An Integrated Cryptography–Steganography Model for Confidential Data Transmission with Compression and Forensic Detection

Ms. Swapnali S. Shinde[1], Ms. Sakshi B. Wagh[2], Ms. Tejaswi P. Pasalkar[3], Dr. Rohan R. Kubde[4]
[1,2,3]*Dept. of E&TC, Sinhgad Institute of Technology and Science, Narhe, Pune*
[4]*Asst. Professor, Dept. of E&TC, Sinhgad Institute of Technology and Science, Narhe, Pune*

***Abstract*—Secure communication over open networks requires both confidentiality and concealment to prevent unauthorized access and detection. This paper presents a multi-layer secure communication framework that integrates cryptographic encryption, image steganography, and forensic analysis. The system employs AES-based encryption using the Fernet module, enhanced with password-based key derivation (PBKDF2) to strengthen key security. Prior to encryption, the plaintext message is compressed to improve embedding efficiency. The encrypted payload is embedded into digital images using an optimized Least Significant Bit (LSB) technique implemented with NumPy for high performance and minimal distortion. A user-friendly interface developed using Streamlit enables secure message transmission and retrieval. Additionally, a forensic analysis module is incorporated to detect hidden data using LSB visualization and histogram analysis. Experimental evaluation demonstrates high imperceptibility, accurate data recovery, and strong resistance to unauthorized access. The proposed system provides a practical and efficient solution for secure and covert digital communication.**

***Index Terms*—Steganography, Cryptography, AES, LSB, PBKDF2, Data Security, Steganalysis**

I. INTRODUCTION

With the rapid growth of digital communication, ensuring data security has become a critical requirement. Traditional encryption techniques provide confidentiality but fail to conceal the existence of sensitive information. Encrypted data often attracts attention, making it vulnerable to interception and analysis. Steganography addresses this limitation by embedding secret data within a cover medium such as an image, thereby hiding its existence. However, steganography alone cannot guarantee security if the hidden data is extracted. Therefore, combining cryptography with steganography provides a dual-layer defense mechanism.

This paper proposes an enhanced secure communication framework that integrates AES-based encryption, optimized LSB steganography, and forensic analysis. The system also incorporates password-based key derivation and data compression to improve security and efficiency. The objective is to design a robust, user-friendly, and scalable system capable of secure data transmission with minimal perceptual impact.

II. LITERATURE REVIEW

Image steganography has evolved significantly as researchers strive to improve data security, imperceptibility, and robustness. Sharma and Kumar [1] evaluated the traditional Least Significant Bit (LSB) substitution method, noting its high embedding capacity but vulnerability to statistical attacks. To enhance security and image quality, Chen and Liu [2] proposed a Pixel-Value Differencing (PVD) technique that adaptively embeds data based on local pixel variations, achieving better imperceptibility at the cost of reduced capacity.

In frequency-domain approaches, Khan and Ahmed [3] utilized the Discrete Cosine Transform (DCT) to embed information in mid-frequency coefficients, improving resistance to compression and filtering. Similarly, Reddy and Singh [6] implemented the Discrete Wavelet Transform (DWT) to achieve higher robustness and spatial-frequency localization, though with increased computational complexity.

Hybrid methods such as that of Patel and Gupta [4] combined edge detection with LSB matching, embedding data in complex regions to minimize detection probability. Lee and Kim [5] explored Spread Spectrum Image Steganography (SSIS), distributing hidden data across multiple frequencies for greater robustness but lower capacity.
Incorporating modern encryption strengthens steganographic security. The AES-GCM standard ensures confidentiality and integrity [7], while PBKDF2 provides secure key generation from user passwords [8].

III. METHODOLOGY

The proposed Steganography-Based Secure Communication System is designed as a multi-layer security framework that integrates cryptographic protection with data hiding techniques. The system operates in two primary phases — embedding (sender side) and extraction (receiver side) — while maintaining data confidentiality, integrity, and concealment.
The implementation has been enhanced with modern features such as password-based key derivation, data compression, optimized embedding algorithms, and a graphical user interface for improved usability.

Sender Side:
The sender-side workflow begins with the user providing a plaintext message and selecting a cover image (preferably lossless formats such as PNG to preserve pixel accuracy).

