Lab 02- CS458

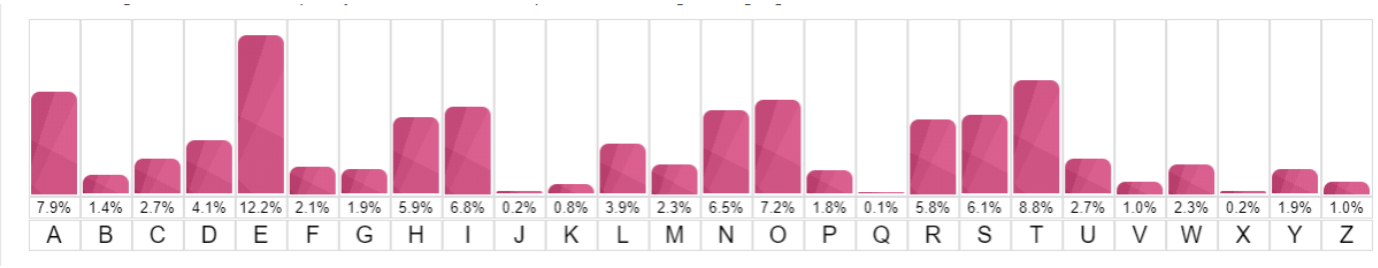
Task 1 Frequency Analysis

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

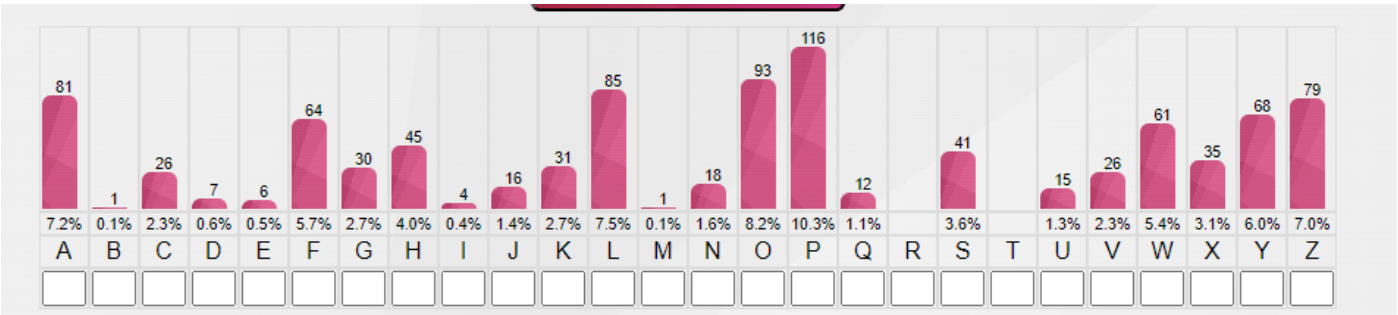
The given Ciphertext is as follows: -

```
hfcnkopw ahyplhp ya wznysgj hxzlvylv oxp qfwgs qyox lpq spdpgfncploa xznnplylv pdpwj
szj z wyvfwfka pskhzoyfl hfceylylv oxp oxpfwj fu ylufwczoyfl zls hfcnkozoyfl qyox
xzlsafl ajaopca zls afuoqzwp spayvl ya oxp ipj of akhhpaa za flp fu oxp fgspao hfcnkopw
ahyplhp spnzwocploa yl oxp hxyhzvf zwpz oxp ha spnzwocplo zo yyo xza z gflv xyaofwj fu
cppoylv oxya hxzggplvp oxwfkvx mkzgyoj pskhzoyfl yl aczgg hgzaawffc pldywflcploa zgflv
qyox ylopwlaxyn zls wpapzwhx fnnfwoklyoypa yl ylskaowj zls lzoyflzg gzefwzofwypa
yyo aoksploa qfwi qyox fkw uzhkgoj fl qfwgshgzaa wpapzwhx yl zwpza oxzo ylhgksp szoz
ahyplhp syaowyekops ajaopca ylufwczoyfl wpowypdzg hfcnkopw lpoqfwiylv ylopggyvplo
ylufwczoyfl ajaopca zls zgvfwyoxca
oxp spnzwocplo fuupwa ezhxpgfw fu ahyplhp czaopw fu ahyplhp nwfupaayflzg czaopw zls nxs
spvwppa ngka vwzskzop hpwoyuyhzopa zhhpgpwzops hfkwapa zls lflspvwpp aoksj nzwooycp
aoksploa hzl ozip pdplylv hgzaapa zls gflvsyaozlhp aoksploa hzl pzwl 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)

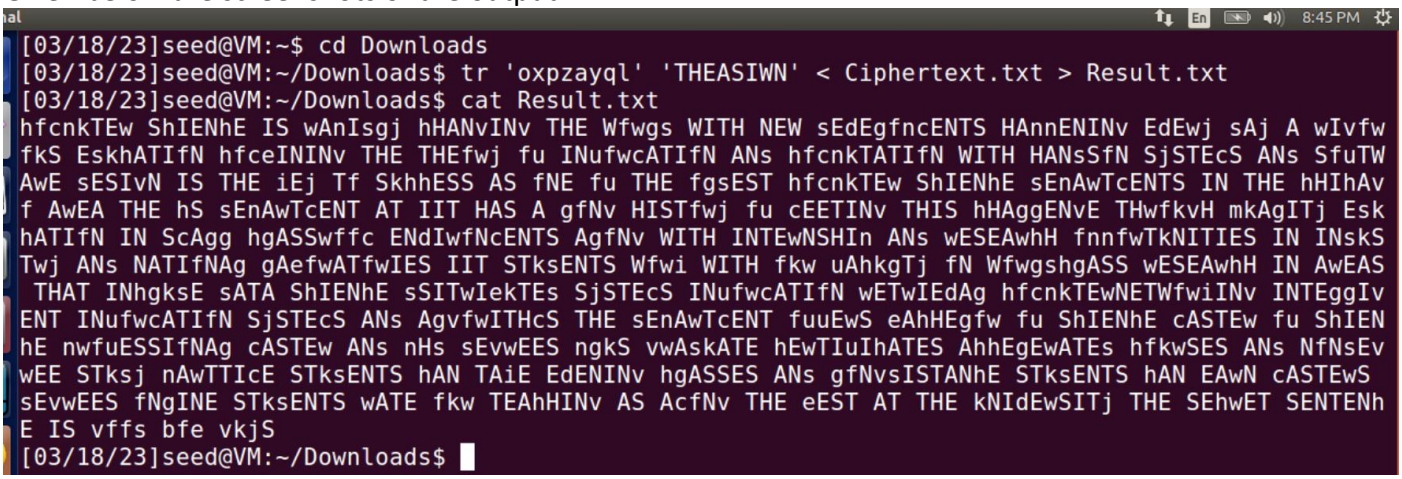


From the graphs above, we can infer the following: -

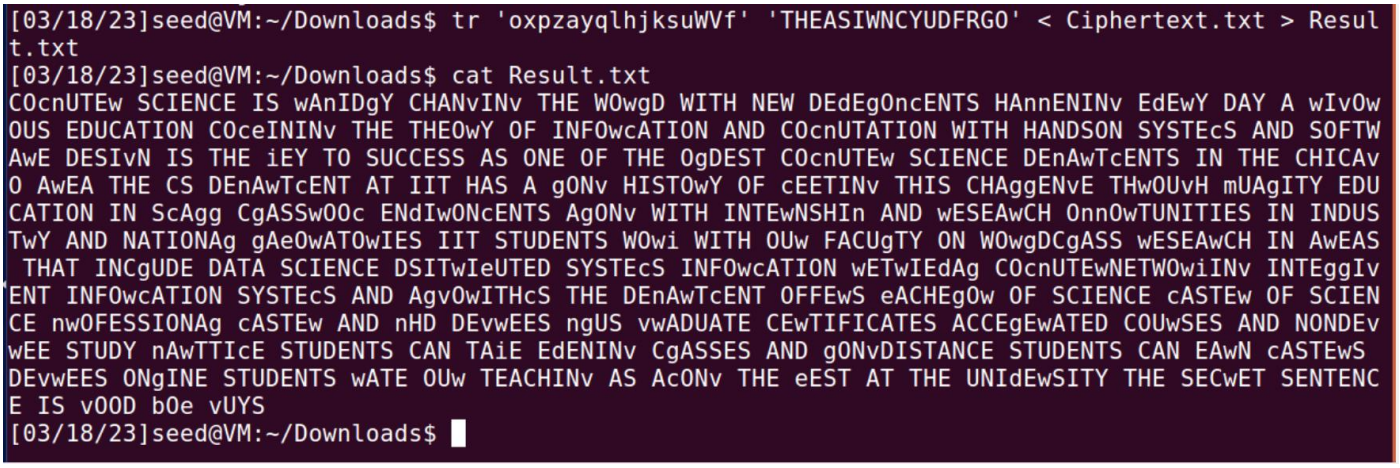
1. The frequency of P in Fig 2 is the highest and E in Fig 1 is the highest. So, P in ciphertext can be replaced by E, i.e., p->e
2. Similarly, the following letter with high frequency is O in Fig 2 and T in Fig 1. So, we can assume that O in the ciphertext is replaced by t, i.e., o->t
3. Most frequently 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, flp, 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



From this we can guess words like
SjSTEcS = Systems
Tf = to
ANs = and
STksENTS = students
wATE = rate



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 'oxpzayqlhjksuwvfcngeidbm' '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 RIGOR
OUS EDUCATION COMBINING THE THEORY OF INFORMATION AND COMPUTATION WITH HANDSON SYSTEMS AND SOFTW
ARE DESIGN IS THE KEY TO SUCCESS AS ONE OF THE OLDEST COMPUTER SCIENCE DEPARTMENTS IN THE CHICAG
O AREA THE CS DEPARTMENT AT IIT HAS A LONG HISTORY OF MEETING THIS CHALLENGE THROUGH QUALITY EDU
CATION IN SMALL CLASSROOM ENVIRONMENTS ALONG WITH INTERNSHIP AND RESEARCH OPPORTUNITIES IN INDUS
TRY AND NATIONAL LABORATORIES IIT STUDENTS WORK WITH OUR FACULTY ON WORLDCLASS RESEARCH IN AREAS
 THAT INCLUDE DATA SCIENCE DSITRIBUTED SYSTEMS INFORMATION RETRIEVAL COMPUTERNETWORKING INTELLIG
ENT INFORMATION SYSTEMS AND ALGORITHMS THE DEPARTMENT OFFERS BACHELOR OF SCIENCE MASTER OF SCIEN
CE PROFESSIONAL MASTER AND PHD DEGREES PLUS GRADUATE CERTIFICATES ACCELERATED COURSES AND NONDEG
REE 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 SENTENC
E 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

