IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Cryptographic Image Protection Using Rsa, Aes And Chaos Algorithm

Author:
[1]M. MANASA ,
M.Tech. Scholar,
Department of Computer Science and Engineering,
School of Engineering,
Malla Reddy University,
Hyderabad, Telangana, India

Under the esteemed guidance of:
[2] Dr. ARUN SINGH CHOUHAN,
Associate Professor,
Department of Computer Science and Engineering,
School of Engineering,
Malla Reddy University,
Hyderabad, Telangana, India

Abstract: In today's digital communication era, protecting image data from unauthorized access and cyber threats is crucial. This project introduces a strong image encryption framework based on Super Encryption. It combines three effective cryptographic techniques: Chaos-based scrambling, AES (Advanced Encryption Standard), and RSA (Rivest-Shamir-Adleman) encryption. This layered approach ensures complete protection while maintaining both confidentiality and integrity of image data during transmission and storage.The first security layer uses a Chaos-based algorithm, specifically the Logistic Map, to create a highly sensitive pseudo-random key stream. This stream adds randomness by changing pixel positions and intensities, making it very hard to reverse without the right parameters. This step offers strong resistance to statistical analysis and brute-force attacks by disrupting the natural structure of the image.The second layer uses the AES-128 algorithm, a single key is used for the encryption. Operating in EAX mode, AES not only encrypts the chaos-scrambled image but also creates an authentication tag to ensure data integrity and detect tampering. This layer strengthens the encryption process by using multiple rounds of nonlinear transformations and key mixing.The final layer uses RSA algorithm to protect the AES session key and nonce. These are encrypted with the recipient's RSA public key, making sure that only the authorized party can decrypt and access the sensitive credentials. This guarantees a crytographic key exchange process, even over untrusted channels.The suggested hybrid encryption method, combining the unpredictability of Chaos, the efficiency of AES, and the secure key distribution of RSA, offers a solid solution for secure image handling. It is especially suitable for sensitive areas like medical imaging, surveillance, defense communication, and cloud storage, where data privacy and accuracy are essential.Experimental tests show that this system provides high security with little effect on image quality. The encrypted images seem statistically random, and recovery is impossible without the exact RSA keys, AES parameters, and Chaos initial conditions. The results confirm that this framework is a reliable and scalable method for current image encryption needs.

Keywords- Image Encryption, Chaos Theory, AES, RSA, Super Encryption, Logistic Map, Hybrid Cryptography, Secure Image Transmission, Data Integrity, Confidentiality.

I. INTRODUCTION

1.1 Background

A major issue for computer networks is preventing important information from getting into the hands of unauthorized users. For this reason, encryption techniques were introduced. Most encryption methods are easy to implement and are widely used in information security. Over the last decade, the use of computer networks has grown dramatically, and this trend continues. New networks are being set up and connected to the global internet. The internet is commonly considered the early form of a global information highway. Today, the information transmitted over the internet is not just text; it also includes multimedia like images and audio, with images being the most common. However, as images are used more extensively, their security becomes increasingly important. For instance, protecting military image databases, securing confidential video calls, and safeguarding personal online photo albums are all vital.

With the growth in computer processor power and storage, unauthorized access has become easier. Consequently, image security has become a significant issue in today's computing landscape. Most traditional and modern cryptosystems are designed to protect text data. The original plain text is turned into cipher text, which is a hidden form of the message, stored or sent over the network. Once received, the cipher text can be converted back into the original plain text using a decryption algorithm.

Images, however, differ from text. Traditional cryptosystems like RSA and DES can encrypt images directly, but this is not advisable for two reasons.

• First, the size of an image is always much larger than that of text. Therefore, traditional cryptosystems take much longer to encrypt image data.

• Secondly, the recovered text should be identical to the original message. This requirement isn't necessary for images because, due to human perception, a decrypted image with slight distortion is usually acceptable.

A digital image consists of a two-dimensional grid of elements called pixels, where each pixel holds an intensity value and a specific position defined by its row and column.

An image can be encrypted using PYTHON and an encoder. Each pixel is represented by 8 bits, or 1 byte. Using PYTHON, these pixel values can be turned into bytes. These byte values serve as input to the encoder. The 128-bit encoder then converts these bytes into corresponding encoded bytes. The encoded values are then changed into decimal values for pixels. This process is repeated for each pixel to create a two-dimensional text array that corresponds to the pixel values.

To protect the stored two-dimensional data, they must first be changed into one-dimensional arrays before applying various traditional encryption techniques. The raster sequence of image data can be encrypted into blocks using a block cipher or a stream cipher. A product cipher can also be used for encrypting a file of image data. However, it is generally more efficient to encrypt an image after applying some compression techniques, as this reduces computational requirements and increases processing speed, which is crucial in real-time scenarios.

1.2 Problem Statement

1. Real-time image encryption needs methods that provide strong security while keeping fast processing speeds. Complex algorithms can slow down performance.

2. Traditional encryption methods like Triple DES and Blowfish do not work well for image data because they require a lot of computation and do not handle image-specific features like pixel correlation effectively.

3. To address these challenges, techniques such as chaos-based and hybrid encryption are preferred. They offer better speed, flexibility, and stronger protection against visual data attacks**.**

1.3 Objective of the Study

Three key characteristics in the field of information security are privacy (unauthorized users cannot access a message), integrity (unauthorized users cannot alter a message), and availability (messages are accessible to authorized users).

An ideal image cryptosystem should not only be flexible in its security methods but also deliver high overall performance. The goal of this study is to develop an image cryptosystem that, in addition to the characteristics mentioned above, also has the following traits:

Objective -1:

The system should be computationally secure, meaning it should take an extremely long time to break. In other words, unauthorized users should not be able to access sensitive images.

Ojective-2:

Encryption and decryption should be quick enough to keep system performance intact, meaning the algorithm should be simple enough for users with a personal computer to execute.

The security mechanism should be widely accepted to design a cryptosystem similar to a commercial product and should be flexible.

1.4 Related Works

Securing image data presents unique challenges compared to text, due to the structural and statistical properties of visual content. Over the years, researchers have proposed various encryption methods tailored specifically for images. Some of these methods focused on spatial transformation, while others utilized mathematical models to achieve randomness and security. One notable approach introduced by researchers involved the use of chaotic systems, particularly 2D chaotic maps. These systems were applied to large image blocks to scramble pixel positions and values, making the data unrecognizable. While effective for bulk data, these schemes often required padding for smaller images, which increased the overall file size and processing overhead—making them less suitable for real-time use cases.

