

Task-02



“Pixel Manipulation for Image Encryption



Table of Contents

Introduction:

- Purpose of the tool
- Features

Conclusion

- Summary
- Acknowledgments

Getting Started

- Prerequisites
- Installation

Usage Instructions:

- Running the tool
- Input parameters

Tool Overview:

- Pixel Manipulation Techniques
- Encryption Process
- Decryption Process

Technical Details:

- Libraries Used
- Encryption/Decryption Key
- Image Processing

Sample Output

- Screenshots of tool execution

1. INTRODUCTION

Purpose of the Tool

- The Image Encryption Tool serves the purpose of providing a straightforward image encryption solution using pixel manipulation.
- It offers various pixel manipulation techniques to enhance image

Features

- Pixel manipulation for image encryption.
- Swapping pixel values and applying basic mathematical operations.
- Command-line interface for user interaction.



2. GETTING STARTED

Prerequisites

- Ensure the following libraries are installed:
- Pillow (PIL)
- NumPy
- tqdm
- colorama

Install the libraries using the provided script or manually using pip.

Installation

Run the following command to install the required libraries:

Kali:\$ `pip install Pillow numpy tqdm colorama`



3. USAGE INSTRUCTION

Running the Tool

Execute the tool using the following command:

```
Kali:$ python image_encryption_tool.py
```

Input Parameters

- Encryption/Decryption Choice: 'e' for encrypt, 'd' for decrypt.
- Image Path: Path to the target image file.
- Output Path: Path to save the result.
- Encryption/Decryption Key: Integer value for the encryption/decryption process.



4. TOOL OVERVIEW

Pixel Manipulation Techniques

The tool offers pixel manipulation techniques, including swapping pixel values and applying basic mathematical operations.

Encryption Process

The encryption process involves selected pixel manipulation techniques based on user input, enhancing image security.

Decryption Process

Decryption reverses the selected pixel manipulation, restoring the original image.



5. TECHNICAL DETAILS

Libraries Used

- Pillow (PIL): Image processing library.
- NumPy: Numerical computing library.
- tqdm: Progress bar library.
- colorama: Terminal color library for ASCII art.

Encryption/Decryption Key

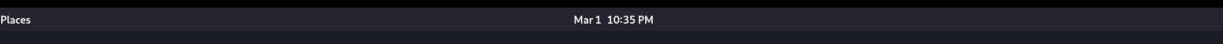
The key is a user-provided integer used to seed the random number generator for pixel manipulation.

Image Processing

The tool utilizes NumPy to convert images to arrays, apply pixel manipulations, and save the processed images.



6. SAMPLE OUTPUT



The screenshot shows a Kali Linux terminal window. The title bar reads "Kali Linux". The top bar indicates the date and time as "Mar 1 10:35 PM". The terminal prompt is "root@kali: ~/Intern/prodigy/task02-ImageManipulation". The user has entered the command "ls", and the output is displayed as follows:

```
(root@kali)-[~/Intern/prodigy/task02-ImageManipulation]
# ls
IMG-ENCRY-DECRY.py  Samples  builds  img1.jpeg  img2.jpeg
```

