UNIVERSITY
OF
CALCUTTA

B.Sc (H) 6th Semester Examination 2021

In Computer Science

Triple Security Message Cryption System Using Cryptography Ciphers & Steganography Techniques

Paper Name: **Project Work**

Paper Code: **CMS-A-CC-6-14-P**

Registration No.: **012-1112-0806-18**

012-1111-1318-18

012-1112-0933-18

Roll No.: **183012-21-0121**

183012-21-0106

183012-21-0125

Supervisor's Certificate

This is to certify that **Mr. Goutam Biswas**, **Mr. Sayan Sarkar** & **Mr. Rupam Das**, students of B.Sc. Honours in Computer Science of Asutosh College under the University of Calcutta have worked under my supervision and guidance for their Project Work and prepared a Project Report with the title "**Triple Security Message Cryption System Using Cryptography Ciphers & Steganography Techniques**". Their work is genuine and original to the best of my knowledge.

Place: Kolkata

Date: 7th August, 2021

Signature:

Name: **Smt. Atrayee Chatterjee**

Designation: **Faculty Member,**

Department of Computer Science,

Asutosh College

Acknowledgement

We do extend our sincere appreciation, first to the University of Calcutta and also to the **Department of Computer Science**, **Asutosh College** for giving us an opportunity to do this wonderful research work on the topic "**Triple Security Message Cryption System Using Cryptography Ciphers & Steganography Techniques**".

We are grateful to our respected HoD, **Dr. Samir Malakar**, for motivating us to complete this project with complete focus and attention.

We are sincerely thankful to our Project Guide **Smt. Atrayee Chatterjee**, Faculty Member, Department of Computer Science, Asutosh College**,** under whose guidance we have successfully completed this project and time spent with her had been a great learning experience. We thank her for constant encouragement, warm responses and filling every gap with valuable ideas has made this project successful. She made it possible for us to put all my theoretical knowledge to work out on this topic.

We wish to record our sincere thanks to our parents and to our fellow friends for their continuous co-operation and inspiration within the limited time frame.

Abstract

This Project Work or Paper describes the designing and implementation of Triple Security Message Cryption. Sending Messages and Information has been and will always be of utmost importance, whether in times of Peace or War, Public use or Private use, Security purposes or Common purposes. And for this purpose, it is needed to well-protect the Information from the unwanted receivers. This paper focuses on encrypting data, information and messages to prevent the same.

Messages were, are and will be the staple in every generation, and thus come the matter of privacy and security.

Keeping this issue in mind, we have tried to ensure data security by enhancing Data Encryption.
With the help of two different techniques, the data or information goes through a series of processes and modifies the data before data transfer and the recipient receives the encrypted data.
Then using the same Cryption system, he can decrypt it and obtain the message while ensuring the security of the message.

Table of Contents

Introduction

Domain Description

Cryptography: Cryptography is a method/technique of securing information and communications through use of codes so that only that person for whom the information is intended can understand it and process it. It provides for secure communication in the presence of malicious third-parties—known as *adversaries*. Thus, preventing unauthorized access to information. The prefix "crypt" means *hidden* or *vault* and suffix "graphy" means *writing*.
In Computer Science, Cryptography refers to secure information and communication techniques derived from Mathematical concepts and a set of rule-based calculations called *Algorithms*, to transform messages in ways that are hard to decipher. These deterministic algorithms are used for *cryptographic key generation*, *digital signing*, verification to protect *Data Privacy*, Web Browsing on the Internet, and confidential communications such as *Credit Card* transactions and *E-mail*.

Steganography: It is the practice of hiding a secret message inside of (or even on top of) something that is not secret. That something can be just about anything you want. These days, many examples of steganography involve embedding a secret piece of text inside of a picture. Or hiding a secret message or script inside of a Word or Excel document. The root "steganos" is Greek for "hidden" or "covered," and the root "graph" is Greek for "to write." The purpose of steganography is to conceal and deceive. It is a form of covert communication and can involve the use of any medium to hide messages. It's not a form of cryptography, because it doesn't involve scrambling data or using a key. Instead, it is a form of data hiding and can be executed in clever ways. Where cryptography is a science that largely enables privacy, steganography is a practice that enables secrecy – and deceit.