Another technique incorporated spatial scan patterns to traverse and rearrange pixel data using predefined paths. While this method increased complexity and confusion, it did not apply standard encryption layers or address secure key exchange, leaving it vulnerable to advanced cryptanalysis techniques. Additionally, compression was not part of the process, which resulted in larger encrypted outputs. Some encryption strategies focused on bit-level and pixel-level permutations. By rearranging the bits, pixels, and blocks of an image based on random indices, these methods could disrupt the visual structure of the image effectively. However, they lacked a cryptographically sound key management system and were mostly effective only against casual attacks.

For robust security in real-world applications, additional cryptographic layers were needed. Other works used vector quantization (VQ)—an image compression technique—combined with simple encryption schemes. While this reduced data size and made transmission more efficient, it didn't provide strong encryption guarantees. The absence of symmetric or asymmetric encryption standards in such systems meant they couldn't resist sophisticated attacks or provide formal security proofs.

To address these limitations, the present project proposes a hybrid encryption system that integrates three layers: Chaos-based scrambling, AES-128 encryption, and RSA encryption. The chaos layer ensures pixel-level randomness and confusion, AES provides fast and secure data transformation with built-in authentication, and RSA securely transmits the AES key and nonce using public-key cryptography.

This combination results in a secure, layered encryption model that is computationally efficient and highly resistant to modern attack techniques. It also supports real-time operation and is well-suited for applications such as secure image storage, medical imaging, and defense-related communications.

Furthermore, the proposed model can be extended to support colored images, larger resolutions, and future upgrades in cryptographic protocols. This adaptability and scalability make it a promising solution in the evolving field of multimedia security**.**

1.5 Limitation of the study

- The current image encryption system uses AES-128 bit encryption as the main symmetric encryption layer in the hybrid Chaos, AES, RSA architecture. While AES-128 is widely seen as secure and efficient, the limited key size of 128 bits may not be enough for ultra-high-security applications or the future threats from quantum computing. The algorithm's key space could greatly expand by adding AES-192 or AES-256. This would improve resistance against brute-force attacks but would also increase computational complexity and resource usage.
- The chaos-based scrambling mechanism in the system relies on the Logistic Map. Although this is effective, it is a single-dimensional chaotic function. It provides a decent level of randomness and initial confusion, but it may not be as secure as multi-dimensional chaotic systems, such as Lorenz, Henon, or hyper-chaotic systems, especially when facing advanced statistical attacks. Furthermore, the predictable nature of chaos, when parameters are known, could create risks if the key generation method is not sufficiently hidden.
- The non-linearity achieved through the AES S-box and chaos-induced transformations currently provides strong but not optimal avalanche effects. The S-box used is static and standard in AES, which might be predictable in some situations. A dynamically generated S-box using chaos-based sequences could increase non-linearity and resistance to algebraic attacks, thus boosting the overall security of the encrypted image. The current setup may show limited diffusion in high-resolution images, where even slight correlations could exist between neighboring pixels.
- RSA only encrypts the AES key and nonce. While this ensures secure key exchange, it faces performance limitations because of its high computational cost, especially during key generation and when encrypting large key material. Future implementations might consider using Elliptic Curve Cryptography (ECC) or post-quantum key exchange algorithms to decrease key size, speed up encryption, and improve scalability.
- To enhance overall security and efficiency, future enhancements may include integrating multi-dimensional chaotic systems, replacing RSA with ECC or post-quantum cryptography, and introducing dynamic S-box generation.

1.3 Thesis Organisation

This thesis is divided into five chapters, each focusing on a different aspect of the project and implementation of the hybrid image encryption system based on Chaos theory, AES-128, and RSA encryption.

Chapter 1: Introduction

This chapter provides an overview of why secure image transmission is important, the motivation for combining different cryptographic techniques, and the goals of the project. It also outlines the scope, limitations, and significance of the proposed system.

Chapter 2: Literature Survey

This chapter reviews existing image encryption techniques, covering classical symmetric and asymmetric methods as well as modern chaos-based and hybrid systems. It compares the strengths and weaknesses of various approaches and highlights the gap that this project aims to fill.

Chapter 3: Research Methodology

This section details the methodology used to develop the encryption and decryption framework. It explains how Chaos-based scrambling confuses pixels, how AES is used for block encryption, and how RSA facilitates secure key exchange. Key generation, preprocessing, encryption, and decryption processes are discussed thoroughly.

Chapter 4: Results and Discussions

This chapter presents the experimental results, including interface screenshots, encryption and decryption outputs, and visual comparisons of original and encrypted images. It assesses performance based on metrics like encryption time, entropy, and correlation coefficients, discussing how well the proposed system works in real-time applications.

Chapter 5: Conclusion and Future Work

The final chapter summarizes the main contributions of the project and emphasizes the benefits of using a multi-layered encryption approach. It also suggests areas for future improvement, such as expanding the system to include video encryption, optimizing it for embedded or mobile devices, and adding post-quantum cryptographic algorithms to improve long-term security.

II. LITERATURE SURVEY

Tong, X., Liu, X., Pan, T., Zhang, M., & Wang, Z. (2024) proposed an image encryption method based on a chaotic map and random scrambling diffusion. The process starts with sparsifying the plaintext using wavelet packet transform (WPT). Then, a one-dimensional chaotic map constructs the measurement and permutation matrices. A bidirectional random scrambling algorithm, based on chaotic magic transformation (CMT), permutes the sparsified plaintext image. After that, the permuted image undergoes compression using the chaotic measurement matrix, followed by diffusion through chaotic pixel diffusion. Finally, the encrypted and compressed image gets embedded in a carrier image using the WPT embedding algorithm [1].

