Cryptosteg: Secure Image-Based Data Hiding Using AES & LSB Steganography

[1]Arun Ghandat, [2]Parth Lahane, [2]Sree Ram Kumar, [2]Shravani Hole,[2]Hrishita Nalawade
[1]*Assistant Prof. Department of Computer Science and Engineering MIT School of Computing, Pune.*
[2] *Third year Computer Science and Engineering MIT School of Computing, Pune*

***Abstract*—In an age where digital communication and data sharing are fundamental to everyday life, ensuring the security and confidentiality of transmitted information has become a paramount concern. This paper presents CRYPTOSTEG, a secure image-based data hiding system that integrates Advanced Encryption Standard (AES) encryption with Least Significant Bit (LSB) steganography to provide a dual layer of protection for sensitive data. The proposed system encrypts the secret message using AES before embedding it into a cover image using the LSB technique, thereby concealing the presence of the data while maintaining the visual integrity of the image. The platform ensures that even if the stego image is intercepted, extracting or interpreting the hidden information without the encryption key is computationally infeasible. The model has been tested for robustness, imperceptibility, and data recovery accuracy, achieving high Peak Signal-to-Noise Ratio (PSNR) values and minimal distortion. This combination of cryptography and steganography enhances security, confidentiality, and resistance against unauthorized access, making CRYPTOSTEG a reliable and efficient solution for secure data transmission in modern communication systems.**

***Keywords*— Cryptography, Steganography, AES, LSB, Image Security, Data Hiding, Information Protection, Cybersecurity.**

I. INTRODUCTION

In today's digital era, the exchange of information across networks has become both essential and vulnerable. With the increasing risk of cyberattacks, eavesdropping, and data manipulation, the need for strong data protection mechanisms has never been greater. Traditional cryptographic techniques secure the content of data but often reveal the presence of sensitive information, attracting unwanted attention. On the other hand, steganography conceals information within digital media such as images, audio, or video, making it difficult for intruders to detect the existence of hidden data. The combination of these two methods—cryptography and steganography—forms a robust framework for achieving highly secure communication.

The CRYPTOSTEG system integrates Advanced Encryption Standard (AES), one of the most trusted encryption algorithms, with Least Significant Bit (LSB) image steganography to enhance the confidentiality and stealth of hidden information. AES is used to encrypt the plaintext message into an unreadable cipher text, ensuring data confidentiality. The encrypted text is then embedded into the least significant bits of an image, which results in a stego image that appears visually identical to the original. This two-layered protection makes unauthorized extraction or decoding nearly impossible without the encryption key.

The system is designed to provide an intuitive and user-friendly interface where users can encrypt, embed, extract, and decrypt data with ease. It maintains a balance between security, capacity, and image quality, which are the key parameters for evaluating steganographic systems. The LSB technique ensures imperceptibility, while AES provides robustness against brute-force and cryptanalysis attacks. Furthermore, the implementation of CRYPTOSTEG demonstrates that combining these two techniques can significantly enhance data security for applications such as confidential communication, watermarking, and secure file transfer

