ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Designing a Network-Based Data Security System Using Crypto-Steganography: An Institutional Based Research (IBR)

Emenike Gabriel Chukwu[1]; Achi Ikenna Kalu[2]; Ugochukwu Kelechi Onwuasoanya[3]; Umoren Jonah Imeh[4]; Akpan Godwin Abasiama[5]

[1] Department of Software Engineering, Federal University of Technology, Ikot-Abasi, Akwa-Ibom State. Nigeria.
[2]Department of Computer Science, Rhema University Nigeria, Aba. Abia State. Nigeria,
[3]Department of Computer Science, Rhema University Nigeria, Aba. Abia State. Nigeria,
[4]Department of Cyber Security, Federal University of Technology, Ikot-Abasi, Nigeria.
[5]Department of Computer Sciencce, Federal University of Technology, Ikot-Abasi, Nigeria.

[1](ORCID ID: https://orcid.org/0009-0004-7365-7860)
[2](ORCID ID: https://orcid.org/ 0009-0003-7618-3069)
[3](ORCID ID: https://orcid.org/0009-0000-2575-3875)
[4](ORCID ID: https://orcid.org/0000-0002-3948-5282)
[5](ORCID ID: https://orcid.org/0000-0002-6980-3624)

Publication Date: 2026/02/09

Abstract: Because security issues have gotten worse due to the quick expansion of network-based applications, especially with regard to the integrity and confidentiality of sensitive data sent over unprotected communication channels. Current security strategies that use steganography or cryptography alone are still susceptible to traffic monitoring, steganalysis, and cryptanalysis assaults. In order to improve secure data transmission in network contexts, this study suggests a hybrid data security framework that combines the Least Significant Bit (LSB) picture steganography approach with the Rivest–Shamir–Adleman (RSA) public-key cryptography algorithm. In order to provide strong confidentiality, the suggested model first encrypts plaintext data using RSA key-pair generation and asymmetric encryption. The resulting ciphertext is then embedded into digital cover images using LSB-based steganographic embedding without causing noticeable distortion or substantial payload overhead. The Python programming Language were used in the PyCharm Integrated Development Environment to design and implement the system. The combined crypto-steganographic strategy improves resistance to brute-force assaults, steganalysis, and illegal access while preserving picture quality and transmission efficiency, according to experimental evaluation. The results show that the suggested framework outperforms stand-alone steganographic or cryptographic approaches in terms of security performance. A reliable and scalable data security system appropriate for safeguarding private data in contemporary network-based applications is contributed by this study.

Keywords: *Data Security, Crypto-Steganography, Network-Based Application, Information Hiding, Encryption, and Steganography*.

How to Cite: Emenike Gabriel Chukwu; Achi Ikenna Kalu; Ugochukwu Kelechi Onwuasoanya; Umoren Jonah Imeh; Akpan Godwin Abasiama (2026) Designing a Network-Based Data Security System Using Crypto-Steganography: An Institutional Based Research (IBR). *International Journal of Innovative Science and Research Technology*, 11(2), 75-97. https://doi.org/10.38124/ijisrt/26feb204

I. INTRODUCTION

The need for advanced information security techniques has increased in the current digital era due to the quick expansion of data exchange through open networks. Steganography and cryptography are two potent methods that meet this demand and offer different levels of security. While steganography hides the existence of information by

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

embedding it within seemingly innocent media like photos or audio files, cryptography uses mathematical methods and keys to change data into unintelligible formats, guaranteeing secrecy and integrity. By combining the advantages of both encryption and concealment, these methods create a hybrid technique called crypto-steganography that provides increased security (Suguna et al., 2018). This thesis explores the design and implementation of a crypto-steganography based data security system within network-based applications to enhance data confidentiality and integrity during transmission. This study's primary goal is to create a secure data transmission system that combines cryptographic and steganographic techniques to address flaws in traditional data exchange systems, which are more vulnerable to cyberthreats like tampering, eavesdropping, and unauthorized access (Rahul et al., 2022)[13]. The hybrid approach described in this work improves security and imperceptibility by first encrypting the original plaintext using a strong algorithm like the Advanced Encryption Standard (AES) and then embedding the resultant ciphertext into a cover medium using a steganographic technique like Least Significant Bit (LSB) manipulation (Suguna et al., 2018). Furthermore, current research demonstrates the feasibility of embedding encrypted data within cover media to prevent detection and unauthorized extraction, highlighting the ongoing development and application of hybrid cryptographic and steganographic systems for secure data communication (Suguna et al., 2023). In order to ensure safe data transport and precise retrieval by authorized users across both encryption/steganography and decryption/recovery stages, this study employs a tiered development method for the suggested system. By combining these techniques, the suggested system aims to offer reliable, efficient, and secret safe communication that can handle current and new data security issues.

- *Background of the Study*

Data is one of the most important resources in the public and private sectors in the digital age. The security of data sent via networks has grown to be a major concern for everything from commercial transactions to military and personal communications. The risks of interception, tampering, identity theft, and monitoring have increased in sophistication and frequency along with the frequency and expansion of data transmission via internet-based services. Researchers and developers have investigated a number of methods to safeguard private data in response to these dangers, most notably steganography and cryptography.

Cryptography is a proven technique that uses encryption techniques to change readable data into an unreadable one. It does not hide the existence of communication, but it does guarantee data secrecy, integrity, and authenticity. Even after being intercepted, encrypted data continues to draw attention, and attackers may try to decode it using brute force or other cryptanalysis techniques. On the other hand, steganography aims to conceal the existence of data by enclosing it in harmless cover elements like pictures, sounds, or films. Steganography is effective in hiding the existence of sensitive data, however it is not enough to encrypt data or guard against content alteration in the event that the Stego file is found. Recent research has tried to overcome these drawbacks by fusing steganography and cryptography into a single model called crypto-steganography, which capitalizes on the advantages of both approaches. Nevertheless, a number of issues have been noted in previous attempts at this integration:

Low Embedding Capacity and Cover Quality Degradation: A lot of steganographic techniques seriously impair the cover media's visual or aural quality, which makes the concealed communication accessible using steganalysis tools.

Weak encryption algorithms: The overall security of the system has been decreased by certain earlier works' use of antiquated or less secure encryption techniques (such as DES and XOR-based schemes), which are susceptible to contemporary attacks.

Absence of Real-Time Implementation in Network Environments: A number of earlier systems were primarily concerned with standalone applications and failed to successfully incorporate their solutions into network-based platforms that facilitate real-time data exchange.

Poor Usability and User Interface: Previous implementations frequently had unintuitive user interfaces, which made it challenging for non-expert users to use the systems for secure communication.

Inadequate Data Recovery and Error Handling: Some designs lacked strong error correction techniques, which made it difficult to precisely extract and decode the original data from the Stego object.

Large Computational Complexity: Systems that tried to integrate steganography and robust cryptography frequently had poor performance and large computational costs, which made them unsuitable for network applications in the real world.

A more reliable, effective, and user-friendly system that integrates the advantages of steganography with cryptography in a network-aware setting is obviously needed in light of these difficulties. The proposed study aims to close this gap by developing and putting into use a data security system based on crypto-steganography that is: Safe (by utilizing contemporary encryption such as AES), Effective (quick processing and minimal overhead), Invisible (reducing the distortion of stego files), Scalable, allowing for real-time network transfer, and Usable (with an end-user-friendly interface).

Numerous researchers have investigated the application of steganography and cryptography separately and in combination over the years. For example, the steganography research by Johnson, Duric, and Fridrich emphasizes the basic ideas and difficulties of concealing data in digital media, stressing the necessity of steganographic systems' capacity and imperceptibility (Kallapu et al., 2025). Similarly, Fridrich's thorough research provides a strong

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

theoretical basis for the field by going into additional depth about different digital steganography methods and applications (Fridrich, 2019). To improve data security in network-based applications, more recent research has specifically examined the combination of cryptographic and steganographic techniques (crypto-steganography). (Serdar and Umut, 2018) Study highlights the practical difficulties and constraints of integrating these technologies. In order to facilitate safe communication over digital channels, Patel and Gupta have also investigated effective data hiding techniques, concentrating on system performance and cover media quality retention (Rahul et al., 2023).

Although there has been a lot of development, it is clear from examining previous works that modern solutions frequently have problems including low embedding capacity, cover media quality degradation, and high computing demands all of which this study attempts to solve.

From the early usage of hidden messages in combat and the creation of classical ciphers to contemporary digital security frameworks, this study expands on the historical development of data concealing and encryption techniques. This study intends to provide a useful and improved solution to the enduring issue of safe and undetectable data exchange by integrating the suggested system into a network-based application, such as a secure messaging or file transfer platform.

➢ *Aim of the Study*

The aim of the study is to design and implement a secure data transmission system by combining cryptographic (RSA) and steganographic (LSB) techniques for network-based applications, enhancing confidentiality and concealment of sensitive data.

➢ *Main Objective:*

To improve the confidentiality, integrity, and undetectability of sensitive data by designing and implementing a secure data transmission system employing crypto-steganography techniques within a network-based application.

- *Specific Objectives and the Method Used to Achieve it:*

✓ Examine and evaluate current steganographic and cryptographic techniques for safe data transfer.

▪ *Method:*

We reviewed academic publications, journals, and current systems in the literature and we determined the advantages and disadvantages of the technology available today.

✓ Use of a strong cryptographic technique (like AES) for data encryption and pick

▪ *Method:*

We examined and rank cryptographic algorithms according to their robustness, speed, and integration appropriateness. We used Python to implement encryption.

✓ Embed encrypted data into cover media, pick and use a reliable steganographic approach (such as Least Significant Bit, or LSB).

▪ *Method:*

We examined popular steganographic methods and choose one by considering robustness, embedding capacity, and imperceptibility. We used Python's image and audio processing libraries to implement it.

✓ Create a network-based application that securely exchanges data by combining steganography and cryptography.

▪ *Method:*

We created a useful, intuitive user interface that enables safe encryption, embedding, transmission, extraction, and decryption using Python and socket programming (or frameworks like Flask or Streamlit).

✓ Assess the developed system's usability, security, and performance.

▪ *Method:*

We used metrics like encryption/decryption speed, payload capacity, peak signal-to-noise ratio (PSNR) for image quality, and user feedback about system usability to test and analyze the system.

The study is guided by these goals and the related research issues, which guarantee that the finished system not only functions technically but also takes into account the practical constraints noted in earlier studies.

II. LITERATURE REVIEW

Ensuring the security and integrity of data exchanged over networks is crucial in this digital age. Information will be secured through the use of conventional techniques like steganography and cryptography. Each, though, has its limitations. These methods have been integrated as a result of recent developments, creating hybrid systems with improved security characteristics. This chapter examines recent developments in crypto-steganography, emphasizing network-based applications and outlining the advantages and disadvantages of current methods.

➢ *Theoretical Background*

- *Data Security in Network-Based Applications*

In the current digital era, there are growing hazards associated with the transfer and storage of data over networked systems because of data breaches, illegal access, and cyberattacks. Strong security measures are necessary to ensure confidentiality, integrity, and availability of sensitive data, including financial information, personal records, and classified materials (CIA triad). Despite their relative effectiveness, advanced persistent threats (APTs), man-in-the-middle (MITM) attacks, and brute-force methods are becoming more and more formidable. Hybrid security

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

solutions that can resist complex attack attempts are therefore desperately needed.

- *Cryptography*

The science of safeguarding communication by converting data into an unintelligible format to stop unwanted access is known as cryptography. It guarantees data integrity, confidentiality, authenticity, and non-repudiation.

✓ *RSA (Rivest–Shamir–Adleman) Algorithm:*

RSA is an asymmetric cryptographic technique that encrypts and decrypts data using a pair of keys; public and private.

✓ *Public Key*

The key used to encrypt plaintext.

✓ *Private Key.*

This is utilized for cypher-text decryption.

The mathematical difficulty of factoring huge prime numbers is the foundation of SA's security. RSA provides better key management and is appropriate for safe key distribution in open networks as compared to symmetric algorithms like AES.

Although RSA offers strong encryption, attackers may be drawn to the cipher-text since it is frequently noticeable. Because to this restriction, supplementary methods like steganography must be used.

