaintext is encrypted separately, so the damage to the file will only affect a few blocks. However, ECB mode is less secure, so it's not recommended for most applications. If you are using Cipher Block Chaining, Cipher Feedback, or Output Feedback modes, your encrypted data is likely to be corrupted and spread throughout the file. However, these modes offer stronger security than the less common Encryption Block Chaining mode, and if the corruption is limited to a specific area of the file, it may be possible to recover it using the correct key and initialization vector. If a file is corrupted, it may be possible to recover some of the information it contains, depending on the encryption mode used, the extent and nature of the corruption, and any additional information or context.

Encryption using AES-128 cipher.

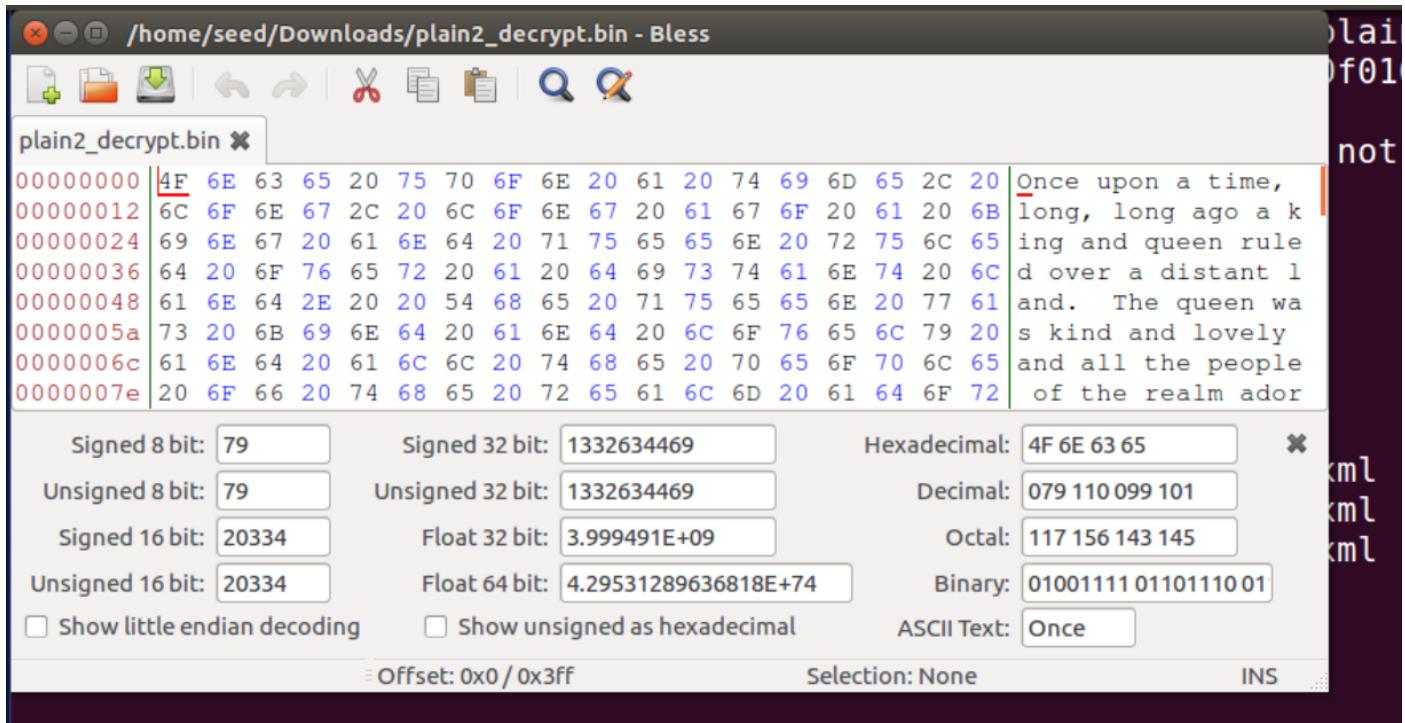
1. CBC

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain2.txt -out plain2_encrypt  
.bin -K 00010203040506070809aabccddeeff -iv 0a0b0c0d0e0f010203040506070809
```



Decryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -in plain2_encrypt.bin -out plain2_decrypt.bin -K 00010203040506070809aabccddeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ bless plain2_decrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



```
[03/22/23]seed@VM:~/Downloads$ cat plain2_decrypt.bin
?
QQ@QQ@QyQQAQQ long, long ago a king and queen ruled over a distant land. The queen was ki
nd and lovely and all the people of the realm adored her. The only sadness in the queen's li
fe was that she wished for a child but did not have one.

One winter day, the queen was doing needle work while gazing out her ebony window at the new
fallen snow. A bird flew by the window startling the queen and she pricked her finger. A si
ngle drop of blood fell on the snow outside her window. As she looked at the blood on the sn
ow she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow,
lips as red as blood, and hair as black as ebony."

Soon after that, the kind queen got her wish when she gave birth to a baby girl who had skin
white as snow, lips red as blood, and hair black as ebony. They named the baby princess Snow
White, but sadly, the queen died after giving birth to Snow White.