Sahin, M. E. (2023) developed a hybrid image encryption scheme that combines memristive chaotic systems with AES and RSA algorithms to improve data security. The proposed method uses a memristive chaotic map to create highly unpredictable sequences that scramble image data, increasing randomness and confusion. This chaotic pre-processing is followed by AES encryption for efficient pixel-by-pixel data protection, while RSA is used for secure key exchange to ensure confidentiality during transmission. The hybrid model takes advantage of the speed of symmetric encryption, the key security of asymmetric encryption, and the complexity of chaotic behavior to create a strong multi-layered encryption framework. Simulation results showed strong resistance to differential and statistical attacks, along with high entropy and low correlation in the encrypted images, indicating the scheme's effectiveness for secure image applications **[2].**

Ping, H. (2022). looked at how data encryption technologies can improve network information security, especially for protecting sensitive data during transmission. The study assesses various encryption methods, including both symmetric and asymmetric approaches, and analyzes their effectiveness in preventing unauthorized access, tampering, and data breaches. The paper highlights the increasing importance of encryption in modern communication systems, particularly in wireless environments where data is more exposed to interception. By evaluating encryption performance in terms of speed, security, and adaptability, the study offers insights into selecting appropriate techniques for real-time and high-volume data settings. The research concludes that strong encryption not only ensures confidentiality but also builds trust in digital systems, making it a key element in any cybersecurity strategy

Hamza, A., & Kumar, B. (2020) offered a detailed review of three common encryption standards: DES, AES, and RSA. They focused on their underlying structures, performance characteristics, and security levels. The paper outlines how each algorithm has evolved, noting DES as an early standard with known weaknesses, AES as a modern symmetric cipher valued for its speed and strength, and RSA as an important asymmetric algorithm suitable for secure key distribution. Through comparative analysis, the authors look at computational

complexity, key sizes, and how each method stands up to cryptanalytic attacks. The study notes that while AES is often favored for data encryption due to its efficiency, RSA remains vital for secure communication channels where key confidentiality is essential. This review helps researchers and developers grasp the strengths and limitations of each standard, guiding them in choosing the most suitable encryption technique based on specific application needs.

Yasser, I., Khalifa, F., Mohamed, M. A., & Samrah, A. S. (2020) This paper presents a new image encryption technique that uses several chaotic maps to improve the security of digital images. The scheme combines the features of different chaotic systems to create highly unpredictable keys and complex transformation patterns. By using a mixed approach, the method boosts resistance to common attacks, like brute-force and statistical analysis. The authors perform extensive testing of the encryption algorithm, measuring its performance through metrics such as histogram analysis, correlation coefficients, and entropy measures. The results show that the hybrid chaotic system offers strong encryption with effective confusion and diffusion characteristics, making it suitable for secure image transmission. This work adds to the ongoing research on chaos-based cryptography by demonstrating how multiple chaotic maps can be synchronized effectively for high-security encryption uses [5]. .

III. DESIGN AND IMPLEMENTATION

The research methodology describes the clear steps taken to design, implement, and evaluate a hybrid image encryption system that uses Chaos-based algorithms, AES-128, and RSA. This approach ensures a systematic way to process secure image data while keeping it efficient and accurate.

To implement this methodology, the project starts by choosing suitable cryptographic algorithms based on their strengths. Chaotic maps are used for scrambling, AES provides block-level encryption, and RSA is for secure key transmission. A modular system is created using Python and web technologies to bring together each layer of encryption. Simulated testing occurs on grayscale images to evaluate the system's performance in terms of speed, accuracy, and resistance to attacks. The encrypted results are also assessed both visually and statistically to ensure data confidentiality and system strength.

3.1 Design

3.1.1 System Design for Encryption

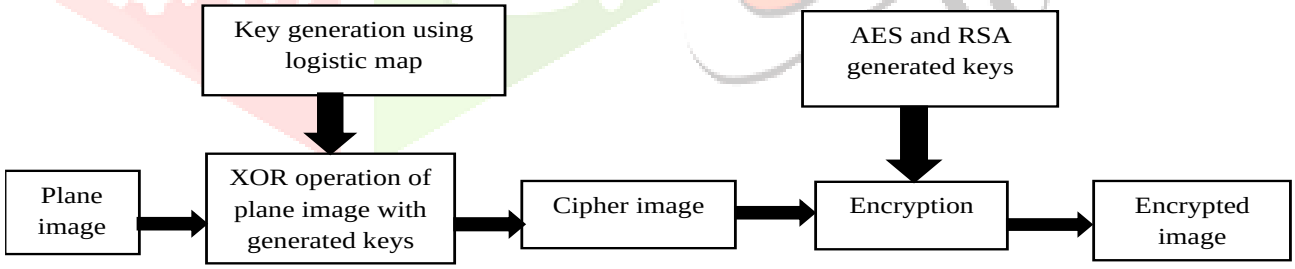

Fig 3. 1 System Design for Encryption

To improve security when sending images, the proposed method encrypts the original image using a combination of RSA, AES, and a chaos-based logistic algorithm. RSA offers strong key security, but it uses a lot of computing power for large data encryption. To enhance performance and speed, AES encryption becomes the main data encryption layer because of its efficiency and fast processing ability. Before applying AES, the image undergoes initial encryption through a chaotic logistic map, which scrambles pixel values

in a way that is very sensitive to initial parameters. This makes the image unrecognizable and helps it resist brute-force and statistical attacks. The AES-encrypted output is further secured by RSA, which encrypts the AES key and nonce. This ensures that only the intended recipient can decrypt the image. The key benefit of this approach is that the image is protected through multiple layers of encryption.

If an attacker manages to bypass one layer, like AES, they still cannot retrieve the image without knowing the chaotic key parameters and the RSA private key. This layered security greatly improves image protection, even against serious threats, and keeps the content confidential during transmission.