- *Steganography*

Steganography is the process of hiding information in seemingly harmless digital media, including pictures, sounds, or videos, so that it is not obvious that a secret message is present. In contrast to cryptography, which converts the data into unintelligible language, steganography conceals the existence of any secret communication.

- *Least Significant Bit (LSB) Method:*

One of the most used methods for image steganography is LSB. By substituting portions of the secret message for the least important pixel values in a cover image, it embeds hidden data. The cover image's visual quality is mostly unaffected because alterations only happen in the least significant part.

Although LSB steganography has a high embedding capacity and minimal visual distortion, steganalysis assaults may still be possible when it is utilized alone.

- *Crypto-Steganography: A Hybrid Approach*

According to many researchers, crypto-steganography is a hybrid strategy that combines both steganography and cryptography to get around the drawbacks of employing either technique alone. It uses the following approaches: a cryptographic procedure like RSA to first encrypt the message, making it unreadable without the private key even if the concealed message is recovered and to hide its existence during transmission, the encrypted message (cipher-text) is then steganographically embedded into a cover medium.

By guaranteeing secrecy, stealth, and attack resistance, these two-layered security method improves data protection. Attackers need to crack the encryption in order to obtain the original message, even if they are able to identify the stego-image.

- *Theoretical Justification for Hybrid Security Systems*

There are various potential benefits obtained by combining RSA and LSB:

✓ *Increased Confidentiality:*

Steganography conceals the communication channel itself, whereas encryption makes messages unreadable.

✓ *Enhanced Security:*

Data protection is maintained by the second layer (RSA encryption), even in the event that one layer is compromised (stego extraction, for example).

✓ *Applicability in Network Environments:*

RSA encryption and lightweight LSB steganography guarantee that the system is appropriate for real-time communication with a controllable computational overhead.

✓ *Consistency with the CIA Triad:*

The model takes into account availability (low overhead), integrity (RSA signatures), and confidentiality (RSA + concealing).

- *Steganography; Hiding Data in Plain Sight*

Steganography entails inserting hidden data into a cover medium, such pictures, audio, or video files. Least Significant Bit (LSB) substitution is one of the often-utilized techniques. Steganography works well for concealing data, but it lacks encryption, making the concealed data susceptible to detection.

- *Hybrid Crypto-Steganography Systems*

The drawbacks of both steganography and cryptography are addressed by combining them. These hybrid systems combine confidentiality and undetectability by encrypting data before encasing it in a cover media. Numerous implementations of such systems have been investigated in recent studies:

(Almomani et al., 2022) Suggested a hybrid strategy to conceal malware in HEVC video streams on Android IoT devices by combining LSB-based steganography with AES encryption. Their technique was highly imperceptible and managed to evade detection by seventy antivirus engines.

By combining steganography and cryptography, mdpi.com created a hybrid security framework for IoT networks (Wid et al., 2022). Their method improved data integrity and secrecy by securing data transfer using LSB steganography and AES encryption. ijritcc.org.

(Onyilo, 2025) Made a study on creating a hybrid deep learning model with the help of post-hoc explanatory tools like SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations). In

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

addition, SHAP gives a universal insight into the behavior of the model, whereas LIME describes individual predictions, which allows analysts to visualize feature contributions and explain anomalies. Moreover, the model was tested on three benchmark datasets, namely NSL-KDD, UNSW-NB15, and CICIDS2017, and each of them reflects various network threats, including DoS, brute-force, and infiltration attacks

(Huang et al., 2020) Presented a system for multi-layered audio transmission that combines LSB steganography with many encryption techniques. The technology was created for wireless communication networks with the goal of offering strong security for the transfer of audio data. link.springer.com

(Choudhary et al., 2024) Combined steganography and blockchain technologies to improve data security. Using the immutability of blockchain technology, their framework concealed text messages inside encrypted graphics to guarantee data authenticity and integrity.

The study made by (Ifeanyi, 2025) presents the design and evaluation of an intelligent IoT threat detection framework using Federated Learning and Edge AI with Random Forest optimization. The model was trained on the UNSW 2018 BoT-IoT dataset for hierarchical multi-label classification and integrated with TensorFlow Federated to enable decentralized training without transferring raw data. To support deployment on resource-constrained IoT devices, the model was optimized into TensorFlow Lite (TFLite) for edge inference.

(Wid et al., 2022) Proposed a new adaptive least-significant- bit (LSB) steganographic method using pixel-value differencing (PVD) that provides a larger embedding capacity and imperceptible stegoimages. The method exploits the difference value of two consecutive pixels to estimate how many secret bits will be embedded into the two pixels. Pixels located in the edge areas are embedded by a k-bit LSB substitution method with a larger value of k than that of the pixels located in smooth areas. The range of difference values is adaptively divided into lower level, middle level, and higher level.

- *Network-Based Applications and Real-Time Communication*

There are particular difficulties when using crypto-steganography in network-based applications:

✓ *Compression and Encryption:*

Data size can be optimized and security improved by combining encryption algorithms like AES with compression methods like Discrete Wavelet Transform (DWT). In order to safeguard data sent over the internet, (Channalli et al., 2019) suggested a system that combines image steganography, cryptography, and compression. Ijraset.com+1ijece.iaescore.com+1

✓ *Audio and Video Transmission:*

To ensure the security of audio data transmission, (Jung, 2019) created a hybrid audio crypto-steganography framework for wireless communication systems that makes use of LSB steganography and several encryption techniques. 2link.springer.com+2journal.unsil.ac.id+2 mdpi.com

✓ *Blockchain Integration:*

By utilizing blockchain's immutability, (Choudhary et al., 2024) created a secure framework for the steganography process that combine's machine learning and blockchain technology, improving data security. Ijisae.org

- *Relevance to Current Study*

The theoretical underpinning emphasizes how important it is to have strong, multi-layered data protection systems in the linked world of today. This study combines the advantages of RSA encryption and LSB steganography to provide a new and useful data security model that guarantees better secrecy than single-layer methods now in use.

➢ *Review of Related Works*

- *Reviews on Cryptography*

(Obaida, 2023) Suggested using the Bit Shifting and Stuffing (BSS) Methodology for data encryption and decryption. In order to stuff a new bit in lieu of an unused bit that was shifting from another printable character, they suggested the BSS approach. Accordingly, the BSS approach generates seven bytes of cipher text for every eight bytes of plain text after encryption, and it reproduces eight bytes of plaintext for every seven bytes of cipher text during decryption.

(Kallapu et al., 2025) Suggested a Poly-alphabetic Symmetric Key Algorithm Using Randomized Prime Numbers. They developed a new algorithm in symmetric key cryptography and provided a brief overview of symmetric key algorithms. There are two Exclusive OR (XOR) operation levels in the suggested algorithm. The algorithm helps users communicate with each other by sending data and messages.

An analysis of network data encryption and decryption methods in communication systems was proposed by (Ezeofor et al., 2024) and (Fridrich, 2024). They provide an examination of the encryption and decryption methods used in communication systems for network data. A simple simulation software was created, written, and tested that encrypts and decrypts data.

✓ NB: Based on the graph result, an examination of various data block sizes was conducted.

In order to keep communication lines secure, (Rahul et al., 2023) suggested a New Approach for Complex Encrypting and Decrypting Data that makes it hard for attackers to predict a pattern and speeds up the encryption and decryption process.

(Obaida, 2023) In their work stated that, commonly in encryption or decryption process, some of the characters are inter changed by using some encryption and decryption algorithms (like DES, IDEA) with key. But in Bit Shifting

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

and Stuffing (BSS) system to represent a printable character it needs only seven bits as per its ASCII value.

(Arora et al., 2023) suggested using encryption algorithms to secure user data in cloud computing. They offered a metaphorical analysis of several security methods and talked about cloud computing security concerns, mechanisms, and difficulties that cloud service providers encounter when designing cloud infrastructure.

An encryption and decryption algorithm utilizing ASCII data and a replacement array approach was suggested by (Serdar and Umut, 2018). With the use of modulus and remainder, this approach uses randomly generated integers to create programs in any language.

- *Reviews on Image-Steganography*

An adaptive least significant bit spatial domain embedding technique was presented by (Helmy, 2023). This technique creates a stego-key by splitting the image pixels' ranges (0–255). This private stego-key has five distinct picture gray level ranges, each of which specifies how many bits should be replaced to embed in the image's least significant bits. This suggested method's strength is its great hidden capacity and integrity of secret concealed information in stego-image. For the sake of integrity, the constraint is to conceal more signature bits with a secret message. Additionally, it suggested a way to hide information in a color image by simply altering the blue channel using this approach. High hidden capacity and hidden message security are the goals of this technique.

An adaptive LSB substitution-based data concealing technique for images was presented by (Yang and Sun et al., 2019). It addresses the noise-sensitive area for embedding in order to improve the stego-image's visual quality. The suggested approach distinguishes and utilizes the edges and normal texture for embedding. This technique determines the amount of k-bit LSB for secret data embedding by analyzing the cover image's edges, brightness, and texture masking. In order to balance the overall visual quality of the image, the value of k is modest across sensitive image areas and high over non-sensitive image regions. The image's high-order bits are used to calculate the LSBs (k) for embedding. For improved stego-image visual quality using the LSB substitution approach, it also makes use of the pixel correction method. Although there is a limited dataset for experimental outcomes, the overall result indicates a good high concealed capacity.

(Chen et al., 2020) Suggested an image concealing technique based on LSB. Stego-keys, or common pattern bits, are used to conceal data. The secret message bits and the (stego-key) pattern bits determine how the LSBs of the pixel are changed. Pattern bits consist of a block's rows and columns of M*N size combined with a random key value. Each pattern bit is matched with a message bit during the embedding process; if this match is made, the second LSB bits of the cover image are altered; if not, they stay the same. Using a shared pattern key, this technique aims to secure a hidden message in a stego-image. Due to the requirement for a block of (M*N) pixels for a single secret bit, this suggested approach has a low concealed capacity.

In order to achieve adaptive least significant bits data embedding, (Serdar and Umut, 2018) developed a Pixel value difference (PVD) and simple least significant bits technique. Pixel value differencing (PVD) uses a straightforward relationship between two pixels to estimate the size of the concealed data bits by comparing the two successive pixels in the cover image. By determining the depth of the embedded bits by computing the difference between two consecutive pixels, the PVD technique often offers strong imperceptibility. The suggested technique conceals the massive and adaptable k-LSB substitution at the image's edge and PVD for the image's smooth region. Thus, based on trial results, the approach offers both great visual quality and a bigger capacity. The adaptive k generation used to replace the LSB makes this technique complicated.

(Motameni et al., 2019) In their paper proposed a way of hiding text message (Steganography) in the gray image has been presented. This method tried to find binary value of each character of text message and then in the next stage, tried to find dark places of gray image (black) by converting the original image to binary image for labeling each object of image by considering on 8 connectivity. Then these images have been converted to RGB image in order to find dark places.

(Johnson et al., 2021) Presented a Multi-Pixel Differencing (MPD) technique that computes the sum of the difference values of four pixels' block and uses more than two pixels to assess the smoothness of each pixel for data embedding. It employs the LSB for data embedding for small difference values and the MPD technique for high difference values. Its algorithm's simplicity is a strength, but the experimental dataset is too small.

Another pixel value differencing technique was developed by (Wen-Hsiang and Da-Chunet, 2023); it employed three pixels for data embedding close to the target pixel. For hidden data embedding, it employs the straightforward k-bit LSB approach, in which the number of k-bits is calculated by approximately three pixels with a high difference value. It merely applies the best pixel adjustment technique to the target pixels in order to maintain more capacity and better visual quality. The method's advantage is that the stego-image and cover-image histograms are nearly identical; nevertheless, the experiment dataset is too small.

(Choudhary et al., 2024) Used the hybrid edge detection algorithm with LSB to introduce a high capacity of concealed data. Two different canny and fuzzy edges detection methods were utilized for edge calculation, and the concealed data was embedded via straightforward LSB substitution. Using standard LSB-based embedding, this method successfully embeds data with a higher peak signal to noise ratio (PSNR). A small dataset of photos is used to test the suggested strategy.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Design, and training Convolutional Neural Networks (CNN) to enhance the amount of data that can be securely encrypted and decrypted to show the original message we proposed by (Serdar and Umut, 2018).