The system processes the input through multiple sequential stages:

1. Input Acquisition:
The user uploads a carrier image and enters the secret message via a web-based interface. An optional password can also be provided to enhance security through user-defined key generation.

2. Integrity Preparation:
A SHA-256 hash of the uploaded image is computed to create a unique fingerprint. This helps in verifying file authenticity and detecting any modifications during transmission.

3. Key Generation Mechanism:
The system supports two key generation strategies:

• Automatic Key Generation:
A secure random key is generated using the Fernet module.

• Password-Based Key Derivation:
If a password is provided, a strong encryption key is derived using PBKDF2 with SHA-256 hashing and multiple iterations, making it resistant to brute-force attacks.

4. Message Compression:
Before encryption, the plaintext message is compressed using a lossless compression algorithm. This reduces the data size and increases embedding efficiency, allowing more information to be stored within the image.

5. Encryption Process:
The compressed message is encrypted using AES-based Fernet encryption, which ensures both confidentiality and integrity through authenticated encryption mechanisms.

6. Data Formatting:
A structured payload is created by attaching a fixed-size header containing the length of the encrypted data. This enables accurate extraction at the receiver side.

7. LSB-Based Data Embedding:
The encrypted payload is embedded into the Least Significant Bits of image pixels. Unlike traditional implementations, this system uses optimized NumPy-based vectorized operations for efficient bit manipulation, significantly improving performance and reducing processing time.
The embedding process ensures that pixel modifications are minimal, thereby maintaining visual similarity between the original and stego-image.

8. Stego-Image Generation:
The modified pixel array is reconstructed into an image and saved in PNG format to avoid compression artifacts. The final output includes:
• Stego-image (containing hidden encrypted data)
• Encryption key (if auto-generated)

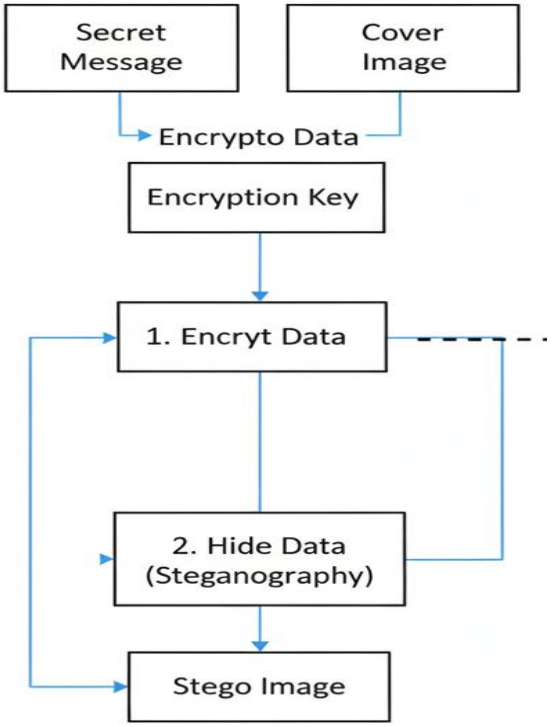

Fig.1: sender side process,

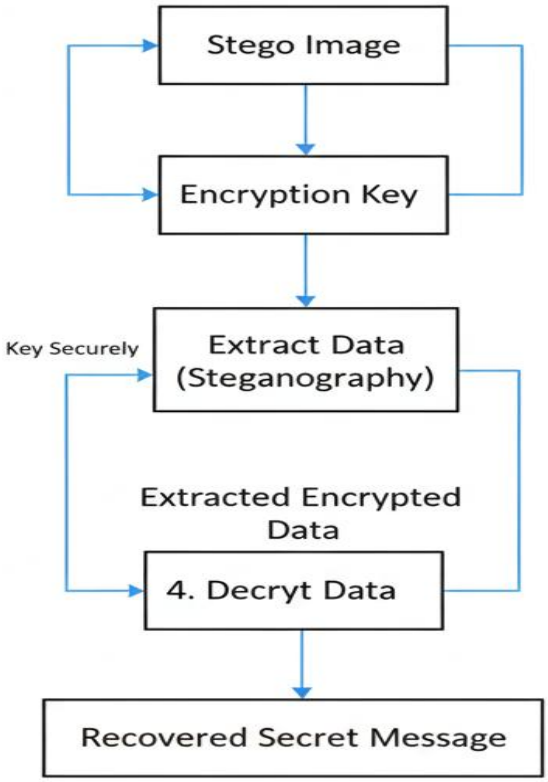

Fig.2: receiver side process,

The receiver-side workflow reverses the embedding process to recover the hidden message securely.

