Project Report

# Steganography Tool with AES Encryption

# Introduction

With the rapid rise of digital communication, safeguarding private information has become increasingly important. Although traditional encryption is effective at protecting the contents of a message, the visibility of the encrypted data itself can invite attackers to attempt decryption. To address this limitation, steganography offers a way to disguise the existence of sensitive information by hiding it within ordinary-looking digital media.  
  
This project focuses on creating a Steganography Tool that can conceal either text or complete files inside images using the Least Significant Bit (LSB) technique. To strengthen security, it incorporates AES-GCM (Advanced Encryption Standard in Galois/Counter Mode) encryption, which not only encrypts but also verifies data integrity. The combined use of steganography and encryption ensures that the message remains both invisible and secure.  
  
The tool was built with a Tkinter-based Graphical User Interface (GUI) for desktop platforms and also deployed online for wider use. It enables anyone, regardless of technical expertise, to embed and retrieve information safely.

# Abstract

This project highlights the fusion of steganography and cryptography within a simple yet secure application. Steganography relies on the LSB method to embed secret information by adjusting the least significant bits of image pixels, producing imperceptible visual differences in the output image.  
  
AES-GCM encryption strengthens this approach by ensuring confidentiality (data can only be accessed with the correct key) and integrity (any alteration of the hidden information is detected).  
  
The application was implemented in Python 3 using Pillow for image manipulation, PyCryptodome for encryption, and Tkinter for GUI design. It supports embedding both short text and full files, with the convenience of drag-and-drop functionality. Testing confirmed that the tool works best with non-lossy image formats such as BMP and PNG. Additionally, an online demo version was deployed for demonstration:  
https://advanced-steganograp-f3g1.bolt.host

# Tools Used

Python 3 – Primary programming language for development.  
  
Pillow – Library used for image processing and pixel modification.  
  
PyCryptodome – Provides AES-GCM encryption and decryption algorithms.  
  
Tkinter – GUI framework for desktop application design.  
  
tkinterdnd2 – Enables drag-and-drop support.  
  
Bolt Host – Cloud platform for deploying the online tool.  
  
GitHub – Version control and collaborative code hosting platform.

# Project Development Process

1.Requirement Analysis  
  
Set the goal: allow users to securely hide text and files inside images.  
  
Identified the need for encryption in addition to steganography to enhance data security.  
  
2.Design Phase  
  
Designed workflow: Input → Optional Encryption → Embedding into Image → Output Image.  
  
Created two functionalities: Text Mode for short hidden messages and File Mode for embedding entire files.  
  
Outlined GUI features like image selection, file input, message fields, and passphrase entry.  
  
3.Implementation  
  
Developed LSB algorithms for embedding and extracting data.  
  
Added AES-GCM encryption and decryption for secure communication.  
  
Built the interface with Tkinter, including file dialogs and drag-and-drop.  
  
4.Testing  
  
Verified embedding and extraction across different image sizes.  
  
Tested AES encryption with correct and incorrect keys to ensure reliability.  
  
Confirmed best performance with PNG and BMP formats.  
  
5.Deployment  
  
Packaged the Python application as a desktop tool.  
  
Hosted an online demo version via Bolt Host for browser-based access.

# Conclusion

The developed Steganography Tool effectively integrates data hiding and strong encryption to create a secure method of communication. By embedding hidden text or files within an image and applying AES-GCM encryption, the tool ensures that sensitive information is both undetectable and inaccessible without authorization.