**A   
Mini Project Report**

**on  
"** **IMAGE ENCRYPTION USING RSA"**

*Submitted in partial fulfilment of requirement for the award of the of*

**MASTER OF TECHNOLOGY**I­n **COMPUTER SCIENCE AND ENGINEERING**

**(CYBER SECURITY)**

Submitted By **B AKHIL AKASH 1005-23-742503**

Under the Guidance of

**A.GAYATHRI**

Assistant Professor

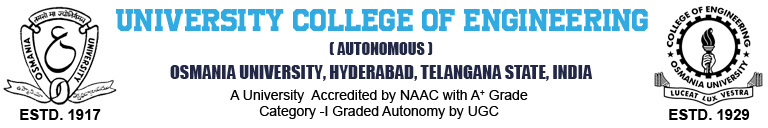
****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**UNIVERSITY COLLEGE OF ENGINEERING(AUTONOMOUS)**

**OSMANIA UNIVERSITY, HYDERABAD**

**JULY – 2024**

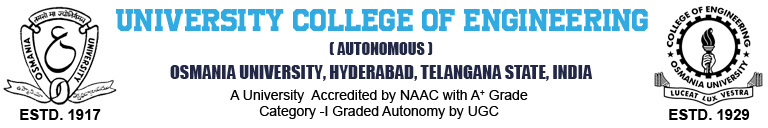


****

**CERTIFICATE**

This is to certify that the thesis entitled **"IMAGE ENCRYPTION USING RSA",** submitted by **B AKHIL AKASH** Bearing **ROLL NO 100523742503**, student of Department of computer science engineering, University college of engineering, Osmania university, Hyderabad, is a record of bonafide research work under my supervision and I consider it worthy of consideration for the award of the degree of Master of Technology of the Institute.

|  |  |
| --- | --- |
| Project Guide (Dept. of CSE)  **Asst. Prof. A.GAYATHRI**  Osmania University | **Head of Department, CSE**  **Prof. Dr. P.V. Sudha**  Osmania University |



****

**DECLARATION**

I **B AKHIL AKASH** bearing Roll No. **1005-23-742503**, hereby declare that the results presented in this project is the bonafide work done and carried out by me during the year 2024 in partial fulfilment of the **academic** requirements for the award of **"Master of Technology"** in Computer Science and Engineering, Osmania University, Hyderabad, TELANGANA(INDIA).This project was done under the supervision of **A.GAYATHRI** Asst. Professor.

Further, I declare the report has not been submitted to any other University for the award of any other degree.

|  |  |
| --- | --- |
| **Date:**  **Place: Hyderabad** | **Name and Signature of The Student**  **B AKHIL AKASH (1005-23-742503)** |

**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be incomplete the people who made it possible and whose constant guidance and encouragement crowns all efforts with success. They have been a guiding and source of inspiration towards the completion of the project.

It is my privilege and pleasure to express profound sense of respect, gratitude and indebtedness to my project guide, **A GAYATHRI,** Assistant Professor, Department of Computer Science and Engineering, **OSMANIA UNIVERSITY,** who have supported us throughout my project with patience and knowledge and guided me with valuable inputs and suggestions & indefatigable inspiration, guidance, cogent discussion, constructive criticisms, and encouragement throughout this dissertation work.

I express my sincere gratitude to Head of the Dept **Prof. Dr. P.V SUDHA** of Computer Science and Engineering, **OSMANIA UNIVERSITY,** for her suggestions, motivations and providing excellent infrastructure and a conducive atmosphere for completing this project successfully. I convey my heartfelt thanks to the lab staff for allowing me to use the required equipment whenever needed.