The prompt is now "(root@kali)-[~/Intern/prodigy/task02-ImageManipulation] #", with the cursor positioned after the hash symbol.

```
(root@kali)-[~/Intern/prodigy/task02-ImageManipulation]
# python3 IMG-ENCRY-DECRY.py
Do you want to (e)ncrypt or (d)ecrypt an image? e
root, enter the path of the image: img1.jpeg
Do you want to save the result in the same location as the original image? (y/n): y
Enter the encryption/decryption key: 14321

[[M/[ ]Y [ ]T]]N[/
[ ]K/[ ]V [ ]T [ ]K]

Encrypting
Progress: 100%|██████████████████████████████████████████████████████████████████████████| 5/5 [00:05<00:00, 1.00s/s]

Encryption complete.
Result saved as 'Encrypted_file.jpg'.
File saved at: Encrypted_file.jpg

Thank you for using my script Oxploited19

(root@kali)-[~/Intern/prodigy/task02-ImageManipulation]
#
```




```
Kali Linux
Applications Places
Mar 1 10:40 PM
root@kali: ~/Intern/prodigy/task02-ImageManipulation

(root@kali)-[~/Intern/prodigy/task02-ImageManipulation]
# python3 IMG-ENCRY-DECYR.py
Do you want to (e)ncrypt or (d)ecrypt an image? d
root, enter the path of the image: Encrypted_file.jpg
Do you want to save the result in the same location as the original image? (y/n): y
Enter the encryption/decryption key: 14321

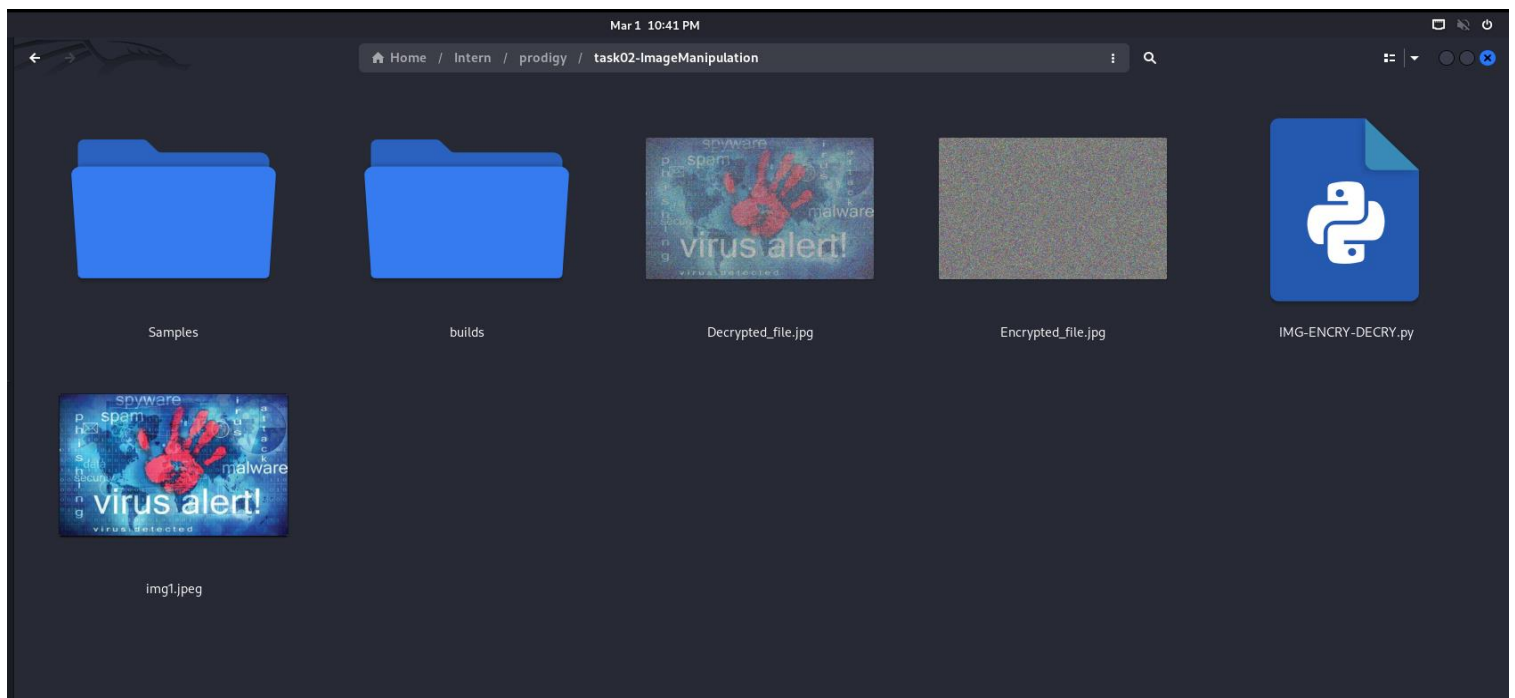
DECRYPTING

Decrypting
Progress: 100% | 5/5 [00:05<00:00, 1.00s/s]

Decryption complete.
Result saved as 'Decrypted_file.jpg'.
File saved at: Decrypted_file.jpg

Thank you for using my script Oxploited19

(root@kali)-[~/Intern/prodigy/task02-ImageManipulation]
#
```



7. CONCLUSION

Summary

The Image Encryption Tool provides a simple yet effective solution for image encryption using pixel manipulation. It offers various techniques for enhanced image security.

Acknowledgments

I thank the Prodigy Infotech for making me work onto my programming skills and achieve this task.

