Luke Evans

February 27, 2021

Lab 5 – Secret Key Encryption

*1.1: Frequency Analysis*

We run the cipher text through the frequency analysis website at https://www.dcode.fr/frequency-analysis:

Occurency and Frequency Analysis  
1-grams  
% calculated | % expected

| **↑↓** | **↑↓** | **↑↓** | **↑↓** |
| --- | --- | --- | --- |
| N | 488× | 12.41% |  |
| Y | 373× | 9.49% |  |
| V | 348× | 8.85% |  |
| X | 291× | 7.4% |  |
| U | 280× | 7.12% |  |
| Q | 276× | 7.02% |  |
| M | 264× | 6.72% |  |
| H | 235× | 5.98% |  |
| T | 183× | 4.66% |  |
| I | 166× | 4.22% |  |
| P | 156× | 3.97% |  |
| A | 116× | 2.95% |  |
| C | 104× | 2.65% |  |
| Z | 95× | 2.42% |  |
| L | 90× | 2.29% |  |
| B | 83× | 2.11% |  |
| G | 83× | 2.11% |  |
| R | 82× | 2.09% |  |
| E | 76× | 1.93% |  |
| D | 59× | 1.5% |  |
| F | 49× | 1.25% |  |
| S | 19× | 0.48% |  |
| K | 5× | 0.13% |  |
| J | 5× | 0.13% |  |
| O | 4× | 0.1% |  |
| W | 1× | 0.03% |  |
| **#N : 26** | **Σ = 3931.0** | **Σ = 100.01** | **#N : 26** |

Standard English frequencies are as follows:

Array

(

[e] => 0.127

[t] => 0.0906

[a] => 0.0817

[o] => 0.0751

[i] => 0.0697

[n] => 0.0675

[s] => 0.0633

[h] => 0.0609

[r] => 0.0599

[d] => 0.0425

[l] => 0.0403

[c] => 0.0278

[w] => 0.0276

[u] => 0.0276

[m] => 0.0241

[f] => 0.0223

[g] => 0.0202

[y] => 0.0197

[p] => 0.0193

[b] => 0.015

[v] => 0.0098

[k] => 0.0077

[j] => 0.0015

[x] => 0.0015

[q] => 0.001

[z] => 0.0007

)

Using the website <https://www.guballa.de/substitution-solver>, we can easily feed in the cipher text and get the decryption key as well as the original text:



The original text is:

the oscars turn on sunday which seems about right after this long strange

awards trip the bagger feels like a nonagenarian too

the awards race was bookended by the demise of harvey weinstein at its outset

and the apparent implosion of his film company at the end and it was shaped by

the emergence of metoo times up blackgown politics armcandy activism and

a national conversation as brief and mad as a fever dream about whether there

ought to be a president winfrey the season didnt just seem extra long it was

extra long because the oscars were moved to the first weekend in march to

avoid conflicting with the closing ceremony of the winter olympics thanks

pyeongchang

one big question surrounding this years academy awards is how or if the

ceremony will address metoo especially after the golden globes which became

a jubilant comingout party for times up the movement spearheaded by

powerful hollywood women who helped raise millions of dollars to fight sexual

harassment around the country

signaling their support golden globes attendees swathed themselves in black

sported lapel pins and sounded off about sexist power imbalances from the red

carpet and the stage on the air e was called out about pay inequity after

its former anchor catt sadler quit once she learned that she was making far

less than a male cohost and during the ceremony natalie portman took a blunt

and satisfying dig at the allmale roster of nominated directors how could

that be topped

as it turns out at least in terms of the oscars it probably wont be

women involved in times up said that although the globes signified the

initiatives launch they never intended it to be just an awards season

campaign or one that became associated only with redcarpet actions instead

a spokeswoman said the group is working behind closed doors and has since

amassed million for its legal defense fund which after the globes was

flooded with thousands of donations of or less from people in some

countries

no call to wear black gowns went out in advance of the oscars though the

movement will almost certainly be referenced before and during the ceremony

especially since vocal metoo supporters like ashley judd laura dern and

nicole kidman are scheduled presenters

another feature of this season no one really knows who is going to win best

picture arguably this happens a lot of the time inarguably the nailbiter

narrative only serves the awards hype machine but often the people forecasting

the race socalled oscarologists can make only educated guesses

the way the academy tabulates the big winner doesnt help in every other

category the nominee with the most votes wins but in the best picture

category voters are asked to list their top movies in preferential order if a

movie gets more than percent of the firstplace votes it wins when no

movie manages that the one with the fewest firstplace votes is eliminated and

its votes are redistributed to the movies that garnered the eliminated ballots

secondplace votes and this continues until a winner emerges

it is all terribly confusing but apparently the consensus favorite comes out

ahead in the end this means that endofseason awards chatter invariably

involves tortured speculation about which film would most likely be voters

second or third favorite and then equally tortured conclusions about which

film might prevail

in it was a tossup between boyhood and the eventual winner birdman

in with lots of experts betting on the revenant or the big short the

prize went to spotlight last year nearly all the forecasters declared la

la land the presumptive winner and for two and a half minutes they were

correct before an envelope snafu was revealed and the rightful winner

moonlight was crowned

this year awards watchers are unequally divided between three billboards

outside ebbing missouri the favorite and the shape of water which is

the baggers prediction with a few forecasting a hail mary win for get out

but all of those films have historical oscarvoting patterns against them the

shape of water has nominations more than any other film and was also

named the years best by the producers and directors guilds yet it was not

nominated for a screen actors guild award for best ensemble and no film has

won best picture without previously landing at least the actors nomination

since braveheart in this year the best ensemble sag ended up going to

three billboards which is significant because actors make up the academys

largest branch that film while divisive also won the best drama golden globe

and the bafta but its filmmaker martin mcdonagh was not nominated for best

director and apart from argo movies that land best picture without also

earning best director nominations are few and far between

Observation: In this task we had to decrypt a cipher text back to its original text with no previous knowledge of the key. We knew that the text was English, only contained letters, was all lower case, and still included spaces. The cipher text was created using simple substitution cipher which allowed us to exploit the English character frequency analysis to help us decrypt it. One method would be to write our own tool to use the known English frequencies and do the substitution through trial and error. We used a web tool to do the analysis and suggest the encryption key. We used the key to decrypt the ciphertext back to clear text.

Explanation: Simple substitution ciphers are vulnerable to frequency analysis of the ciphertext to the known frequencies of the clear text message language. Messages using this type of encryption can easily be decrypted. To make this more difficult you can using many substitution keys depending on the occurrence of a letter.

*2.1: Encryption Using Different Ciphers and Modes*

First we can look at the plaintext that we will test using the different encryption algorithms.

A picture containing text, outdoor, electronics

Description automatically generated

Here we issue the command to encrypt the text using **128-bit AES encryption using ECB** mode. Then we decrypt and run a diff to ensure we are able to reproduce our ciphertext without any corruption:

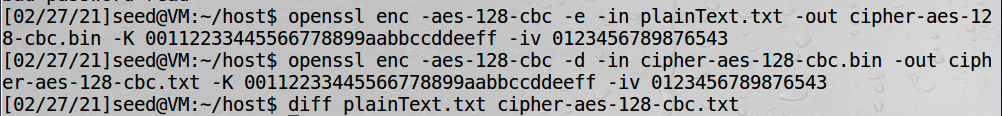
Text

Description automatically generated

A picture containing text, electronics

Description automatically generated

Next, we issue the command to encrypt the text using **128-bit AES encryption in CBC** mode. Then we decrypt and run a diff to ensure we are able to reproduce our ciphertext without any corruption:



A picture containing text, outdoor, electronics, keyboard

Description automatically generated

Next, we issue the command to encrypt the text using **128-bit AES encryption in OFB** mode. Then we decrypt and run a diff to ensure we are able to reproduce our ciphertext without any corruption:

Text

Description automatically generated

A picture containing text, outdoor

Description automatically generated

We can see all of the files we created with the different encryption algorithms:

Text

Description automatically generated

Observation: In this task we encrypted and decrypted files using the openssl library. After performing the encryption and decryption we then used the diff tool to compare the decrypted text to the original plaintext to verify there was no corruption during the process. We used three different encryption modes all 128-bit AES block ciphers in ECB, CBC, OFB modes.