```
Terminal
[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 000102030405
06070809aabbccddeeff -iv 0a0b0c0d0e0f0102030405060708090
[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 000102030405
06070809aabbccddeeff
[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..\\OOsQ.=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  00010203040
506070809aabbccddeeff -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 0102030405060708 09
[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 0102030405060708 09
[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
809aabbccddeeff -iv 0a0b0c0d0e0f0102030405060708 09
[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
809aabbccddeeff -iv 0a0b0c0d0e0f0102030405060708 09
[03/20/23]seed@VM:~$ xxd cipher.bin
00000000: ee00 df03 c150 1b98 c951 4f43 3c19 795a  .....P...QOC<.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
809aabbccddeeff -iv 0a0b0c0d0e0f0102030405060708 09
[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  ..+...O......u{.
00000020: 3e25 1e53 dab2 e88a 45ed fe8a 08a2 b3c3  >%.S....E.......
00000030: 7ec7                                     ~.
[03/20/23]seed@VM:~$
```

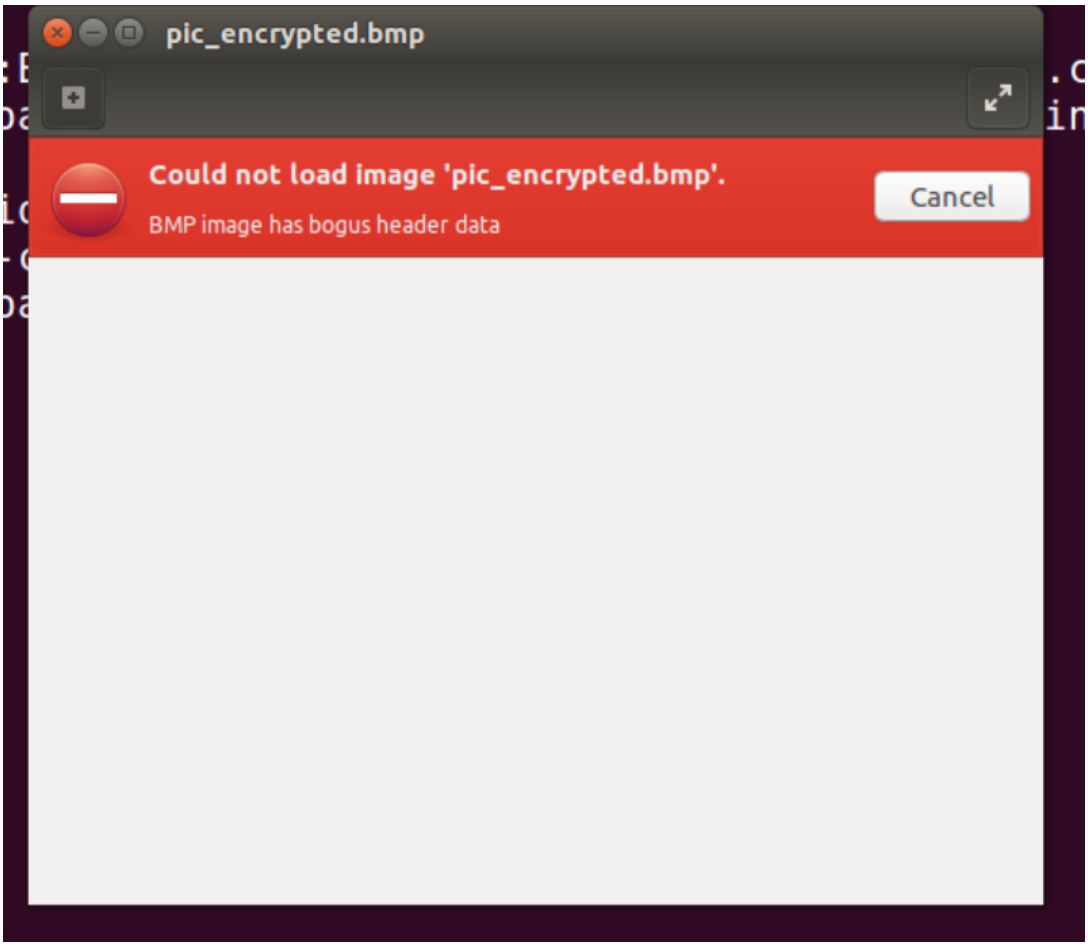
Task 3: Encryption Mode – ECB V/s CBC

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

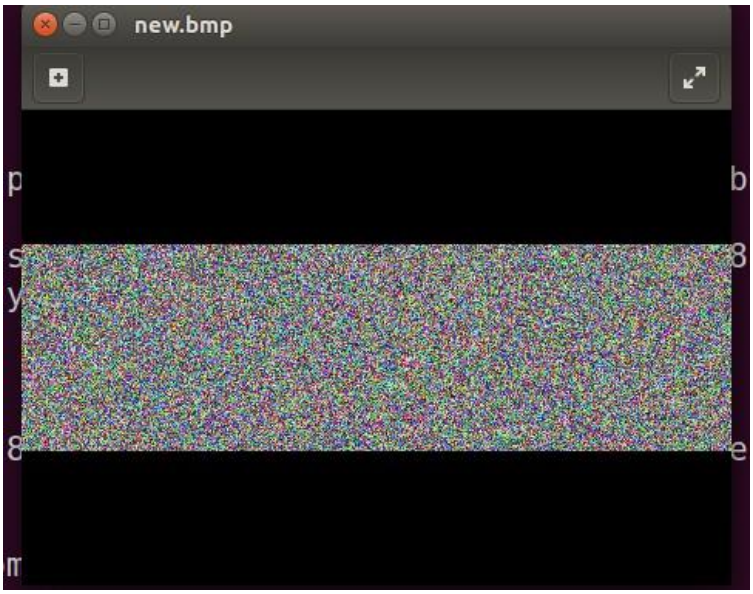
1. CBC

Encrypting picture with AES-128-CBC cipher

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -in pic_original.bmp -out pic_encrypt
ed.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$
```

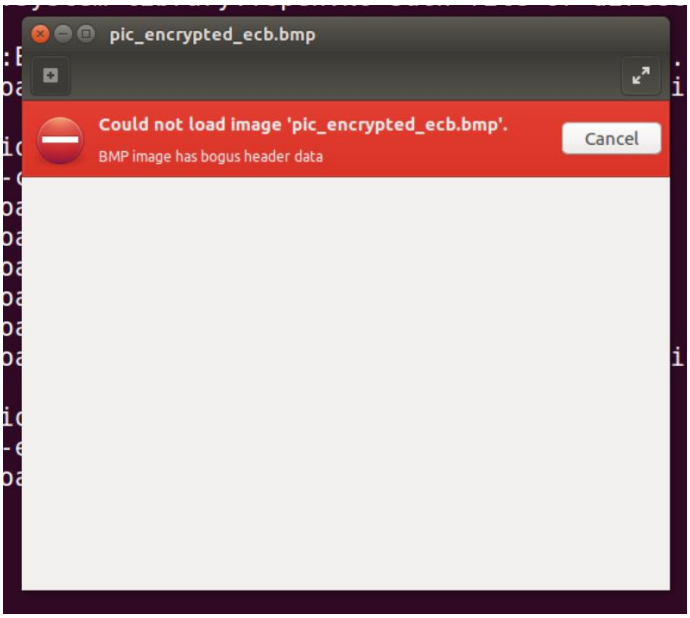


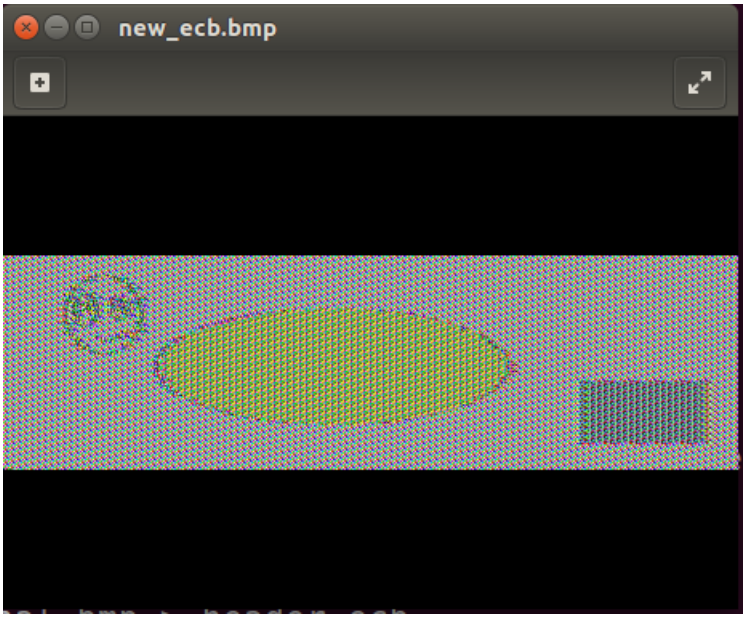
We can see that the image is completely encrypted, and we cannot identify the original image from this. This image is not at all clear.

