

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
Belagavi-590018, Karnataka



**A
SYNOPSIS
ON**

“Image Steganography Tool”

*Submitted in partial fulfillment for the award of the degree in Bachelor of
Engineering in Computer Science & Engineering*

Submitted by

**SUVAM SHARMA
SAURABH SHARMA TIMILSINA
SHREYA KUMARI
RISHAV UPADHYAY**

**1BH20CS046
1BH20CS034
1BH20CS040
1BH20CS029**



BANGALORE TECHNOLOGICAL INSTITUTE

(ISO 9001:2015 Certified Institute)

Sarjapur Road, Kodathi, Bangalore-560035

Department of Computer Science & Engineering

Table of contents

1.	Introduction	1
2.	Problem Statement	1
3.	Objectives	2
4.	Methodology	2
	a) Requirement Identification	
	b) Algorithm Used	
5.	Expected Outcome	4
6.	References	6

1. Introduction

Image Steganography Tool is a project created as a tool for secret communication among people where data privacy is denied by the government or any form. It can be very useful in the places where cryptography in internet is not allowed to be used. In such places, if you want to have a secure communication, you can use this tool that will take a message from users, encrypt it first and hide it into the provided image and provides a new image with hidden data in it. The quality change in image, or the hidden messages won't be visible to any naked eye nor will the hidden messages be detected by any of the tools or technology that is used to filter the internet traffic.

This tool is based on python and has a CLI, which is super simple to use and to work upon by any people. This tool needs no any deep knowledge or understanding about any technology. It simply takes message, password to encrypt the message and image in which message is to be hidden.

2. Problem Statement

There are many countries in the world like Iran, Cuba and Sudan, where SSL (Secure Socket Layer), the standard internet encryption technology, is strictly prohibited to be used within their cyber space. So, in those countries, whatever you communicate among your friends or anyone, internet traffic is not encrypted. So attacks like Network Sniffing and Man in The Middle attacks are always possible. Even worse, your government or any person with resource can access your network and your personal raw unencrypted data, as it was never encrypted nor will it be.

So, if you want to share some secret messages, like your organizations server passwords, other credentials, or anything you want to share secretly will be always in risk of being broadcasted by hackers. This tool solves the problem as the message will be hidden in the image and there is no way to extract the message other than the image steganography tool. Even if the third person intercepts your traffic, they will just see the image and won't get your secret messages.

3. Objectives

- To provide a means of secret communication to those people who want to have a secret communication with anyone they want.
- To protect the privacy of the people in the internet where standard encryption technology like SSL cannot be used.

4. Methodology

a. Requirement Identification

i. Study of Existing System

Research upon existing system on the technology it was found that most of the systems are based on freemium approach and everyone cannot afford the tool, as they are not so cheap. Also they tend to hide the message in image in plain text without any encryption. This makes the message vulnerable, as analyst can study the pixel data of image and may uncover the message, in plain text. Here are some of the tools that I have studied about:

- **SteganPEG** is a lightweight software application built specifically for helping users hide sensitive data inside JPG images in only a few steps. It sports a clean and simple interface that offers only a few configuration settings to tinker with. The program gives you the possibility to embed photos into the working environment using the built-in browse function, so you cannot rely on “drag and drop” operations. SteganPEG allows you to encrypt data by setting up passwords, preview the picture in the primary panel, as well as insert multiple items to be hidden in the photos. What’s more, the utility is able to indicate how much space is occupied by the hidden files so you can calculate if there’s available space for adding more items. The photo, which embeds the sensitive data, can be exported to JPG file format, provided that you have specified the filename and saving directory.

- **OpenStego** can hide any data within an image file. It also does watermarking image files with an invisible signature. It can be used to detect unauthorized file copying. This tool has standard licensing on watermark technology and the image with message may be distinguished by any people.

- **Our Secret** is yet another image Steganography tool which allows hiding files, text, and message in photos. If these files are combined with Images, then the final image will be of more size. But this tool makes sure that the size of the final image is not abnormal.