Hence,

Cryptography: ***"The discipline which embodies principles, means and methods for the transformation of data in order to hide its information content, prevent its undetected modification, or prevent its unauthorized use"***

Steganography: ***"A method of hiding a secret message inside of other data."***

Motivation

Cryptography can be two types:- ***Symmetric*** and ***Asymmetric***. With symmetric cryptography, the same key is used for both encryption and decryption. A sender and a recipient must already have a shared key that is known to both. Key distribution is a tricky problem and was the impetus for developing asymmetric cryptography.

With Asymmetric crypto, two different keys are used for encryption and decryption. Every user in an asymmetric cryptosystem has both a public key and a private key. The private key is kept secret at all times, but the public key may be freely distributed.

Data encrypted with a public key may only be decrypted with the corresponding private key. So, sending a message to John requires encrypting that message with John's public key. Only John can decrypt the message, as only John has his private key. Any data encrypted with a private key can only be decrypted with the corresponding public key. Similarly, Jane could digitally sign a message with her private key, and anyone with Jane's public key could decrypt the signed message and verify that it was in fact Jane who sent it.

Keeping this in mind, we proceeded in the work of our project paper. Also, to add extra security and protection, we hide the encrypted message inside an image to perceive as 'hidden text'.

Scope of Work

Purpose: To increase difficulty of accessing information by unauthorized adversaries.

Project Scope:

Project Name	Triple Security Message Cryption System
Background to proposed work	Multiple use of Cryptography ciphers
Objectives	To increase difficulty of accessing information by unauthorized adversaries
Deliverables	Encrypted Message, Image containing the encrypted message
Milestones	• Selection of compatible Cipher techniques • Joining of Cipher techniques to encrypt data • Applying Steganography techniques • Final execution of program Assembling a suitable Algorithm for the program
Reports	• Check on Cipher techniques • Stay in touch with user for Cipher keys • Check on using multiple Cipher processes • Checking of the accuracy of encrypted data Check on message carrying Image produced
Technical Requirements	OS: Windows 7,8,10 or above, Linux Software: Python 3.0 or above/PyCharm

Background/Review of Related Work

Now-a-days, the Data and Information security is one from the most challenges that face the organizations that need to transfer sensitive or private data online. According to a survey, the number of hackers or online data thief increased rapidly in the last years. The hackers focus on stole the sensitive data such as credit cards numbers and organizations secrets. Thus, the organizations always afraid from the security level of data transferring channels.

Cryptography is playing a major role in data protection in applications running in a network environment that used to secure the online transferred data. But the main weakness of Cryptography method is that the hackers can detect the encrypted messages and try to decrypt these messages through many ways such as automatic counters or random tests based on mathematic calculations. So to improve the level of security, Steganography is included as a supportive security method. The main aim of Steganography is to maximize the difficulty of detect the encrypted data that transferred online. Therefore, the encrypted data can be hidden in media file before send this file through online application. The receiver can ex-tract the encrypted data from the media file and decrypt the data using the secret keys. Thus, the hackers will face difficulty to discover encrypted data the transferred online. Steganography is the art of communicating in a way which hides the existence of the communication. Cryptography scrambles a message so it can't be understood; the Steganography hides the message so it can't be seen. So, the combined cryptography and steganography into one system increases the security and confidentiality of it.

Hybrid Cryptography and Steganography are the latest methods that have contributed greatly towards the improvement of security of message transmission.

Khider Nassif Jassim (Department of Statistics. Faculty of Management and Economics, Wasit University, Al-Kut, Iraq) and **Zico Pratama Putra** (School of Electronic Engineering and Computer Science, Queen Mary University of London) proposed a multitasking system for "Improving the cryptography security level using supportive method which is Steganography". [5]

"There are four stages represent the methodology of this paper; (1) encrypt the original texts using RSA algorithm, (2) hide the encrypted texts in Image files, (3) extract the encrypted texts from Image files, and (4) decrypt the original texts using decryption key of RSA algorithm"

It is expected to improve the security level of the online transferred textual data. The performance of the final results will be evaluated through compare the Image files quality before and after hide the data in these files. The quality of the original and stego Image files need to be same or near in order to maximize the difficulty of detect that there data hide in these files. [5]

We referred to this project for ideas and motivation and the techniques used were the main inspiration towards the foundation of our Project Work.