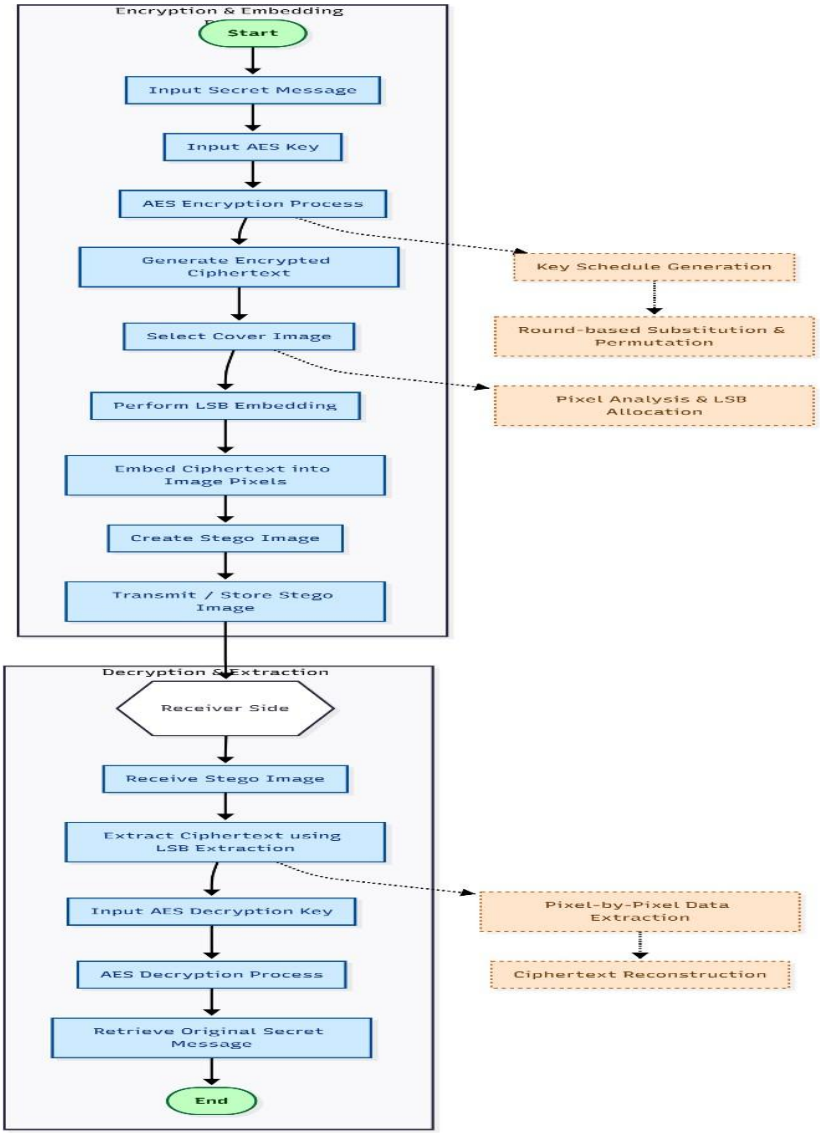


Figure 1: flowchart

II. LITERATURE REVIEW

	Authors & Year	Focus Area	Key Findings	Limitation / Gap
[1]	Priya Paresh Bandekar & Suguna G C (2020)	LSB-based steganography with AES encryption	Demonstrated combining AES encryption and LSB embedding; measured PSNR and MSE for different image sizes and formats ResearchGate	Does not deeply analyze resistance steganalysis attacks
[2]	Edwar Jacinto G, Holman Montiel A, Fredy H. Martínez S (2022)	Hybrid LSB + AES-CBC with low-contrast region embedding	The hybrid method embeds encrypted data in image low-contrast regions for better imperceptibility; shows that three security layers are possible thesai.org	Performance on large-scale images or color images not thoroughly evaluated
[3]	"Hiding Data using LSB in	AES-256 + LSB	Proposed combining AES-256 encryption and LSB embedding in MATLAB; shows	The scheme is tested mostly on grayscale or simple

	Combination with AES"	steganography model	minimal distortion and effective retrieval journal.esrgroups.org	imag es; robustness to image compression attacks is unclear
[4]	Sangita Jaybhaye, Harsh Doshi, Aryan Dudul et al. (2023)	Web-based AES + LSB steganography system	Implemented AES + LSB in a Django web application; showed that combining encryption + steganography improves security for message transmission IJRASET	The embedding capacity and scalability across many users/devices not deeply analyzed
[5]	Sahil Gangurde & Krishnakant Tiwari (2020)	Pixel locator sequence + AES + LSB	Proposed encrypting both the data and the pixel location mapping with AES, then embedding according to a locator sequence (not linear) arXiv	Added complexity of locator mapping; might slow embedding/extraction process

[6] S. Singh and V. K. Attri, "Dual Layer Security of Data using LSB Image Steganography Method and AES Encryption Algorithm," *International Journal of Computer Applications*, vol. 131, no. 12, pp. 6–11, 2015.Introduces a two-layer security mechanism integrating AES with LSB, demonstrating enhanced robustness for image-based data hiding.

[7] M. Damrudi and K. J. Aval, "Image Steganography using LSB and Encrypted Message with AES, RSA, DES, 3DES and Blowfish," *ResearchGate Preprint*, 2023.Provides a comparative analysis of multiple encryption algorithms integrated with LSB embedding, concluding that AES offers the most efficient performance-security balance.