**B AKHIL AKASH (1005-23-742503)**

|  |  |  |
| --- | --- | --- |
|  | **Table of Contents** |  |
| **Chapter** | **Title** | **Page No.** |
|  | Abstract | 1 |
| 1 | Introduction | 2 |
|  | 1.1 Scope of the Project | 2 |
|  | 1.2 Objective | 2 |
|  | 1.3 Problem Definition | 2 |
|  | 1.4 Significance of Image encryption and Decryption | 2 |
| 2 | Literature Survey | 3-4 |
| 3 | System Requirements | 5 |
|  | 3.1 Software Requirements | 5 |
|  | 3.2 Hardware Requirements | 5 |
| 4 | System devlopment |  |
|  | 4.1 Rsa Algorithm | 6 |
|  | 4.2 Steps of Rsa Algorithm | 7-9 |
|  | 4.3 Rsa Algorithm Illustration | 10-18 |

|  |  |  |
| --- | --- | --- |
|  | **Table of Contents** |  |
| **Chapter** | **Title** | **Page No.** |
|  | 4.4 Proposed Algorithm | 19-20 |
|  | 4.5 Advantages Of Rsa Algorithm | 21 |
|  | 4.6 Miller-Rabin Test | 21 |
|  | 4.6 Encryption | 22 |
|  | 4.7 Decryption | 23 |
| 5 | Performance Analysis | 24 |
|  | 5.1 Practical Work | 25-27 |
|  | 5.2 screenshots of Implementation. | 28-33 |
| 6 | Conclusion | 34 |
|  | References | 35 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **LIST OF FIGURES** | | |
| **Fig. No.** | **Title** | **Page No.** |
| 1 | Encryption and Decryption | 20 |
| 2 | Plain text to cipher text | 20 |
| 3 | Encryption | 22 |
| 4 | Decryption | 23 |
| 5 | Image used for Encryption and after decryption | 24-25 |
| 6 | Screenshots of Implementation | 28-32 |

# ABSTRACT

In today's world, secure online data transmission takes precedence over other activities. Various algorithms exist to provide the computational difficulty that makes cracking the key to identify a unique message difficult. Many researchers have applied various cryptographic algorithms for secure data transmission, and various hybrid cryptographic algorithms have been proposed to improve the level of information security. Key management plays an important role in implementing cryptographic algorithms. For this reason, we applied image encryption technology that uses a random image as a key. I used a random image as a key and encrypted another image as information using the RSA algorithm. The proposed method is compared with conventional approaches and concludes that encryption algorithms implemented using images as keys offer more security in terms of encryption and decryption times.

Keywords: Rsa ,Encryption,Decryption, Cryptographic

# 1.INTRODUCTION

#### 1.1 Scope of the Project

This project delves into the realm of image encryption and decryption utilizing the RSA algorithm, aiming to secure digital images against unauthorized access. The project's scope includes designing an encryption system, implementing the RSA algorithm, and validating the system's effectiveness through rigorous testing and analysis. The project encompasses both software and hardware requirements and addresses potential security vulnerabilities.

#### 1.2 Objective

The primary objective of this project is to develop a robust encryption and decryption system for digital images using the RSA algorithm. This entails creating secure encryption keys, implementing the encryption process to convert images into unreadable formats, and designing a decryption process to revert images to their original state for authorized users. The objective is to ensure data confidentiality, integrity, and authenticity in the storage and transmission of digital images. These are ways to protect digital images to meet your security goals. Confidentiality, Integrity and Availability (CIA). The RSA algorithm is the foundation of cryptosystems (a set of cryptographic algorithms used for a particular security service or purpose), enables public-key cryptography, and is widely used to protect sensitive data. such as the Internet. The purpose of image encryption schemes is to obtain the highest quality hidden image to keep the information secret.

#### 1.3 Problem Definition

With the proliferation of digital images across various platforms, ensuring their security and privacy has become a paramount concern. Unauthorized access, alteration, and theft of digital images can lead to significant privacy breaches and misuse. This project addresses these challenges by employing the RSA algorithm for image encryption, thereby safeguarding images from unauthorized access and ensuring that only authorized individuals can decrypt and access the original images.

