Dr. Palvi Aggarwal

CS 4351/5352: Computer Security

Assignment 4

Total Points: 100 Points

Submission Date: March 24, 2024, at 11:59 P.M.

Instructions: Please carefully read the following instructions.

- Please add your name and student ID as part of your assignment.
- Please include a detailed description of your observations and findings for each task in your report.
- Please submit your report as a PDF file and your code as a zipped folder.
- Please use the following naming convention for submission: Name_AssignmentNumber.
- If you wish to use one or more of your grace days, please check with the professor or the teaching assistant (TA) at **least one day prior to the deadline** to ask if you can utilize these days.
- If you have CASS accommodation, please contact the professor and the teaching assistant (TA) to provide this accommodation.

(15 points) Task 1: Caesar Cipher

A) **Message:** Tvsxigx csyv hexe amxl gevi erh geyxmsr. Epaecw ywi wxvsrk erh yrmuyi tewwasvhw jsv iegl eggsyrx, vikypevpc ythexi, erh oiit csyv hizmgiw erh wsjxaevi yt xs hexi

- Write a Python program to determine the Caesar cipher's shift key that was used to encode this message.
- Once you have identified the key, decode the message to its original text.

B) Plain Text Message: Programming is like cooking. Even if you follow the recipe perfectly, there's always a chance you'll end up with a big mess.

- Use the program you completed in section A and encode the plain text message using Caesar Cipher and using the shift key 7. Provide the encrypted message as your answer.

(15 points) Task 2: Frequent Analysis

Message: lrvmnir bpr sumvbwvr jx bpr lmiwv yjeryrkbi jx qmbm wi bpr xjvni mkd ymibrut jx irhx wi bpr riirkvr jx ymbinlmtmipw utn qmumbr dj w ipmhh but bj rhnvwdmbr bpr yjeryrkbi jx bpr qmbm mvvjudwko bj yt wkbrusurbmbwjk lmird jk xjubt trmui jx ibndt wb wi kjb mk rmit bmiq bj rashmwk rmvp yjeryrkb mkd wbi iwokwxwvmkvr mkd ijyr ynib urymwk nkrashmwkrd bj ower m vjyshrbr rashmkmbwjk jkr cjnhd pmer bj lr fnmhwxwrd mkd wkiswurd bj invp mk rabrkb bpmb pr vjnhd urmvp bpr ibmbr jx rkhwopbrkrd ywkd vmsmlhr jx urvjokwgwko ijnkdhrii ijnkd mkd ipmsrhrii ipmsr w dj kjb drry ytirhx bpr xwkmh mnbpjuwbt lnb yt rasruwrkvr cwbp qmbm pmi hrxb kj djnlb bpmb bpr xjhhjcwko wi bpr sujsru msshwvmbwjk mkd wkbrusurbmbwjk w jxxru yt bprjuwri wk bpr pjsr bpmb bpr riirkvr jx jqwkmcmk qmumbr cwhh urymwk wkbmvb

A) Compute the relative frequency of all letters A...Z in the ciphertext. You may want to use a tool such as the open-source program CrypTool for this task. However, a paper and pencil approach are also still doable.

B) Decrypt the ciphertext with the help of the relative letter frequency of the English language.

(10 points) Task 3: DES

A) Explain the process of encryption and decryption in the Data Encryption Standard (DES) algorithm.

(10 points) Task 4: AES

A) Explain the AES algorithm in general terms, mentioning its main features, such as the length of the key, the size of the block, and the number of rounds.

(15 Points) Task 5: Encryption using Different Ciphers and Modes

In this task, you will use various encryption algorithms and modes. You can use the following `openssl enc` command to encrypt/decrypt a file. To see the manuals, you can type `man openssl` and `man enc`.

```
$ openssl enc -ciphertype -e -in plain.txt -out cipher.bin \
            -K 00112233445566778889aabbccddeeff \
            -iv 0102030405060708
```

Please replace the `ciphertype` with a specific cipher type. In this task, you should try at least **3 different ciphers**. You can find the meaning of the command-line options and all the supported cipher types by typing "`man enc`". Please create your own plain text file and show the output after running different cipher types. Discuss your observations.

(15 Points) Task 6: Encryption Mode – ECB vs. CBC

The file `pic_original.bmp` is included in the Labsetup.zip file, and it is a simple picture. We would like to encrypt this picture, so people without the encryption keys cannot know what is in the picture. Please encrypt the file using the ECB (Electronic Code Book) and CBC (Cipher Block Chaining) modes, and then do the following:

- Let us treat the encrypted picture as a picture and use a picture viewing software to display it. However, For the `.bmp` file, the first 54 bytes contain the header information about the picture, we must set it correctly, so the encrypted file can be treated as a legitimate .bmp file. We will replace the header of the encrypted picture with that of the original picture. We can use the bless hex editor tool (already installed on our VM) to directly modify binary files. We can also use the following commands to get the header from p1.bmp, the data from p2.bmp (from offset 55 to the end of the file), and then combine the header and data together into a new file. *Please show the steps and discuss the observations.*

```
$ head -c 54 p1.bmp > header
$ tail -c +55 p2.bmp > body
$ cat header body > new.bmp
```

- Display the encrypted picture using a picture viewing program (we have installed an image viewer program called eog on our VM). Can you derive any useful information about the original picture from the encrypted picture? *Please explain your observations and provide a screenshot of the encrypted picture.*

(20 points) Task 7: Error Propagation – Corrupted Cipher Text

To understand the error propagation property of various encryption modes, we would like to do the following exercise:

1. Create a text file that is at least 1000 bytes long.
2. Encrypt the file using the AES-128 cipher.
3. Unfortunately, a single bit of the 55th byte in the encrypted file got corrupted. You can achieve this corruption using the bless hex editor.
4. Decrypt the corrupted ciphertext file using the correct key and IV.

Please answer the following question: How much information can you recover by decrypting the corrupted file, if the encryption mode is ECB, CBC, CFB, or OFB, respectively? Please answer this question before you conduct this task, and then find out whether your answer is correct or wrong after you finish this task. Please provide justification.