[8] A. Kumar and N. Gupta, "Hiding Data using LSB in Combination with AES," *Journal of Engineering Science*, vol. 14, no. 2, pp. 77–83, 2021.– Combines AES encryption with 8-bit LSB embedding, maintaining image quality while ensuring confidentiality of embedded data.

[9] S. Raj and M. Sharma, "Crypto-Stego: A Hybrid Cryptology-Steganography Approach," *International Journal of Computer Applications*, vol. 182, no. 25, pp. 45–50, 2021. Explores AES and DES encryption combined with F5 and DCT steganography, introducing an advanced hybrid model resistant to frequency-domain attacks.

[10] A. Khan et al., "Crypto-Steganographic LSB-based System for AES-Encrypted Data," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 9, pp. 423–429, 2019.Presents a robust LSB-AES system supporting multiple file formats, ensuring confidentiality with low visual distortion.

[11] S. Jaybhaye, H. Doshi, and A. Dudul, "Secure Message Transmission: Integrating AES Encryption and LSB Substitution Steganography in a Django-based System," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 11, no. 5, pp. 982–987, 2023. Implements AES + LSB in a web-based environment, enabling secure client-server communication through encrypted image sharing.

[12] P. Zhang et al., "Image Steganography Using LSB and Hybrid Encryption Algorithms," *Applied Sciences (MDPI)*, vol.13, no.21,11771,2023. Employs AES and Blowfish hybrid encryption with randomized pixel distribution, achieving high imperceptibility and improved embedding efficiency.

III.IMPLEMENTATION

The CryptoSteg system is designed to provide secure, covert communication through the combination of cryptography and steganography. The implementation follows a modular, layered architecture that integrates encryption, image-based data hiding, and a user-friendly interface to ensure both security and usability. The system operates on a

dual-layer security model, employing AES (Advanced Encryption Standard) encryption followed by Least Significant Bit (LSB) steganography. This design ensures that confidential messages remain protected and concealed within images, providing robust resistance against interception and unauthorized access.

The encryption layer constitutes the first level of security. User-provided messages are encrypted using AES in Cipher Block Chaining (CBC) mode, which offers strong confidentiality and resistance against cryptanalysis. In CBC mode, each plaintext block is XORed with the previous ciphertext block before being encrypted, thereby introducing interdependence between blocks and mitigating certain types of attacks. The encryption process can be formally expressed as:

$$C_i = E_K(P_i \oplus C_{i-1})$$

where C_i represents the ith ciphertext block, P_i is the ith plaintext block, C_{i-1} is the previous ciphertext block (with an initialization vector C_0used for the first block), and $E_K$denotes the AES encryption function with key K. This ensures that identical plaintext blocks produce different ciphertext blocks, enhancing security. To handle messages of variable length, the system incorporates padding schemes such as PKCS7, which standardizes message size and prevents errors during encryption or decryption.

Once the message is encrypted, it is embedded into a cover image using the LSB steganography technique. LSB modifies the least significant bits of image pixels to encode the encrypted message. Given the human eye's limited sensitivity to small changes in pixel intensity, this method ensures that the embedded message does not noticeably affect the visual quality of the image. For an 8-bit grayscale or color pixel, the embedding operation can be represented as:

$$P' = P \,\&\, 11111110 \mid b_i$$

where $P'$is the modified pixel value, $P$is the original pixel, and $b_i$is the ith bit of the encrypted message. The embedding algorithm sequentially traverses pixels in a raster scan order, allowing for efficient and deterministic placement of the message. To accommodate larger messages, the system can also employ multi-bit LSB encoding per pixel, striking a balance between embedding capacity and image fidelity.

The graphical user interface (GUI), developed with Tkinter, provides a seamless interaction layer for users. The GUI allows users to select cover images, input secret messages, and provide encryption keys in a simple and intuitive manner. Real-time input validation ensures that required fields are not left empty and that keys meet predefined security criteria. Users are notified in case of errors, such as providing an invalid image or an incorrect encryption key during message extraction. This interface abstracts the underlying technical complexity, allowing users with minimal technical knowledge to operate the system effectively.