Methodology

Problem Formulation: In this Section, we detail the general structure of three different ciphers focusing on Cryptography. We then describe how to encrypt and decrypt data and finally introduce the encoding and decoding of text into a given image.

Algorithm Description: Write a suitable Algorithm to Encode and Decode a message into an Image, after Encrypting the input message thrice using separate Cipher techniques. Show the Decryption and Decoding as well to prove accuracy and viability.

Design Description:

Cryptography Ciphers used:

1. OneTime Pad Cipher
2. Substitution Caesar Cipher
3. Columnar Transposition Cipher

Steganography Process used: Image-based Encoding & Decoding

OneTime Pad Cipher: One Time Pad algorithm is also known as Vernam Cipher. It is a method of encrypting alphabetic plain text. It is one of the Transposition techniques which converts plain text into ciphertext. The relation between the key and plain text, the length of the key should be equal to that of plain text. In this mechanism, we assign a number to each character of the Plain-Text. [3]

eg:-	A	B	C	D	E	F	G	H	I	J
	0	1	2	3	4	5	6	7	8	9
	K	L	M	N	O	P	Q	R	S	T
	10	11	12	13	14	15	16	17	18	19
	U	V	W	X	Y	Z				
	20	21	22	23	24	25				

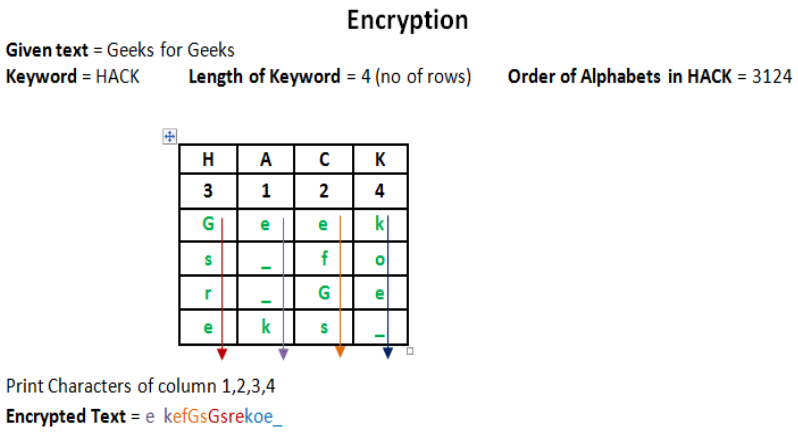
Plain text — H E L L O → 7 4 11 11 14
Key — M O N E Y → 12 14 13 4 24

Plain text + key → 19 18 24 15 38
→ 19 18 24 15 12 (= 38 – 26)
Cipher Text → T S Y P M

Substitution Caesar Cipher: This Cipher is one of the earliest & simplest method of encryption technique. In a Substitution cipher, any character of plain text from the given fixed set of characters is substituted by some other character from the same set depending on a key. For example, with a shift of 1, A would be replaced by B, B would become C, and so on. [3]

eg:- Plain Text: I am studying Data Encryption
Key: 4
Output: M eq wxyhCmrk Hexe IrgvCtxmsr

Columnar Transposition Cipher: A Transposition Cipher is a method of encryption by which the positions held by units of plaintext (which are commonly characters or groups of characters) are shifted according to a regular system, so that the ciphertext constitutes a permutation of the plaintext.



In Columnar Transposition Technique, the message is written out in rows of a fixed length, and then read out again column by column, and the columns are chosen in same scrambled order. Both the width of the rows and the permutation of the columns are usually defined by a keyword. [3]

Image Based Steganography: Steganography is the method of hiding secret data in any image/audio/video. In a nutshell, the main motive of steganography is to hide the intended information within any image/audio/video that doesn't appear to be secret just by looking at it.
The idea behind image-based Steganography is very simple. Images are composed of digital data (pixels), which describes what's inside the picture, usually the colors of all the pixels. Since we know every image is made up of pixels and every pixel contains 3-values (red, green, blue).

