

PROJECT REPORT ON

Image Security Using Encryption



GRAPHIC ERA

DEEMED TO BE UNIVERSITY

Submitted to :Department of Computer Science and Technology

Bachelor of Technology-CSE

Batch (2020-2024)

Submitted by

Name of the student:

Mr. Naman Manjkhola

Enrollment Number:

GE-202016861

Roll no:2016861

Under the Guidance of

Dr. Manoj Diwakar



ACKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude to entire faculty at Department of Computer Science and Information Technology, Graphic Era Deemed to be University; Dehradun who evaluated the project from time to time and gave me valuable suggestions as to how to improve the project.

I am grateful to **Dr. Manoj Diwakar** Graphic Era Deemed to be University, Dehradun; for his supervision, encouragement, inspiration and guidance. Working under him is being an enriched experience.

In all, I found congenial work environment in Graphic Era University, Dehradun and this project completion will mark a new beginning for me in the coming days.

I am highly indebted to Graphic Era University for providing me the required infrastructure and facilities to accomplish the given task.

Naman Manjkhola

Bachelor of technology-CSE

2020-2024

Graphic Era University

Problem Statement

Image security using encryption.

Development Environment

- PyCharm Community edition 2021.2.3

Language used

- Python(3.8.0-64bit)-libraries included are:
 1. opencv-python 4.5.5.62 for image data manipulation.
 2. tkinter module for graphical interface of application.
 3. numpy module for storing and manipulating image data.
 4. os module for working with image directories.

GOAL AND APPROACH

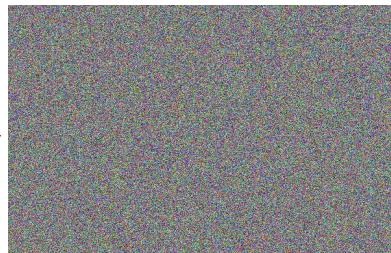
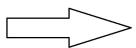
The goal of this project is to achieve a way to transmit some sensitve image from sender to receiver in a non-readable format in order safeguard from information leak or modification.

This can be achieved through image encryption. The application let's us create an encrypted file using chaotic XOR technique in which the sender encrypts the data using two distinct keys ranging from 1-255. First the image pixels are XORed with a set of disinct keys created using logistic map, then byte data of image is XORed with keys provided by the sender.

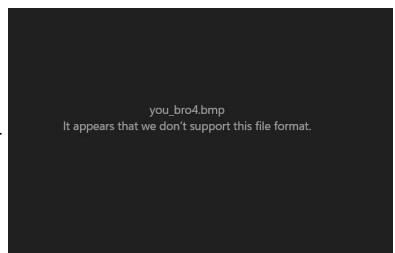
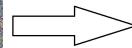
When the receivers receives the encrypted data, it is decrypted by providing same keys used by the sender to encrypt it, wrong set of keys would result in corrupted data.



original image



step1 :XORed pixels
step2 :XORed binary data x2



Encrypted image

Requirements

The system shall encrypt the given image to an unreadable format. This is done using chaotic encryption function.

The system shall decrypt the received encrypted image to a readable format. This is done using chaotic decryption function. The output image should be same as the original image.

Encrypted and Decrypted images will be in .bmp format so as to keep image data intact during encryption/decryption

Two keys are provided for encryption/ decryption process in range 1-255.

Performance Requirements

Decryption should not take more than 10-15 seconds.

If the decryption takes more than 10 seconds, then discard the message (because the message might have been corrupted during transmission) and ask sender to re-send it.

For smooth and efficient encryption, preferred image size is less than 5MB.

Encryption is done using two encryption keys. Decryption will happen only when same encryption keys are used at the receiver's side.

Application Usability

Python based GUI Tkinter is used for creating the interface which is user-friendly so someone with basic computer knowledge can easily operate using the interface.