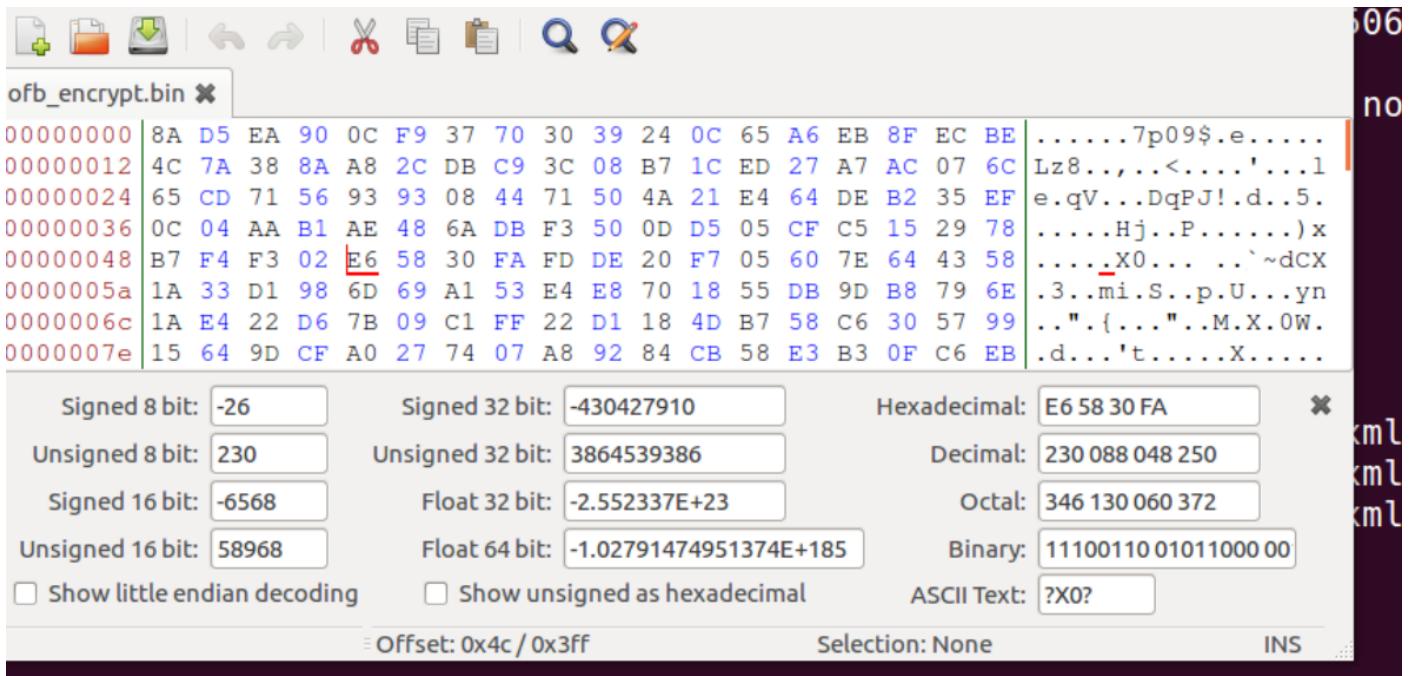
Soon after, the king married a new woman who was beautiful, but as well proud and cruel. She
ha[03/22/23]seed@VM:~/Downloads$
```

Hence, we can see that when we changed just one bit in the block of 128 bits, the whole block of data became corrupted and was lost.

2. OFB

Encryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ofb -e -in plain2.txt -out ofb_encrypt.bin -K 00010203040506070809aabbcdddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ bless ofb_encrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences.
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



Decryption

```
[03/22/23]seed@VM:~/Downloads$ 
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ofb -d -in ofb_encrypt.bin -out ofb_decrypt.bin -K 00010203040506070809aabbcdddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ bless ofb_decrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences.
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
```

/home/seed/Downloads/ofb_decrypt.bin - Bless

```
[03/22/23]seed@VM:~/Downloads$ cat ofb_decrypt.bin
Once upon a time, lon, long ago a king and queen ruled over a distant land. The queen was kind and lovely and all the people of the realm adored her. The only sadness in the queen's life was that she wished for a child but did not have one.

One winter day, the queen was doing needle work while gazing out her ebony window at the new fallen snow. A bird flew by the window startling the queen and she pricked her finger. A single drop of blood fell on the snow outside her window. As she looked at the blood on the snow she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow, lips as red as blood, and hair as black as ebony."

Soon after that, the kind queen got her wish when she gave birth to a baby girl who had skin white as snow, lips red as blood, and hair black as ebony. They named the baby princess Snow White, but sadly, the queen died after giving birth to Snow White.