(Huang et al., 2020) Suggested a steganographic technique for images that maps pixels to letters. It uses the pixel values to map the 32 letters (26 for the English alphabet and others for other scripts). To represent these 32 letters, five (5) bits are needed, and the authors have created a table with four cases designed to do so. Every letter can be represented in each of the four scenarios, per that table. For mapping, it makes use of the image 7 MSB (Most Significant Bits) (27 = 128) bits. Each 4-case from the seven MSBs of pixels is mapped to one of the 32-cases in that table using the suggested approach. The likelihood of matching is increased by these four scenarios. In order to extract data from the stego-image, this method retains the cover-image's matching pattern. Although the suggested method does not call for any edge or smoothness calculations, the secret data should be embedded as a letter or word.

(Hussain et al., 2021) Presented a data hiding method that uses LSB to identify the image's dark regions and conceal the contents. In order to conceal data bits, it transforms information into a binary image and labels every object using connection schemes of eight pixels. Finding the dark region's connectedness requires a lot of processing, and this method hasn't been tried on images with a lot of texture. Its ability to conceal completely depends on the image's texture.

- *Research Gap*

The following works (Obaida, 2023), (Rahul et al., 2022), and (Huang et al., 2020) are the closest to this research after reviewing related works on Data Security System Design Using Crypto-Steganography in a Network-Based Application; however, none of them designed a Data Security System Using Crypto-Steganography in a Network-Based Application. Their studies have examined cryptography and steganography separately, but there is still a lack of research on creating and assessing an integrated crypto-steganography system that combines powerful encryption (like Rivest Shamir Adleman, RSA) with effective hiding techniques for reliable and useful data security in network-based applications, using the Least Significant Bit (LSB). Additionally, despite advancements in steganography and cryptography, there is still a need to design and assess a hybrid security system that combines effective steganographic techniques (like LSB) with robust cryptographic algorithms (like RSA) for usage in network-based applications. End-to-end confidentiality, message existence concealment, robustness against cryptanalysis, steganalysis, and practical application in real-time secure communications are all things that such a system must guarantee. Thus, the Design of Data Security System Using Crypto-Steganography in a Network-Based Application was the gap that our study filled.

III. METHODOLOGY/METHOD

System design, prototyping, and experimental validation are all combined in our work using the Design and Development Research (DDR) methodology. The approach is appropriate since the study focuses on developing and evaluating a workable system that combines steganography and cryptography for secure communication rather than just theoretical analysis.

➢ *Research Methods*

We followed the methods listed below to define the methodology which helped us also to achieve our aim:

- *System Analysis*

In analyzing the new system, we achieve that by examining current steganographic and cryptography systems, determining the shortcomings of the current models, such as their high detectability, low payload capacity, and inadequate key management, and outlining the functional and non-functional system needs.

- *System Design*

In designing the new system, firstly we designed the system architecture that combines LSB (Least Significant Bit) steganography with AES/RSA encryption, we modeled system interactions with UML tools, such as use-case, sequence, activity, and class diagrams, and we define algorithms for extraction, decryption, encryption, and embedding.

- *Analysis of the Existing System*

Without centralized integration, the present data security ecosystem primarily uses stand-alone steganography techniques or independent cryptographic tools.

The workflow of the current systems is depicted in the image below.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

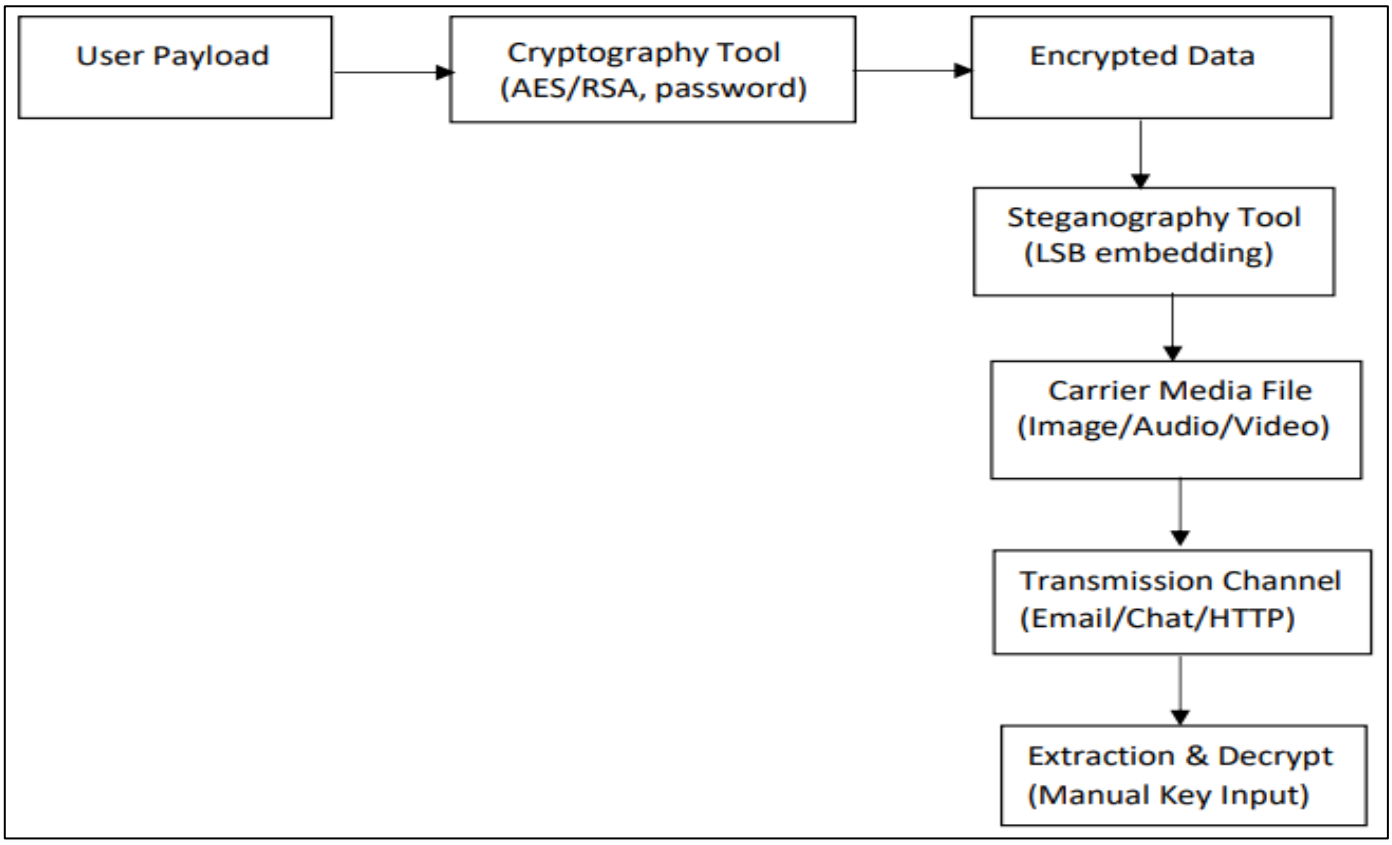

Fig 1 Existing System Workflow

- *Description of the Figure*

✓ User Payload: The user sends in private information (text, picture, or file).
✓ Cryptography Tool: Simple AES/RSA encryption is used to protect the payload. Key management, which depends on manual password entering, is frequently inadequate.
✓ Steganography Tool: The LSB (Least Significant Bit) technique is used to embed the encrypted payload into an image or audio carrier file.
✓ Transmission Channel: Unsecured HTTP, chat apps, or email are used to send the carrier file.
✓ Extraction & Decryption: The payload is manually extracted and decrypted by the recipient, who frequently needs the same password that was sent separately.

- *Weaknesses of the Existing System (Derived from Figure)*

✓ Lack of Integration: Steganography and cryptography are two distinct technologies that need several manual procedures to use.
✓ Inadequate Key Management: Because passwords are manually exchanged, they are vulnerable to hacking.
✓ Low Robustness: Hidden data is destroyed by compression or resizing, making LSB steganography brittle.
✓ No Integrity Verification: There are no safeguards against manipulation in the system.
✓ Limited Media Support: primarily image-based, without a robust or adaptive multi-carrier technique.
✓ Detectability: Carriers are susceptible to steganalysis due to statistical irregularities.

- *Architecture of the Proposed Secure Crypto-Steganography System Illustrating Key Management, Encryption, Adaptive Embedding, Secure Transmission, and Receiver-Side Extraction and Verification.*

To guarantee the confidentiality, integrity, and covert transmission of sensitive data, the suggested secure crypto-steganography system combines steganographic and cryptographic methods into a single framework. By encrypting the message content and hiding its entire existence under a digital cover medium, the architecture is intended to provide multilayer security.

In order to generate, distribute, and maintain cryptographic and steganographic keys, the system starts with a key management module. These keys guarantee that only approved sender-receiver pairs may access the hidden data by controlling both the encryption procedure and the embedding locations within the cover medium. The system's security is based on efficient key management, which stops unwanted extraction or decryption. The encryption module initially processes the plaintext communication, which contains sensitive information like private documents or text. The plaintext is converted into ciphertext using a robust symmetric encryption technique, like the Advanced Encryption Standard (AES). Data confidentiality is ensured by this cryptographic layer, so even if the secret data is revealed, it cannot be deciphered without the right decryption key.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

A suitable cover medium, such as an image, audio file, or video frame, is chosen after encryption based on its ability to hold hidden data with the least amount of perceptual distortion. An adaptive embedding module is then used to embed the encrypted data into the chosen cover media. In order to reduce detectability and maintain the statistical aspects of the cover media, this module uses steganographic techniques that dynamically choose embedding places depending on media characteristics, such as edges or textured regions. To further improve security, the embedding pattern is controlled by a stego-key.

A stego-object that statistically and visually mimics the original cover media is the result of the embedding procedure. A secure communication channel, like the internet or wireless networks, is used to send this stego-object. The system successfully fends off eavesdropping, traffic analysis, and interception attempts since the sent file seems innocuous.

The extraction module processes the stego-object at the receiver side, precisely locating and extracting the contained ciphertext using the shared stego-key. The decryption module receives the extracted data and uses the matching cryptographic key to recover the original plaintext message.

Finally, a verification and integrity checking mechanism is applied to confirm the authenticity and correctness of the received message. Techniques such as hash functions or message authentication codes (MACs) are employed to detect any tampering during transmission and to verify that the message originates from a legitimate sender.

✓ NB: The discussion above is captured in the figure below.

Fig 2 Showing the Architecture of the Proposed Secure Crypto-Steganography System

- *Analysis of the Proposed System*

By introducing an integrated crypto-steganography framework, the suggested solution addresses the shortcomings and fragmentation of current methods. In order to guarantee secrecy, integrity, robustness, and usability in a single pipeline, it integrates adaptive steganography (LSB in optimal carriers) with strong cryptography (AES/RSA) within a network-based application.