#### 1.4 Significance of Image Encryption and Decryption

Image encryption and decryption are crucial for protecting sensitive visual information from unauthorized access and tampering. By converting images into encrypted formats, the project ensures that even if images are intercepted during transmission or storage, they remain inaccessible without the correct decryption key. This enhances the security of personal, corporate, and governmental visual data, preventing data breaches and maintaining privacy

## 2.LITERATURE SURVEY

Information security is a serious issue for modern online applications. When data is transmitted via online media, it must be protected against unauthorized access. Many researchers have proposed many security algorithms and hybrid approaches to information security. Faster algorithms and high-performance computing techniques provide ways to decrypt data even after security algorithms are implemented. Therefore, various researchers are interested in image security. Analyzing network traffic involves managing various resources where bandwidth plays a critical role. Researchers have proposed a dynamic bandwidth management technique for on-demand services. Since images require more bandwidth than textual information, the proposed operations can be integrated with dynamic resource management technology to provide faster and more secure services. Today, mixed media data travels extensively across the web in various structures such as images, audio, video, and text. Computer communication via the Internet makes all information transparent and available to all customers. Data security is therefore an important and critical task. As data moves from one place to another over the Internet, it must be protected from unauthorized access. So, the most important thing is to transfer your photos safely. There are multiple ways to store images, including encryption, watermarking, custom watermarking, cryptanalysis, and steganography**.** RSA encryption of images typically involves treating the image data as a stream of bytes or pixels. Each pixel value or block of pixels can be encrypted using the recipient's public key, ensuring that only the holder of the corresponding private key can decrypt the image back to its original form. This methodical approach ensures confidentiality during transmission and storage, essential in scenarios where sensitive or proprietary image data must be securely communicated.

Performance considerations play a crucial role in the practical implementation of RSA for image encryption. The computational overhead associated with RSA encryption increases with larger key sizes, impacting encryption and decryption speeds, especially when dealing with high-resolution images or real-time applications. Techniques such as hybrid encryption, where RSA is used in conjunction with symmetric algorithms like AES (Advanced Encryption Standard), help mitigate these performance issues by leveraging the efficiency of symmetric encryption for bulk data while using RSA for secure key exchange and data integrity verification.

Security analysis forms another critical aspect of the literature survey, evaluating the robustness of RSA-based image encryption against various cryptographic attacks. While RSA is renowned for its security based on the difficulty of factoring large numbers, vulnerabilities such as chosen-plaintext attacks or exploits leveraging known image patterns require careful consideration. Research efforts focus on enhancing RSA's resilience to these threats through improved key management practices, randomization techniques, and adaptive encryption schemes tailored to the characteristics of image data.

| **S.No** | **Title** | **Methodology / Algorithm** | **Pros** | **Cons** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | A new approach for image encryption in the modified RSA cryptosystem using MATLAB | Utilized a modified RSA algorithm in MATLAB for image encryption | **Pros**: Enhanced security by modifying traditional RSA | **Cons**: Complexity in implementation | **2016** |
| 2 | Methods of network security and improving the quality of service—a survey | Survey on various network security methods and QoS improvement techniques | **Pros**: Comprehensive analysis of security techniques | **Cons**: Lacks practical implementation details | **2015** |
| 3 | RSA cryptography algorithm using linear congruence class | Proposed the use of linear congruence class in RSA cryptography | **Pros**: Improved computational efficiency | **Cons**: Limited to theoretical analysis | **2016** |
| 4 | In-depth packet inspection using a hierarchical pattern matching algorithm | Employed hierarchical pattern matching for packet inspection | **Pros**: Increased accuracy in detecting malicious packets | **Cons**: High computational overhead | **2010** |
| 5 | A new authentication scheme with elliptical curve cryptography for IoT environments | Introduced an authentication scheme using elliptical curve cryptography for IoT | **Pros**: Enhanced security in IoT environments | **Cons**: Requires high computational power | **2018** |
| 6 | Network traffic management using dynamic bandwidth on demand | Discussed dynamic bandwidth management techniques for network traffic | **Pros**: Efficient bandwidth utilization | **Cons**: Implementation complexity | **2017** |

**3.System Requirements:**

### 3.1SOFTWARE REQUIREMENT:

The project requires a robust development environment with the following software components:

Programming language: Python

Integrated Development Environment (IDE): PyCharm, or Visual Studio Code

Libraries: Cryptography libraries such as PyCryptodome for Python

Image processing libraries: OpenCV

### 3.2HARDWARE REQUIREMENTS:

The system should be implemented on hardware with sufficient processing power and storage capacity:

Processor: Intel i5 or higher

RAM: 4 GB or higher

Storage: 500 GB HDD or SSD

Graphics: Dedicated graphics card for handling image processing tasks

.