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$
```





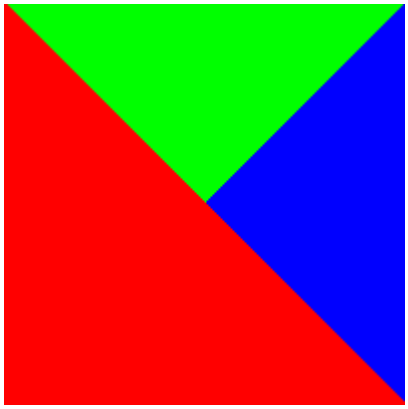
In encrypted image mode, we can still see the outline of the original image, whereas, in cipher mode (ECB), each block is encrypted separately, which results in different outputs. With CBC, the location of the block is also taken into consideration, so it is better than ECB.

Task 2.3.1

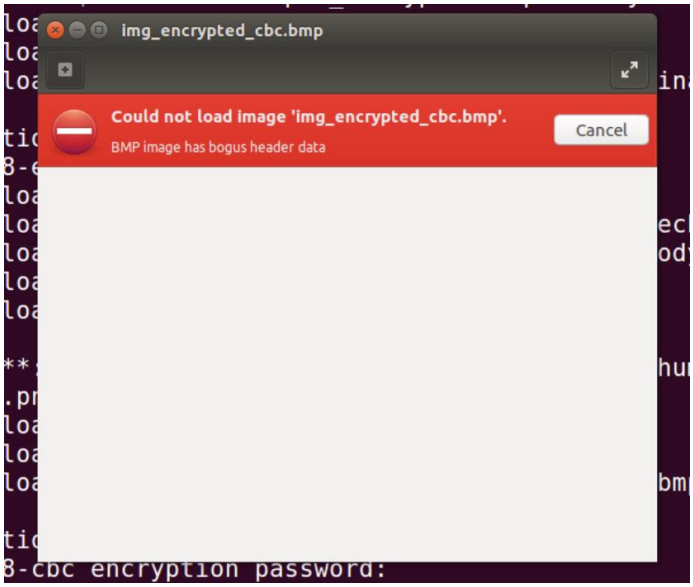
Select a picture of your choice, repeat the experiment above, and report your observations.

1. For CBC

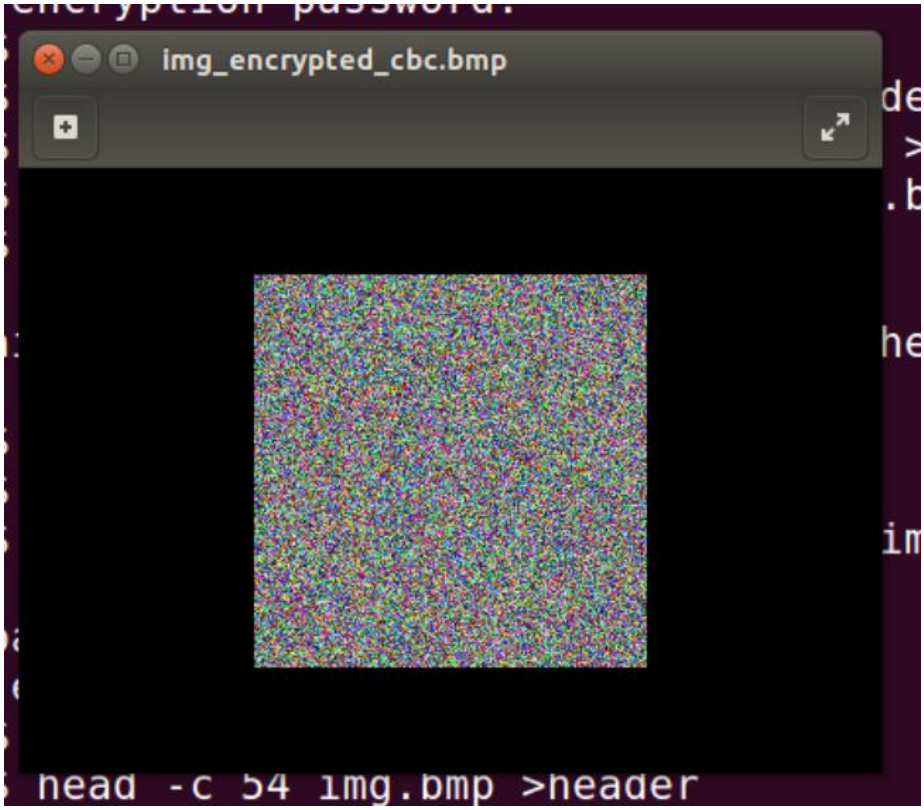
   The picture that is being encrypted is as follows: -



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



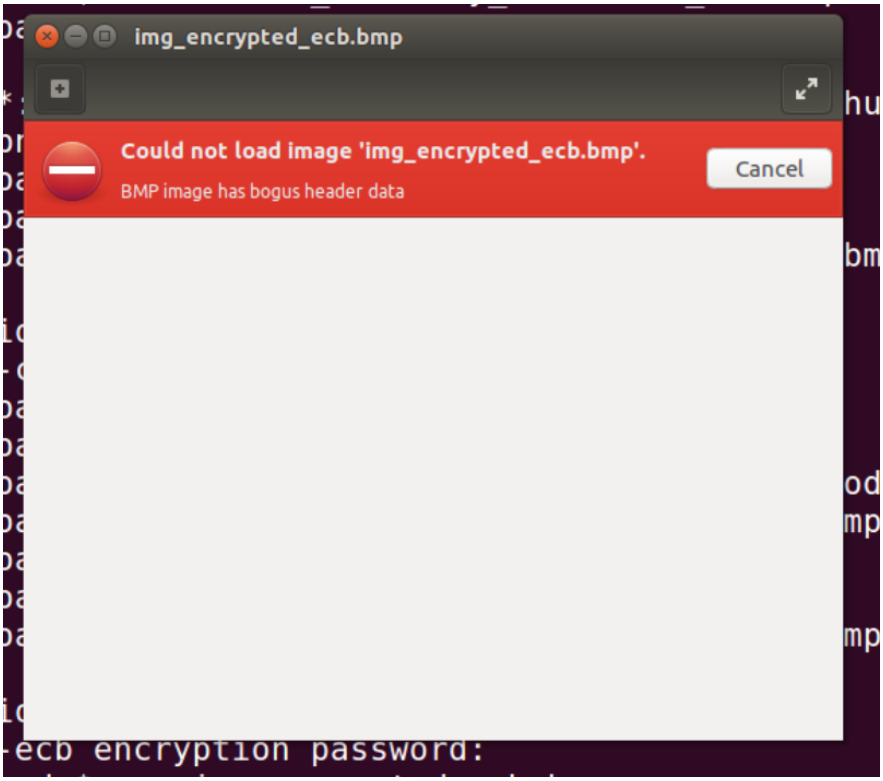

```
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_cbc.bmp
[03/22/23]seed@VM:~/Downloads$ head -c 54 img.bmp >header
[03/22/23]seed@VM:~/Downloads$ tail -c +55 img_encrypted_cbc.bmp > body
[03/22/23]seed@VM:~/Downloads$ cat header body > img_encrypted_cbc.bmp
[03/22/23]seed@VM:~/Downloads$ eog img_encrypted_cbc.bmp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



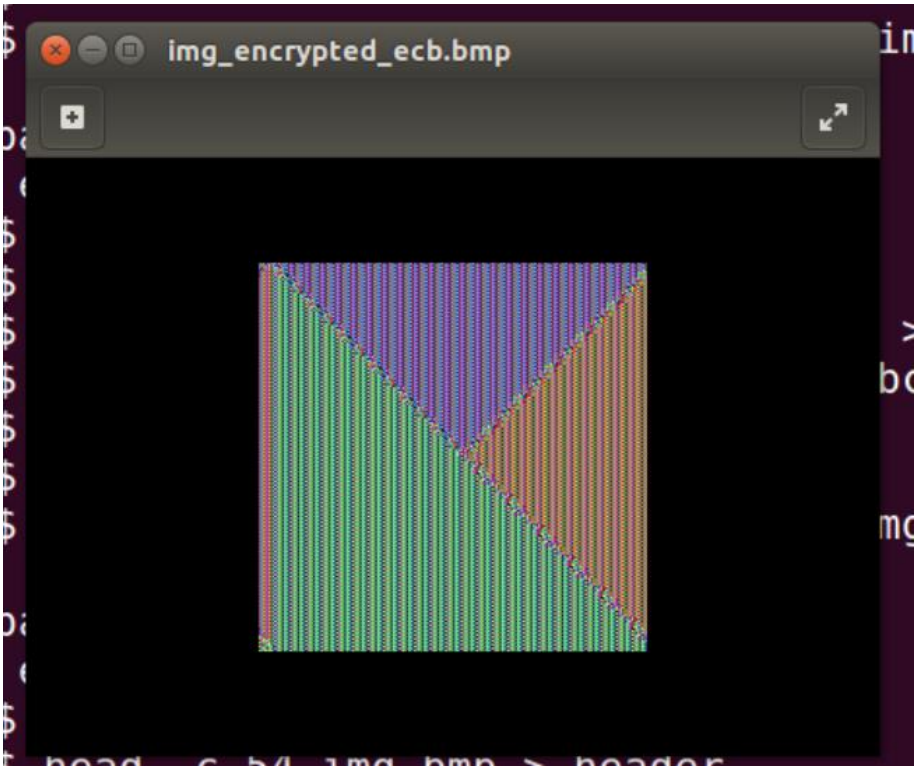
2. ECB

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 000102030405060708
09aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in f2.txt -out f2_encrypt.txt -K 000102030405060708
09aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in f3.txt -out f3_encrypt.txt -K 000102030405060708
09aabbccddeeff -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 000
102030405060708090aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -nopad -in f2_encrypt.txt -out f2_decrypt.txt -K 000
102030405060708090aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -nopad -in f3_encrypt.txt -out f3_decrypt.txt -K 000
102030405060708090aabbccddeeff -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  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
0000000   1   2   3   4   5
0000005
[03/22/23]seed@VM:~/Downloads$ hexdump -c f1_encrypt.txt
0000000   ﷿ 215   W   ﷿   ﷿   ^   ﷿ 003   >   w   ﷿ 030   - 201   ﷿   $
0000010
[03/22/23]seed@VM:~/Downloads$ hexdump -c f1_decrypt.txt
0000000   1   2   3   4   5  \v  \v  \v  \v  \v  \v  \v  \v  \v  \v  \v
0000010
[03/22/23]seed@VM:~/Downloads$
```

For f2