Encode the Data :
Every byte of data is converted to its 8-bit binary code using ASCII values. Now pixels are read from left to right in a group of 3 containing a total of 9 values. The first 8-values are used to store binary data. The value is made odd if 1 occurs and even if 0 occurs.

For Example :
Suppose the message to be hidden is ' Hii '. Since the message is of 3-bytes, therefore, pixels required to encode the data is 3 x 3 = 9. Consider a 4 x 3 image with a total 12-pixels, which are sufficient to encode the given data.

[(27, 64, 164), (248, 244, 194), (174, 246, 250), (149, 95, 232), (188, 156, 169), (71, 167, 127), (132, 173, 97), (113, 69, 206), (255, 29, 213), (53, 153, 220), (246, 225, 229), (142, 82, 175)]

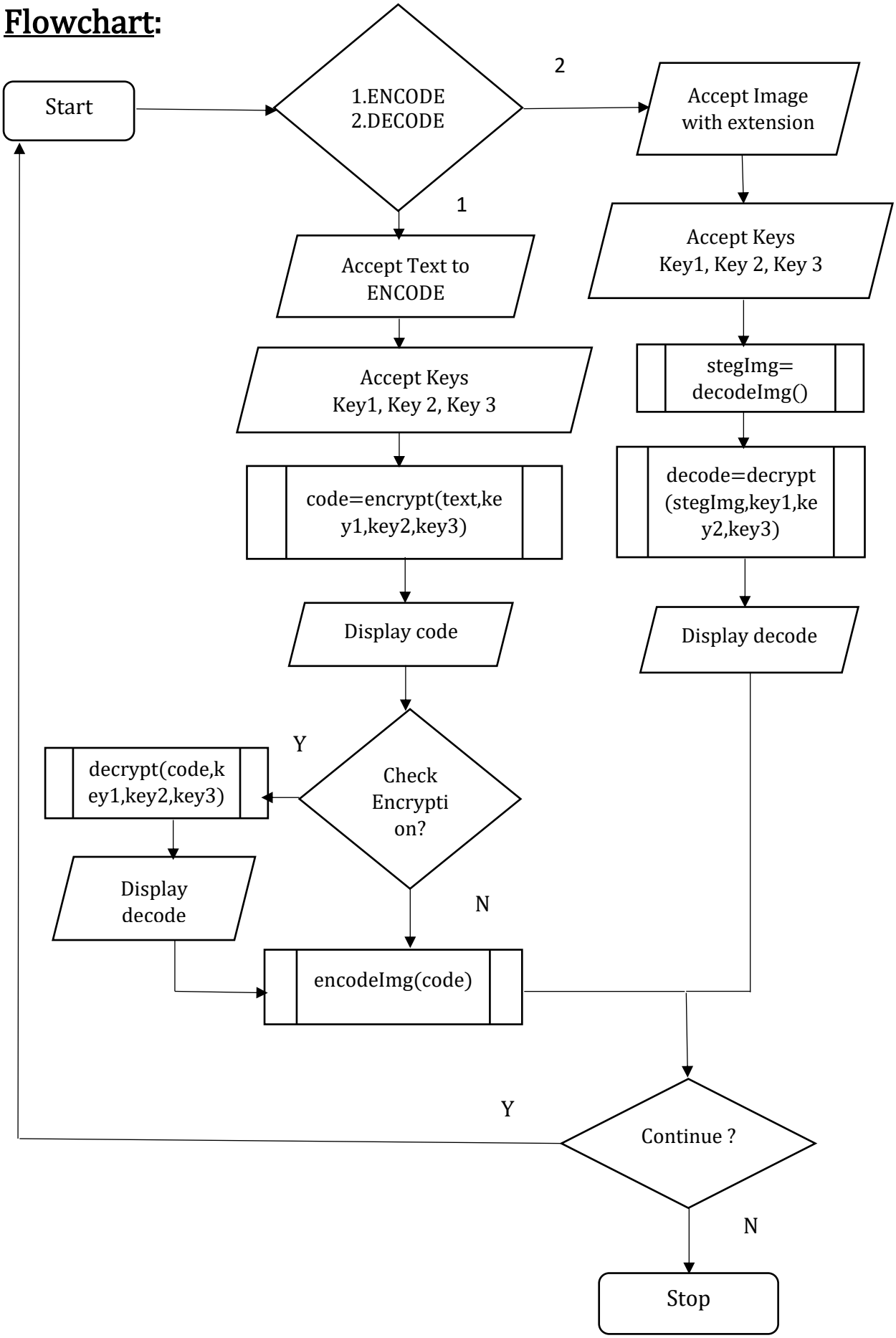
ASCII value of ' H ' is 72 whose binary equivalent is 01001000.
Taking first 3-pixels (27, 64, 164), (248, 244, 194), (174, 246, 250) to encode. Now change the pixel to odd for 1 and even for 0. So, the modifies pixels are (26, 63, 164), (248, 243, 194), (174, 246, 250). Since we have to encode more data, therefore, the last value should be even. Similarly, 'i' can be encoded in this image.
The new image will look like :