[03/22/23]seed@VM:~/Downloads$
```

In this, we can see that the bit that we replaced (the corrupted bit) got a garbage value. However, the rest of the block's data remained the same.

3. CFB

Encryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cfb -e -in plain2.txt -out cfb_encrypt.bin -K 00010203040506070809aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ bless cfb_encrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences.
Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```

/home/seed/Downloads/cfb_encrypt.bin - Bless

cfb_encrypt.bin

00000000	8A D5 EA 90 0C F9 37 70 30 39 24 0C 65 A6 EB 8F 33 C57p09\$.e...3.
00000012	3A 3D B9 5F 88 09 B9 55 6E 09 F5 6D 00 B6 B9 4F 62 2A	:=._...Un..m....Ob*
00000024	A1 28 66 B7 58 A1 34 36 D9 7C 4F D2 ED C9 8F FD 34 E1	.(f.X.46. o.....4.
00000036	2D 40 16 74 3C 4B 4E CE D5 AF 02 3F A6 A9 20 61 6A 7A	-@.t<KN....?.. ajz
00000048	AA 94 27 DD 9B 16 ED EA 4A 46 47 64 64 C0 36 5D 75 B9	..'.....JFGdd.6]u.
0000005a	A5 7F 69 F5 31 E2 E5 EE 00 A0 3E 0A B5 E6 6D 0E 4B F9	..i.1.....>...m.K.
0000006c	09 C4 FB 35 F7 83 03 FA 28 F3 55 77 B5 3E B7 D5 93 32	...5....(.Uw.>....2
0000007e	E0 7A 3B 82 75 CD CB 2E 9A 7C DC DD F7 59 D2 B8 31 BE	.z;.u....Y..1.

Signed 8 bit: -118 Unsigned 8 bit: 138 Signed 16 bit: -29995 Unsigned 16 bit: 35541

Signed 32 bit: -1965692272 Unsigned 32 bit: 2329275024 Signed 32 bit: -2.059939E-32 Unsigned 32 bit: 1.82452642453778E-256

Hexadecimal: 8AD5EA90 Decimal: 138213234144 Octal: 212325352220 Binary: 100010101101010111

Show little endian decoding Show unsigned as hexadecimal ASCII Text: ????

Offset: 0x0 / 0x3ff Selection: None INS

Directory '/home/seed/.config/bless/plugins' not found.

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cfb -d -in cfb_encrypt.bin -out cfb_decrypt.bin -K 00010203040506070809aabcccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ bless cfb_decrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```

Bless /home/seed/Downloads/cfb_decrypt.bin - Bless

cfb_decrypt.bin

00000000	C5 6E 63 65 20 75 70 6F 6E 20 61 20 74 69 6D 65 1F 9F	nce upon a time..
00000012	D4 03 15 4C 57 03 5F 78 D0 8A 35 FE D9 32 20 61 20 6B	...LW._x..5..2 a k
00000024	69 6E 67 20 61 6E 64 20 71 75 65 65 6E 20 72 75 6C 65	ing and queen rule
00000036	64 20 6F 76 65 72 20 61 20 64 69 73 74 61 6E 74 20 6C	d over a distant l
00000048	61 6E 64 2E 20 20 54 68 65 20 71 75 65 65 6E 20 77 61	and. The queen wa
0000005a	73 20 6B 69 6E 64 20 61 6E 64 20 6C 6F 76 65 6C 79 20	s kind and lovely
0000006c	61 6E 64 20 61 6C 6C 20 74 68 65 20 70 65 6F 70 6C 65	and all the people
0000007e	20 6F 66 20 74 68 65 20 72 65 61 6C 6D 20 61 64 6F 72	of the realm ador

Signed 8 bit: -59 Signed 32 bit: -982621339 Hexadecimal: C5 6E 63 65
 Unsigned 8 bit: 197 Unsigned 32 bit: 3312345957 Decimal: 197 110 099 101
 Signed 16 bit: -14994 Float 32 bit: -3814.212 Octal: 305 156 143 145
 Unsigned 16 bit: 50542 Float 64 bit: -2.93897234642349E+26 Binary: 11000101 01101110 01
 Show little endian decoding Show unsigned as hexadecimal ASCII Text: ?nce

Offset: 0x0 / 0x3ff Selection: None INS

Directory '/home/seed/.config/bless/plugins' not found

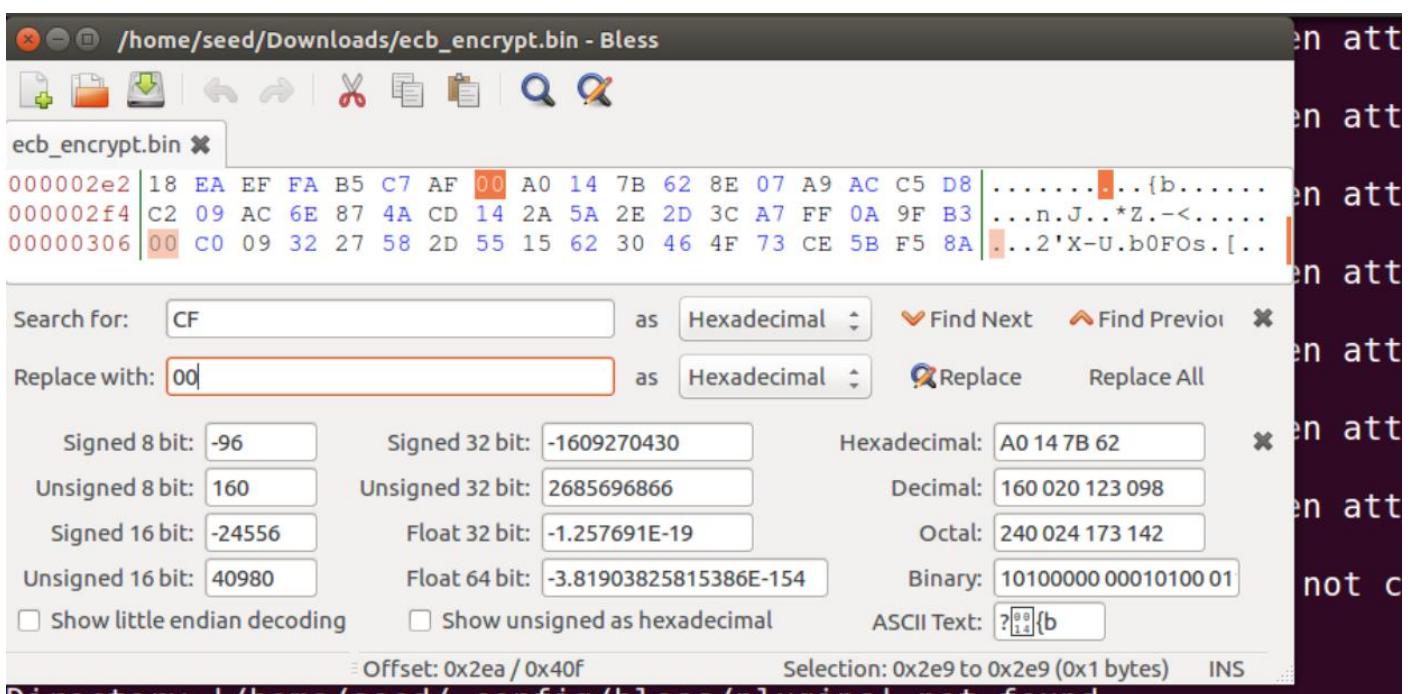
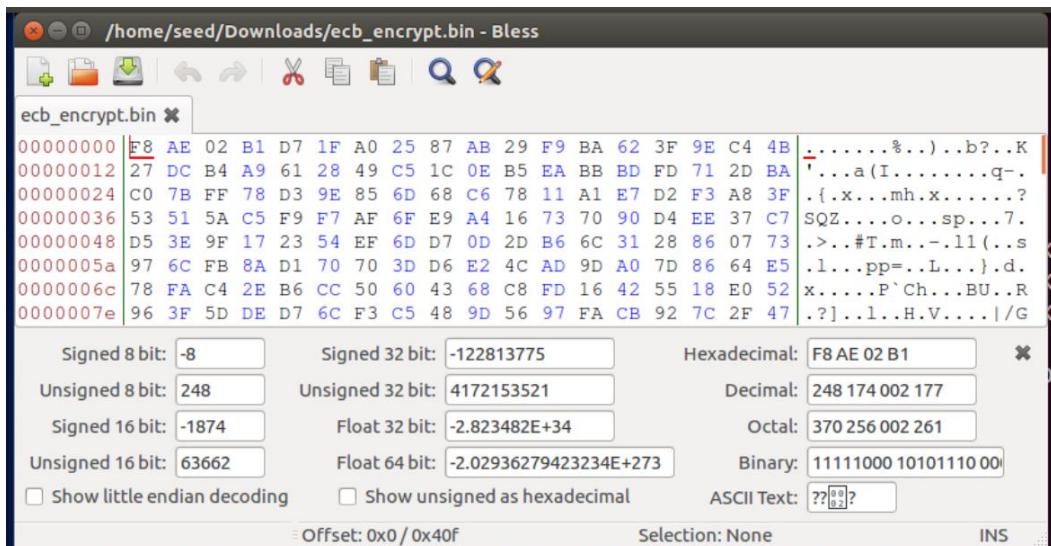
```
[03/22/23]seed@VM:~/Downloads$ cat cfb_decrypt.bin
Once upon a time a king and queen ruled over a distant land. The queen was kind and lovely and all the people of the realm adored her. The only sadness on the g
zoo's that she wished for a child but
she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow, lips as red as blood, and hair as black as ebony."
When she gave birth to a baby girl who had skin white as snow, lips red as blood, and hair black as ebony. They named her Snow White.
Soon after, the king married a new woman who was beautiful, but as well proud and cruel. She ha[03/22/23]seed@VM:~/Downloads$
```

In this one, we can see that when we changed just one bit in our block of data, it became corrupted and lost its information.

4. ECB

Encryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ecb -e -in plain2.txt -out ecb_encrypt.bin -K 00010203040506070809aabccddeeff
[03/22/23]seed@VM:~/Downloads$ bless ecb_encrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences .
Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
```