```
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2.txt
0000000   1   2   3   4   5   6   7   8   9   0
000000a
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2_encrypt.txt
0000000   ﷿   /   ﷿   ~   ﷿   ﷿ 225   | 224   +   ﷿   ﷿ 237   ﷿   ﷿   ﷿
0000010
[03/22/23]seed@VM:~/Downloads$ hexdump -c f2_decrypt.txt
0000000   1   2   3   4   5   6   7   8   9   0 006 006 006 006 006 006
0000010
[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   ?   :   ?  \v   ?   ?   ? 221 236   ?   ?   k 023 177 020 203
0000010   >   3   (   H   .   *   ?   b   '   q   ?   ?   h   ?   4   ?
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 020 020 020 020 020 020 020 020 020 020 020 020 020 020 020 020
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 k
ind and lovely and all the people of the realm adored her.  The only sadness in the queen's l
ife 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
 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$
```

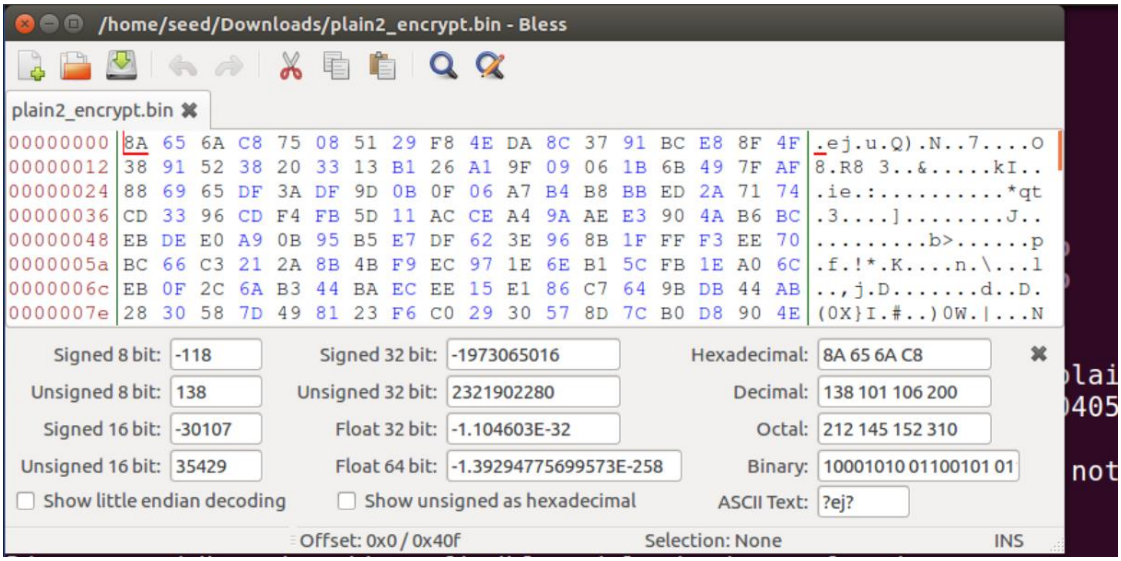
How much information can you recover by decrypting the corrupted file, if the encryption mode is ECB, CBC, CFB, or OFB, respectively?

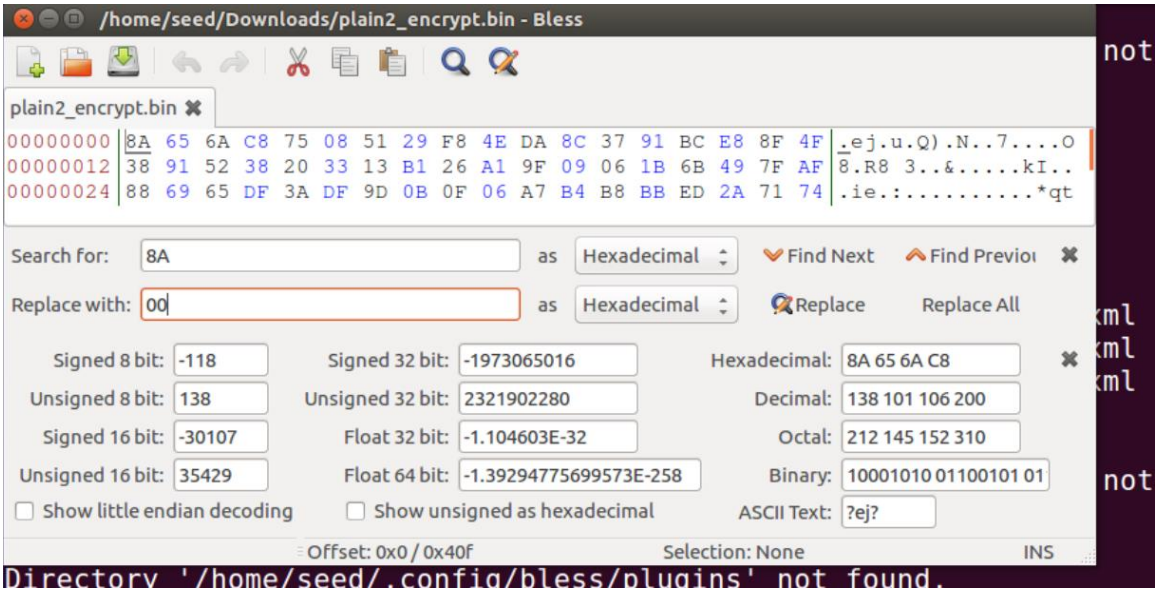
Answer- Decryption of a corrupted file can depend on several factors, including the encryption mode used, the extent and nature of the corruption, and the availability of any additional information or context. If you're using encryption in ECB mode, it's easier to get some information out of a damaged 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

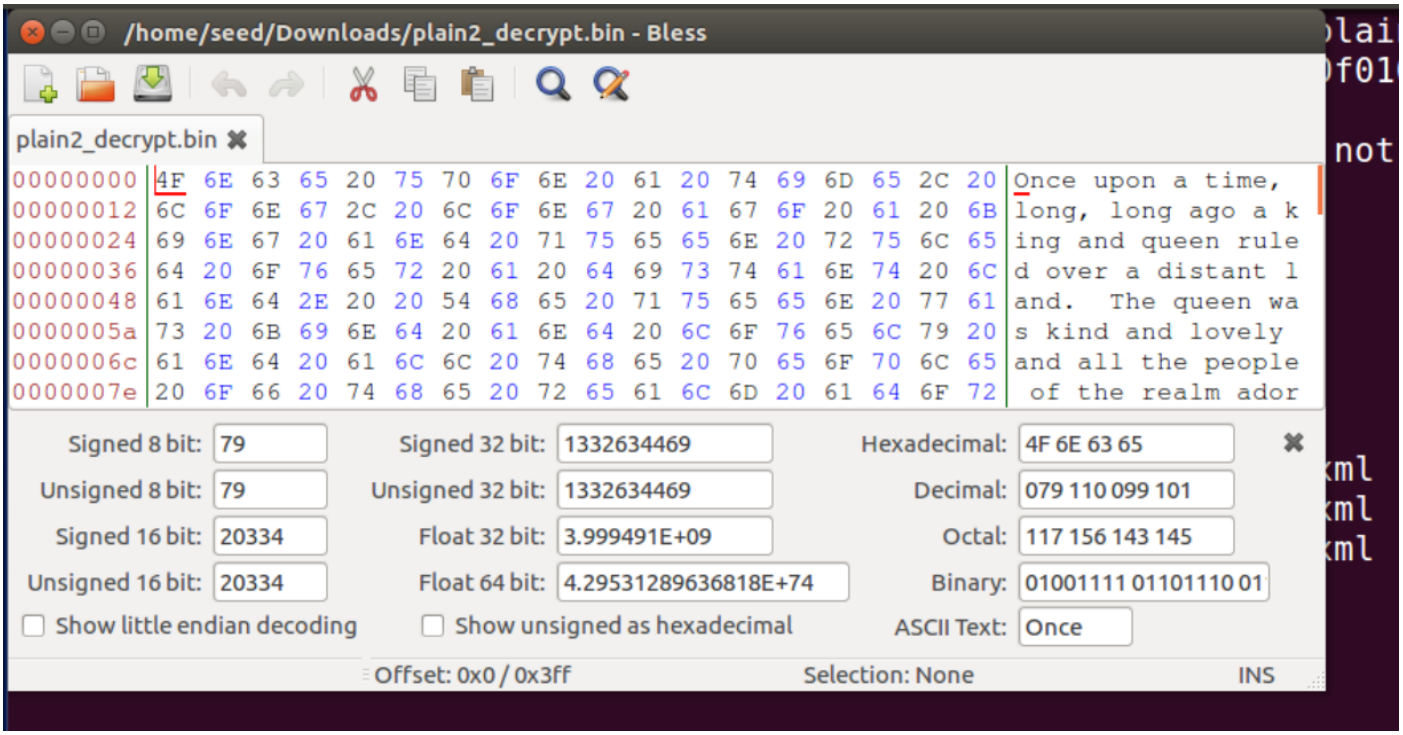
```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain2.txt -out plain2_encrypt
.bin -K 000102030405060708090aabbccddeeff -iv 0a0b0c0d0e0f010203040506070809
```





Decryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -d -in plain2_encrypt.bin -out plain2
_decrypt.bin -K 000102030405060708090aabbccddeeff -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$
```



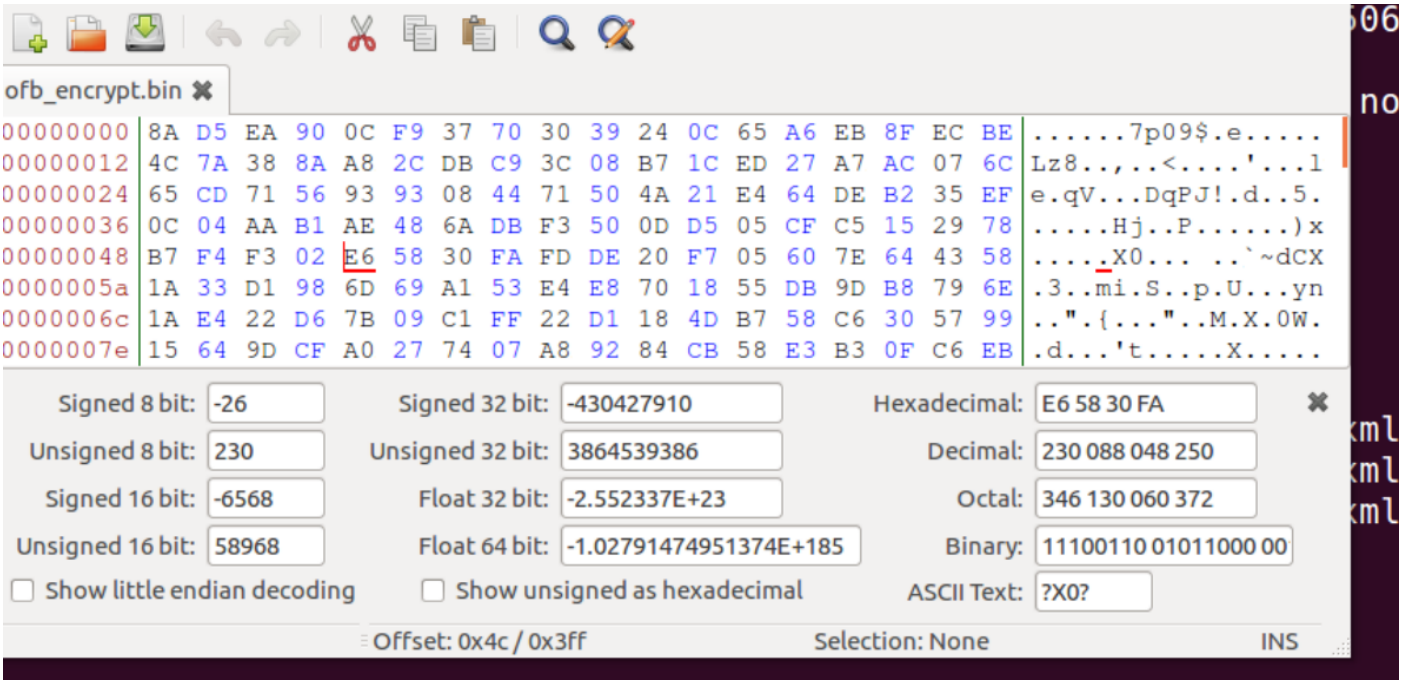

```
[03/22/23]seed@VM:~/Downloads$ cat plain2_decrypt.bin
?
 Q???@??▒▒?y??A?? long, long ago a king and queen ruled over a distant land.  The queen was ki
nd and lovely and all the people of the realm adored her.  The only sadness in the queen's li
fe was that she wished for a child but did not have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

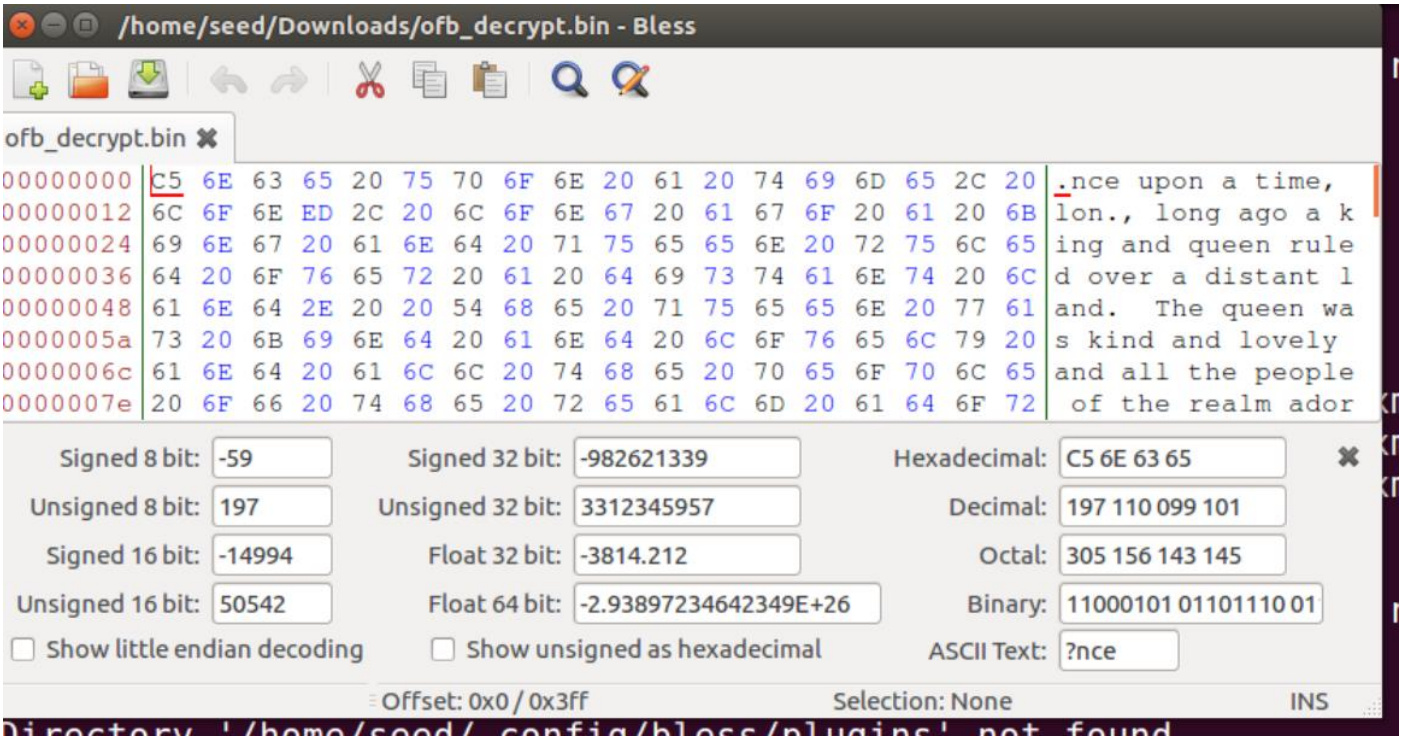
Encryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ofb -e -in plain2.txt -out ofb_encrypt.bi
n -K 00010203040506070809aabbccddeeff -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_decry
pt.bin -K 00010203040506070809aabbccddeeff -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.
```