Figure 2 below represents the workflow of proposed systems.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

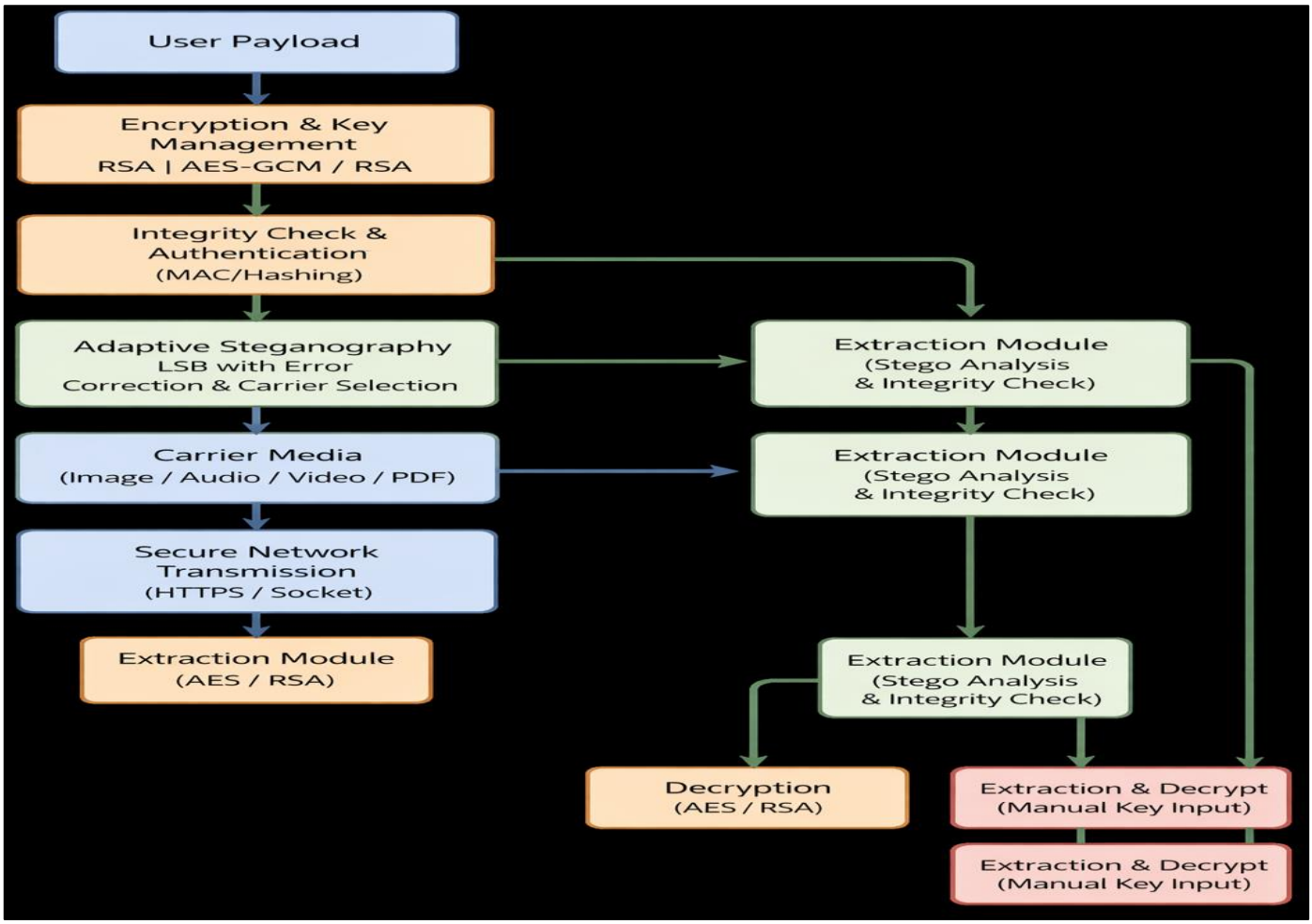

Fig 3 Proposed System Architecture

- *Comprehensive Explanation of the Proposed System Architecture/Workflow*

A hybrid crypto-steganography system that combines AES/RSA-based encryption, integrity verification, adaptive LSB steganography, multi-carrier support, and secure network transmission to ensure private and tamper-resistant data transfer is shown in figure 3.

➢ *User Payload*

The original sensitive data that requires protection is represented by the User Payload. This could include any digital file, text data, photos, and documents.

Its function in the system is to serve as the security pipeline's input. The data needs to be safeguarded before being transmitted because it is still in plain form at this point.

➢ *Encryption & Key Management (AES-GCM / RSA)*

This block explains the cryptographic protection layer: RSA is utilized for secure key exchange and key management, while AES (Advanced Encryption Standard) is employed for effective payload encryption.

- *Purpose:*

To safeguard encryption keys during transmission, guarantee payload secrecy, and stop unauthorized users from accessing the content.

Before concealing the user payload inside a carrier, this stage transforms it into ciphertext.

➢ *Integrity Check & Authentication (MAC / Hashing)*

This section describes the cryptographic protection layer: AES (Advanced Encryption Standard) is used for efficient payload encryption, while RSA is used for safe key management and exchange.

- *Purpose:*

To prevent unauthorized users from viewing the content, ensure payload confidentiality, and protect encryption keys during transmission.

This step converts the user payload into ciphertext before hiding it inside a carrier.

➢ *Adaptive Steganography (LSB with Error Correction & Carrier Selection)*

The encrypted payload is hidden using an enhanced Least Significant Bit (LSB) steganography approach. The document highlights three enhancements:

- *Carrier Selection*

The system chooses the most suitable carrier based on payload size and type.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

- *Error Correction*

Error correction coding is applied to: Improve robustness and Allow recovery even after compression or noise.

- *LSB Embedding*

Data is embedded in the least significant bits of the carrier in a controlled manner.

✓ *Purpose:*

Conceals the existence of the encrypted data and Improves resistance to distortion and steganalysis.

➢ *Carrier Media (Image / Audio / Video / PDF)*

The system supports multiple carrier formats, including: Images, Audio files, Video files, and PDF documents.

- *Purpose:*

Enhances flexibility, makes detection harder since attackers cannot assume a fixed carrier type, and allows adaptation to different application scenarios the output at this stage is a stego-object that appears normal.

➢ *Secure Network Transmission (HTTPS / Socket)*

The system supports a variety of carrier formats, such as images, audio files, video files, and PDF documents.

- *Purpose:*

To increase flexibility, make detection more difficult because attackers cannot assume a fixed carrier type, and enable adaptation to various application scenarios. At this point, the output is a stego-object that looks normal.

➢ *Extraction Module (Stego Analysis + Integrity Check)*

This indicates that the system uses the MAC or hash to verify integrity and extracts concealed data via steganalysis on the receiving end.

- *Purpose:*

Verifies the authenticity of the extracted payload, identifies manipulation prior to decryption, and stops the processing of harmful or damaged data.

➢ *Decryption (AES / RSA)*

This means that the extracted encrypted payload is decrypted using AES (with the session key) and RSA (for key recovery, if necessary) once integrity has been confirmed.

- *Purpose:*

To ensure that only authorized recipients can access the data and to restore the original user payload.

➢ *Extraction & Decrypt (Manual Key Input)*

This block indicates that key input may be manual, which means that final data recovery requires human intervention and the recipient expressly gives the decryption key or private key.

- *Purpose:*

This draws attention to the shortcomings of the current design, such as the possibility of human error in manual key input and the potential for greater security and usability improvements through automation.

Moreover, the figure describes a layered security architecture that: sensitive data is encrypted, its integrity is confirmed, it is concealed inside multimedia carriers, it is securely transmitted, and it is extracted, verified, and decrypted at the recipient. Although formatting is to blame for the figure's repetition, it logically depicts a single end-to-end secure operation.

- *Advantages of the New System over the Existing System*

Table 1 Advantages of Proposed System over Existing System

Aspect	Existing System	Proposed System
Key Management	Manual password sharing	RSA/ECC secure exchange
Security	AES only, no integrity	AES-GCM + MAC (Confidentiality + integrity)
Steganography	Naïve LSB	Adaptive LSB + ECC + Multi-carrier
Robustness	Fragile, easily broken	Resistant to Compression/attacks
Workflow	Disjointed tools	Unified, automated pipeline
Usability	Error-prone manual steps	Guided user interface (Python/Streamlit)

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

- *The Proposed System Design*

The suggested approach combines network-based transmission, steganographic embedding, and cryptographic encryption into a hybrid crypto-steganography framework. By using an incremental and iterative system development life cycle (SDLC) methodology, it allows system components to be refined gradually until optimal performance is attained. The technique ensures both confidentiality and imperceptibility, while maintaining usability and resilience in real-world communication circumstances.

✓ *Overview of the New System Architecture*

The system consists of three core modules:

✓ *Cryptography Module:*

Encrypts plaintext messages using AES-256 symmetric encryption and before embedding, data is converted into cipher-text to ensure confidentiality.

✓ *Steganography Module:*

This module embeds encrypted data in cover images (such as PNG files) using the Least Significant Bit (LSB) technique and by maintaining excellent image quality while concealing information, imperceptibility is ensured.

✓ *The Network Transmission Module:*

Enables safe data transfer over a local area network (LAN) or internet-based socket application and oversees the transmission and reception of stego-images between servers and clients.

- *Methodological Steps and Instrument/Tools*

The suggested system's design and implementation adhere to a Design and Development Research (DDR) methodology, which is organized into steps that are supported by the right tools and technology. Every step is chosen with care to solve the flaws found in the current system while guaranteeing that the fundamental objectives of usability, robustness, confidentiality, and imperceptibility are met.

✓ *Data Encryption*

- Step: Before embedding, encrypt sensitive data.
- Method: Use the AES-256 symmetric encryption technique, a strong block cipher that is widely used in the business and is renowned for its effectiveness and security.
- Instruments: PyCryptodome: A Python package that encrypts and decrypts data using AES.
- Justification: This step makes sure that the payload cannot be read without the right decryption key, even if the stego-object is intercepted.

✓ *Data Embedding (Steganography)*

- Step: Use a cover file to conceal the encrypted contents.
- Method: Insert ciphertext into a PNG image's pixel values using the Least Significant Bit (LSB) approach.
- Tools: OpenCV and PIL (Python Imaging Library), for image processing and pixel manipulation
- Justification: LSB embedding ensures imperceptibility and maintains the image's visual quality while providing sufficient payload capacity. Due of its lossless compression, which guards against data loss during embedding, PNG is recommended.

✓ *Network Transmission*

- Step: Transmit the stego-image across a network.
- Method: Build a client-server communication system for transmitting and receiving stego-images.

✓ *Tools:*

- Python Sockets: For low-level TCP/IP communication.
- Flask/Flask-SocketIO: For implementing web-based communication.
- Requests library: For handling HTTP transmissions.
- Justification: This supports both local area networks (LANs) and internet-based applications by guaranteeing the safe, real-time interchange of stego-images.

✓ *Data Extraction and Decryption*

- Step: Extract and decrypt the concealed message at the recipient's end.
- Method: Use reverse-LSB decoding to extract the embedded ciphertext and to recover the original message, use AES decryption with the appropriate key.
- Tools**:** PyCryptodome (AES decryption).
- PIL/OpenCV (for extraction).
- Rationale: This guarantees precise retrieval of the concealed message while preventing data loss.

✓ *System Evaluation*

- Step: Evaluate the system performance to validate effectiveness.
- Metrics Used: PSNR (Peak Signal-to-Noise Ratio): To measure image quality and imperceptibility (≥ 40 dB indicates good quality).
- Encryption/Decryption Time: To assess computational efficiency.
- Transmission Time: To evaluate network latency and throughput.
- Success Rate of Retrieval: To verify accuracy under different conditions (message size, carrier type, network quality).
- Tools: Python testing scripts for performance benchmarking and OpenCV for pre- and post-steganography image analysis.
- Justification: Assessment guarantees that the system is not only operational but also realistic, effective, and safe.

✓ *User Interface Development*

- Step: Give users an easy-to-use interface.
- Method: Design a simple and interactive GUI or web interface.
- Tools:
- Tkinter: For desktop GUI applications.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

- Streamlit or Flask: For lightweight, web-based interfaces.
- Rationale: By improving usability, this step enables even non-technical people to utilize the system.

- NB: The above discussion is captured in the table below:

Table 2 Methodological Steps and Tools

Objective	Method	Tools/Technologies
Encrypt sensitive data	AES-256 encryption	PyCryptodome
Embed encrypted data	LSB steganography in PNG	PIL, OpenCV
Transmit data	Client-server socket/web app	Python Sockets, Flask-SocketIO
Extract & decrypt	Reverse LSB + AES decryption	PyCryptodome, OpenCV
Evaluate performance	PSNR, latency, retrieval success	Python scripts, OpenCV
User interface	GUI/web interface	Tkinter, Streamlit, Flask

- *System Block Diagram*

The crypto-steganography framework's logical workflow is depicted in the New System Block Diagram. It demonstrates in detail how the network-based program protects, conceals, transmits, and retrieves plaintext data. Every block is essential to maintaining robustness, confidentiality, and imperceptibility.

✓ *Input Message*

The user starts the process by entering an input, which could be a document, text message, or any other sensitive information that needs to be protected. The message is still in plaintext at this point and could be intercepted if sent directly.