[(26, 63, 164), (248, 243, 194), (174, 246, 250), (148, 95, 231), (188, 155, 168), (70, 167, 126), (132, 173, 97), (112, 69, 206), (254, 29, 213), (53, 153, 220), (246, 225, 229), (142, 82, 175)]

Decode the Data :

To decode, three pixels are read at a time, till the last value is odd, which means the message is over. Every 3-pixels contain a binary data, which can be extracted by the same encoding logic. If the value if odd the binary bit is 1 else 0. [4]

Flowchart:


Start
1.ENCODE
2.DECODE
2
Accept Image with extension
1
Accept Text to ENCODE
Accept Keys Key1, Key 2, Key 3
stegImg= decodeImg()
Accept Keys Key1, Key 2, Key 3
code=encrypt(text,key1,key2,key3)
decode=decrypt (stegImg,key1,key2,key3)
Display code
Display decode
Y
Check Encryption?
decrypt(code,key1,key2,key3)
Display decode
N
encodeImg(code)
Y
Continue ?
N
Stop

Implementation

MAIN ALGORITHM:

1. First we import packages OneTimePad.
2. Import package String.
3. Import package Math.
4. Also from PILLOW, we import package Image.
5. Storing all alphabets (lowercase and uppercase) in all letters using string.ascii_letters.
6. Then define the necessary processes in functions:
   a) Subencrypt()
   b) Subdecrypt()
   c) transEncrypt()
   d) transDecrypt()
   e) encrypt()
   f) decrypt()
   g) genData()
   h) modPix()
   i) encode_enc()
   j) encodeImg()
   k) decodeImg()
7. For the body, display a message for the start of user interaction.
8. Assign char value 'Y' to variable 'choice'.
9. Start a while loop with condition being variable 'choice' is equal to 'y' or 'Y'.
10. Display a message for user input for Encrypting or Decrypting. Take integer input in variable 'a'.
11. Start if with condition variable 'a' equals 1(1 denotes Encryption).
12. Accept text to encrypt.
13. Display message to input three keys for encryption as asked.
14. Accept Shift integer value for Substitution cipher key.
15. Accept String value for Transposition cipher key.
16. Accept String value for OneTimePad cipher key.
17. Use function encrypt() with the input text and three keys as parameters.
18. Store it in variable 'code'.
19. Start an if with choice 'ch' from user input, to check the correctness of the encrypted code.
20. Use function decrypt() with the encrypted text 'code' and three keys as parameters.
21. Store it in variable 'decode'.
22. Display the decoded text 'decode' for user verification.
23. Close if.
24. Proceed by encoding the encrypted text 'code' in the image.

25. Use function encodeImg(code) to do step 24. The image extension should be provided by user.
26. Display that the Image has been encrypted.
27. Accept ‘choice’ from user to Continue.
28. Close if.
29. Start elif with condition variable ‘a’ equals 2 (2 denoted Decryption).
30. Use function decodeImg() and store it in variable ‘stegoImage’. The image extension should be provided by user.
31. Display message to input three keys for decryption as asked.
32. Accept Shift integer value for Substitution cipher key.
33. Accept String value for Transposition cipher key.
34. Accept String value for OneTimePad cipher key.
35. Use function decrypt() with ‘stegoImage’ and the keys as parameters.
36. Display the Decoded message obtained from the image.
37. Accept ‘choice’ from user to Continue.
38. Close elif.
39. Start else.
40. Raise an Exception for wrong ‘choice’ input.
41. Close while loop.
42. Display a thankful message for using the program.