The input file is preferably a JPEG file, maintaining small size of the image. Even audio files can be carried using image files with the help of Our Secret. Use the same tool to unhide the data.

- **RSteg** is yet another Steganography tool developed using Java. You need to have Java installed on your machine to run RSteg. Another striking advantage is its portable feature. Hence no need to install it, just run and the software windows pop up. Performing Steganography using RSteg is simple. You need an Image file, text to be encrypted and password to be set for decryption. The final output is stored as PNG. Plug the image into the same Steganography detection tool for decryption along with a password.

ii. Requirements Analysis

Software configuration

- Operating System: Linux, Unix, Mac
- Technology: Python, Bash
- Modules: PIL, OS, time, cryptography.fernet.Fernet, base64, hashlib

Hardware configuration

- Pentium IV Processor
- 2 GB RAM
- 120 GB HDD
- Monitor
- Input devices

b. Algorithm Used

In the existing system, BlowFish Algorithm and RSA Algorithm were used. BlowFish Algorithm has small block size, which is considered unsecured. The computational complexity of RSA Algorithm makes it a relatively less effective and resource heavy algorithm. RSA Algorithm is considered weak due to its vulnerability to collision attacks. The best algorithm for Image Steganography is Least Significant Bit(LSB) algorithm as it performs the embedding operation of message along with the image file where each pixel has a size of 3 bytes.

The algorithm used for the project is Red Channel Least Significant Bit Method for

Steganography (R-Channel LSB Method). This algorithm converts the user input text into binary form and hides the data bit by bit in the LSB of 8 bit color code or red channel from the each pixel of image provided.

Step by step algorithm:

Algorithm to hide message in image:

Step 1: Start

Step 2: Take text and image input from users

Step 3: Encode the text provided into binary form

Step 4: While encoded data is not hidden

For each pixel in image

Get red color code

Encode the color code into 8 bit binary

Replace the two bits from LSB of color code with two bit MSB of encoded text Update the pixel data

If encoded data is hidden

Encoded data is hidden

Step 5: Stop

Algorithm to extract message from image:

Step 1: Start

Step 2: Take the image with hidden message

Step 3: For each pixel in image

Get red color code

Encode the color code into 8 bit binary

Append the two bit from LSB of the encoded color code to a string

Step 4: Segment the copied data into seven bits each

Step 5: Convert the segmented bits it into corresponding ASCII character

Step 6: Stop

5. Expected Outcome

After the completion of the project, we are expecting the following outputs which will help to obtain our objective of developing the system. Also many drawbacks of existing system will be covered by our system. After completion of project, this system will be

able to:

- Provides a platform for secret communication.
- Encrypt the user message with their password in their local machine and hide the data in image.
- Extract the hidden messages and decrypt with the pre-shared key.
- Save the extracted text and give the path of saved text file to users.

References

1. Hide and Seek : An Introduction to Steganography
Available : <https://ieeexplore.ieee.org/document/1203220>
Authors : N. Provos, P. Honeyman
2. SteganPEG Steganography + JPEG
Available : <https://ieeexplore.ieee.org/document/6172093>
Authors : Velagalapalli Lokeswara Reddy, Arige Subramanyam, Pakanati Chenna Reddy
3. Implementation of Android Based Text to Image Steganography using 512-Bit Algorithm with LSB Technique
Available : <https://ieeexplore.ieee.org/document/9733441>
Authors : Nazmun Nahar, Md. Kawsher Ahmed, Tareq Miah, Shahriar Alam
4. Image Steganography using Blowfish Algorithm and Transmission via Apache Kafka
Available : <https://ieeexplore.ieee.org/document/9716292>
Authors : Shivam Illasariya, Parth Patel, Vatsal Patel, Swapnil Gharat
5. A combination of RSA Algorithm with Image Steganography to Ensure Enhanced Encryption
Available : <https://ieeexplore.ieee.org/document/10085371>
Authors : B. Karthikeyan, B. BharathKumar, G. Manikandan, R. Seethalakshmi