In this encryption process, a Bitwise Exclusive OR operation combines a set of image pixels with a public and private key, changing for each pixel set. The cipher keys are generated independently on the sender and receiver sides based on the Chaos and RSA Encryption Key Expansion process. Consequently, only the public key is shared, and the private key remains secret. Experiments with standard benchmark images from the USC-SIPI database indicate that this method provides strong resistance against brute force attacks, key sensitivity tests, and statistical cryptanalysis**.**

3.1.2 System Design for Decryption

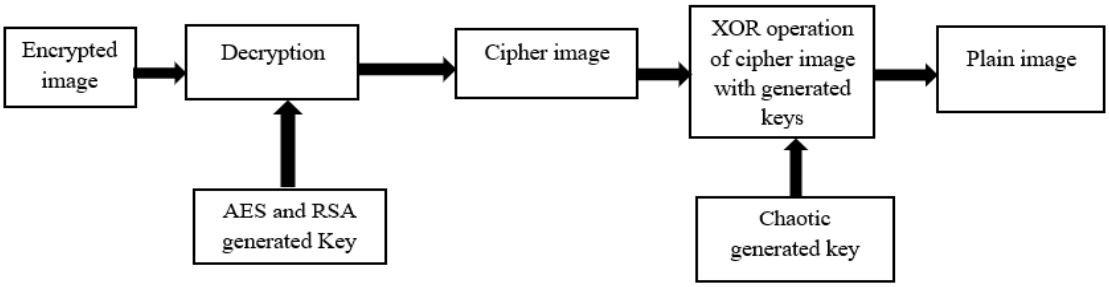

Fig 3. 2 System Design for Decryption

The decryption process starts when the receiver supplies three key inputs: the encrypted image, the RSA private key, and the RSA-encrypted AES credentials, which contain the AES key and nonce. First, the system uses the RSA private key to securely decrypt these credentials. Next, the system uses the recovered AES key and nonce to decrypt the image data, which was originally encrypted in EAX mode.

This stage recovers the chaos-scrambled version of the original image. Once AES decryption is successful, the final step involves applying the opposite operations of the chaotic logistic map.

The chaotic parameters that scrambled the pixels are applied in reverse to correctly reorder the pixel positions and intensities. This structured decryption process ensures that the data remains confidential, safe, and correct. Using RSA for secure key exchange, AES for quick decryption, and Chaos for non-linear scrambling ensures the original image is reconstructed completely, as long as the correct keys are provided. The system is designed to perform these steps automatically, giving users a smooth and secure experience when recovering images.

3.2 Image Encryption Using Chaos, AES, and RSA Algorithm

The proposed system implements a super encryption model that combines the strengths of three powerful encryption techniques—Chaos-based scrambling, AES symmetric encryption, and RSA asymmetric encryption—to provide multi-layered protection for image data. Each layer in this hybrid cryptosystem plays a specific role in ensuring data confidentiality, integrity, and security during transmission or storage.

- Chaos-based encryption offers pixel-level confusion and randomness.
- AES provides high-speed, block-based encryption of image content.
- RSA securely encrypts and transmits the AES key and nonce, ensuring only authorized users can perform decryption.

This layered encryption strategy achieves high computational efficiency, secure key exchange, and robust resistance to cryptographic attacks.

3.3.1 Key Generation

1. RSA Key Generation

- Generate a public-private key pair using two large prime numbers.

Public Key (e, n): Used for encrypting the AES key and nonce.

Private Key (d, n): Used for decrypting the AES key and nonce.

- Ensures secure transmission of session-based AES credentials.

2. AES Key Generation

- A random 128-bit AES key is generated for each encryption session.
- A unique nonce (number used once) is also created for integrity protection in AES EAX mode.
- Both key and nonce are securely encrypted with the RSA public key for protected delivery.

3. Chaos-Based Key Generation

A chaotic system is used to generate a dynamic pseudo-random key stream:

- Select a chaotic map: Logistic, Henon, or Lorenz.
- Initialize the system with secret parameters (initial condition x_0 and control parameter r).
- Iterate the map to generate a chaotic sequence.
- Create the key stream by scaling, quantizing, or transforming the sequence into usable pixel-level encryption values.

These three components RSA keys, AES credentials, and Chaos keys form the foundation for the hybrid encryption process.

3.3.2 Image Preprocessing

Before encryption, the image undergoes transformation to enhance confusion and diffusion, which are vital for secure encryption.

Common preprocessing steps include:

Steps:

1)Pixel Shuffling: Reorder pixel positions using chaotic permutations.

2)Block Permutation: Split the image into blocks and rearrange them randomly.

3)Normalization: Standardize pixel values, if required, for downstream algorithms.

4)Matrix Conversion: Convert image into a numerical array (e.g., NumPy matrix) for manipulation.

This steps breaks visual patterns and prepares the image for chaos-based and AES encryption stages.

3.2.3 Chaos-Based Encryption

The chaos-generated key stream is applied to the image using several transformation methods:

- XORing: Each pixel value is XORed with the corresponding chaotic value.
- Addition/Subtraction Modulo 256: Ensures pixel values remain in valid grayscale range.
- Substitution: Replace pixel values using a chaotic substitution table (e.g., S-box).
- Combination: Apply multiple chaotic operations in a sequence for enhanced complexity.

These steps create a scrambled image that loses any direct visual resemblance to the original content.

3.3.4 AES Encryption (Second Layer Encryption)

Once the chaos scrambling is complete, the resulting image matrix is encrypted using AES-128 in EAX mode:

- The AES key and nonce generated earlier are used.
- AES operates on 128-bit blocks, applying 10 rounds of:
    a) SubBytes (nonlinear substitution)
    b) ShiftRows (permutation)
    c) MixColumns (column mixing)
    d) AddRoundKey (XOR with round keys)

- EAX mode ensures:

Confidentiality through encryption and Integrity through an authentication tag

The output of this phase includes:

Ciphertext (encrypted image data)

Authentication Tag

Nonce (used once for session security)

3.3.5 RSA Encryption (Key Protection Layer)

The final encryption step is to secure the AES credentials:

- Encrypt AES Key and Nonce using RSA public key.
- This prevents unauthorized access, as only the recipient with the RSA private key can decrypt the AES key and nonce.
- The encrypted AES credentials are packaged alongside the encrypted image.