FUNCTIONS USED:

1. def Subencrypt(plain_text,key) : This function accepts a plain text and a key for Substitution Encryption.
   Input:
   1. A String of both lower and upper-case letters, called PlainText.
   2. An Integer denoting the required key.
   Procedure:
   1. Create a list of all the characters.
   2. Create a dictionary to store the substitution for all characters.
   3. For each character, transform the given character as per the Substitution encrypting rule, i.e, adding Shift key value to Ascii value of the letters present in the text to create Cipher text. Use for loop to carry out this process.
   4. Start a for loop to generate Cipher text. Add spaces where necessary using ‘join’.
   5. Return the new string generated.
2. def Subdecrypt(cipher_text,key) : This function accepts a cipher text and a key for Substitution Decryption.
   Input:
   1. A String of both lower and upper-case letters, called CipherText.

2. An Integer denoting the required key.

Procedure:

1. Create a list of all the characters.
2. Create a dictionary to store the decryption for all characters.
3. For each character, transform the given character as per the Substitution decrypting rule, i.e, subtracting Shift key value to Ascii value of the letters present in the text to create Plain text. Use for loop to carry out this process.
4. Start a for loop to generate Plain text. Add spaces where necessary using 'join'.
5. Return the new string generated.

3. def transEncrypt(text,key): This function accepts a plain text and a key for Transposition Encryption.

Input:

1. A String of both lower and upper-case letters, called PlainText.
2. A String Input value denoting the required key.

Procedure:

1. Track key index.
2. Create a string variable to store the transposition for all characters.
3. Calculate column of the matrix (length of the key).
4. Calculate the maximum row of the matrix (ceil value of text length/column length).
5. Add padding character '_' for empty cells of matrix.
6. Create matrix and insert message characters row-wise.
7. Read matrix column-wise using key order.
8. Return the new string generated.

4. def transDecrypt(cipher,key): This function accepts a cipher text and a key for Transposition Decryption.

Input:

1. A String of both lower and upper-case letters, called Cipher Text.
2. A String Input value denoting the required key.

Procedure:

1. Track key index. Track text indices.
2. Create a string variable to store the decryption for all characters.
3. Calculate column of the matrix (length of the key).
4. Calculate the maximum row of the matrix (ceil value of text length/column length).
5. Convert key into list and sort alphabetically to access each character by its alphabetical position.
6. Create an empty matrix to store deciphered message.
7. Use for loops and arrange the matrix column wise according to permutation order by adding into new matrix.

8. Convert decrypted message matrix into a string.
9. Return the new string generated.

5. def encrypt(text, key1 ,key2 ,key3): This function accepts a plain text and keys for triple Encryption.

   Input:

   1. A String of both lower and upper-case letters, called Plain Text.
   2. An Integer Shift value and two String Input values denoting the required keys respectively.

   Procedure:

   1. Use function Subencrypt() with plain text and Integer input key and store it in ‘cipher1’.
   2. Use function transEncrypt() with ‘cipher1’ and String input key and store it in ‘cipher2’.
   3. Use function onetimepad.encrypt( ) from OneTimePad package with ‘cipher2’ and String Input key as parameters and store it in ‘cipher3’.
   4. Return ‘cipher3’, i.e. the triple Encrypted Cipher text.

6. def decrypt(cipher, key1 ,key2 ,key3): This function accepts a Cipher text and keys for triple Decryption.

   Input:

   1. A String of both lower and upper-case letters, called Cipher Text.
   2. An Integer Shift value and two String Input values denoting the required keys respectively.

   Procedure:

   1. Use function onetimepad.decrypt( ) from OneTimePad package with cipher text and Integer input key and store it in ‘cipher3’.
   2. Use function transDecrypt() with ‘cipher3’ and String input key and store it in ‘cipher2’.
   3. Use function Subdecrypt() with ‘cipher2’ and String Input key as parameters and store it in ‘cipher1’.
   4. Return ‘cipher1’, i.e. the triple Decrypted Plain text.