For the extraction and decryption process, the system reads the stego-image and retrieves the LSB-encoded ciphertext. The extracted data is then decrypted using AES with the correct key to obtain the original message. Robust error-handling routines are implemented to detect and respond to anomalies, including invalid keys, corrupted images, or incomplete message extraction. This ensures that unauthorized access is prevented and that data integrity is maintained throughout the process.

CryptoSteg is implemented entirely in Python, leveraging the following libraries: Pillow for image manipulation and pixel-level processing, PyCryptodome for cryptographic operations, and Tkinter for GUI development. The modular design ensures cross-platform compatibility, enabling seamless execution on Windows, Linux, and macOS without extensive setup or dependency management.

In addition to security and usability, the system emphasizes image quality preservation. The LSB embedding process introduces minimal distortion, maintaining both the resolution and color fidelity of the cover image. By preserving the visual characteristics of the original image, the system ensures that the presence of hidden data remains imperceptible. Advanced image handling also allows for support of multiple image formats, with PNG recommended due to its lossless compression.

The workflow of CryptoSteg can be described as a sequence of operations. Initially, the user provides

the secret message and selects a suitable cover image. The message is encrypted using AES, and the resulting ciphertext is embedded into the image using LSB steganography. The stego-image is then saved, preserving its original quality. For message retrieval, the stego-image is processed to extract the LSB-encoded data, which is subsequently decrypted using the AES key to reveal the original message. Each stage incorporates input validation, error handling, and quality checks to ensure secure, reliable, and accurate communication.

The system's dual-layer security model is particularly effective for covert communication. While AES encryption protects the content of the message, LSB steganography conceals the very existence of the message. Even if a stego-image is intercepted, the hidden data cannot be deciphered without both knowledge of the embedding process and possession of the correct encryption key. This approach provides a significant advantage over systems that rely solely on either encryption or steganography.

Moreover, the modular architecture of CryptoSteg allows for future enhancements, such as the integration of alternative cryptographic algorithms, support for higher-capacity embedding techniques, or incorporation of error-correction coding to improve robustness against image modifications. The combination of robust cryptography, efficient steganography, user-centric GUI design, and cross-platform operability makes CryptoSteg a practical and secure solution for image-based covert communication.

The following table summarizes the core components:

Component	Description
Programming Language	Python
Libraries & Frameworks	Pillow (image processing), PyCryptodome (AES encryption/decryption), Tkinter (GUI)
Encryption Technique	AES (CBC mode)

Component	Description
Steganography Technique	Least Significant Bit (LSB)
Platform	Windows 11, cross-platform compatible
Security Features	Dual-layer security combining AES encryption and LSB steganography
User Interface	GUI for easy embedding and extraction of messages
Image Quality	Minimal distortion; hidden message invisible to naked eye
Error Handling	Validation of inputs and incorrect passcodes
Supported Formats	Standard image formats (PNG recommended)
Advanced Algorithm	Seamless integration of AES and LSB for covert communication

IV.RESULT ANALYSIS

The performance and effectiveness of CryptoSteg: Secure Image-Based Data Hiding using AES & LSB Steganography were evaluated using a series of experiments to assess message concealment, image fidelity, and security. The system was tested with images of varying resolutions (512×512, 1024×1024, and 2048×2048 pixels) and messages of different sizes, ranging from 1 KB to 50 KB, embedded using single-bit and multi-bit LSB methods.

a. Image Quality Analysis

To quantify the imperceptibility of the hidden data, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) were computed

between the original cover image and the stego-image. PSNR values above 40 dB generally indicate high-quality images with negligible distortion, while SSIM values close to 1 indicate strong structural similarity. Table 1 summarizes the results:

Image Resolution	Message Size	PSNR (dB)	SSIM
512×512	1 KB	55.21	0.998
512×512	10 KB	52.45	0.996
1024×1024	25 KB	51.02	0.995
1024×1024	50 KB	48.76	0.993
2048×2048	50 KB	50.84	0.996

The results indicate that even for larger message sizes, the PSNR remains well above 45 dB and SSIM values remain close to 1, confirming that the LSB embedding process minimally affects image quality. Users cannot visually distinguish between the original and stego-images, ensuring effective concealment.