**4.SYSTEM DEVELOPMENT**

**4.1 RSA Algorithm**

Developed by Ron Rivest, Adi Shamir, and Leonard Adleman in 1977, the RSA algorithm is one of the most widely used asymmetric key encryption algorithms. Naming is based on the last name of the developer. This algorithm enables both information security and authentication.

Public key cryptosystems are primarily used to protect data in transit. RSA is an asymmetric key encryption algorithm that uses a public/private key pair. The sender encrypts the message with the public key and the recipient decrypts the message with the recipient's private key.

The reason for using the RSA algorithm is the main problem with factoring large integers. Security comes from the product of two large prime numbers in implementing the algorithm. The most complicated part of RSA encryption is the generation of public and private keys. Two prime numbers, p and q, are generated using the Rabin-Miller primality test algorithm. The connection between private and public keys is established by two prime numbers. Key sizes are often expressed in bits.

**4.2Steps of RSA Algorithm**

The RSA algorithm involves three major steps during implementation. They are as follows:

Generation of key

Encryption

Decryption.

# Key Generation

The first stage of the RSA algorithm is key generation, which includes public and private generation. As the name suggests, the public key is the public key that is visible to everyone and is used to participate in encrypting messages. Image transmissions are encrypted with your public key and can be decrypted with your private key. A key for the RSA algorithm can be created using the following steps:

First, select the two different prime numbers that are p and q.

For safety, prime integers p and should be selected with the same bit-length. Prime integers are efficiently found by primality testing.

Then, calculate the value of n that is n=pq.

n is the modulus that is used for equally the public and private keys. Its length is known as key length that is usually stated in bits.

Then compute Euler’s totient of n.

ϕ(n)=ϕ(p)ϕ(q) =(p−1) (q−1) =n−(p+q−1);

here, ϕ is Euler’s totient function. This rate is kept private

Then, choose an integer e that 1 <e<ϕ(n) and gcd (e,ϕ(n)) =1; i.e., e and ϕ(n) are coprime.

e is out as a key which is kept public.

e has a brief bit-length and slight Hamming weight outcomes in more effective encryption. However, minor e values have been published to become less locked in some settings.

Determine d as d≡e−1(mod ϕ(n)),

i.e., d which is the modular multiplicative inverse of e (modulo ϕ(n)). This is performed as, solve d given by d·e≡1(mod ϕ(n)).

That is calculated using an extended Euclidean algorithm. It uses the pseudo-code in the modular integers section; inputs a and n correspond to e and ϕ(n), respectively.

Evaluate the value of d which is kept as the private key. The public key involves the modulus of n and e. The private key has the modulus of n and d, and it is kept secret. p,q, and ϕ(n) values are kept secret because these values can be used for calculating d.

# Encryption

***c≡me(mod n) c=cipher text m=plain text e=public key d=private key***

# C.Decryption

***m≡cn(mod n) c=cipher text m=plain text e=public key d=private key***

# 4.3RSA Algorithm Illustration

This method is the traditional method used for encryption and decryption using text or numeric keys. Here I proposed the same algorithm as the previous one with the simple modification of using images as keys instead of text or numbers as keys.The following is our proposed algorithm:

# Let's take some examples of RSA encryption algorithm:

**Example 1:**

This example shows how we can encrypt plaintext 9 using the RSA public-key encryption algorithm. This example uses prime numbers 7 and 11 to generate the public and private keys.

Explanation:

Step 1: Select two large prime numbers, p, and q.

p = 7

q = 11

Step 2: Multiply these numbers to find n = p x q, where n is called the modulus for encryption and decryption.