Image Encryption Algorithm (Chaos + AES + RSA)

Input:

Original image I

RSA public key (e, n)

Output:

Encrypted image C

Encrypted AES key and nonce (RSA_enc)

AES authentication tag (Tag)

begin

Step 1: Preprocessing

Convert image I to grayscale if colored

Normalize and reshape I as a flat array or matrix

Step 2: Chaos Key Stream Generation

Select chaotic map (e.g., Logistic Map)

Set initial values (x_0, r)

Generate chaotic sequence Kc

Scale Kc to match pixel range $[0, 255]$

Step 3: Chaos-Based Scrambling

for each pixel i in I do

I_chaos[i] ← I[i] XOR Kc[i]

end for

Step 4: AES Key Generation

Generate random 128-bit AES key K_AES

Generate unique nonce N_AES

Step 5: AES Encryption

Encrypt I_chaos using AES-128 (EAX mode) with K_AES and N_AES

Obtain:

Ciphertext C

Authentication tag Tag

Step 6: RSA Encryption of AES Credentials

Concatenate AES key and nonce → K_pack

Convert K_pack to integer m

RSA_enc ← $m\text{^}e \ mod \ n$

Output C, RSA_enc, Tag

End

Encryption Time Overview

The encryption time is the duration needed to change the original image into its encrypted form using the hybrid Chaos, AES, and RSA algorithm. The times vary slightly with image size, but they remain efficient. For example, encrypting a 1 KB image takes about 3.25 ms, while a larger file of 2 MB takes around 4.60 ms. These results show that the encryption method scales well and is suitable for real-time applications, even for medium to large images.

Table 3. 1Encryption Time Table

Image Size	Encryption Time (ms)
1 KB	3.25
10 KB	3.99
100 KB	2.57
1 MB	3.46
2 MB	4.60

3.3 Image Decryption Using Chaos, AES, and RSA Algorithm

Decryption is the reverse of the encryption pipeline. It proceeds as follows:

Step 1: RSA Decryption

- Use the RSA private key to decrypt the AES key and nonce.
- This step is mandatory before AES decryption can occur.

Step 2: AES Decryption

- Use the decrypted AES key and nonce to perform EAX-mode decryption.
- The ciphertext is decrypted to retrieve the chaos-scrambled image.
- The authentication tag is validated to ensure the image has not been tampered with.

Step 3: Chaos-Based Decryption

- Regenerate the same chaotic key stream using the original seed and parameters.
- Apply the inverse operations:

Reverse substitution

Undo pixel permutation

Reverse XOR or modulo transformations

This fully restores the original image matrix from its scrambled form.

Step 4: Image Reconstruction

- Convert the final decrypted matrix back into image format (e.g., using OpenCV or PIL).
- The user receives the original, secure image without quality loss or artifacts.

This triple-layer encryption model achieves strong security through nonlinearity, key diversity, and authenticated encryption. Even if one layer is compromised, the remaining layers maintain the overall security of the image data.

Image Decryption Algorithm (RSA + AES + Chaos)

Input: Encrypted image C

Encrypted AES key and nonce (RSA_enc)

AES authentication tag (Tag)

RSA private key (d, n)

Output: Decrypted original image I_recovered

begin

Step 1: RSA Decryption of AES Credentials

m ← RSA_enc^$d\ mod\ n$

Extract AES key K_AES and nonce N_AES from m

Step 2: AES Decryption

Decrypt ciphertext C using AES-128 in EAX mode with K_AES and N_AES

Validate Tag for integrity check

Output: Chaos-scrambled image I_chaos

Step 3: Chaos Key Stream Regeneration

Select same chaotic map and parameters (x_0, r)

Regenerate chaotic sequence Kc

Scale Kc to match pixel range $[0, 255]$

Step 4: Reverse Chaos-Based Decryption

for each pixel i in I_chaos do

I_recovered[i] ← I_chaos[i] XOR Kc[i]

end for

Output I_recovered

End

Decryption Time Overview

Table 3. 2 Decryption Time

Image Size	Decryption Time(ms)
1 KB	1.13
10 KB	1.06
100 KB	1.12
1 MB	1.00
2 MB	1.14

Decryption time measures how long it takes to restore the original image from the encrypted data using RSA decryption, AES reversal, and Chaos unscrambling. The times are consistently low across all sizes. For instance, decrypting a 1 KB image takes just 1.13 ms, and a 2 MB image takes 1.14 ms. This shows that the decryption process is quick and efficient. Such performance ensures a smooth and secure user experience in practical use.

3.5 Security

The hybrid encryption approach in our project integrates the strengths of RSA, AES, and Chaos-based encryption to ensure a high level of security for image data.

- RSA secures the AES key and nonce using asymmetric encryption. Since RSA relies on the computational difficulty of factoring large prime numbers, the key exchange process remains protected from brute-force and key interception attacks.
- AES (Advanced Encryption Standard) is used to encrypt the image data after it has been scrambled using chaotic transformations. AES-128 provides strong resistance to known cryptographic attacks, including linear and differential cryptanalysis. It also ensures fast and secure encryption of large data blocks with minimal performance overhead.
- Chaos-based Encryption enhances the unpredictability and randomness of the encryption process. Due to its sensitivity to initial conditions and the non-linear behavior of chaotic systems, even a slight change in the key or image leads to completely different encrypted outputs. This makes statistical and brute-force attacks nearly impossible.

By combining these three layers, the system achieves:

- Strong key confidentiality
- High data integrity and confidentiality
- Resistance to cryptanalysis and unauthorized access

This layered structure ensures that even if one component is compromised, the overall system retains its security, making it highly suitable for sensitive applications like secure image transmission and digital archiving.

IV. RESULTS AND DISCUSSIONS

This chapter highlights the performance of the hybrid image security system. It combines chaotic scrambling, AES-128 symmetric encryption, and RSA asymmetric encryption to protect images in multiple layers. We carried out simulations to validate each part of the system. These tests focused on user interaction, encryption accuracy, and decryption reliability. We also assessed the web-based platform for usability, response time, and data security. Key components include user authentication, encryption workflow, key

management, and image restoration, along with supporting results. This chapter shows how the system functions in a controlled environment and confirms that each module operates as expected.

4.1 Home Page Dashboard

Describes application functionality with navigation to Encrypt and Decrypt pages. The Fig 4.1 Home Page Dashboard is the main interface of our image security web application. It gives users a clear overview of the system's purpose, features, and navigation options. This design focuses on being more informative and easy to use. This allows for smooth access to the primary functions: encryption and decryption

Fig 4. 1 Home Page Dashboard

4.2 Login & Registration Pages

Secure login and account creation with form validation. The Login and Registration module is the primary security layer of our web-based image encryption and decryption system.

As shown in Fig 4.2 Login Page it ensures that only verified users can access features related to uploading, encrypting, and decrypting sensitive images.