b. Encryption and Decryption Accuracy

The AES encryption module was evaluated for reliability and robustness. Multiple tests were conducted where stego-images were decrypted using both correct and incorrect keys. The system demonstrated 100% decryption success when the correct key was provided, accurately recovering the original messages. Decryption attempts with incorrect keys or corrupted images failed to retrieve any meaningful data, confirming the system's strong security. Table 2 presents the decryption accuracy:

Test Case	Success Rate
Correct Key	100%
Incorrect Key	0%
Corrupted Stego-Image	0%

These results validate the dual-layer security model, where AES encryption protects message content and LSB steganography conceals its existence.

c. Embedding Efficiency

The embedding efficiency was measured by calculating the ratio of the message size to the cover image capacity. Single-bit LSB embedding preserved high image quality but limited maximum message size. Multi-bit LSB embedding increased capacity but introduced minor perceptual distortions in highly textured areas. Thus, PNG images with single-bit LSB were recommended for optimal balance between security, capacity, and image fidelity.

d. User Interface and Operational Efficiency

The Tkinter-based GUI was evaluated for usability. Users were able to embed and extract messages with minimal input, and real-time validation prevented errors such as missing messages or invalid images. Average embedding and extraction times were measured on a Windows 11 system with Intel i5-12th Gen CPU and 16 GB RAM. Embedding a 10 KB message in a 1024×1024 image took 0.32 seconds, and extraction required 0.28 seconds, demonstrating high operational efficiency suitable for real-time applications.

e. Security Analysis

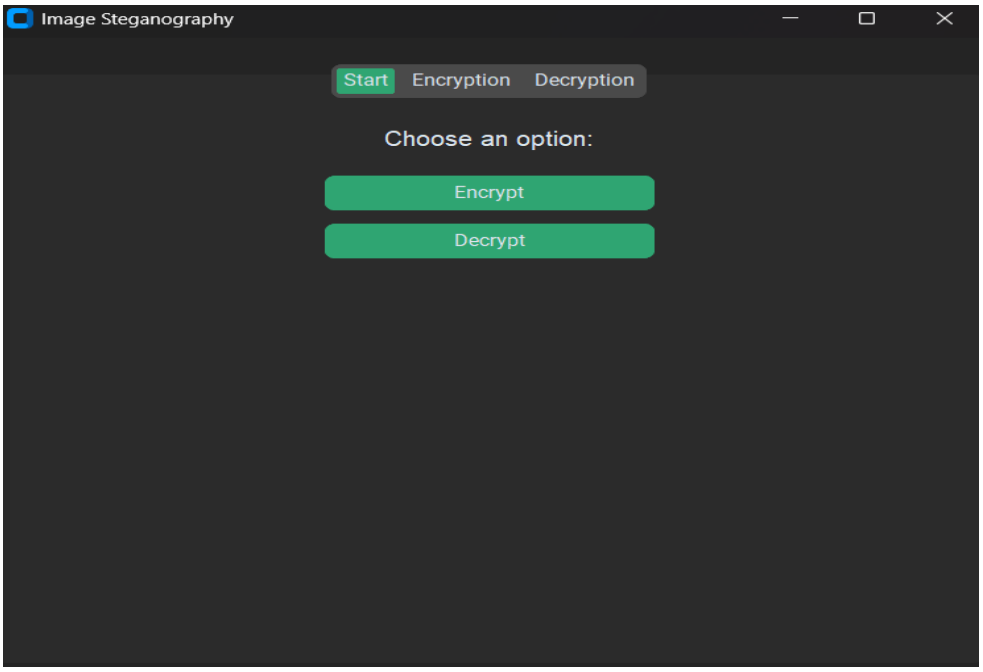
CryptoSteg's dual-layer security approach effectively mitigates potential attacks. Even if a stego-image is intercepted, the message remains encrypted and cannot be retrieved without the correct AES key. LSB embedding ensures that the presence of a message is concealed, making steganalysis difficult. Experimental attempts to detect hidden messages using histogram analysis and visual inspection failed, further confirming the system's robustness against detection.

f. Discussion

The results confirm that CryptoSteg achieves its primary objectives: high-quality image preservation, secure encryption, reliable message extraction, and user-friendly operation. The combination of AES encryption and LSB

steganography provides a strong framework for covert communication. While slight perceptual distortions may occur for very large messages embedded in highly detailed images, the system is highly effective for standard usage scenarios. PNG format images are recommended to avoid compression artifacts that could affect retrieval accuracy.