7. def genData(data): This function accepts Encrypted Cipher text as data and returns the list of binary codes of given data.

   Input:

   1. A String of both lower and upper-case letters, namely the Encrypted Cipher text as data.

   Procedure:

   1. Create a list to store binary codes of given data.
   2. Start a for loop.
   3. Transform each character of data into Unicode first,and then format it to binary codes respectively and append them into the list. Use append(format(ord(i), ‘08b’) for this step.
   4. Close for loop.
   5. Return the list.

8. def modPix(pix,data): This function accepts contents of the image as a sequence object containing pixel values as data and modifies the pixels according to the 8-bit binary data.
   Input:
   1. Pixel values from the Image.
   2. A String of both lower and upper-case letters, namely the Encrypted Cipher text as data.
   Procedure:
   1. Convert the data into list of binary codes using function genData() and store it in 'datalist'.
   2. Get length of 'datalist'. Access and store the input pixel values.
   3. Start a for loop and extract 3-pixels at a time.
   4. Start a for loop to change pixel value to odd for 1 and to even for 0.
   5. Eighth pixel of every set tells whether to stop or read further. Start a nested if to check. 0 means keep reading, 1 means that message is over.
   6. Use tuple to keep all changed pixel values together.
9. def encode_enc(new_img, data): This function accepts image and Encrypted Cipher data to encode.
   Input:
   1. Image where data is to be encoded.
   2. A String of both lower and upper-case letters, namely the Encrypted Cipher text as data.
   Procedure:
   1. Calculate and store dimensions of the new image.
   2. Start a for loop and use modPix() function to modify the pixels.
   3. Put new modifies pixel values in the new image using putpixel() function from Image package.
10. def encodeImg(data): This function accepts Encrypted Cipher text as data and encodes the data into the Image.
   Input:
   1. A String of both lower and upper-case letters, namely the Encrypted Cipher text as data.
   Procedure:
   1. Display a message to accept Image name to use with full extension. Store it.
   2. Open image using Image.open() from Image package of PILLOW.
   3. Copy the image into a new variable, possible 'newimg'.
   4. Encode the data using encode_enc() function with 'newimg' and data as the parameters.
   5. Accept a new Image name and Extension for the 'newimg'.
   6. Save the 'newimg' with the new name and Extension.
11. def decodeImg(): This function decodes the Encrypted data from the Image.

Input:

1. Image name and Extension of the Image from where Data can be extracted, is to be provided by user during the execution of this function.

Procedure:

1. Display a message to accept Image name to use with full extension. Store it.
2. Open image using Image.open() from Image package of PILLOW, and open it with read ‘r’ format.
3. Create a string variable to store the data.
4. Extract the pixel values of data and access it using ’iter(image.getdata())’.
5. Start a while loop with condition Boolean ‘true’.
6. To decode, read three pixels at a time till the last value is odd, which means the message is over. Every 3-pixels contain a binary data, which can be extracted by same encoding logic. If the value is odd, binary bit is 1 else 0.
7. Return the data.

Results & Discussion

Encoding (with Viability): Here, an original image has been provided by the user along with three different Cipher keys for respective Ciphers.

Original Image (Nature.jpg)
User provided Keys: 4, Hack, Crypto

Image Encoded with CipherText (Nature1.png)
Keys used: Same as provided by the user

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:23:07) [MSC
v.1927 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more
information.

>>>

==== RESTART: C:/Users/HP/Desktop/Sixth Semester
Project/ProjectCODE.py ====

Welcome To TriPLe SeCuriTy IMAGE EnCODER-DeCODER.

:: Welcome to Steganography ::

Choose:

1. Encode
```

```
2. Decode

1

Enter text to Encrypt: Good and Evil are just a matter of
Perspective

Enter Three Keys for Encryption as followed:

Enter Key for Substitution Cipher(Input Shift Value):4

Enter Key for Transposition Cipher(Input String Value):Hack

Enter Key for OneTimePad Cipher(Input String Value):Crypto

The Encoded Message
:085259001d18630a0a191d15301730505417321b13061306300003151a4f2
60459070c302b1a14060d0a3b522d041930

Do You Want to DeCode the message to Check? Press Y or N: Y

The Decoded Message: Good and Evil are just a matter of
Perspective

Enter image name(with extension) : Nature.jpg

Enter the name of new image(with extension) : Nature1.png

Code has been Encrypted in the Provided Image.