Explanation: The linux openssl program allows us to easily encrypt and decrypt files using a variety of encryption algorithms and their supported modes. This is a nice tool because it abstracts the underlying complexity of the encryption algorithm away from the end user.

*3.1: Encryption Mode ECB vs CBC:*

The original image prior to decryption is shown below:

Shape, circle

Description automatically generated

First, we save the image header:



Then we encrypt the image and add back the header:

Text

Description automatically generated

Below we can view the image that was encrypted with aes-128-ecb algorithm:

Background pattern

Description automatically generated

Then we encrypt the same image but this time we use aes-128-cbc algorithm:

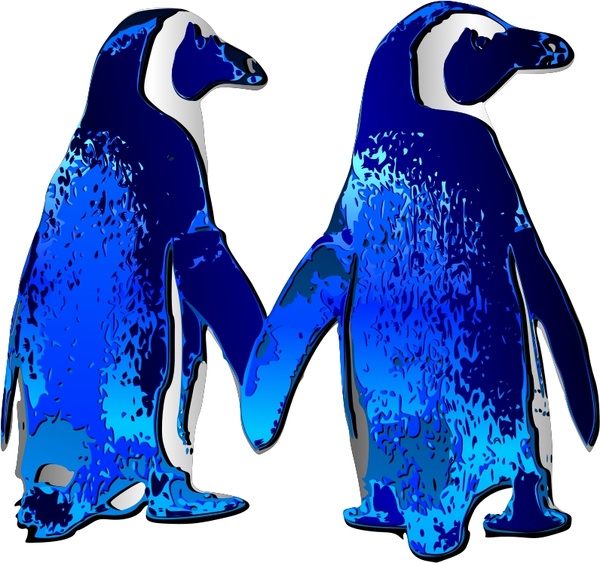
Text

Description automatically generated

Background pattern

Description automatically generated

We will run the test again using the following image:



Then we encrypt the image and add back the header:

Text

Description automatically generated

Below we can view the image that was encrypted with aes-128-ecb algorithm:

A close-up of a map

Description automatically generated with medium confidence

Then we encrypt the same image but this time we use aes-128-cbc algorithm:

Text

Description automatically generated

A close up of a green fabric

Description automatically generated with medium confidence

Observation: In this task we encrypted bitmap images using AES 128 in ECB and CBC modes. The first thing we needed to do is save the header info so that the image viewer will know how to display it. Then we run the encryption on the picture and add back the header. When we view the image in ECB mode, we are still able to see the outline and have an understanding of what the original picture was. In CBC mode, the image is completely obscured. We did the experiment with the test image as well as an image of penguins and we were able to reproduce the results.

Explanation: In this task, we experimented with two different encryption modes of AES. The CBC mode takes in an initialization vector which causes more randomization and does not leak information about the original image. In EBC mode, there is no initialization vector only a key used and information is leaked in the encrypted file.

*4.1: Not required*

*5.1: Error Propagation – Corrupted Cipher Text*

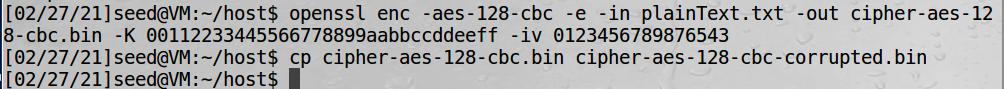
We start with a plaintext file with a size of 1022 bytes:



A picture containing text, newspaper

Description automatically generated

Next we will encrypt our file using **128-bit AES encryption in CBC** mode and make a copy of it so that we can corrupt a single byte:



Before we manually corrupt the file we can see that the 55th byte has a hex value of 0xB4:

A picture containing text, indoor

Description automatically generated

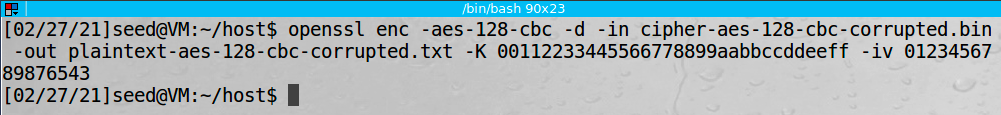
We change the 55th byte to AA:

Graphical user interface, text, application, table, email, Excel

Description automatically generated

Answer before experiment: Just as in the async lecture, I think this will cause the corruption of 1 block and 1 byte.

Next, we decrypt the corrupted file:



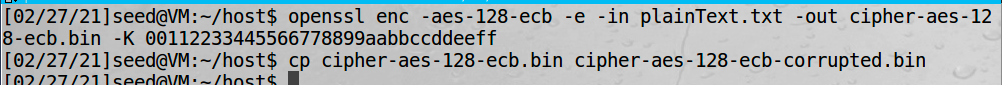
We can then dump the plaintext and see how much of the file has been corrupted by corruption of a single byte in the encrypted file:

A picture containing text

Description automatically generated

We can see from the image above that an entire block is corrupted as well as a single byte in the next block just as we expected. The second block it is harder to see the corruption but the litter z should actually be d. This is expected due to the way the initialization vector is inserted in the function

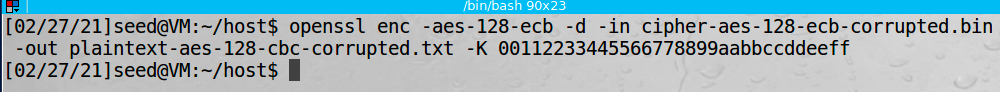
Next we will repeat the experiment using ECB mode. We expect a full block to be corrupted but that is it because there is no IV only a key used in each step.



The 55th byte is A3 and we will change to AA:

A screenshot of a computer

Description automatically generated with low confidence



Then we dump the file and examine the affect:

A picture containing text

Description automatically generated

As expected, only a single block is corrupted. This is why a random initialization vector is useful to increase the level of confusion and diffusion that is introduced by the encryption algorithm.

Next, we will repeat the experiment using OFB mode. Since OFB mode acts like a stream cipher I expect only a single block will be affected by the corruption of a single byte in the encrypted file:

Text

Description automatically generated

The 55th byte was 0x71 and we will change to 0xAA:

Graphical user interface, text, application, table, email, Excel

Description automatically generated

Next we will decrypt and examine the plaintext:

A close-up of a document

Description automatically generated with low confidence

As expected of a stream cipher, only a single byte was affected by corrupting a single byte.

Finally, we will repeat the experiment using CFB mode. CFB mode uses an initialization vector after the encryption and feeds it into the next block after the encryption. Because of this it will behave similarly as CBC with a full block affected and then a single byte only it will be a single byte in the block that was changed and the entire next block

Text

Description automatically generated

The 55th byte was 0xCB and we will change to 0xAA:

A picture containing text, indoor, screenshot

Description automatically generated

Next we will decrypt and examine the plaintext:

Text

Description automatically generated

As expected a single byte is affected and the entire next block. This again has to do with where the IV is XORed with the encryption result and fed into the next block.

Observation: In this task we had to encrypt data with the AES block cipher in different modes including CBC, EBC, CFB, and OFB. After the encryption was done for each algorithm we modified the 55th byte in the encrypted files and changed it to be 0xAA. Then we decrypted the data and examined the affect of changing a single byte in the encrypted file to simulate what would happen if the data were corrupted. All of the algorithms produced different results. Our theory of what would occur and the actual results are all listed in the experiments above.

Explanation: Data corruption during encryption causes different data corruption during decryption depending on which AES mode is being used. The differences have to do with the use of random initialization vectors and where they are XORed and fed into the next block.