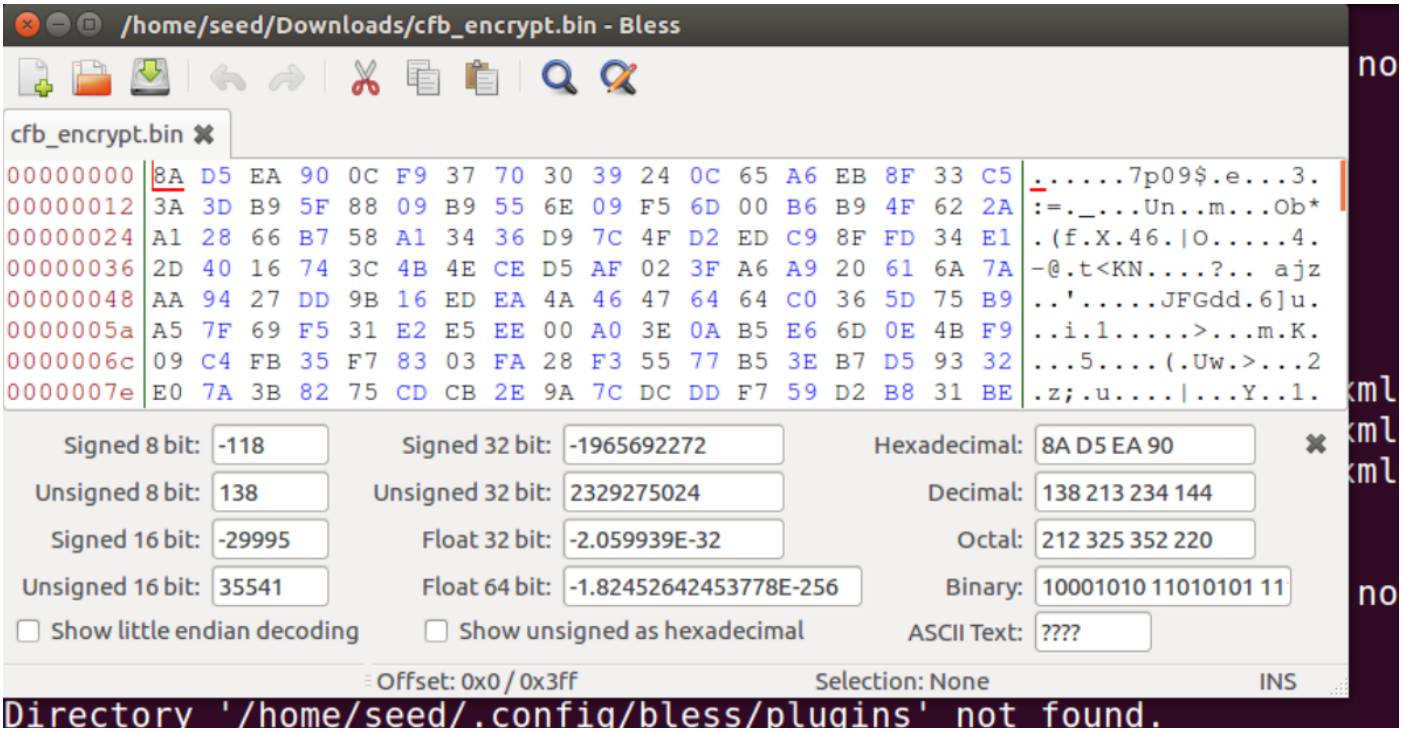

```
[03/22/23]seed@VM:~/Downloads$ cat ofb_decrypt.bin
ønce upon a time, lonø, long ago a king and queen ruled over a distant land.  The queen was k
ind and lovely and all the people of the realm adorød her.  The only sadness in the queen's l
ife was that she wished for a child but did not have one.

One winter day, the quøen was doing needle work while gazing out her ebony window at the new
fallen snow.  A bird flew by the window startlinø 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 aføer 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 birtø 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$
```

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.

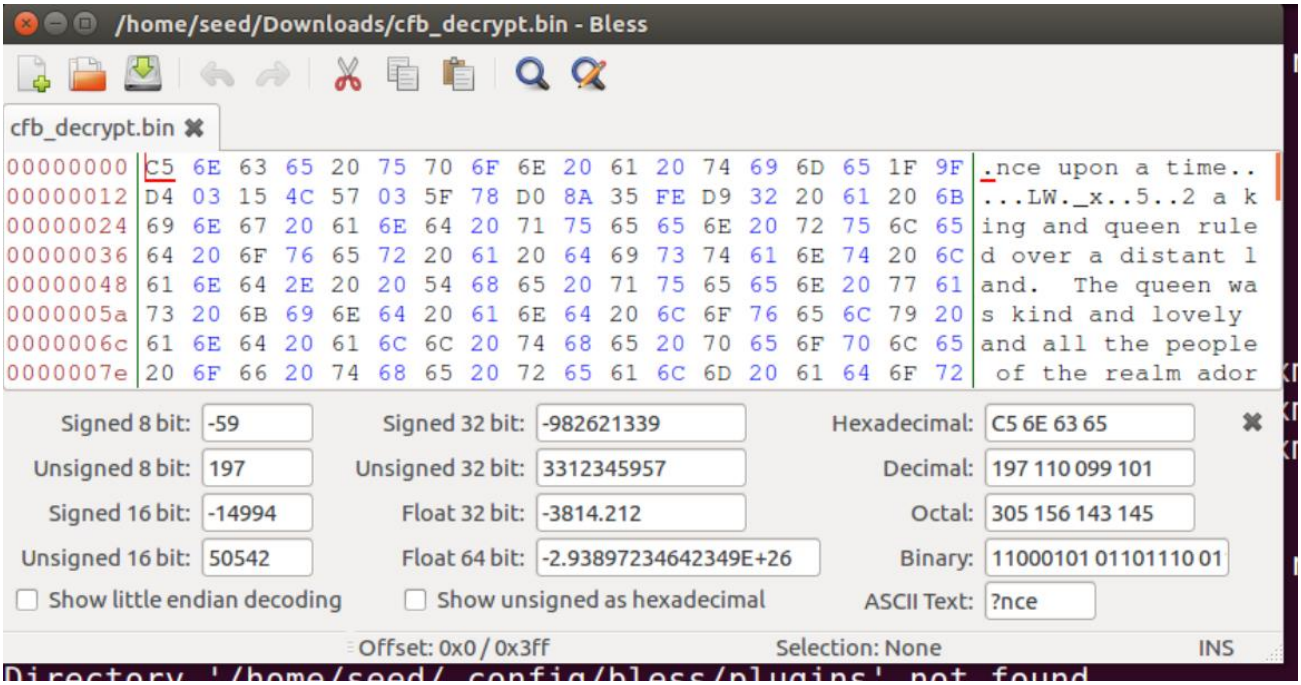
3. CFB

Encryption

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cfb -e  -in plain2.txt -out cfb_encrypt.b
in -K 000102030405060708090aabbccddeeff -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$
```




```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cfb -d -in cfb_encrypt.bin -out cfb_decry
pt.bin -K 000102030405060708090aabbccddeeff -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
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
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
Sharing violation on path /home/seed/.config/bless/preferences.xml
Document does not have a root element.
[03/22/23]seed@VM:~/Downloads$
```



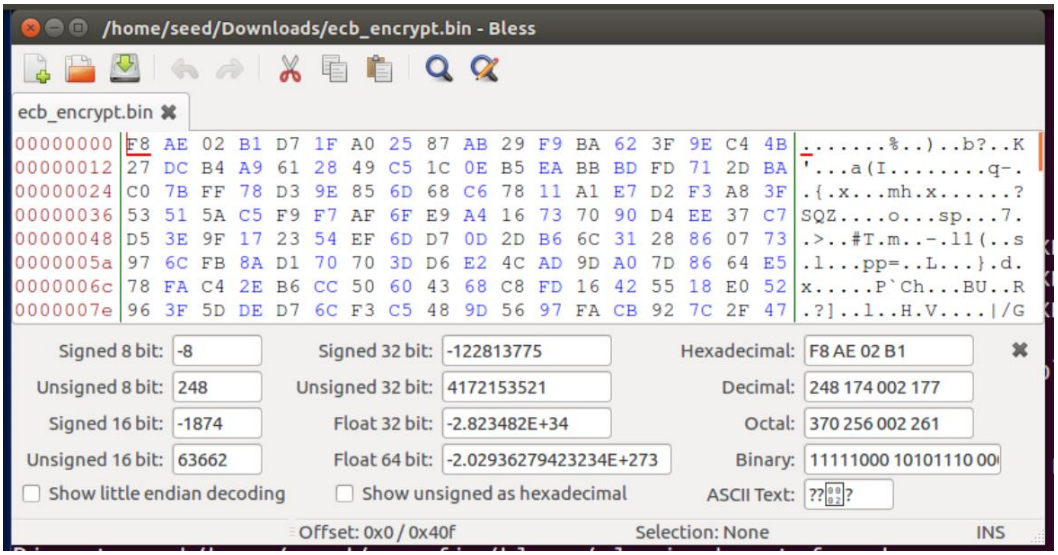

```
[03/22/23]seed@VM:~/Downloads$ cat cfb_decrypt.bin
nce upon a time  W  xЋ5  2 a king and queen ruled over a distant land.  The queen was ki
nd and lovely and all the people of the realm adored her.  The only sadness n theg X
                                                                                 чgl
zs that she wished for a ch ld bu
u  #  a 6ne.