Decryption

```
[03/22/23]seed@VM:~/Downloads$ 
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128 ECB -d -in ecb_encrypt.bin -out ecb_decrypt.bin -K 00010203040506070809aabccddeeff
[03/22/23]seed@VM:~/Downloads$ bless ecb_decrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```

Bless /home/seed/Downloads/ecb_decrypt.bin - Bless

ecb_decrypt.bin x

00000000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 74 69 6D 65 2C 20	Once upon a time, long, long ago a k
00000012	6C 6F 6E 67 2C 20 6C 6F 6E 67 20 61 67 6F 20 61 20 6B	ing and queen rule d over a distant l
00000024	69 6E 67 20 61 6E 64 20 71 75 65 65 6E 20 72 75 6C 65	and. The queen wa
00000036	64 20 6F 76 65 72 20 61 20 64 69 73 74 61 6E 74 20 6C	s kind and lovely
00000048	61 6E 64 2E 20 20 54 68 65 20 71 75 65 65 6E 20 77 61	and all the people
0000005a	73 20 6B 69 6E 64 20 61 6E 64 20 6C 6F 76 65 6C 79 20	of the realm ador
0000006c	61 6E 64 20 61 6C 6C 20 74 68 65 20 70 65 6F 70 6C 65	
0000007e	20 6F 66 20 74 68 65 20 72 65 61 6C 6D 20 61 64 6F 72	

Signed 8 bit: 79 Unsigned 8 bit: 79 Signed 16 bit: 20334 Unsigned 16 bit: 20334 Show little endian decoding

Signed 32 bit: 1332634469 Unsigned 32 bit: 1332634469 Float 32 bit: 3.999491E+09 Float 64 bit: 4.29531289636818E+74 Show unsigned as hexadecimal

Hexadecimal: 4F 6E 63 65 Decimal: 079 110 099 101 Octal: 117 156 143 145 Binary: 01001111 01101110 01 ASCII Text: Once

Offset: 0x0 / 0x3ff Selection: None INS

Directory '/home/seed/.config/bless/plugins' not found