✓ *AES Encryption*

The AES-256 technique is used to encrypt the plaintext message. AES converts the message into ciphertext, making sure that the content cannot be decrypted without the decryption key, even if it is intercepted. Confidentiality is guaranteed by this step.

✓ *Embed in Image (LSB Method)*

Using the Least Significant Bit (LSB) steganography technique, the encrypted ciphertext is embedded into a cover image. This embedding achieves imperceptibility because it does not substantially change the image's visibility. Because it employs lossless compression, which guards against data loss while embedding, the PNG format is recommended.

✓ *Send via Network (Client-Server Transmission)*

A client-server application developed with Flask-SocketIO or Python sockets is used to transmit the stego-image, or image holding the concealed message, over the network. Transmission might take place across a local area network (LAN) or online. The communication channel for safe message delivery is guaranteed by this block.

✓ *Receiver Extracts Encrypted Data*

The secret ciphertext is extracted at the receiver's end by processing the stego-image. The embedded bits are extracted from the carrier picture using reverse-LSB decoding. Even in the case that the stego-image is intercepted, the message is still encrypted at this stage, preventing unwanted access.

✓ *AES Decryption*

The AES algorithm is used to decrypt the collected ciphertext using the appropriate key. This restores the original format of the encrypted data. This step guarantees that the information can only be accessed by the designated recipient who has the right key.

✓ *Original Message Retrieved*

The procedure guarantees that the message is delivered securely, precisely, and imperceptibly, shielding it from eavesdropping or tampering during transmission. The encrypted message is returned to its original plaintext form and displayed to the recipient.

- NB: The discussion above is captured in the figure below

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

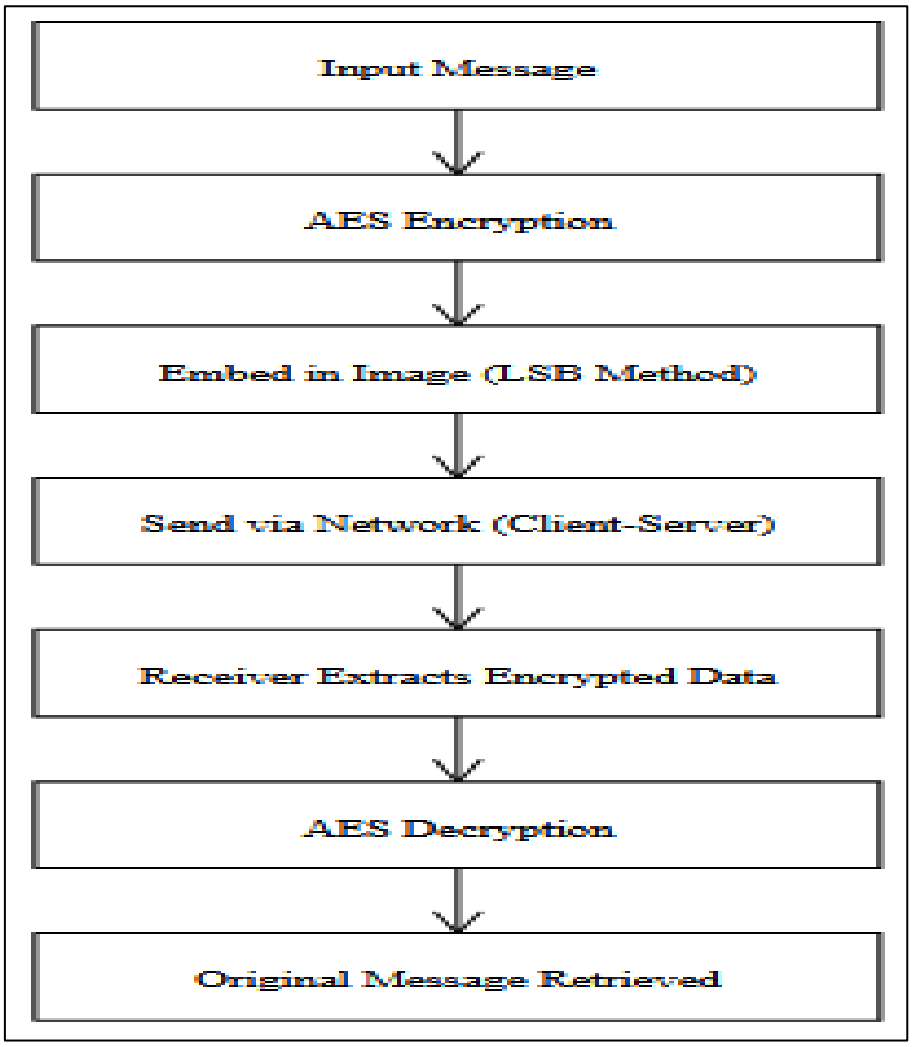

Fig 4 Proposed System Block Diagram

- *System Implementation*

When implementing the new system, Python was used for development, with Streamlit for the user interface and SQL/MySQL for data administration. We Integrate the steganography modules and cryptography libraries (like PyCryptodome), and we create a network-based application prototype for testing in real time.

The analysis and design of the new system suggested that hybrid crypto-steganography data security system are thoroughly covered. It describes the modules, the development technologies, the system architecture, and the flow of operations of the new system. Diagrams that show the new system architecture and the data flow through different components are also included.

- *Choice of Development Environment*

Four main objectives are met by the environment: practicality (quick prototyping, cross-platform, minimal setup cost), networking (reliable client–server transmission), signal quality (correct image processing), and security (strong crypto). Additionally, it facilitates testability, scalability from a single PC to LAN/Internet deployment, and reproducibility (versioned dependencies).

- *Programming Language Used*

We used Python 3.11+ because it has a matured crypto and imaging ecosystems; rapid prototyping; rich scientific tooling; cross-platform (Windows/Linux/macOS); and easy packaging. It has Short development cycle with strong library support and reproducible builds.

- *Implementation Architecture*

A modular client-server design for a data security system that combines network communication, steganography, and encryption is described in the document. By enclosing encrypted data within images before sending them across a network, the method maintains confidentiality.

The architecture places a strong emphasis on a multi-layered security approach:

- ✓ Encryption makes sure that the data is safe even if it is stolen by an attacker.
- ✓ Steganography makes sure that the hidden data is difficult for attackers to find or even suspect.

The approach solves the drawbacks of employing steganography and cryptography separately by integrating

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

them and Real-time secure communication across networks is made possible by the client-server architecture;

➢ *Key Components*

- *User Interface (UI):*

This is where users enter the system. It accepts a cover picture, which is the medium in which the message will be concealed, and a plaintext message, which is the sensitive information to be protected. The user interface (UI) is essential to the user experience since it makes file selection and supplying the secret message simple.

- *The AES Encryption Module*

It is a secret key and the Advanced Encryption Standard (AES) which is used to encrypt the plaintext communication. By adding a cryptographic security layer, this makes sure that the secret message cannot be decrypted without the decryption key, even if it is revealed. AES is selected due to its effectiveness, robustness, and broad use. AES strikes a mix between robust security and computational efficiency.

- *Steganography Module (LSB)*

The Least Significant Bit (LSB) approach is used to embed the encrypted text (ciphertext) into the selected cover image. The least significant bits of an image's pixels are altered via LSB steganography to make the hidden data invisible to the human eye. This guarantees concealment and makes it more difficult for attackers to find out that concealed information is present. LSB steganography offers a low-tech approach to data concealing.

- *Network Transmission Module:*

This module facilitates network-based communication by sending the stego-image from sender to recipient. By sending an image instead of raw data, it lowers suspicion and makes it more difficult for attackers to identify it.

- *Receiver's Stego Analysis Module:*

This module recovers the encrypted data by reversing the embedding process and recovering the concealed ciphertext from the stego-image at the receiver's end.

- *Decryption Module:*

The recovered ciphertext is decrypted via AES using the shared secret key, guaranteeing end-to-end confidentiality by restoring the original plaintext message.

- *Output Module:*

This ensures that the encryption, steganography, and decryption cycle has been successfully completed by displaying the recovered plaintext to the recipient.

✓ NB: The discussion above is capture in the figure below

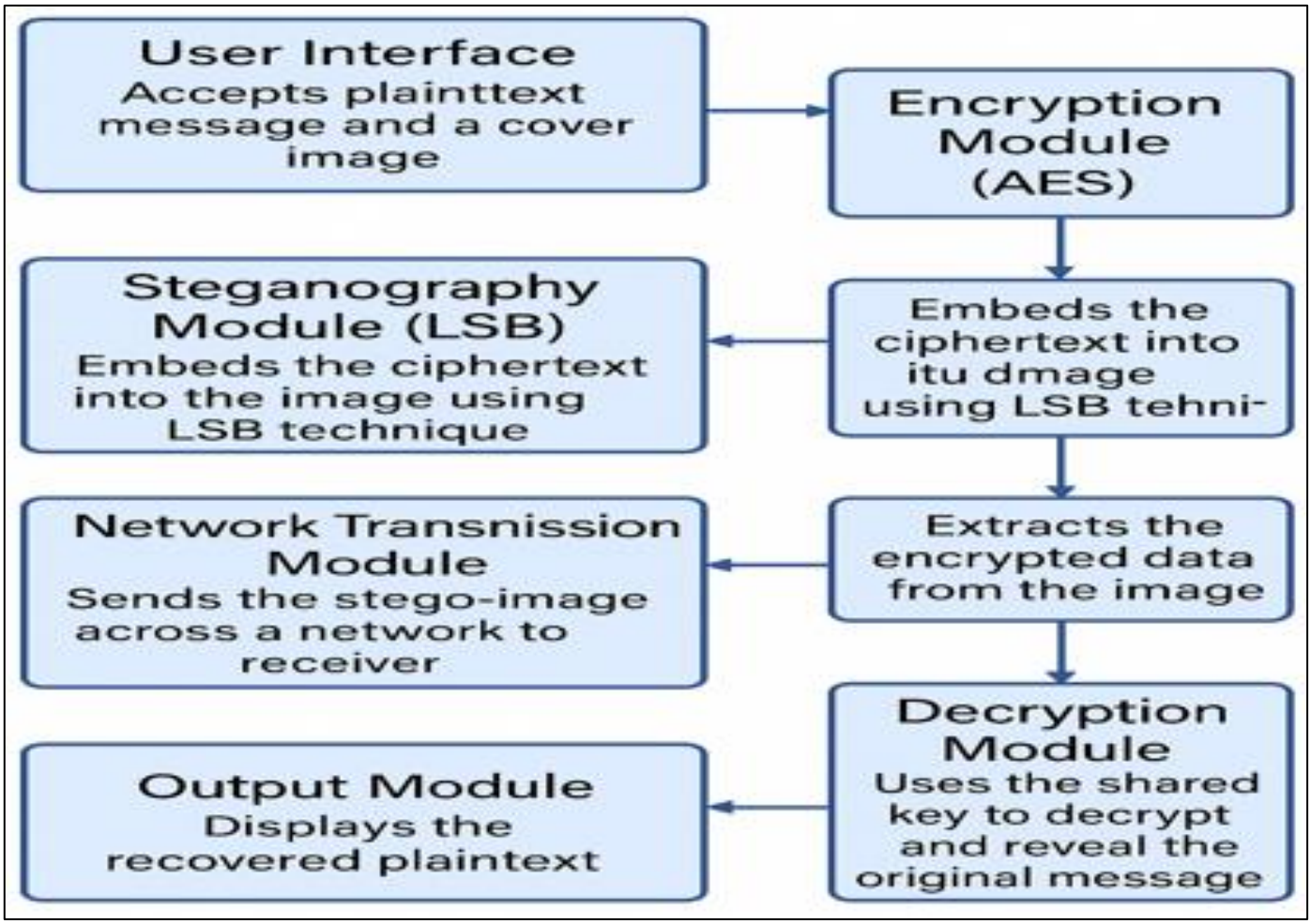

Fig 5 System Architecture Diagram

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

- *Implementation Flowchart*

The flowchart below represents the logical operation of the system:

✓ *Sender Side:*

- Step1: Start
- Step2: Input Message & Image
- Step3: Encrypt Message (AES)
- Step4: Embed Encrypted Message in Image (LSB)
- Step5: Send Stego-Image over Network
- Step6: End