First, we calculate n = p x q

n = 77

Step 3: Choose a number e less than n, such that n is relatively prime to (p - 1) x (q -1). It means that e and (p - 1) x (q - 1) have no common factor except 1. Choose "e" such that 1<e < φ (n), e is prime to φ (n), gcd (e, d (n)) =1.

Second, we calculate

φ (n) = (p - 1) x (q-1)

φ (n) = (7 - 1) x (11 - 1)

φ (n) = 6 x 10

φ (n) = 60

Let us now choose the relative prime e of 60 as 7.

Thus, the public key is <e, n> = (7, 77)

Step 4: A plaintext message m is encrypted using public key <e, n>. To find ciphertext from the plain text following formula is used to get ciphertext C.

To find ciphertext from the plain text following formula is used to get ciphertext C.

C = me mod n C = 97 mod 77

C = 37

Step 5: The private key is <d, n>. To determine the private key, we use the following formula d such that:

De mod {(p - 1) x (q - 1)} = 1

7d mod 60 = 1, which gives d = 43

The private key is <d, n> = (43, 77)

Step 6: A ciphertext message c is decrypted using the private key <d, n>. To calculate plain text m from the ciphertext c the following formula is used to get plain text m.

m = cd mod n

m = 3743 mod 77

m = 9

In this example, Plain text = 9 and the ciphertext = 37

## Example 2:

In an RSA cryptosystem, a particular A uses two prime numbers, 13 and 17, to generate the public and private keys. If the public of A is 35. Then the private key of A is ?

Explanation:

Step 1: in the first step, select two large prime numbers, p and q.

p = 13

q = 17

Step 2: Multiply these numbers to find n = p x q, where n is called the modulus for encryption and decryption.

First, we calculate

n = p x q

n = 13 x 17

n = 221

Step 3: Choose a number e less than n, such that n is relatively prime to (p - 1) x (q -1). It means that e and (p - 1) x (q - 1) have no common factor except 1. Choose "e" such that 1<e < φ (n), e is prime to φ (n), gcd (e, d (n)) =1.

Second, we calculate

φ (n) = (p - 1) x (q-1)

φ (n) = (13 - 1) x (17 - 1)

φ (n) = 12 x 16

φ (n) = 192

g.c.d (35, 192) = 1

Step 3: To determine the private key, we use the following formula to calculate the d such that:

Calculate d = de mod φ (n) = 1

d = d x 35 mod 192 = 1

d = (1 + k.φ (n))/e [let k =0, 1, 2, 3… ]

Put k = 0

d = (1 + 0 x 192)/35

d = 1/35

Put k = 1

d = (1 + 1 x 192)/35

d = 193/35

Put k = 2

d = (1 + 2 x 192)/35 d = 11

The private key is <d, n> = (11, 221)

Hence, private key i.e., d = 11

## Example 3:

An RSA cryptosystem uses two prime numbers 3 and 13 to generate the public key= 3 and the private key = 7. What is the value of cipher text for a plain text?

Explanation:

Step 1: In the first step, select two large prime numbers, p and q.

p = 3

q = 13

Step 2: Multiply these numbers to find n = p x q, where n is called the modulus for encryption and decryption.

First, we calculate

n = p x q

n = 3 x 13 = 39

Step 3: If n = p x q, then the public key is <e, n>. A plaintext message m is encrypted using public key <e, n>. Thus, the public key is <e, n> = (3, 39).

To find ciphertext from the plain text following formula is used to get ciphertext C.

C = me mod n

C = 53 mod 39

C = 125 mod 39

C = 8

Hence, the ciphertext generated from plain text, C = 8.

Example 4:

An RSA cryptosystem uses two prime numbers, 3 and 11, to generate private key = 7. What is the value of ciphertext for a plain text 5 using the RSA public-key encryption algorithm?

Explanation:

Step 1: in the first step, select two large prime numbers, p and q.

p = 3

q = 11

Step 2: Multiply these numbers to find n = p x q, where n is called the modulus for encryption and decryption.