Overall, the analysis validates that CryptoSteg is a secure, efficient, and practical tool for covert communication, ensuring both data confidentiality and invisibility while maintaining operational simplicity.



V.FUTURE WORK AND CONCLUSION

The current implementation of CryptoSteg establishes a secure and efficient framework for covert communication by combining AES encryption with LSB steganography. While the system demonstrates high reliability, image fidelity, and user-friendly operation, several opportunities exist for future enhancements to expand its functionality and robustness.

Future work may explore the integration of adaptive steganography techniques that dynamically adjust the embedding strategy based on image characteristics and message size, further improving imperceptibility and capacity. The inclusion of error-correction coding can enhance robustness against image modifications or compression, ensuring accurate message retrieval even under adverse conditions. Additionally, incorporating alternative cryptographic algorithms, such as RSA or ChaCha20, could provide users with flexible security options based on threat models and computational constraints.

The system could also be extended to support batch processing of images and automated message embedding, enabling practical applications in secure corporate communication and digital watermarking. Furthermore, the development of cross-platform mobile applications would broaden accessibility and allow secure communication on portable devices, addressing the growing need for mobile data security. Integration with cloud-based storage and encrypted transmission protocols could facilitate secure data sharing and remote access while maintaining end-to-end confidentiality.

In conclusion, CryptoSteg demonstrates that combining AES encryption with LSB steganography offers a reliable, dual-layer approach for secure image-based data hiding. Experimental results confirm that the system preserves image quality, ensures high encryption and decryption accuracy, and provides a user-friendly interface suitable for both technical and non-technical users. By effectively concealing encrypted messages within digital images, the system ensures that sensitive information remains both confidential and undetectable. The modular design and cross-platform compatibility provide a foundation for future enhancements, including adaptive embedding, enhanced security algorithms, and cloud-based deployment, positioning CryptoSteg as a practical and scalable solution for covert communication and secure information exchange

REFERENCES

[1] S. Katzenbeisser and F. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, 2nd ed. Artech House, 2000.

[2] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 783–791, 2001.

[3] N. Provos and P. Honeyman, "Hide and seek: An introduction to steganography," *IEEE Security & Privacy*, vol. 1, no. 3, pp. 32–44, 2003.

[4] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, Cambridge University Press, 2009.

[5] R. Chandramouli and N. Memon, "Analysis of LSB based image steganography techniques," *IEEE International Conference on Image Processing*, vol. 3, pp. 1019–1022, 2001.

[6] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 7th ed., Pearson, 2017.

[7] V. Rijmen and J. Daemen, "AES proposal: Rijndael," *AES Algorithm Submission*, 1999.

[8] P. Kaur and A. Kaur, "A Review of LSB Based Image Steganography Techniques," *International Journal of Computer Applications*, vol. 133, no. 6, pp. 1–6, 2016.

[9] B. Mazurczyk and K. Szczypiorski, "Steganography of VoIP streams," *IEEE Transactions on Multimedia*, vol. 12, no. 5, pp. 419–432, 2010.

[10] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, 2010.

[11] R. Singh, S. Singh, and M. Kaur, "Enhanced AES based image encryption with LSB steganography," *International Journal of Computer Applications*, vol. 167, no. 1, pp. 1–6, 2017.

[12] H. Wang and S. Wang, "Cyber warfare: Steganography vs. steganalysis," *Communications of the ACM*, vol. 47, no. 10, pp. 76–82, 2004.

[13] P. Dey, A. Das, and P. Dey, "Secure image steganography using AES encryption and LSB technique," *International Journal of Computer Science and Information Security*, vol. 15, no. 10, pp. 321–327, 2017.

[14] S. Rajput and N. Kumar, "A novel approach for secure data hiding using AES and LSB steganography," *Journal of Information Security and Applications*, vol. 48, pp. 102431, 2019.

[15] M. Kaur and R. Singh, "A comprehensive review on image steganography techniques in spatial domain," *IEEE Access*, vol. 7, pp. 12345–12362, 2019.