1. Input Acquisition:
The receiver uploads the stego-image and provides the encryption key or password through the interface.

2. Key Interpretation:
The system intelligently determines whether the input corresponds to:
- A direct encryption key, or
- A password requiring key derivation

This flexibility improves usability while maintaining security.

3. Data Extraction:
The system reads the Least Significant Bits from the image pixels and reconstructs the embedded bitstream. The header is first decoded to determine the exact length of the hidden data.

4. Decryption Process:
The extracted encrypted data is decrypted using the appropriate key. If the key is valid, the original compressed message is recovered.

5. Decompression:
The decrypted data is decompressed to retrieve the original plaintext message.

6. Integrity Verification:
If the image has been tampered with or an incorrect key is used, the decryption process fails, ensuring data authenticity and preventing unauthorized access.
The system is implemented in Python 3.x, using the Pillow (PIL) library for image handling and the Cryptography module for key generation and encryption. It operates via a simple command-line interface (CLI), ensuring cross-platform compatibility. The modular structure allows independent improvement of cryptographic and steganographic components. AES-based encryption ensures that even if embedded data is intercepted, it remains unreadable without the correct key.

IV. RESULTS AND DISCUSSION

The enhanced Steganography-Based Secure Communication System was successfully implemented using an interactive web-based interface developed with Streamlit. The system integrates multiple security layers, including AES-based

encryption (Fernet), password-based key derivation (PBKDF2), data compression (zlib), and optimized Least Significant Bit (LSB) steganography using NumPy for efficient processing.
During the encryption and embedding phase, the user uploads a carrier image and inputs a secret message through the graphical interface.

The system first generates a SHA-256 hash of the uploaded image to ensure file integrity tracking. Depending on user input, the encryption key is either:
- Automatically generated (random secure key), or
- Derived from a user-provided password using PBKDF2 with 100,000 iterations

The plaintext message is then compressed to reduce payload size and encrypted using AES (Fernet). The encrypted data is embedded into the image using an optimized LSB technique that leverages NumPy-based bitwise operations, significantly improving embedding efficiency compared to traditional pixel-wise methods.

The system successfully produces a stego-image visually indistinguishable from the original, confirming high imperceptibility. Additionally, the system ensures usability by providing a secure key display mechanism and download functionality.

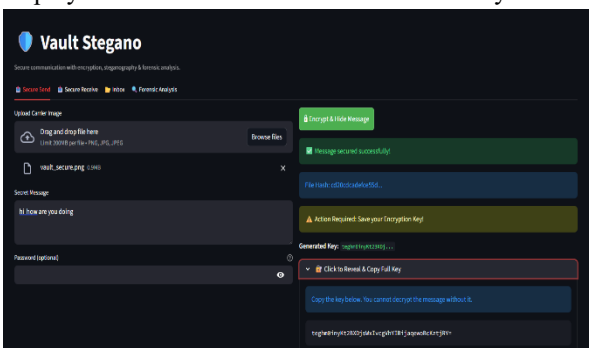

Fig.3: Sender Interface – Encryption and Embedding

During the extraction and decryption phase, the receiver uploads the stego-image and provides either the raw encryption key or password. The system intelligently determines whether the input is a direct key or requires derivation. The embedded data is extracted using LSB decoding, reconstructed using a header-based length mechanism, and decrypted to recover the original message.

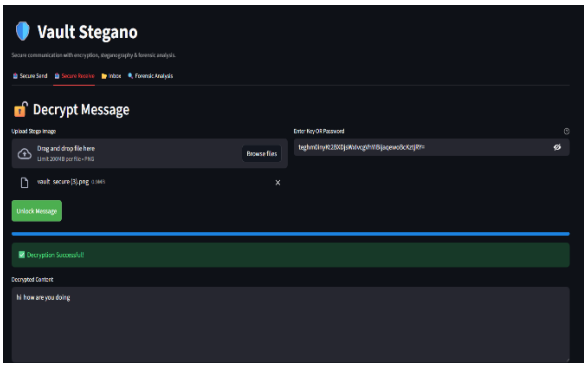

Fig.4: Receiver Interface – Decryption Output

The system demonstrates high reliability, as:
- Messages are recovered without loss
- Incorrect keys or tampered images result in decryption failure
- Compression improves embedding capacity

