LAB 7

Secret-Key Encryption

Task 1: Frequency Analysis

* Generate the encryption key and convert all upper cases to lower cases, and then removed all the punctuations and numbers.

[03/18/22]seed@VM:~/.../Files$ ./seckey.py

**tqnvmaxwjlegrciopdkzfusybh (SECRET KEY)**

[03/18/22]seed@VM:~/.../Files$ tr ’abcdefghijklmnopqrstuvwxyz’ ’tqnvmaxwjlegrciopdkzfusybh’ < plaintext.txt > ciphertext.txt[03/18/22]seed@VM:~/.../Files$ ./freq.py

* By using the tr command above, plaintext.text was encrypted to generate ciphertext.txt

Text

Description automatically generated with low confidence

A picture containing letter

Description automatically generated

* Frequency was analysed using the freq.py which was provided on the generated ciphertext.txt

Text

Description automatically generated

Table

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidence

* Using tr command, we found out the plaintext from cipher text through continuous inspection we get the output.

A picture containing letter

Description automatically generated







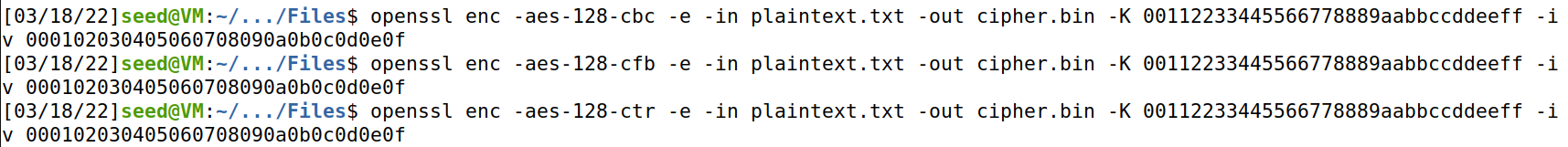




A picture containing letter

Description automatically generated

Task 2: Encryption using Different Ciphers and Modes



* As seen above, encryption was done using **-aes-128-cbc, -aes-128-cfb & -aes-128-ctr** respectively.

Task 6: Initial Vector (IV) and Common Mistakes

* Task 6.1. IV Experiment

We created messages for P1 and P2, encrypted them with the OFB mode, and then used the xor.py tool from the book to xor the hex strings of the ciphertexts and plain texts. After doing so, the message from P2 appeared meaning we are successful. The command used are listed below:

Text

Description automatically generated with medium confidence

* Task 6.2. Common Mistake: Use the Same IV

A picture containing graphical user interface

Description automatically generated

* Task 6.3. Common Mistake: Use a Predictable IV

A screenshot of a computer

Description automatically generated with medium confidence

Task 7: Programming using the Crypto Library