One winter day, the queen was doing needle work while gazin out her ebony-  0  rWK
                                                                              Ee ew fa
llen snow.  A bird flew by the window startling the queen and she pricked her finger.  A sing
le drop of blood fell on the snow outside her window.  As she lo ked at the bl
                                                                          `&  `  {
she said to herself, "Oh, how I wish that I had a daughter that had skin as white as snow, li
ps as red as blood, and hair as black as bony."
W ´    )    |, the kind ueenl\ ]9`D#}
                                                  en she gave birth to a baby girl who had skin wh
ite as snow, lips red as blood, and hair black a ebony.  They n f1NwCmF  l nce s Snow Wh
it j  ^ZRU  $  e queen died after giving birt Q;
                                                        mQ]s sDSoon after, the king married
a new woman who was beautiful, but as well proud and cruel.  She ha[03/22/23]seed@VM:~/Downlo
ads$
```

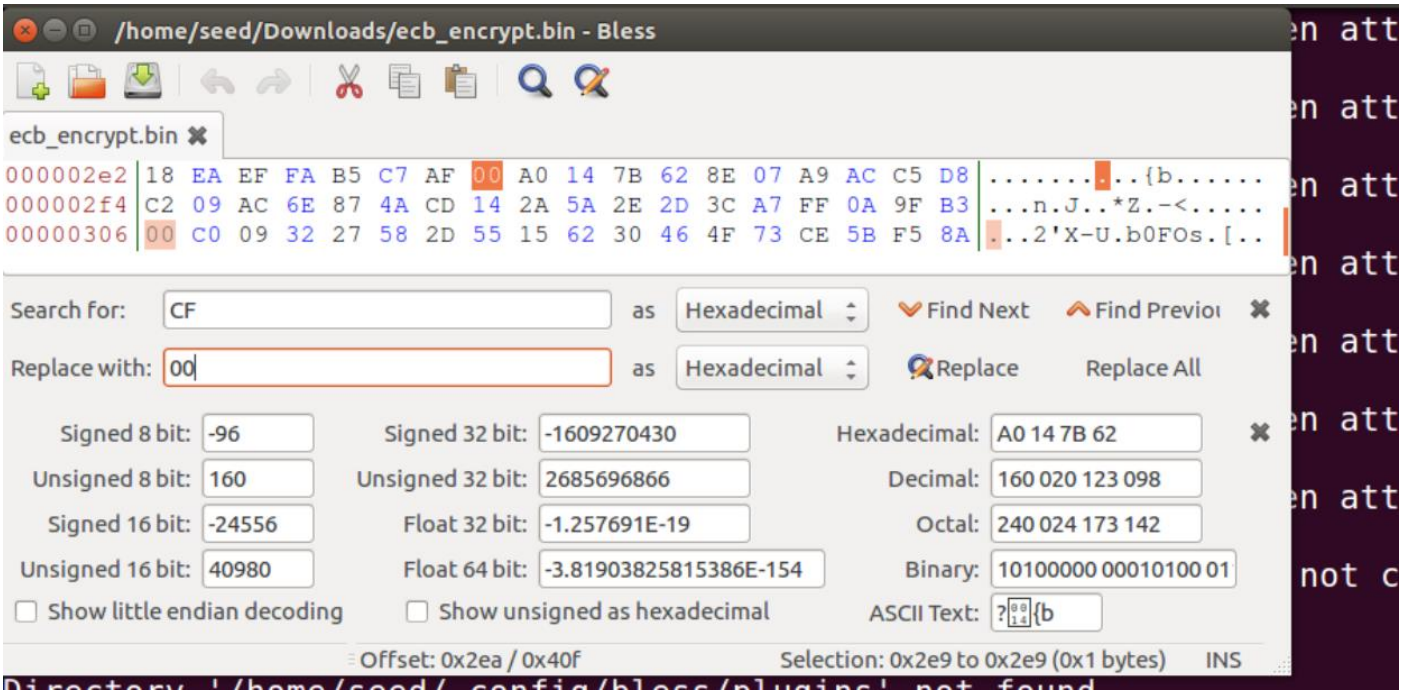
In this one, we can see that when we changed just one bit in our block of data, it became corrupted and lost its information.

4. ECB

Encryption

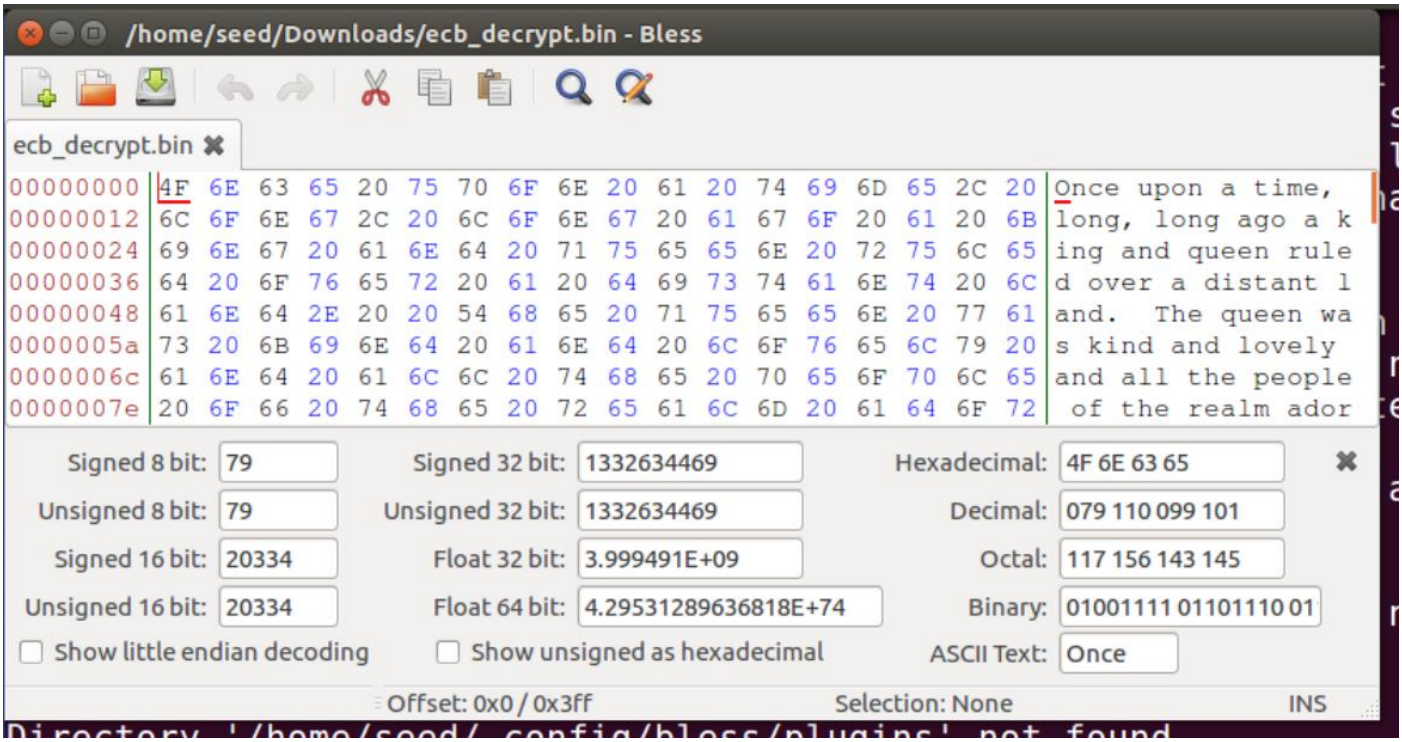
```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-ecb -e -in plain2.txt -out ecb_encrypt.bi
n -K 000102030405060708090aabbccddeeff
[03/22/23]seed@VM:~/Downloads$ bless ecb_encrypt.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 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_decry
pt.bin -K 000102030405060708090aabbccddeeff
[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$ cat ecb_decrypt.bin
Once upon a time, long, long ago a king and queen ruled over a distant land.  The queen was k
ind and lovely and all the people of the realm adored her.  The only sadness in the queen's l
ife 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
wh????K??`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
 ha[03/22/23]seed@VM:~/Downloads$
```

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.

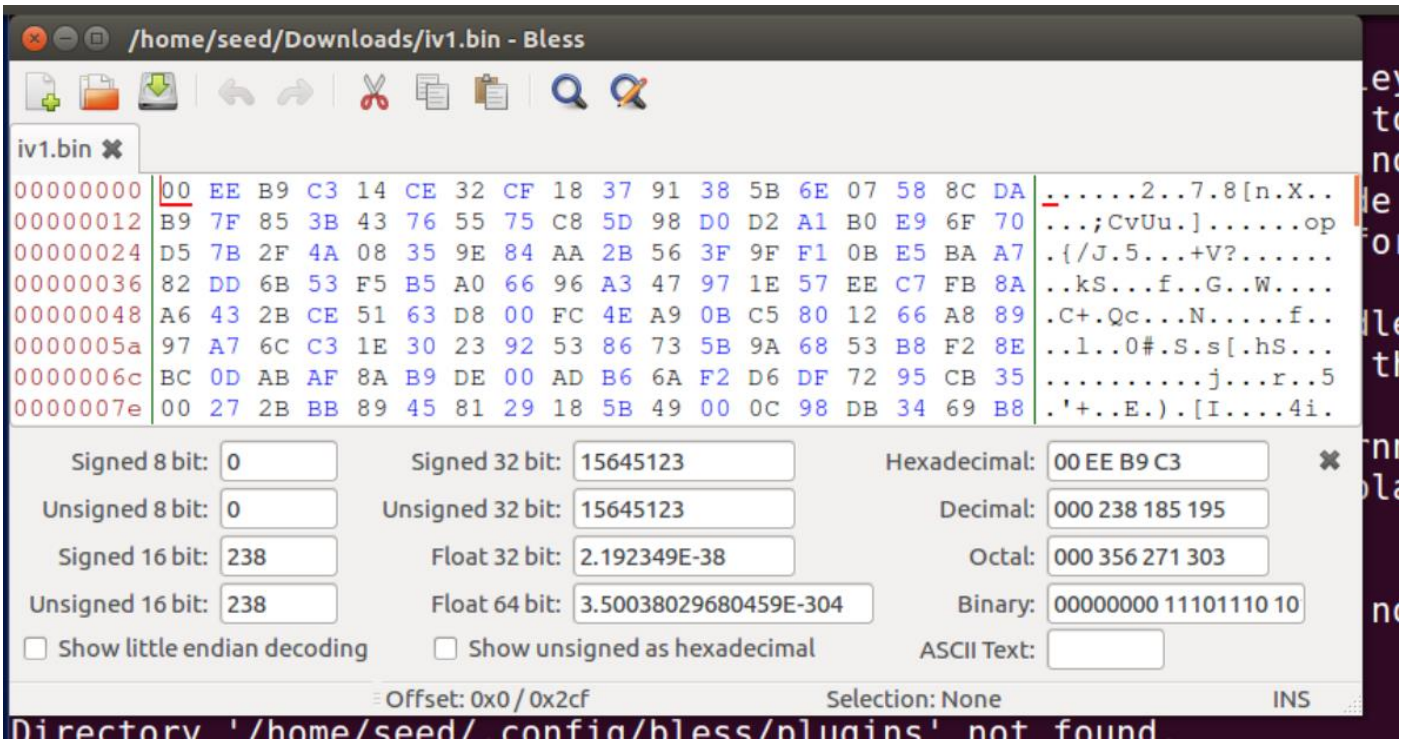
Task 6: Initial Vector (IV)

Task 6.1: Uniqueness of IVs

1. Using different IV's

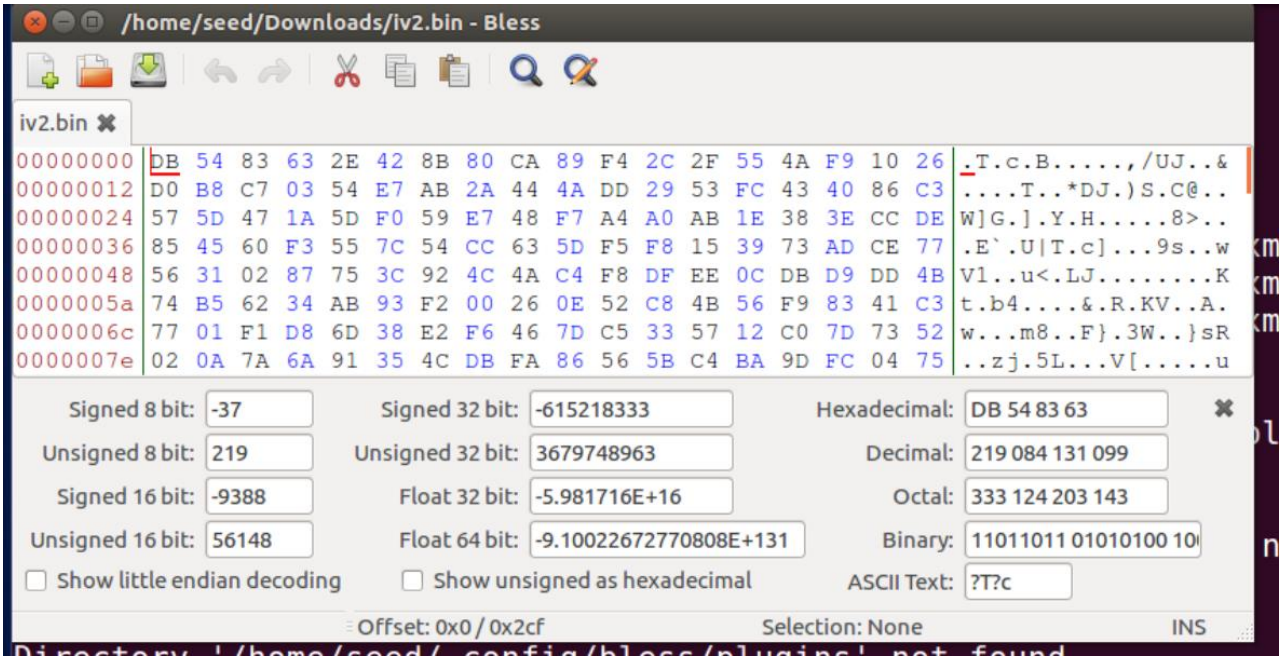
IV 1