First, we calculate

n = p x q

n = 3 x 11

n = 33

Step 3: Choose a number e less than n, such that n is relatively prime to (p - 1) x (q -1). It means that e and (p - 1) x (q - 1) have no common factor except 1. Choose "e" such that 1< e < φ (n), e is prime to φ (n), gcd (e, d (n)) =1.

Second, we calculate

φ (n) = (p - 1) x (q-1)

φ (n) = (3 - 1) x (11 - 1)

φ (n) = 2 x 10

φ (n) = 20

Step 4: To determine the public key, we use the following formula to calculate the d such that:

Calculate e x d = 1 mod φ (n)

e x 7 = 1 mod 20

e x 7 = 1 mod 20

e = (1 + k. φ (n))/ d [let k =0, 1, 2, 3… ]

Put k = 0

e = (1 + 0 x 20) / 7

e = 1/7

Put k = 1

e = (1 + 1 x 20) / 7

e = 21/7

e = 3

The public key is <e, n> = (3, 33)

Hence, public key i.e., e = 3

# 4.4Proposed Algorithm

The proposed algorithm considered an image as information rather than text or numbers, and another image as a key to implement the RSA algorithm.

The steps of the proposed algorithm are as follows:

Consider the image and find out the array format.

Finding the length of an array.

Consider the key image and find out the corresponding array format.

From the key image, any random prime number will be considered as key (For simplicity, we have considered the first prime number as key among the key image array).

Then, the same key generation, encryption, decryption process can be done as the traditional one.

# Key generation

Choose two numbers p and q that are prime and distinct in nature.

For high security purposes, the prime numbers p and q should be randomly taken and must have the same bit-length.

Compute n=pq, where n is used for modulus of both the public and private keys. Its size is expressed in bits which is known as key length.

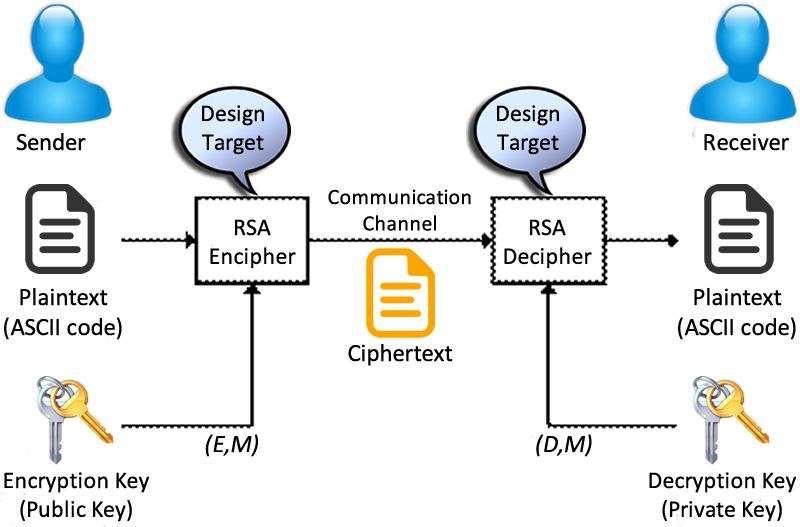
Compute ϕ(n)=ϕ(p)ϕ(q)=(p−1) (q−1) =n−(p+q−1), where ϕis Euler’s totient function.