Application Interface

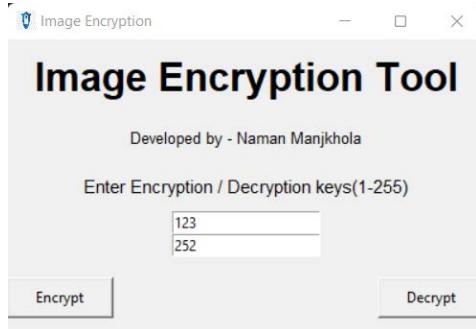


Code link

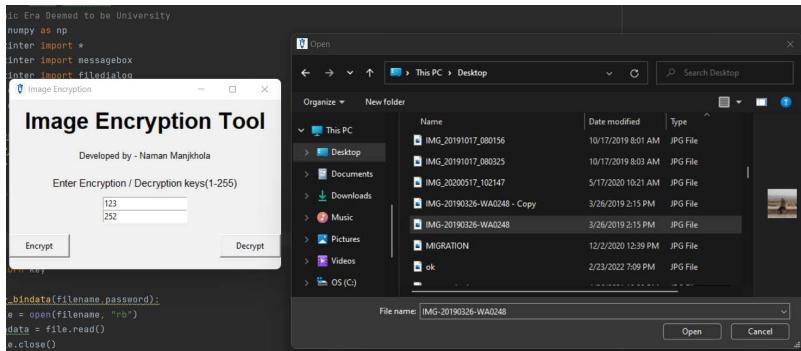
[Click here to see the code](#)

Implementation Screenshots

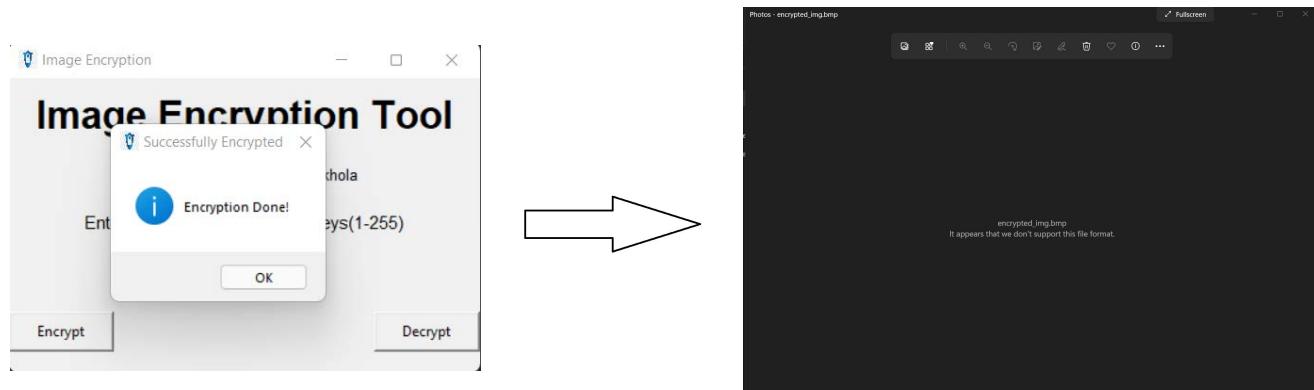
Encryption



step1 :Provide two keys
step2 :Click on Encrypt button



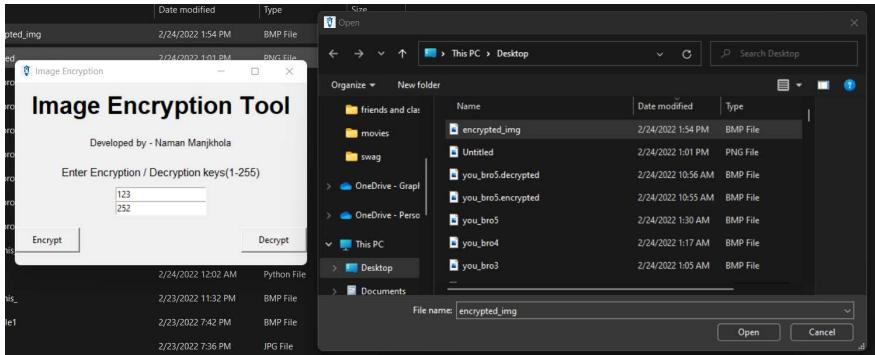
step3 :Select the image



Done!

Encrypted image

Decryption



step1 :Provide valid keys
step2 :Click on Encrypt
step3 :Select encrypted image



Decryption Done!

Conclusion

The encrypts and decrypts the image files accurately. This will help in resolving the problem of data theft and leaks of sensitive information. File obtained after encryption is safe and cannot be easily manipulated and recovered. The solution can be made more complex by applying different sets of algorithm and can be used for military/secrecy purposes.

References

GeeksForGeeks

python docs

python packages