✓ *Receiver Side:*

- Step1: Sstart
- Step2: Receive Stego-Image
- Step3: Extract Encrypted Message (LSB)
- Step4: Decrypt Message (AES)
- Step5: Display Original Message
- Step6: End

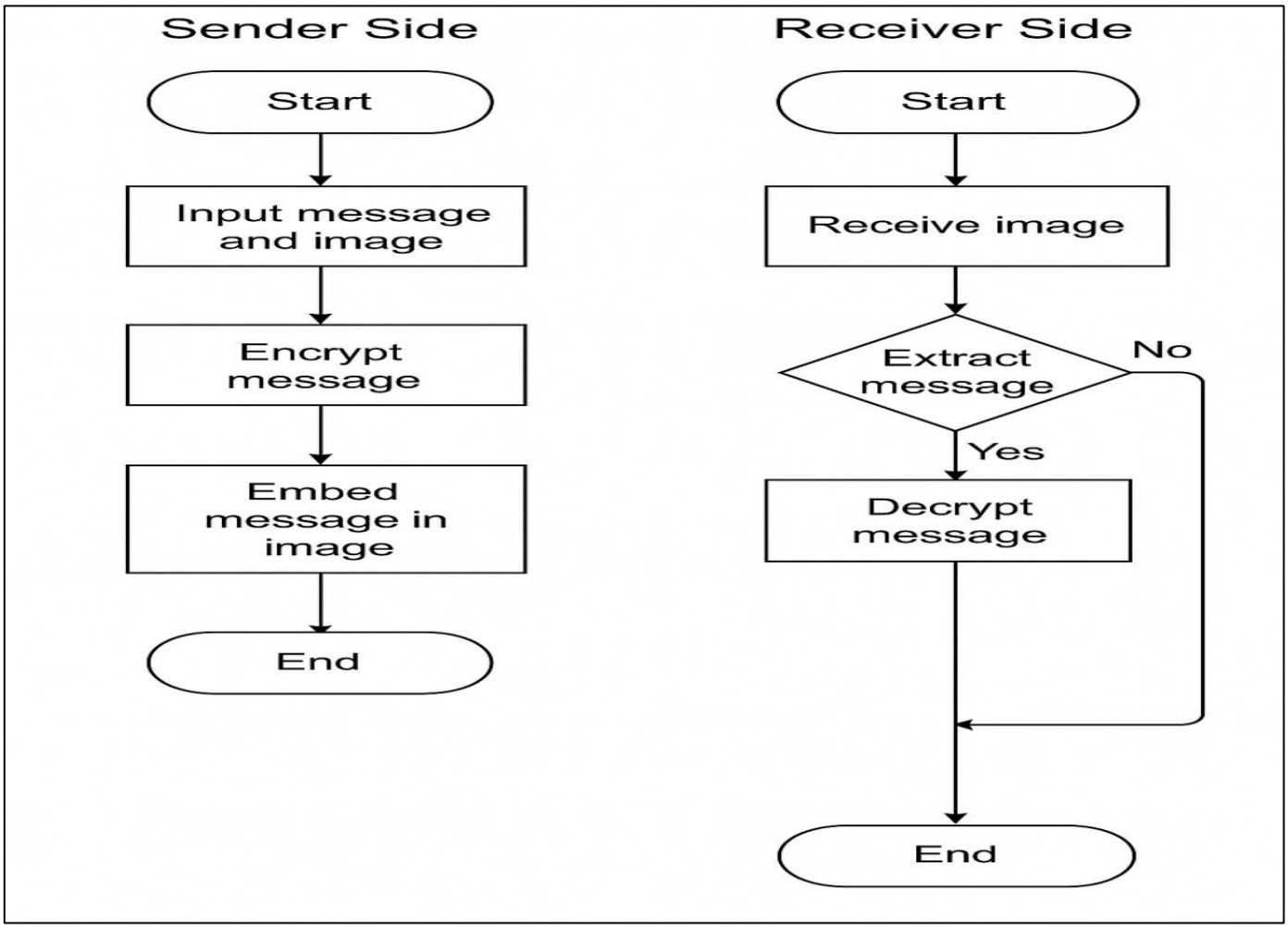

Fig 6 System Implementation Flowchart

✓ *Technologies Used*

- Python: Core development language.
- PyCryptodome: For AES encryption and decryption.
- PIL (Pillow): For image manipulation and LSB steganography.
- Socket / Flask: For client-server network communication.
- Tkinter / Streamlit: User interface (GUI/Web).
- JSON: For sending structured metadata.

✓ *Security Considerations*

- The sharing of secret keys must be secure (for example, via a public key or a pre-shared technique).
- The system is made to withstand casual steganalysis; there is little distortion in the images.
- Even in the event that LSB is compromised, AES provides robust cryptographic protection.

- *System Testing and Validation*

Testing of the new system was done by **v**erifying encryption, embedding, and extraction processes, which is known as functional testing, and we carried out performance assessment with metrics like: PSNR (Peak Signal-to-Noise Ratio) which measures imperceptibility, MSE (Mean Square Error), which measures image distortion, SSIM (Structural Similarity Index) which measures visual similarity, Processing Time, which measures efficiency, and Payload

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Capacity, which measures amount of data hidden without detection, and we measured Security Robustness which shows resistance to cryptanalysis and steganalysis attacks.

- *Significance of the Workflow*

✓ *Confidentiality:*
Prior to embedding, AES encryption ensures confidentiality and LSB embedding in lossless carriers maintains imperceptibility.

✓ *Robustness:*
The system guarantees successful retrieval and withstands simple steganalysis.

✓ *Usability:*
Designed to reduce human mistake by automating encryption, embedding, transmission, and extraction.

✓ *Security-in-Depth:*
Makes it very difficult for unauthorized individuals to gain access by combining two layers of protection (steganography and encryption).

IV. RESULT/DISCUSSION

Screenshots of the developed software's graphical user interface (GUI) are used in this chapter's section to show the implementation's outcomes. The following is what it is:

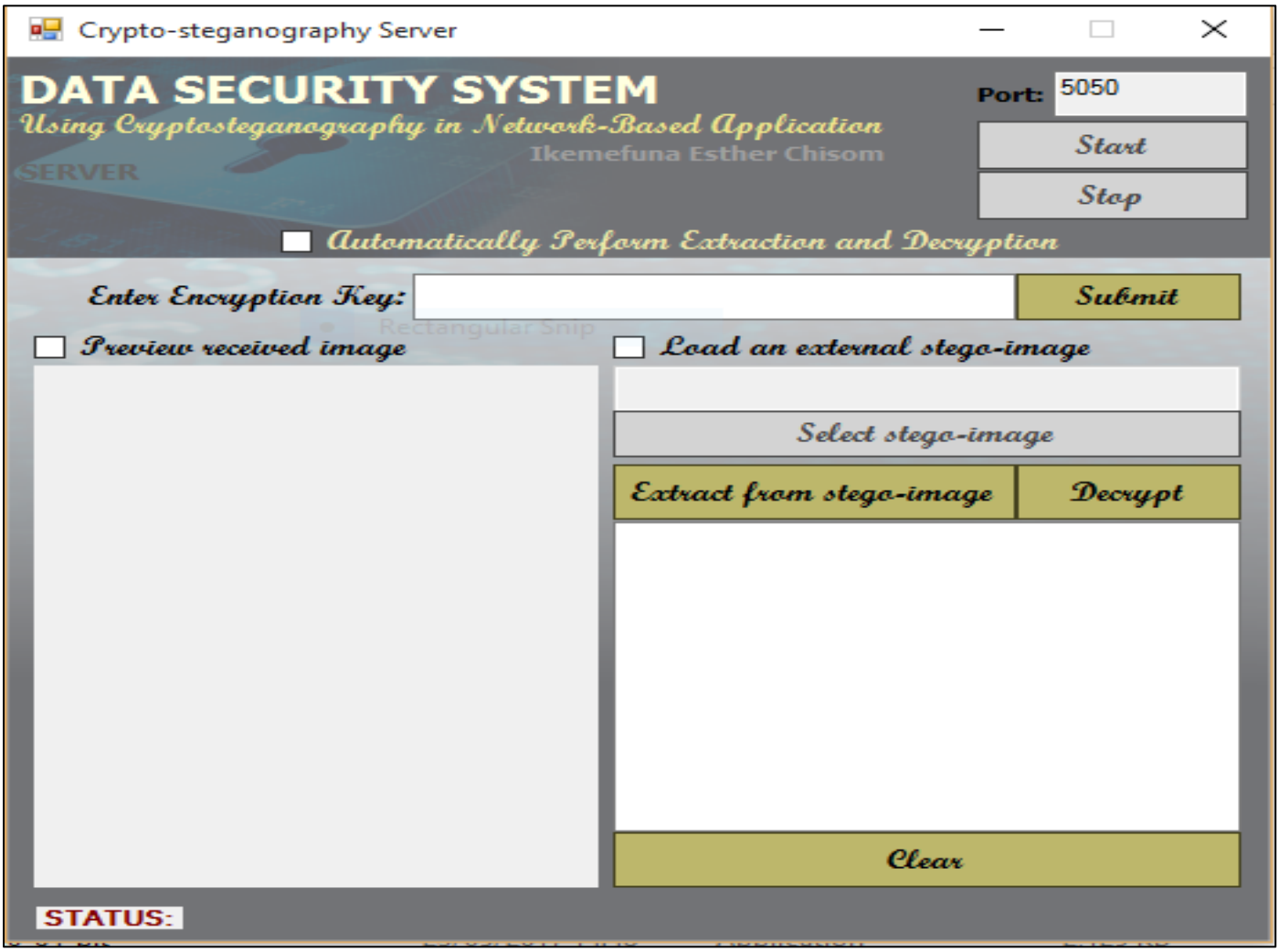

Fig 7 How to Connect to Server Side

The software's server-side session is shown in figure 7 above. The client side is connected to the server side. In order to establish a connection with a client system, it automatically obtains the host system's IP address and port address.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

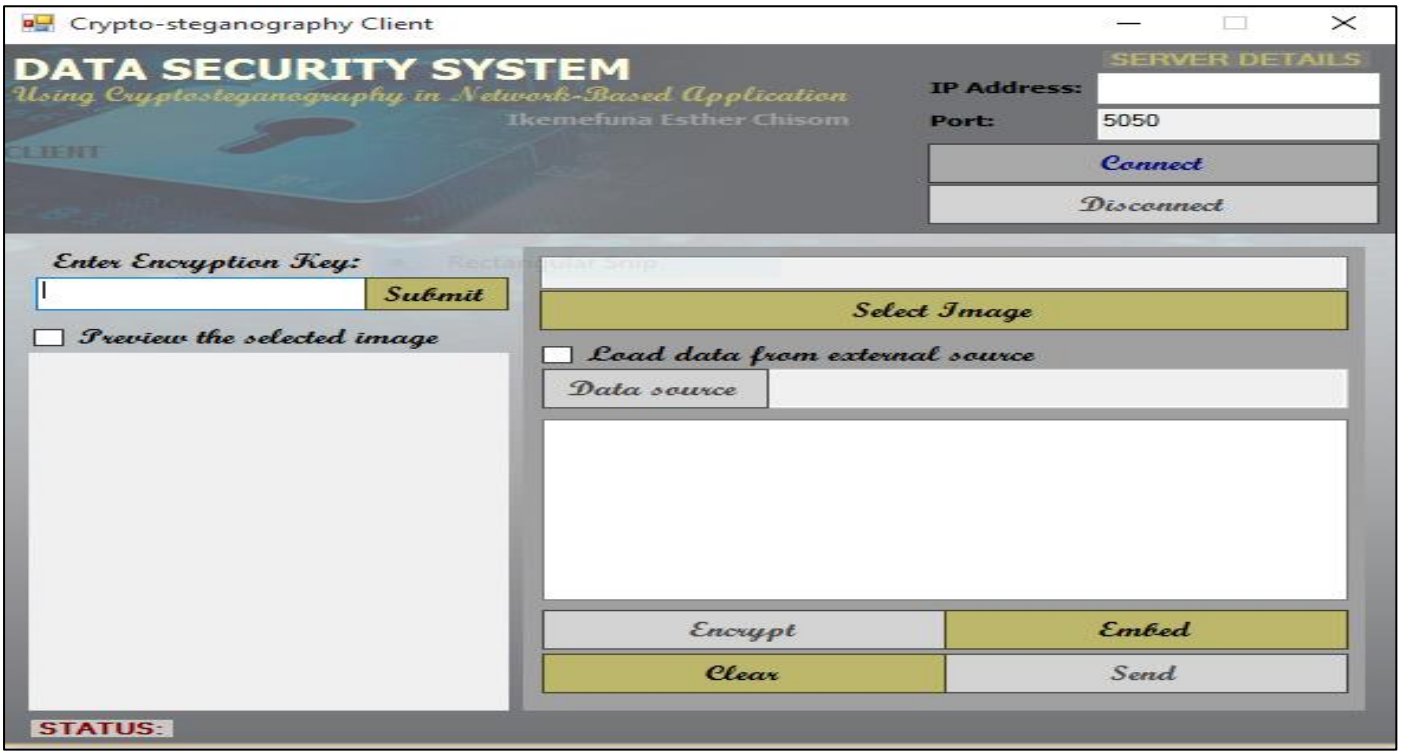

Fig 8 The Client Side.