```
[03/22/23]seed@VM:~/Downloads$ cat ecb_decrypt.bin
Once upon a time, long, long ago a king and queen ruled over a distant land. The queen was kind and lovely and all the people of the realm adored her. The only sadness in the queen's life was that she wished for a child but did not have one.

One winter day, the queen was doing needle work while gazing out her ebony window at the new fallen snow. A bird flew by the window startling the queen and she pricked her finger. A single drop of blood fell on the snow outside her window. As she looked at the blood on the snow she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow, lips as red as blood, and hair as black as ebony."

Soon after that, the kind queen got her wish when she gave birth to a baby girl who had skin wh0000K@`q@*0Las red as blood, and hair black as ebony. They named the baby princess Snow White, but sadly, the queen died after giving birth to Snow White.

Soon after, the king married a new woman who was beautiful, but as well proud and cruel. She
```

In this, we can see that when we changed just one bit in a block of data, the block became corrupted and lost its information.

From this, I conclude that, for modes ECB, CBC, and CFB, if even one bit is corrupted, the data of the entire block containing that bit gets lost. But for OFB, only the corrupted bit will have garbage values, and the rest of the data can be recovered without any loss.

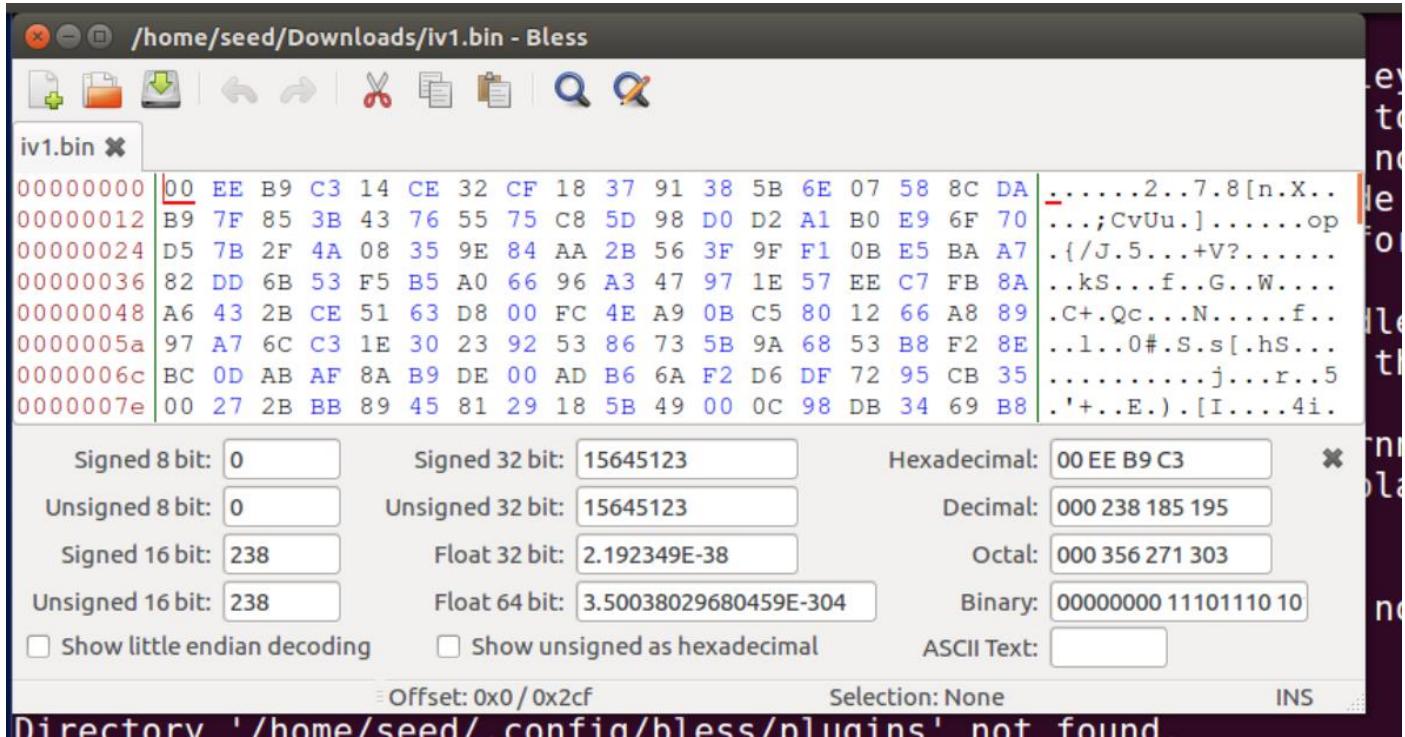
Task 6: Initial Vector (IV)

Task 6.1: Uniqueness of IVs

- Using different IV's

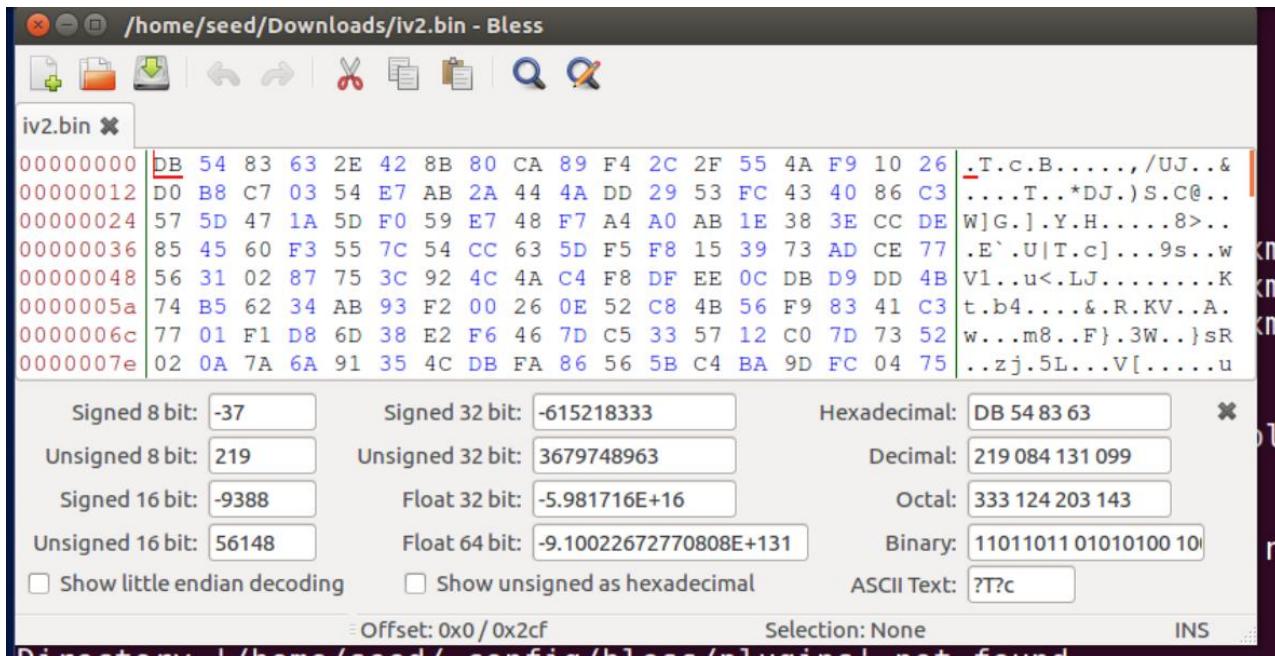
IV 1

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out iv1.bin -K 000
10203040506070809aabbccddeeff -iv 010203040506
[03/22/23]seed@VM:~/Downloads$ bless iv1.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



IV2

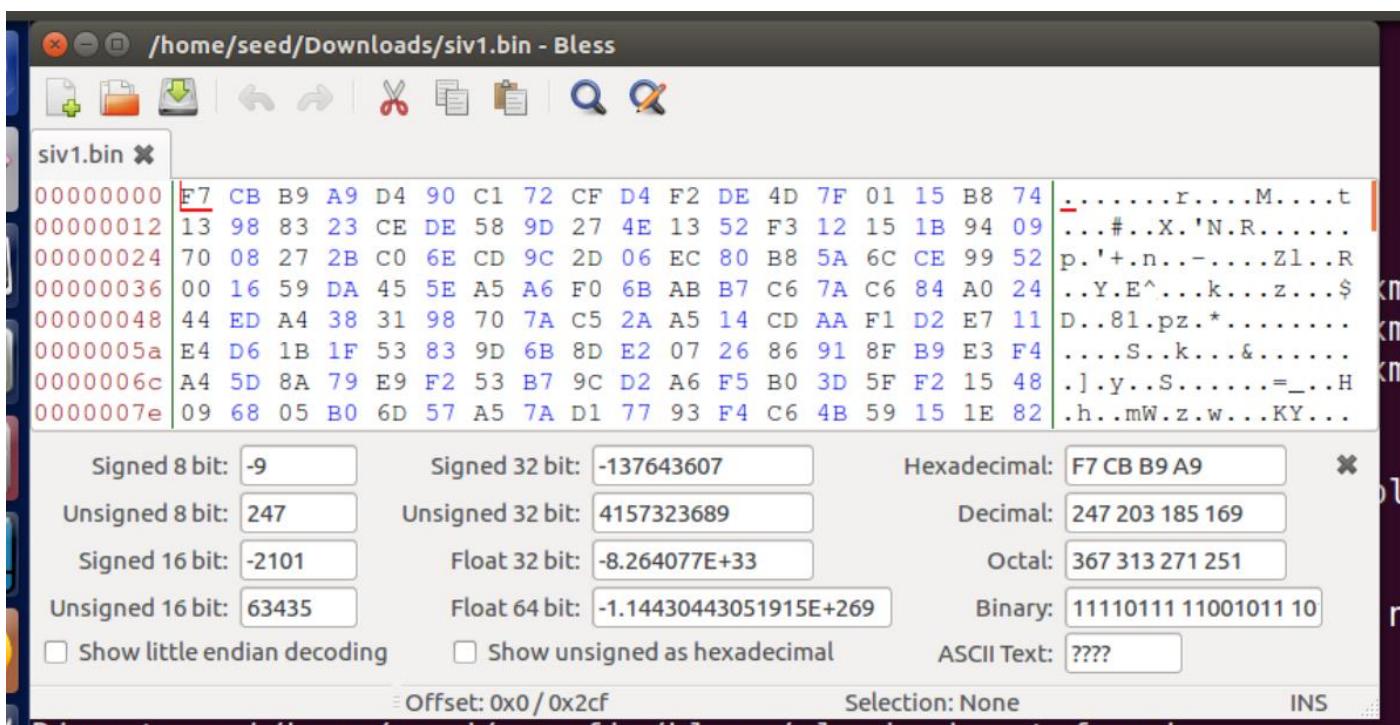
```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out iv2.bin -K 000
10203040506070809aabbccddeeff -iv 0506070809
[03/22/23]seed@VM:~/Downloads$ bless iv2.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$ █
```



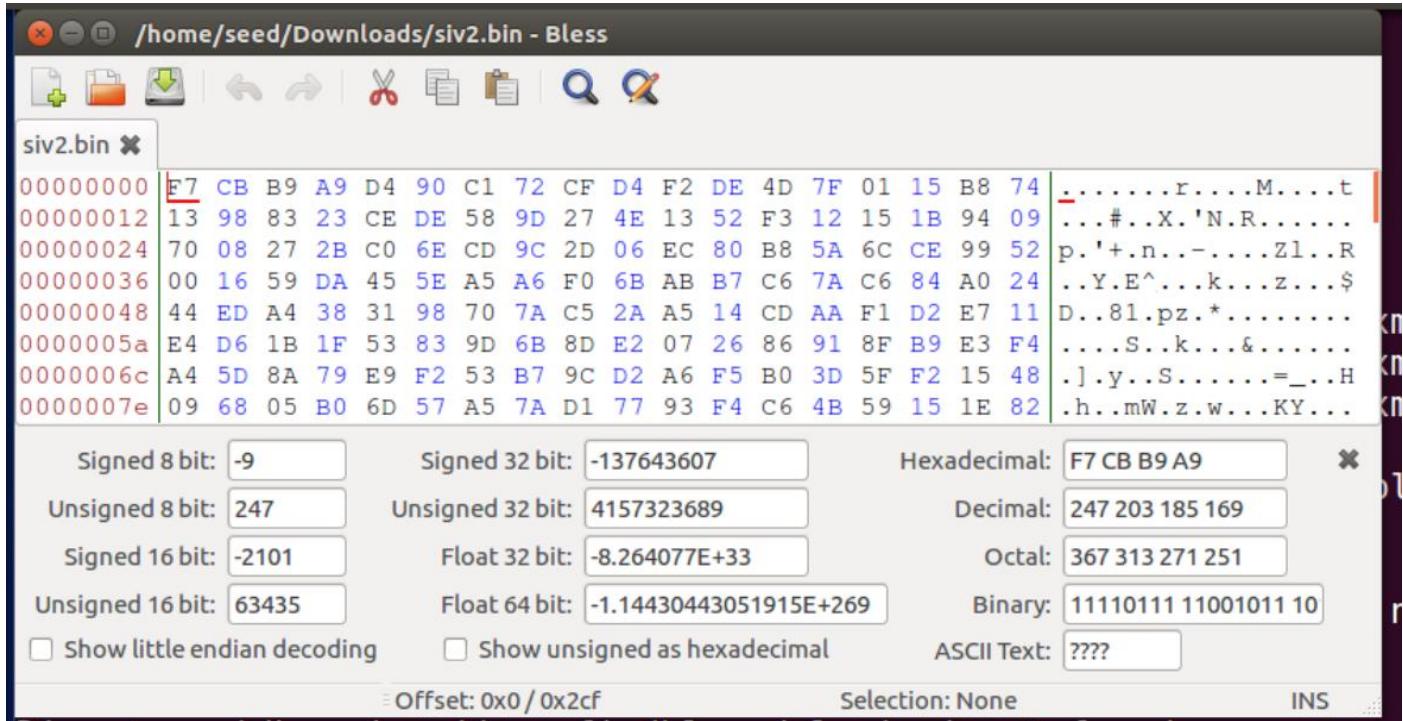
By using different encryption methods on the same piece of text, it will make the message more secure.

2. Same IV's

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out siv1.bin -K 00
010203040506070809aabccddeeff -iv 000123456789
[03/22/23]seed@VM:~/Downloads$ bless siv1.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out siv2.bin -K 00
010203040506070809aabbccddeeff -iv 000123456789