DO YOU WISH TO CONTINUE? Y

:: Welcome to Steganography ::

Choose:

1. Encode

2. Decode

2

Enter image name(with extension) : Nature1.png

Enter Three Keys for Decryption as followed:

Enter Key for Substitution Cipher(Input Shift Value):4

Enter Key for Transposition Cipher(Input String Value):Hack

Enter Key for OneTimePad Cipher(Input String Value):Crypto

The Decoded Message: Good and Evil are just a matter of
Perspective

DO YOU WISH TO CONTINUE? N

THANK YOU FOR USING TriPLe-SeCuriTy EnCODER-DeCODER.

>>>
```

Decoding: Here, an already encoded image has been provided by user along with the three symmetrical Cipher keys (which has been already used to encrypt the message for this image) for decryption of the message.

Provided Image with Cipher Text embedded: lucifer1.png
Provided Keys: 4, TANK, excelsior

```
Python 3.9.0 (tags/v3.9.0:9cf6752, Oct 5 2020, 15:23:07) [MSC
v.1927 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more
information.

>>>

==== RESTART: C:/Users/HP/Desktop/Sixth Semester
Project/ProjectCODE.py ====

Welcome To TriPLe SeCuriTy IMAGE EnCODER-DeCODER.


:: Welcome to Steganography ::

Choose:

1. Encode

2. Decode

2

Enter image name(with extension) : lucifer1.png

Enter Three Keys for Decryption as followed:

Enter Key for Substitution Cipher(Input Shift Value):4

Enter Key for Transposition Cipher(Input String Value):TANK
```

```
Enter Key for OneTimePad Cipher(Input String Value):excelsior
The Decoded Message: Calmness is the hallmark of those who are mighty.
DO YOU WISH TO CONTINUE? N
THANK YOU FOR USING TriPLe-SeCuriTy EnCODER-DeCODER.
>>>
```

Conclusions

From a technical point of view, Cryptography is the solution to many of the Security Challenges that are present in the Internet. The technology exists to solve most of the problems. However, there are several issues that have obstructed the widespread use of cryptography in the Internet. First of all, cryptography, as a science, faces a difficult problem. Most of the algorithms cannot be proven secure. For this reason, there is suspicion around many of the cryptographic algorithms. Another aspect is related to the intellectual property associated with the algorithms. Most algorithms are patented, and only some companies have licensed them for use.

Finally, Cryptography can be used to harm society. Governments are concerned that encryption will make law enforcement and national security goals more difficult to achieve. For example, terrorists could communicate information over the Internet using encryption that law enforcement agencies could not decrypt. Therefore, some governments, such as the U.S., have regulated the export of software containing encryption algorithms. This is a topic of debate, pitting governments against the right to free speech. For example, U.S. export regulations can prevent the publication of cryptographic research. In one court case, in March 1996, Phil Karn filed suite over whether he could export some source code from [SCHN96]. A District Court ruled that "export controls on encryption software are constitutional under the First Amendment" to the U.S. Constitution.

Steganography can protect data by hiding it but using it alone may not guarantee total protection. It is possible that by using a steganocryption technique, enemy detects presence of text message in the image file and then he/she may succeed in extracting information from the picture, which can be disastrous in real life situations.
This is same for plain encryption. In this case by seeing the meaningless appearing sequence of bits enemy can detect that

some illegal message is being sent (unless he/she is a fool), and we may land-up in a problematic situation. However, if one uses both methods, this will lead to 'security in depth'. The message should first be encoded using a strong encryption algorithm and then embedded into a carrier.

References

1. Cryptography and Network Security by Atul Kahate, Tata McGraw Hill
2. Cryptography and Network Security by Behrouz A Forouzan & Debdeep Mukhopadhyay, Tata McGraw Hill
3. https://www.geeksforgeeks.org/cryptography-and-its-types/
4. https://www.geeksforgeeks.org/image-based-steganography-using-python/
5. https://iopscience.iop.org/article/10.1088/1742-6596/1339/1/012061/pdf