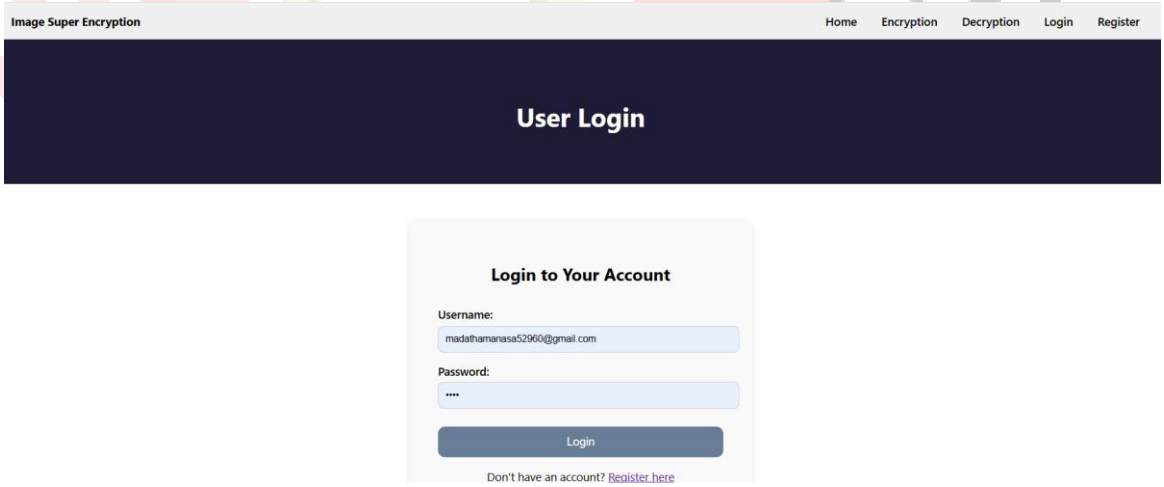

Fig 4. 2 Login Page

4.3 Encryption Output

In this phase, the uploaded image goes through layered encryption using Chaos-based scrambling, AES-128, and RSA. First, a chaotic algorithm scrambles the image to eliminate visual patterns. Next, AES-128 in EAX mode encrypts the pixel data. This process guarantees both confidentiality and integrity. The complete encryption process and user interface for this stage are illustrated in Fig. 4.3. The AES key and nonce are encrypted with RSA algorithm for safe key transmission. Users can download all outputs, including the encrypted image, RSA keys, and encrypted AES credentials, for future decryption.

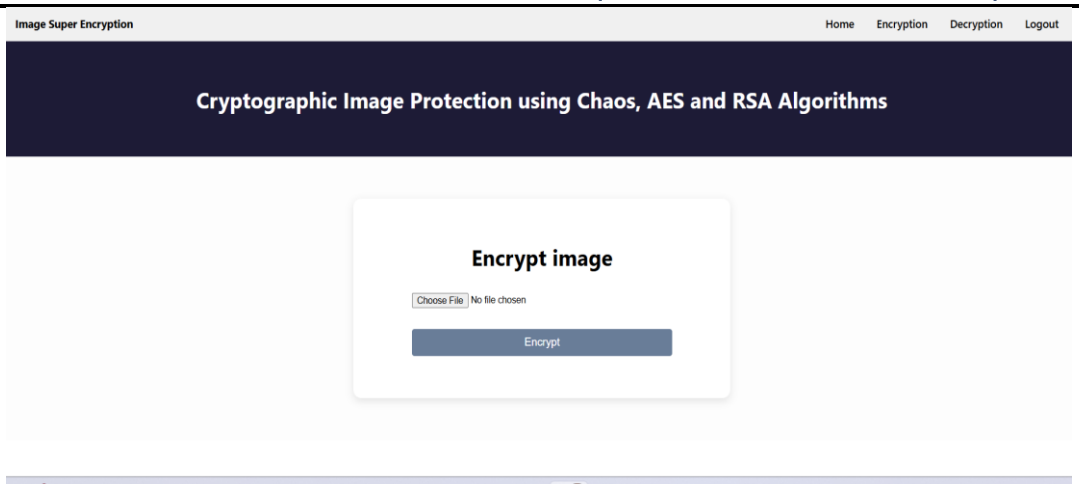

Fig 4. 3 Encryption Page

After encryption:

Once the encryption process is complete, the system securely stores the encrypted image, the generated AES key, and the nonce. These files can be downloaded. The web interface also offers a visual preview, which confirms that encryption was successfully applied. As illustrated in Fig. 4.4 Encrypted Image Page.

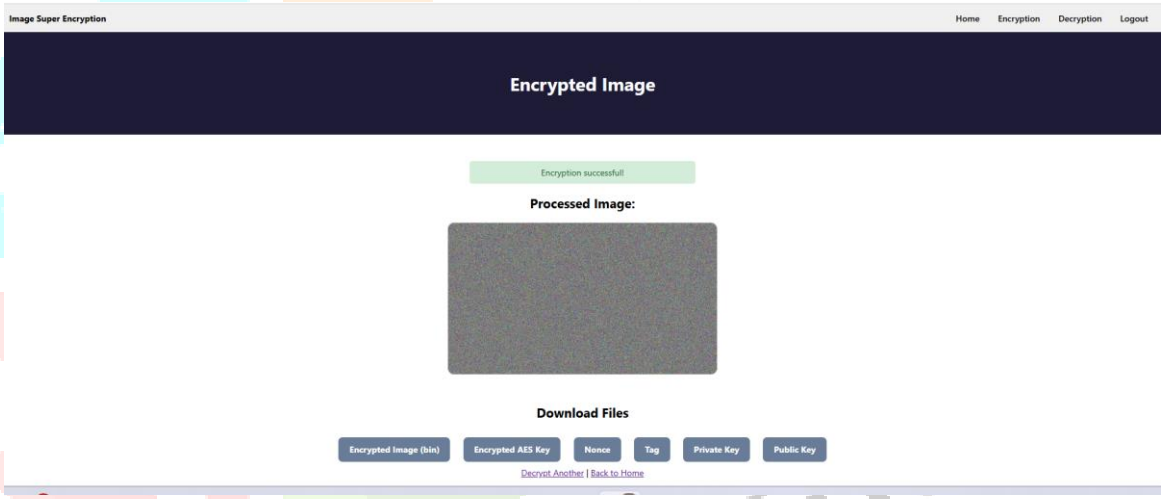

Fig 4. 4 Encrypted Image Page

4.4 Decryption Output

In the decryption phase, users upload the previously encrypted image along with the RSA private key, the encrypted AES key, and nonce. The RSA key decrypts the AES credentials, which are then used to reverse the AES-128 encryption. After that, the chaos-based scrambling is inverted to restore the image to its original form.