[03/22/23]seed@VM:~/Downloads$ bless siv2.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 36.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Directory '/home/seed/.config/bless/plugins' not found.
Could not find file "/home/seed/.config/bless/export_patterns".
Could not find file "/home/seed/.config/bless/history.xml".
Document does not have a root element.
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



Using the same password on different websites can be risky because an attacker could try to guess your password on other websites.

Task 6.2: Common Mistake: Use the Same IV

Plaintext (P1): This is a known message!

Ciphertext (C1): a469b1c502c1cab966965e50425438e1bb1b5f9037a4c15913

Plaintext (P2): (unknown to you)

Ciphertext (C2): bf73bcd3509299d566c35b5d450337e1bb175f903fafc15913

1. Converting P1 to Hexadecimal

Now we get P1 as the following

P1: 546869732069732061206b6e6f776e206d65737361676521

and we are given C1 as

C1: a469b1c502c1cab966965e50425438e1bb1b5f9037a4c15913

So, by this we observe that P1 is short by 2 bits. Hence, we can pad it.

P1: 546869732069732061206b6e6f776e206d65737361676521**00**

If we XOR P1 and C1, we will get Keystream i.e., Ks.

$$P1 \oplus C1 = Ks$$

XOR Calculator

Thanks for using the calculator. [View help page](#).

I. Input: hexadecimal (base 16) ▾