```
[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

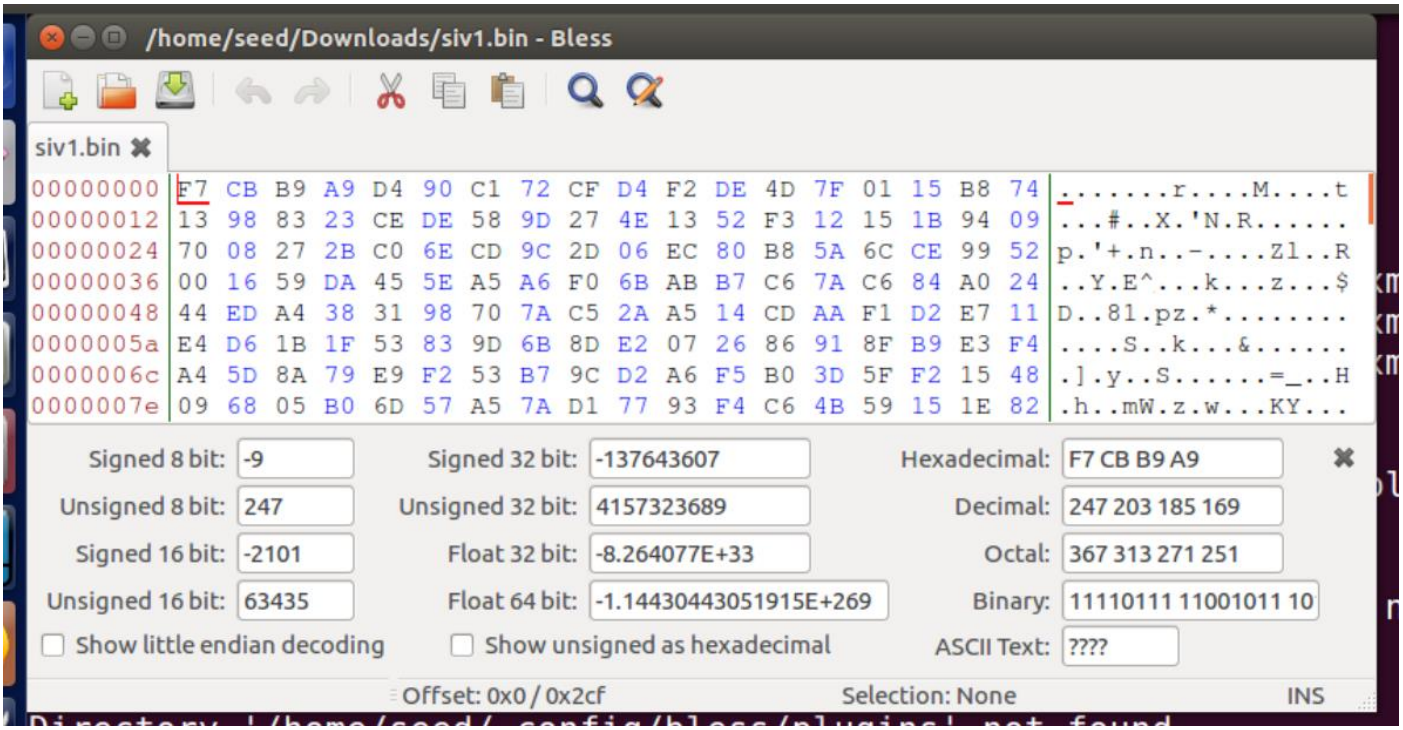
```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out iv2.bin -K 000
10203040506070809aabbccddeeff -iv 0506070809
[03/22/23]seed@VM:~/Downloads$ bless iv2.bin
Unexpected end of file has occurred. The following elements are not closed: pref, preferences
. Line 22, position 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$
```



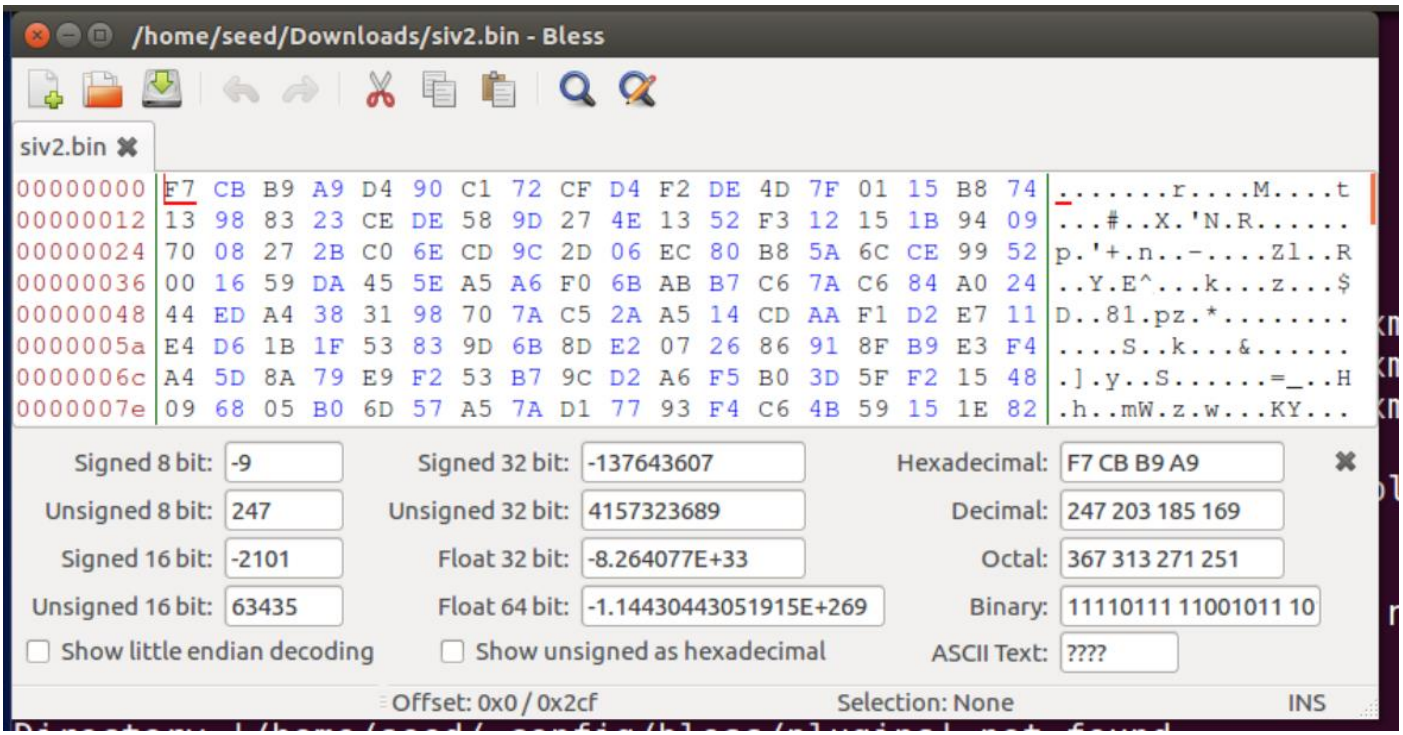
By using different encryption methods on the same piece of text, it will make the message more secure.