A checksum verification confirms that the image's integrity, and the original image is displayed. The system also uses the flash alerts to notify users of successful decryption or any errors that occur during the process.

This secure decryption module guarantees that only authorized users can recover the image with the right keys. It supports strong protection, usability, and real-time feedback across multiple sessions or users.

As illustrated in Fig. 4.5 Decryption Page users upload encrypted image files along with keys. The system checks credentials, decrypts the files using RSA and AES, reverses chaos scrambling, reconstructs the original image, and shows success or error alerts.

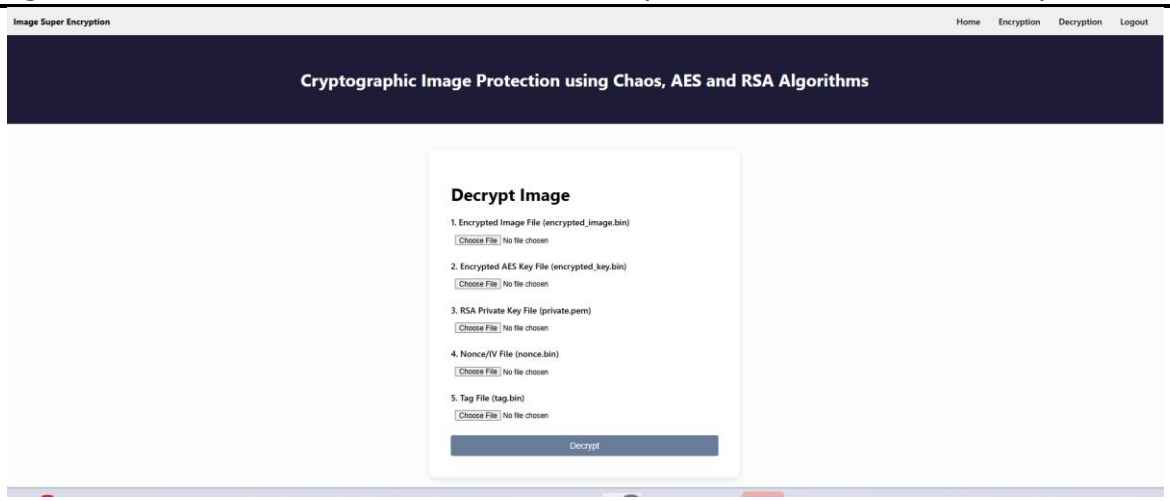

Fig 4. 5 Decryption Page

In our project, decryption begins when the user uploads the encrypted image, the AES key, and the RSA private key. The system first decrypts the AES key and nonce using the RSA private key. Next, it uses the decrypted AES credentials to perform AES decryption and retrieve the chaos-scrambled image. Finally, it applies reverse chaotic operations to accurately reconstruct the original image as illustrated in Fig. 4.6 Decryption Image Page. This process ensures secure and complete image recovery.

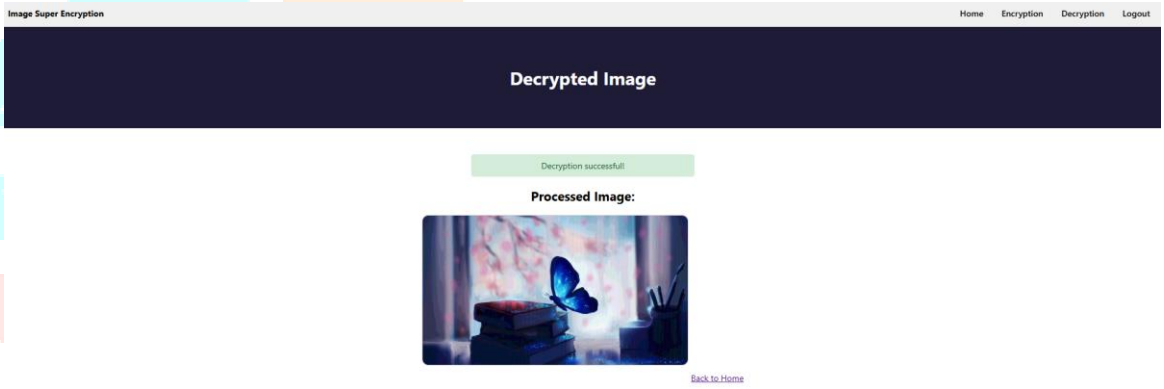

Fig 4. 6 Decryption Image Page

The decryption process brings back the original image by reversing all encryption steps. After the user uploads the encrypted file and keys, the system decrypts and verifies the data securely. It checks for correctness through internal reviews and visual output, providing a simple and reliable method to recover the image. This stage completes the secure image cycle, ensuring both usability and data protection.

V . CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

In today's digital world, where sharing and storing images across networks is common, the need for secure image transmission is critical. Traditional single-layer encryption methods often fail to protect against complex cyber threats. This is especially true for applications that need real-time confidentiality, such as medical imaging, defense systems, and cloud services.

To tackle these issues, this project presents a hybrid image encryption framework. This framework combines Chaos-based scrambling, AES symmetric encryption, and RSA asymmetric key protection into a multi-layered super encryption model. Each layer in this design serves aa a specific security function:

- Chaos encryption adds non-linear confusion, disrupting visual patterns. This makes the image content unrecognizable even before applying standard encryption.
- AES offers the fast, block-wise symmetric encryption. It securely transforms the image while keeping computational efficiency high.
- RSA encryption acts as the outer security layer. It encrypts the AES key and nonce to ensure the secure key exchange between the sender and the recipient using public-key infrastructure.

This three-tier encryption system ensures the data confidentiality, integrity, and key security. It also shows strong resistance against various attacks, including brute-force, statistical, and the differential cryptanalysis. By merging chaotic dynamics with established cryptographic standards, the system offers a reliable and secure image protection method.