```
546869732069732061206b6e6f776e206d65
73736167652100
```

II. Input: hexadecimal (base 16) ▾

```
a469b1c502c1cab966965e50425438e1bb1b
5f9037a4c15913
```

Calculate XOR

III. Output: hexadecimal (base 16) ▾

```
f001d8b622a8b99907b6353e2d2356c1d67e2
ce356c3a47813
```

So, $K_s = f001d8b622a8b99907b6353e2d2356c1d67e2ce356c3a47813$

2. To find P_2 , we will do $K_s \text{ XOR } C_2$.

$$K_s \oplus C_2 = P_2$$

XOR Calculator

Thanks for using the calculator. [View help page](#).

I. Input: hexadecimal (base 16) ▾

```
f001d8b622a8b99907b6353e2d2356c1d67e  
2ce356c3a47813
```

II. Input: hexadecimal (base 16) ▾

```
bf73bcd3509299d566c35b5d450337e1bb17  
5f903fafc15913
```

Calculate XOR

III. Output: hexadecimal (base 16) ▾

```
4f726465723a204c61756e63682061206d697  
373696c652100
```

$P_2: 4f726465723a204c61756e63682061206d697373696c652100$

3. Now, we will convert P_2 from Hex to Ascii

$P_2: 4f726465723a204c61756e63682061206d697373696c652100$

From: Hexadecimal To: Text

Open File

Paste hex numbers or drop file

```
4f726465723a204c61756e63682061206d697373696c652100
```

Character encoding: ASCII

Order: Launch a missile!

$P_2(\text{plaintext}): \text{Order: Launch a missile!}$

If we had used CFB instead of OFB, we would have been able to decrypt only the first block of the keystream, because the rest of the keystream depends on the plaintext blocks.

Task 6.3: Common Mistake: Use a Predictable IV

1.

Encryption method: 128-bit AES with CBC mode. Key (in hex): 00112233445566778899aabbccddeeff (known only to Bob)

Ciphertext (C1): bef65565572ccee2a9f9553154ed9498 (known to both)

IV used on P1 (known to both)

(in ASCII): 1234567890123456

(in hex): 31323334353637383930313233343536

Next IV (known to both)

(in ASCII): 1234567890123457

(In hex): 31323334353637383930313233343537

We need to construct P2 and there is no need to decipher P1. If C1=C2, then we can say that P1='Yes' or P1='No'.

2. We need to find P2 which satisfies the condition:- $IV1 \oplus P1 = IV2 \oplus P2$

Therefore, $P2 = IV1 \oplus IV2 \oplus P1$ Here, we know that the value of the plain text is either Yes or No

$P2 = 31323334353637383930313233343536 \text{ XOR } 31323334353637383930313233343537 \text{ XOR } 596573$

XOR Calculator

Thanks for using the calculator. [View help page](#).

I. Input: hexadecimal (base 16)

31323334353637383930313233343536

II. Input: hexadecimal (base 16)

31323334353637383930313233343537

Calculate XOR

III. Output: hexadecimal (base 16)

1
//

Here, if output is 1, it is Yes or else it is No.

As the size of P1 is not equal to and smaller than IV1, we have to consider that P1 is padded

Without padding, $P2 = 596573$,

Now find the ASCII value for P2, if the output is Yes, that means $C1=C2$ and P1 is also Yes

From To

Hexadecimal Text

Paste hex numbers or drop file

```
596573
```

Character encoding

ASCII

```
Yes
```

The output shown above is yes, which means C1 =C2 and P1 is also Yes

Submitted By: -

Aastha Dhir

A20468022

adhir2@hawk.iit.edu