2. Same IV's

```
[03/22/23]seed@VM:~/Downloads$ openssl enc -aes-128-cbc -e -in plain3.txt -out siv1.bin -K 00
010203040506070809aabbccddeeff -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
0102030405060708090aabbccddeeff -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

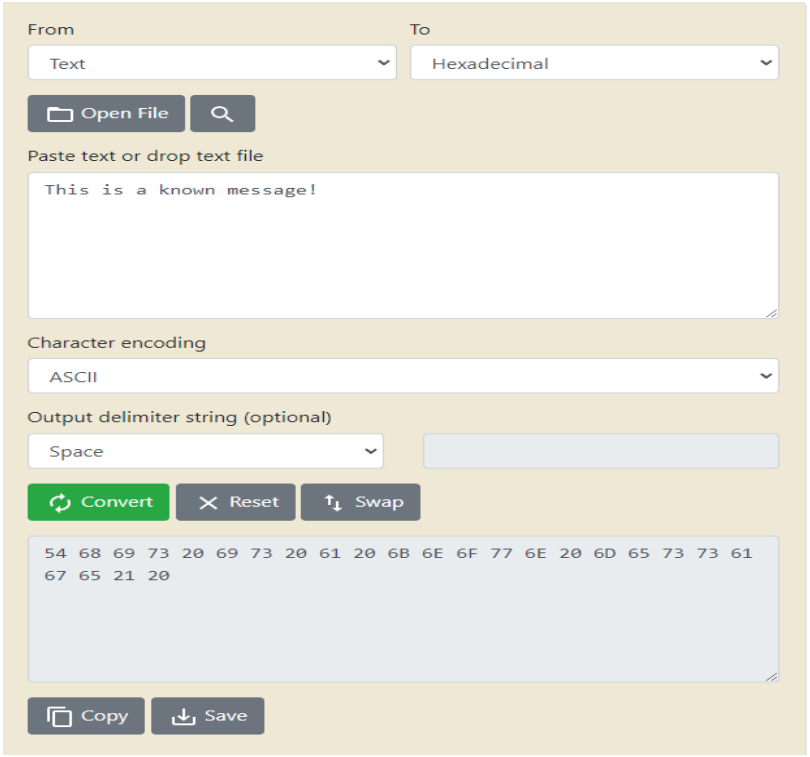
Plaintext (P1): This is a known message!

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

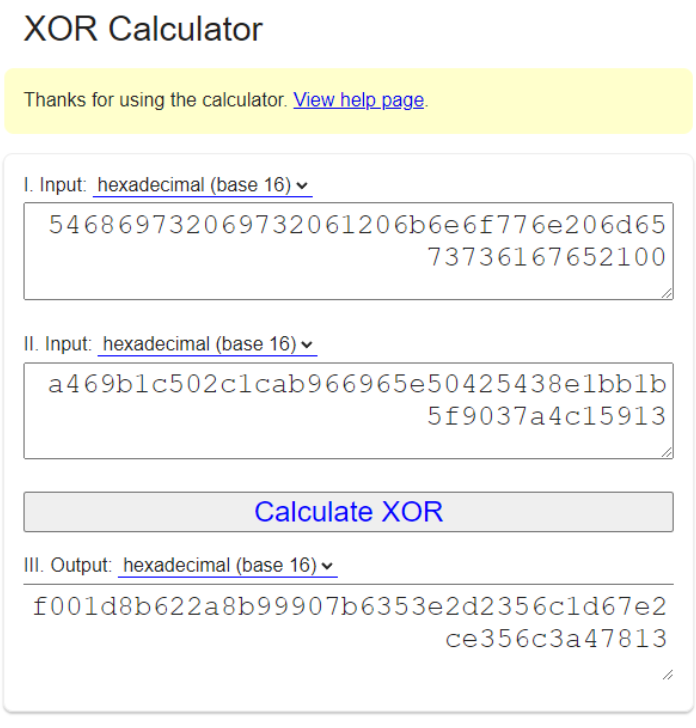
C1: a469b1c502c1cab966965e50425438e1bb1b5f9037a4c15913

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

P1: 546869732069732061206b6e6f776e206d6573736167652100

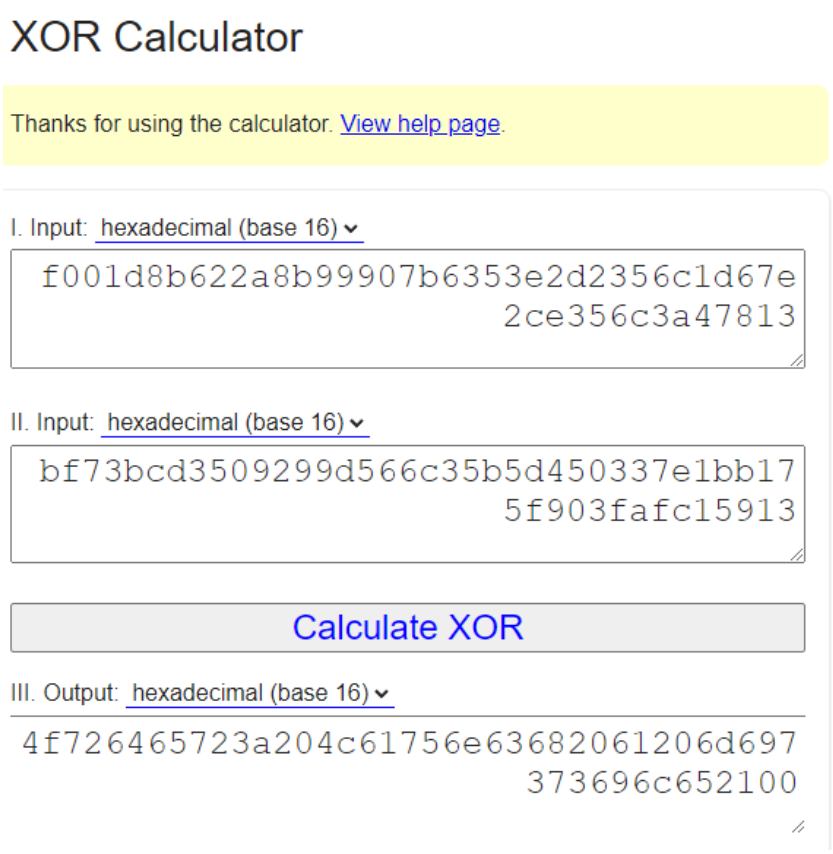
If we XOR P1 and C1, we will get Keystream i.e., Ks.

P1⊕C1 = Ks



So, Ks = f001d8b622a8b99907b6353e2d2356c1d67e2ce356c3a47813

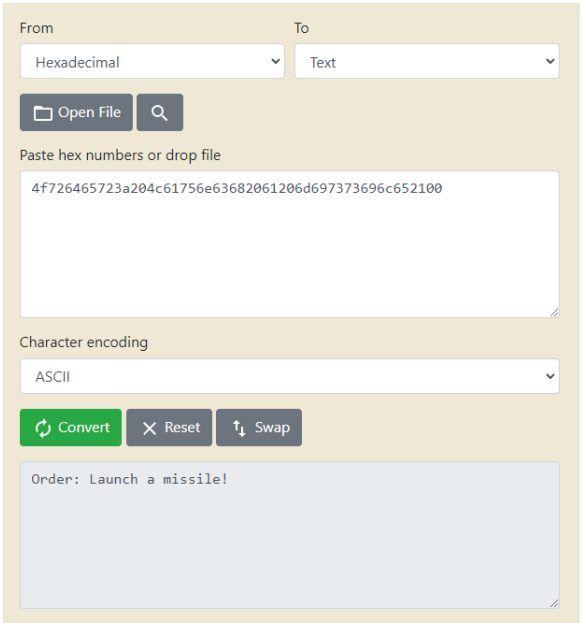
2. To find P2, we will do Ks XOR C2.
   Ks⊕C2 = P2



P2: 4f726465723a204c61756e63682061206d697373696c652100

3. Now, we will convert P2 from Hex to Ascii

P2: 4f726465723a204c61756e63682061206d697373696c652100



P2(plaintext): 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

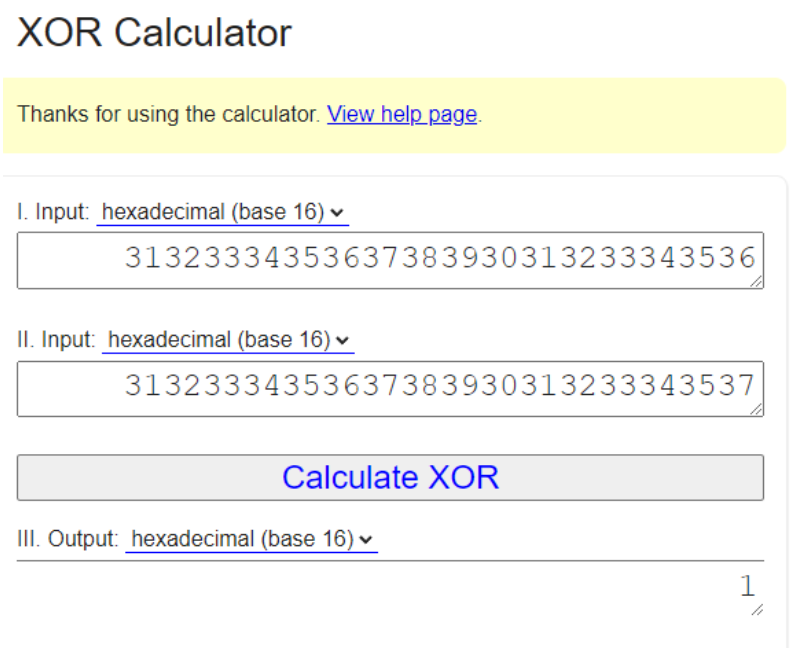
(In hex): 31323334353637383930313233343537

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 ⊕ P1 = IV2 ⊕ P2

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

P2 = 31323334353637383930313233343536 XOR 31323334353637383930313233343537 XOR 596573

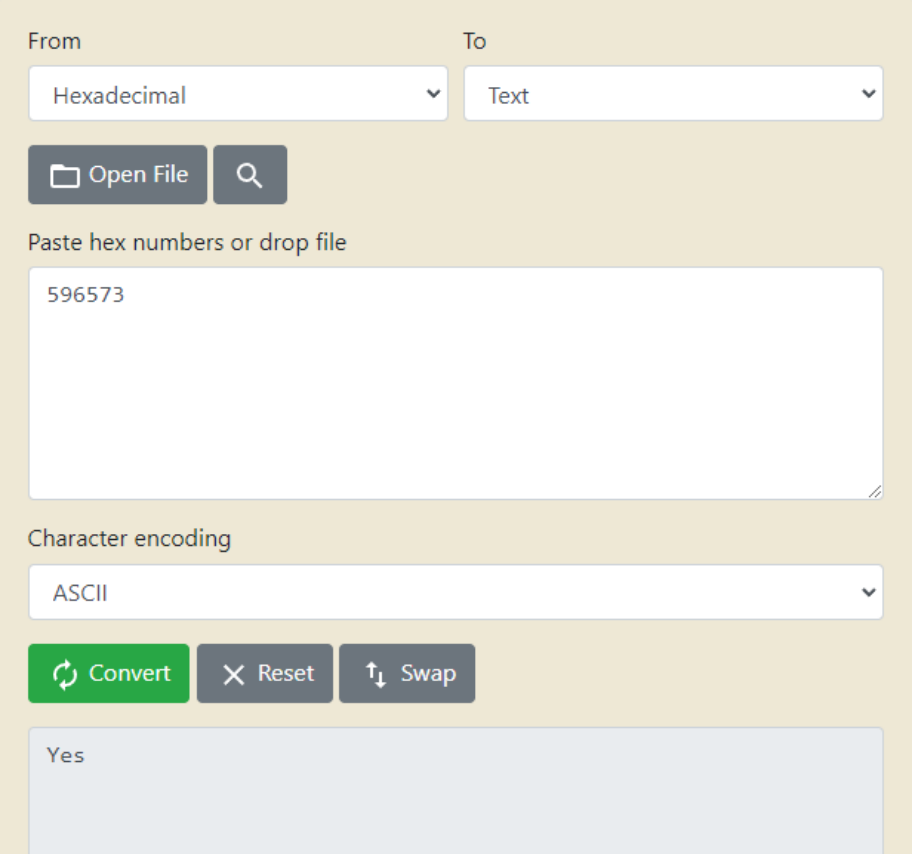


Here, if output is 1, it is Yes or else it is No.

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



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

Submitted By: -

Aastha Dhir

A20468022

adhir2@hawk.iit.edu