A key advancement in this implementation is the forensic analysis module, which evaluates images for hidden data using:
- LSB bit-plane visualization
- Histogram analysis (pixel intensity distribution)

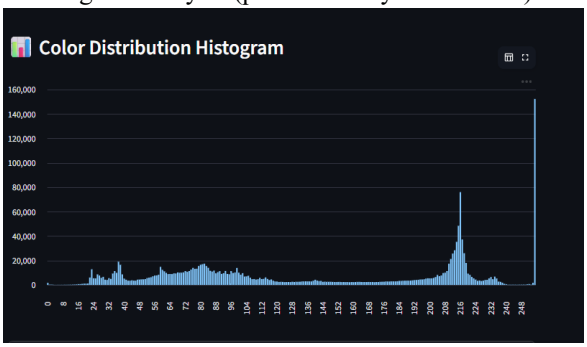

Fig.5: Histogram Analysis of Image

The results show detectable anomalies such as “twin peaks” in histograms and high-entropy noise patterns in LSB planes when data is embedded. This confirms that while the system ensures concealment, advanced analysis can still indicate the presence of hidden data.

V. CONCLUSION

The developed Steganography-Based Secure Communication System successfully demonstrates a robust and modern approach to secure data transmission by combining cryptography, steganography, and intelligent system design.

Unlike traditional implementations, this system enhances security through multiple layers, including AES encryption, password-based key derivation, and compressed payload embedding. The integration of an optimized NumPy-based LSB algorithm improves computational efficiency and scalability, making the system suitable for practical real-world applications.

The introduction of a user-friendly Streamlit interface transforms the system from a theoretical model into a deployable solution, enabling secure communication without requiring deep technical expertise. Furthermore, the inclusion of forensic analysis capabilities provides an additional dimension by allowing detection and evaluation of steganographic content.

Experimental results confirm that the system maintains high imperceptibility, accurate data recovery, and strong resistance to unauthorized access. Even if hidden data is detected, it remains protected due to encryption, ensuring a dual-layer defense mechanism.

In conclusion, the proposed system offers a secure, efficient, and scalable framework for covert communication, effectively addressing modern data security challenges.

Future Scope—

The system can be enhanced by incorporating password-based key derivation, pseudo-random pixel embedding, and support for other media formats such as audio and video to improve robustness and adaptability for broader security applications.

REFERENCES

[1] Sharma and R. Kumar, "An evaluation of LSB-based image steganography techniques," *International Journal of Computer Science and Network Security*, vol. 21, no. 5, pp. 110–115, 2021.

[2] J. Chen and W. Liu, "A robust steganography method based on pixel-value differencing (PVD)," Journal of Information Security and Applications, vol. 68, Art. no. 103234, 2022, doi: 10.1016/j.jisa.2022.103234.

[3] Z. Khan and F. Ahmed, "Secure data hiding in images using the discrete cosine transform," in Proceedings of the IEEE International Conference on Communications (ICC), 2020, pp. 1–6, doi: 10.1109/ICC40277.2020.9148868.

[4] S. Patel and M. Gupta, "Adaptive steganography using edge detection and LSB matching," Multimedia Tools and Applications, vol. 82, pp. 7845–7863, 2023, doi: 10.1007/s11042-022-13854-8.

[5] Y. Lee and H. Kim, "Spread spectrum image steganography (SSIS)," Journal of King Saud University – Computer and Information Sciences, vol. 31, no. 2, pp. 189–195, 2019, doi: 10.1016/j.jksuci.2018.11.002.

[6] P. Reddy and T. Singh, "A novel image steganography technique based on discrete wavelet transform," Procedia Computer Science, vol. 204, pp. 581–588, 2022, doi: 10.1016/j.procs.2022.08.071.

[7] National Institute of Standards and Technology, NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. Gaithersburg, MD, USA: National Institute of Standards and Technology, 2007.

[8] B. Kaliski, PBKDF2: Password-Based Key Derivation Function 2, RFC 2898, Sep. 2000. Available: RFC 2898 Specification