The IP address of the host (server) system is used by the client side of the software to establish a connection with the server. The client provides the encryption key, chooses and previews the image, enters plaintext or chooses a text file from the system, encrypts the text entered, embeds the text in the chosen image to create a stego-image, and sends the stego-image to the client side.

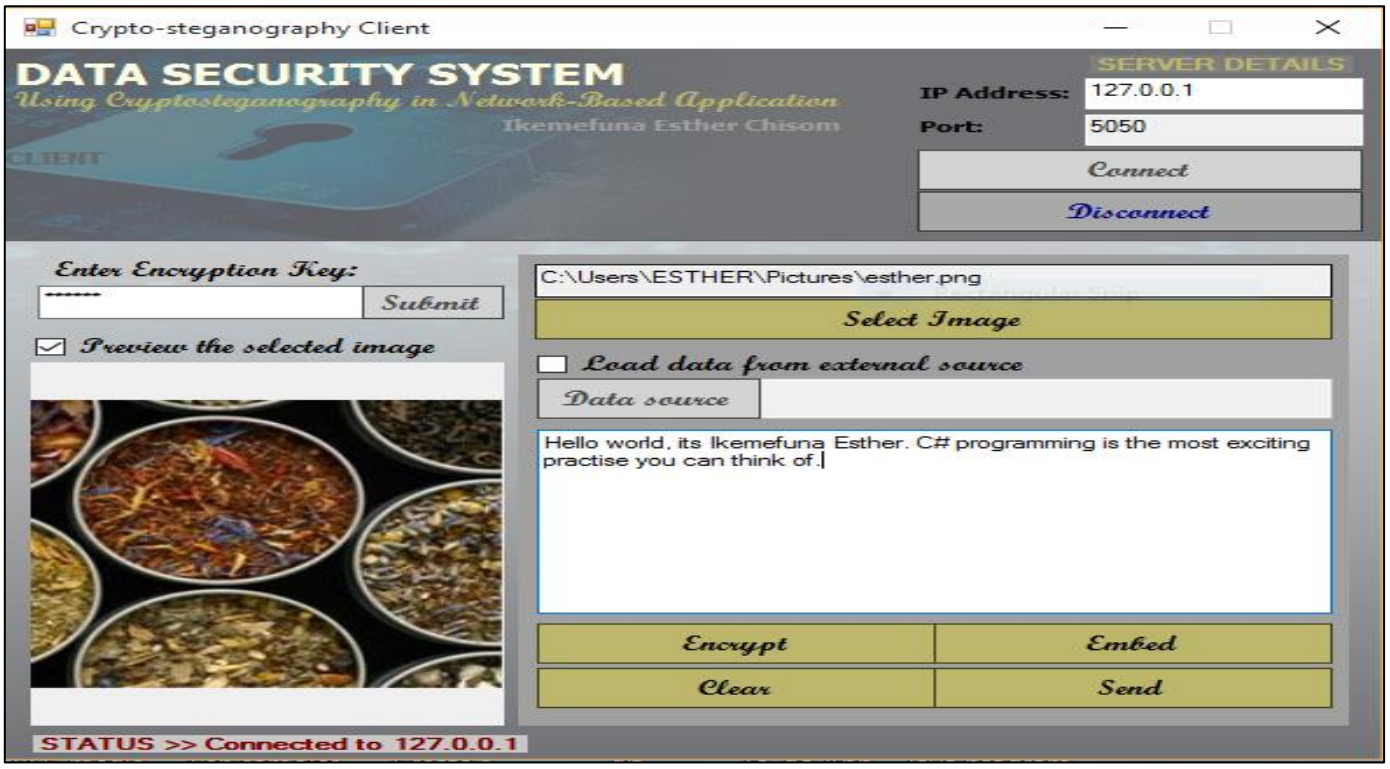

Fig 9 Text Set to be Encrypted

The plain text is stored in the main form's text box prior to encryption. At least six characters must be entered in the password field, which acts as the encryption key, for encryption to take place. After the text was extracted, the server would require the password to decode it.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Fig 10 Encryptions of the Text

As seen in the accompanying image, encryption transforms ordinary text into secret characters. When you click the "encrypt" button, this is accomplished.

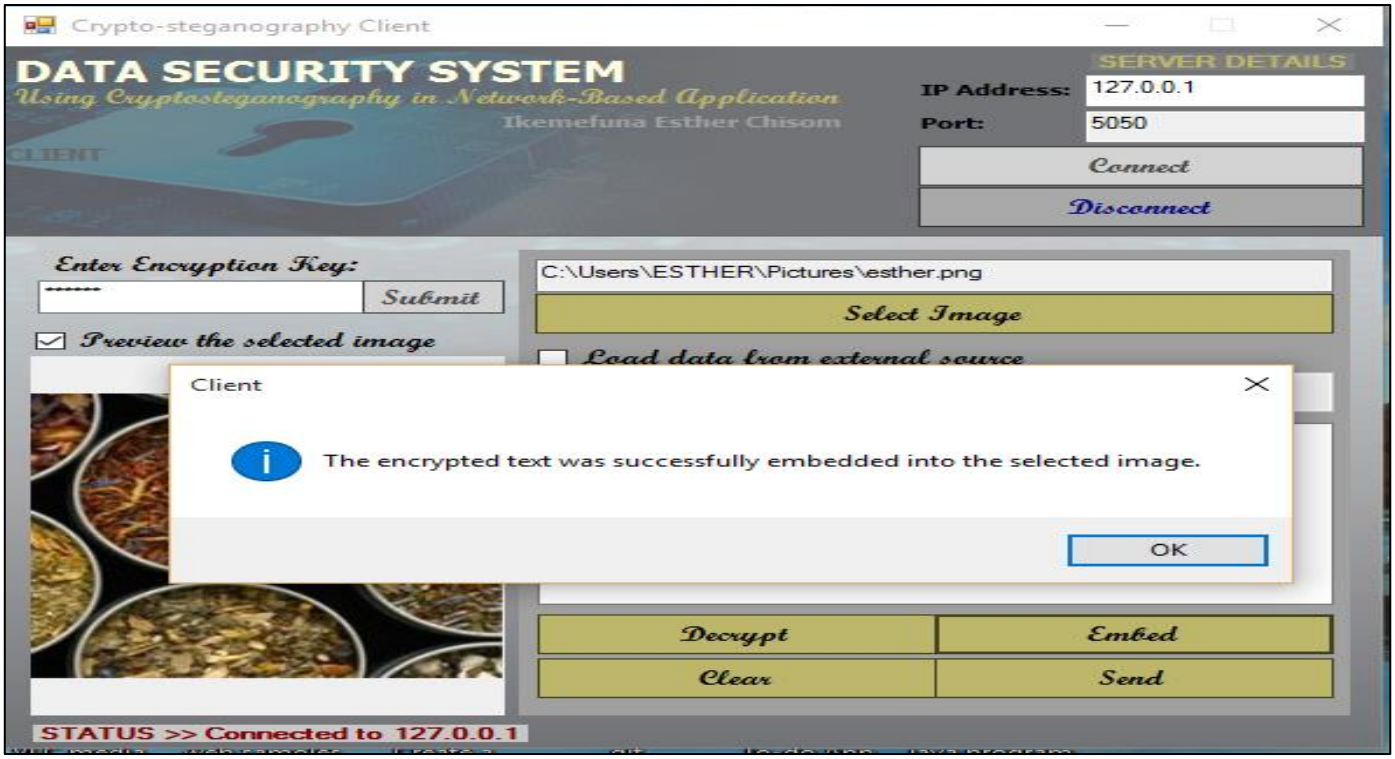

Fig 11 Embedding Encrypted Text in Image

The message box that appears when you click the hide button is seen in the figure above. The chosen image is automatically transformed into a stego-image when the encrypted text is embedded in it using the Embed button.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

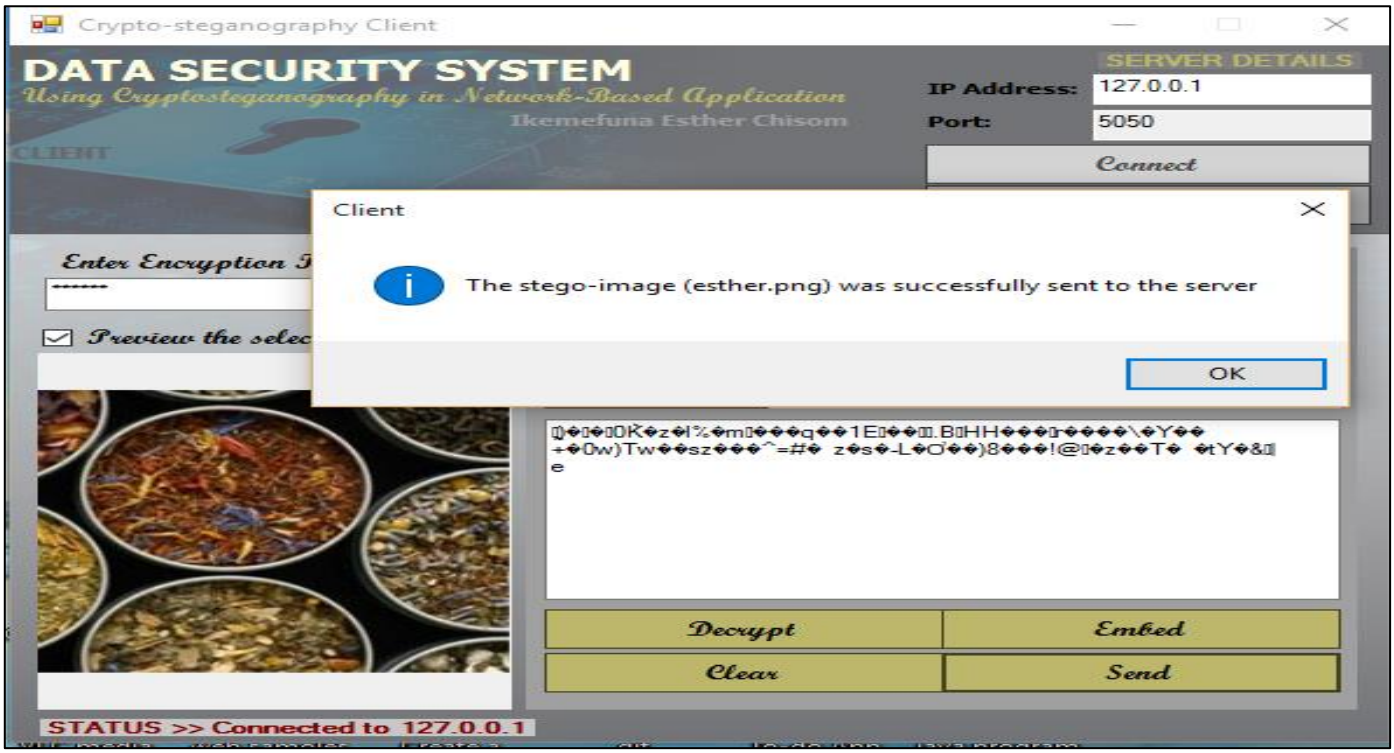

Fig 12 Transmission Stego-Image from Client to Server Side

The stego-image is transmitted to the server side on clicking the send button and a message box is displayed, confirming that the stego-image was successfully sent to the server.