Additionally, the system's modular design supports real-time encryption and decryption. Thus, it is suitable for practical applications in fields such as the Healthcare systems (e.g., protecting radiological or diagnostic images), Military surveillance and intelligence and Encrypted image sharing on communication platforms

Overall, this project demonstrates how effective multi-layer encryption techniques can secure multimedia data and encourages the development of advanced cryptographic tools for today's digital vulnerabilities.

4.2 Future Work

To further enhance the strength, flexibility, and efficiency of the proposed system, the following improvements are planned:

- **Exploration of Diverse Chaotic Maps**

Implement and assess different chaotic systems (e.g., Tent Map, Chebyshev Map, or Rossler Attractor) to see how they affect the randomness and security of the encryption process. The goal is to identify the most efficient chaotic functions regarding entropy and computational load.

- **Optimization for Real-Time Systems**

Integrate optimization algorithms or hardware acceleration (e.g., GPU support, CUDA, or parallel processing) to minimize encryption and decryption latency for high-resolution images. This will enable use in low-latency or edge-computing settings.

- **Advanced Key Management and Rotation Mechanism**

Create automated AES key rotation protocols, manage dynamic nonces, and develop secure RSA key update methods. This will strengthen long-term system security, especially in persistent storage or streaming applications.

- **Formal Security Verification**

Conduct mathematical analysis and formal proofs (e.g., using AVISPA or ProVerif) to validate the cryptographic strength and durability of the system against various attack models. This should include modeling threats like chosen-ciphertext attacks or man-in-the-middle scenarios.

- **Cross-Platform and Mobile Integration**

Develop a lightweight, optimized version of the encryption system for mobile and cross-platform applications. This will make secure image encryption available to users on smartphones and embedded systems. Such portability expands accessibility in real-time communication and low-resource environments. A design that uses less battery and memory will improve performance on handheld devices. A simpler user interface can improve usability across different screen sizes.

REFERENCES

[1] Tong, X., Liu, X., Pan, T., Zhang, M., & Wang, Z. (2024). A visually meaningful secure image encryption algorithm based on conservative hyperchaotic system and optimized compressed sensing. *Multimedia Systems*, *30*(3). https://doi.org/10.1007/s00530-024-01370-4

[2] Sahin, M. E. (2023). Memristive chaotic system-based hybrid image encryption application with AES and RSA algorithms. *Physica Scripta*, *98*(7), 075216. https://doi.org/10.1088/1402-4896/acdba0

[3] Ping, H. (2022). Network Information Security Data protection based on data encryption technology. *Wireless Personal Communications*, *126*(3), 2719–2729. https://doi.org/10.1007/s11277-022-09838-0

[4] Hamza, A., & Kumar, B. (2020). A Review Paper on DES, AES, RSA Encryption Standards. *9th International Conference System Modeling and Advancement in Research Trends (SMART) (Pp. 333-338). IEEE.*, 1–6. https://doi.org/10.1109/smart50582.2020.9336800

[5] Yasser, I., Khalifa, F., Mohamed, M. A., & Samrah, A. S. (2020). A new image encryption scheme based on hybrid chaotic maps. *Complexity*, *2020*, 1–23. https://doi.org/10.1155/2020/9597619

[6] Al-Kadei, F. H. M. S., Mardan, H. A., & Minas, N. A. (2020). Speed up image encryption by using RSA algorithm. *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 1302–1307. https://doi.org/10.1109/icaccs48705.2020.9074430

[7] Alsaffar, D. M., Almutiri, A. S., Alqahtani, B., Alamri, R. M., Alqahtani, H. F., Alqahtani, N. N., Alshammari, G. M., & Ali, A. A. (2020). Image Encryption Based on AES and RSA Algorithms. *In Proc. 2020 3rd Int. Conf. On Computer Applications & Information Security (ICCAIS)*, 1–5. https://doi.org/10.1109/iccais48893.2020.9096809

[8] Mou, J., Yang, F., Chu, R., & Cao, Y. (2019). Image compression and encryption algorithm based on hyper-Chaotic MAP. *Mobile Networks and Applications*, *26*(5), 1849–1861. https://doi.org/10.1007/s11036-019-01293-9

[9] Ping, P., Fu, J., Mao, Y., Xu, F., & Gao, J. (2019). Meaningful encryption: Generating visually meaningful encrypted images by compressive sensing and reversible color transformation. *IEEE Access*, *7*, 170168–170184. https://doi.org/10.1109/access.2019.2955570

[10] Zhu, L., Song, H., Zhang, X., Yan, M., Zhang, L., & Yan, T. (2019). A novel image encryption scheme based on nonuniform sampling in block compressive sensing. *IEEE Access*, *7*, 22161–22174. https://doi.org/10.1109/access.2019.2897721

[11] Advani, N., Rathod, C., & Gonsai, A. M. (2018). Comparative study of various cryptographic algorithms used for text, image, and video. In *Advances in intelligent systems and computing* (pp. 393–399). https://doi.org/10.1007/978-981-13-2285-3_46

[12] Kumar, B. J. S., Raj, V. R., & Nair, A. (2017). Comparative study on AES and RSA algorithm for medical images. *In Proc. 2017 Int. Conf. On Communication and Signal Processing (ICCSP), Chennai, India, 2017*, 0501–0504. https://doi.org/10.1109/iccsp.2017.8286408

[13] Azam, N. A. (2017). A novel fuzzy encryption technique based on multiple right translated AES gray S-Boxes and phase embedding. *Security and Communication Networks*, *2017*, 1–9. https://doi.org/10.1155/2017/5790189

[14] Zhang, Q., & Ding, Q. (2015). Digital Image Encryption Based on Advanced Encryption Standard (AES). *)," in Proc. 2015 Fifth Int. Conf. On Instrumentation and Measurement, Computer, Communication and Control (IMCCC), Qinhuangdao, China, 2015*, 1218–1221. https://doi.org/10.1109/imccc.2015.261

[15] Murillo-Escobar, M., Cruz-Hernández, C., Abundiz-Pérez, F., López-Gutiérrez, R., & Del Campo, O. A. (2014). A RGB image encryption algorithm based on total plain image characteristics and chaos. *Signal Processing*, *109*, 119–131. https://doi.org/10.1016/j.sigpro.2014.10.033

[16] Kumar, N., & Agrawal, S. (2013). "A technical review on Symmetric Key Cryptography Algorithm on Images,." *Int. J. Adv. Res. Comput. Sci. Softw. Eng., 2013.*