Fig 13 how to Extract Encrypted Text from Stego-Image

When the stego-image is received at the server end and the Extract from stego-image button is clicked, the encrypted text that was contained in the stego-image is shown, as seen in the above figure. Because it was encrypted before being embedded into the image, the text displayed in the text box looks encrypted. The client must decode the text in order to read the transmitted message and make it legible and meaningful. Another name for the encrypted text is a cipher-text.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Fig 14 Decrypting the Extracted Text

The decrypted text can be seen in image 14 above. After entering the encryption key and selecting the Decrypt button, the text was decrypted. By doing this, the encrypted text is automatically decrypted and transformed from cipher-text to plain text.

➢ *Result Analysis*

The system was implemented as two software applications**:**

- *The Client Software:*

This is in charge of encrypting the plaintext, incorporating it into a stego-image, and sending it.

- *Server Software:*

Constantly running, awaiting incoming connections, just like in standard client-server configurations. After receiving the stego-image, it decrypts it to reveal the original plaintext and extracts the secret data.

- *The form of Communication Flow is:*

✓ A client uses the IP address and port number of the server to establish a connection.
✓ The client can send the stego-image with the encrypted data after connecting.
✓ After that, the server extracts and decrypts the data, guaranteeing that the private data is safely recovered.

This result analysis demonstrates that our system successfully combines LSB steganography with AES encryption in a network-based setting to secure sensitive data.

V. CONCLUSION AND FUTURE RESEARCH

A reliable method for guaranteeing confidentiality, integrity, and stealth in digital communications is the design and deployment of a data security system utilizing crypto-steganography in a network-based application. Through the combination of steganography (LSB embedding) and cryptography (AES encryption), the system makes sure that the existence of the message is hidden behind cover graphics in addition to its content being safeguarded by encryption. Key security issues including eavesdropping, illegal access, and data manipulation during transmission are addressed by this two-layered method. The logical architecture exhibits a smooth flow of secure data handling and consists of sender-side encryption and embedding as well as receiver-side extraction and decryption. Furthermore, the system's practicality in Internet-based communications is demonstrated by the utilization of network transmission modules. The system's ability to hide and recover communications without sacrificing security or image quality is confirmed by the testing findings. There are still issues with the system in spite of its advantages, namely the requirement for effective key exchange methods and making sure it is resistant to sophisticated steganalysis techniques. However, our study demonstrates that crypto-steganography is a

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

practical and efficient way to protect private information in contemporary communication settings.

➢ *Future Research*

- *Integrate Secure Key Exchange Protocols:*

Implementing key exchange mechanisms such as Diffie-Hellman or RSA will ensure that the AES keys used for encryption are securely transmitted between sender and receiver.

- *Enhance Steganographic Robustness:*

To prevent detection and extraction by steganalysis tools, more advanced embedding techniques or adaptive LSB algorithms should be incorporated.

- *Optimize for Real-Time Use:*

For practical deployment, particularly in web or mobile applications, the system should be optimized for speed and minimal resource consumption.

- *Incorporate Error Handling and Redundancy:*

Introduce error correction codes (ECC) within the steganographic process to ensure accurate data retrieval even if the stego image experiences minor transmission degradation.

- *Extend Support for Multiple File Formats:*

Expanding the system to support various image, audio, or video formats will enhance versatility and real-world usability.

- *Conduct Broader Testing:*

Future work should involve testing across various network conditions, user platforms, and steganalysis tools to evaluate system resilience and usability at scale.

- *User Education and Awareness:*

Train users on best practices for selecting secure keys, using trusted cover images, and maintaining digital hygiene to maximize the system's security potential.

➢ *Declaration*

- Ethical Statement: The submitted work should be original and has not been published elsewhere in any form or language to our knowledge.
- Competing Interest: There is no competing interest of any sort.
- Conflicts of Interest: The authors declare no conflict of interest for both financial and non-financial interests for this article.
- Availability of data and material (data transparency): Applicable on request from the corresponding authors.
- Code availability (software application or custom code): Applicable on request
- Consent to participate: Not Applicable
- Consent for publication: Not Applicable

REFERENCES

[1]. Almomani, I., Alkhayer, A., El-Shafai, W., (2022), a Crypto-Steganography Approach for Hiding Ransomware within HEVC Streams in Android IoT Devices, *Sensors*, *22*(6), 2281; https://doi.org/10.3390/s22062281, s22062281

[2]. Arora, R. and Parashar, A., (2023). Secure User Data in Cloud Computing Using Encryption Algorithms. International Journal of Engineering Research and Applications (IJERA), Vol. 3, 1922-1926.

[3]. Channalli, S. and Jadhav, A. (2019). Steganography an Art of Hiding Data. International Journal on Computer Science and Engineering, IJCSE, vol. 1, no. 3.

[4]. Chen L, et al. (2020), multilevel support vector regression analysis to identify condition-specific regulatory networks. *Bioinformatics*, 26(11):1416-22

[5]. Choudhary, U., and Parul, A., (2024), Image Steganography Combined with Cryptography for Covert Communication, *Jaypee Institute of Information Technology jornal*, C3 2024, August 08–10, 2024, Noida, India.

[6]. Ezeofor, C. J. and Ulasi, A. G (2024). Analysis of Network Data Encryption and Decryption Techniques in Communication Systems. International Journal of Innovative Research in Science, Engineering and Technology, Vol. 3.

[7]. Fridrich, J. (2019). *Steganography in Digital Media: Principles, Algorithms, and Applications.* Cambridge University Press. https://doi.org/10.1017/CBO9781139192903

[8]. F. Ifeanyi, (2025), IoT Threat Detection using Federated Learning and Edge AI with Random Forest Optimization, *Nature Journal of Emerging Sciences, Technologies and Innovations (NATURERUST)*, Vol. 3 No. 2, (2025). https://naturerust.com/index.php/njesti/article/view/10

[9]. H. Motameni, M. Norouzi, M. Jahandar, and A. Hatami (2019), Labeling Method in Steganography, *Proceedings of World Academy 0f Science, Engineering and Technology*, Volume 24 October 2007 ISSN 1307-6884

[10]. Hussain, M. and Hussain, M. (2021). Embedding data in edge boundaries with high PSNR. Proceedings of 7th International Conference on Emerging Technologies (ICET 2011), pp.1-6.

[11]. Huang, J.,Luo, W. and Huang, F., (2020). Edge adaptive image steganography based on LSB matching revisited. IEEE Transactions on Information Forensics and Security, Vol. 2, pp. 201–214.

[12]. Helmy, M. (2023), Audio Transmission Based on Hybrid rypto-steganography Framework for Efficient Cyber Security in Wireless Communication System, Multimedia Tools and Applications (2025) 84:18893–18917 https://doi.org/10.1007/s11042-023-17921-2

[13]. H. V. Patil and V. P. Sonaje 2023, "Crypto-Stego: A hybrid method for encrypting text messages using AES and LSB algorithms," *Int. J. Intell. Syst. Appl.*

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26feb204

Eng., 2024. [Online]. Available: https://ijisae.org/index.php/IJISAE/article/view/7468

[14]. J. Onyilo, (2025), Explainable AI Intrusion Detection System, *Nature Journal of Emerging Sciences, Technologies and Innovations (NATURERUST)*, Vol. 1 No. 1 (2025). DOI: https://doi.org/10.65752/e5fa2b27

[15]. Jung, K. H. and Yoo, K. Y. (2019). Image data hiding method based on multi-pixel differencing and LSB substitution methods. Proc. 2008 International Conference on Convergence and Hybrid Information Technology (ICHIT '08), pp. 355-358.

[16]. Johnson, N. F., Duric, Z., & Fridrich, J. (2021). *Information Hiding: Steganography and Watermarking Attacks and Countermeasures*. Kluwer Academic Publishers.

[17]. Kallapu, B., Janardhan, A.N., Hejamadi, R.M., Shrinivas, K.R.N., Saritha; Ramesh, R.K., Gabralla, L.A. (2025), Multi-Layered Security Framework Combining Steganography and DNA Coding. *Systems 2025*, 13, 341. https://doi.org/10.3390/systems13050341

[18]. Kumar, B. and Dr.Murti, K. (2021). Data Encryption and Decryption process Using Bit Shifting and Stuffing (BSS) Methodology. Journal on Computer Science and Engineering (IJCSE), Vol. 3.

[19]. Obaida, M. and Awad, R., (2023). A New Approach for Complex Encrypting and Decrypting Data. International Journal of Computer Networks and Communications (IJCNC), Vol. 5.

[20]. Patel, A., & Gupta, S. (2019). "Efficient Techniques for Digital Data Hiding and Secure Communication." *International Journal of Computer Applications*, 180(47), 25–31.

[21]. Rahul, P., Shinge, R. and Vineet, S., (2023). An Encryption and Decryption Algorithm Based on ASCII Value of Data. International Journal of Computer Science and Information Technologies, Vol. 5.

[22]. Reddy, S., Sowjanya, Ch., Praveena, P.and Prof.Shalini L (2022). Poly-alphabetic Symmetric Key Algorithm Using Randomized Prime Numbers. International Journal of Scientific and Research Publications, Vol. 2.

[23]. R. Saheb (2024), "A new hybrid approach for image security using steganography and encryption," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, 2024. [Online]. Available: https://ijsrcseit.com/index.php/home/article/view/175

[24]. Serdar S. and Umut A. (2018), LSB Substitution and PVD performance analysis for image steganography, International Journal of Computer Sciences and Engineering, E-ISSN: 2347-2693, Vol.-6, Issue-10, and Oct 2018. DOI: 10.26438/ijcse/v6i10.14

[25]. Sharma, K. G., & Kumar, R. (2018). "Crypto-Steganography: A Fusion Approach for Secure Communication." In *Proceedings of the IEEE International Conference on Information Security* (pp. 135–142).

[26]. Suguna, T., Padma, C., Rani, M. J., and Priya, G. (2018). Hybrid Cryptography and Steganography-Based Security System for IoT Networks. *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(8s), 415–421. https://doi.org/10.17762/ijritcc.v11i8s.7221

[27]. S. Singh, N. K. Sekhon, A. Singh, and S. Gupta (2024), "StegaCrypt: Integrating hybrid cryptography and image steganography," *Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET)*, 2024. [Online]. Available: https://www.ijraset.com/research-paper/stegacrypt-integrating-hybrid-cryptography-and image-steganography

[28]. T. Suguna, C. Padma, M. J. Rani, and G. Priya (2023), "Hybrid cryptography and steganography-based security system for IoT networks," *Int. J. Recent Innov. Trends Comput. Commun. (IJRITCC)*, 2023. [Online]. Available: https://ijritcc.org/index.php/ijritcc/article/view/7221

[29]. Wen-Hsiang T. and Da-Chun W. (2023), A steganographic method for images by pixel-value differencing, *Science Direct AI Journal*, Volume 24, Issues 9–10, June 2003, Pages 1613-1626, https://doi.org/10.1016/S0167-8655(02)00402-6

[30]. Wid, A. A., Ali, S. A., Alaa, K. H. (2022), Hybrid information security system via combination of compression, cryptography, and image steganography, International Journal of Electrical and Computer Engineering (IJECE), Vol. 12, No. 6, December 2022, pp. 6574~6584, ISSN: 2088-8708, DOI: 10.11591/ijece. v12i6.pp6574-6584

[31]. Yang, C. H., and Hath, K. J. (2019). Adaptive Data Hiding in Edge Areas of Images with Spatial LSB Domain Systems. IEEE Transactions on Information Forensics and Security, vol. 3, no. 5, pp. 488-497.

[32]. Yang, H. X. and Sun, G. (2019). A High-Capacity Image Data Hiding Scheme Using Adaptive LSB Substitution. Journal: Radio engineering, vol. 18, no. 4, pp. 509-516.