Choose an integer e such that 1 <e<ϕ(n) and gcd(e,ϕ(n)) =1, i.e., e and

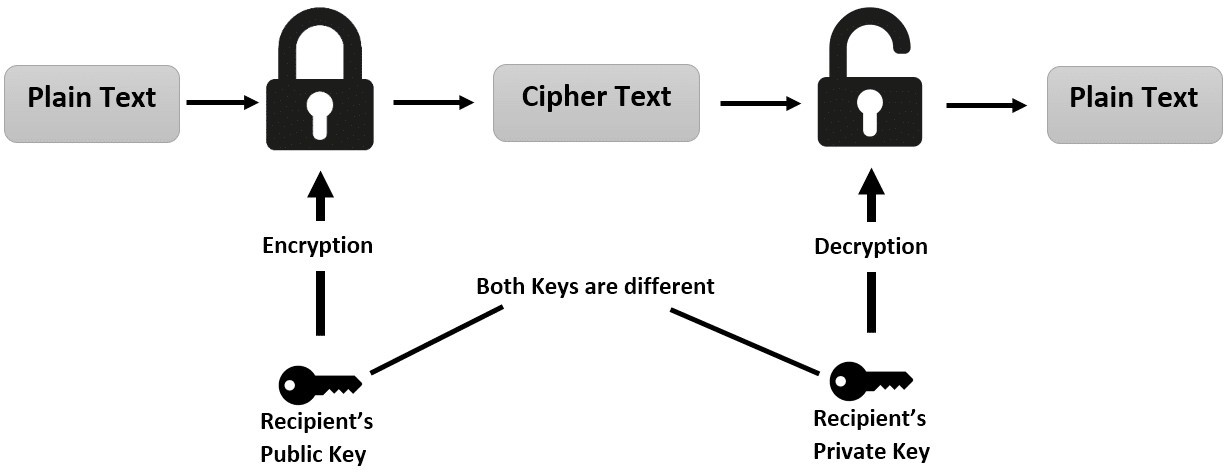
ϕ(n) are coprime.

Determine das d≡e−1(mod ϕ(n));

i.e., d is the multiplicative inverse of e (modulo ϕ(n)).



**Fig 1: Encryption Decryption**



**Fig 2: Plain text to Cipher text**

# 4.5Advantages of RSA Algorithm

No Key Sharing: RSA encryption depends on using the receiver’s public key, so you don’t have to share any secret key to receive

messages from others.

Proof of Authenticity: Since the key pairs are related to each other, a receiver can’t intercept the message since they won’t have the

correct private key to decrypt the information.

Faster Encryption: The encryption process is faster than that of the DSA algorithm.

Data Can’t Be Modified: Data will be tamper-proof in transit since meddling with the data will alter the usage of the keys. And the

private key won’t be able to decrypt the information, hence alerting the receiver of manipulation.

**4.6MILLER-RABIN TEST**

The Miller-Rabin primality test is a probabilistic algorithm used to determine if a given number is a prime. It is widely used due to its efficiency, especially for large numbers, and is often employed in cryptographic applications like RSA key generation.

### How the Miller-Rabin Test Works:

**Decompose n−1n-1n−1**: Express n−1n-1n−1 as 2s×d2^s \times d2s×d, where sss and ddd are integers, and ddd is odd. This step involves factoring out powers of 2 from n−1n-1n−1.

**Random Base Selection**: Choose a random integer aaa between 2 and n−2n-2n−2.

**Witness Calculation**:

Compute x=admod  nx = a^d \mod nx=admodn.

If x=1x = 1x=1 or x=n−1x = n-1x=n−1, nnn is a probable prime.

If not, repeatedly square xxx up to s−1s-1s−1 times and check if xxx becomes n−1n-1n−1. If it does, nnn is a probable prime.

**Repeat**: Perform the test multiple times with different values of aaa. If nnn passes the test each time, it is "probably prime." If it fails even once, nnn is composite.

### Example:

Suppose n=61n = 61n=61:

n−1=60=22×15n-1 = 60 = 2^2 \times 15n−1=60=22×15, so s=2s = 2s=2 and d=15d = 15d=15.

Choose a random base a=5a = 5a=5:

Compute 515mod  61=605^{15} \mod 61 = 60515mod61=60, which is −1mod  61-1 \mod 61−1mod61.

Since the result is −1-1−1, 61 is a probable prime.

The Miller-Rabin test is particularly useful because it provides a high probability of correctness with multiple iterations, making it suitable for practical use in cryptography​ ([Wikipedia](https://en.wikipedia.org/wiki/Miller%E2%80%93Rabin_primality_test))​​ ([incolumitas.com](https://incolumitas.com/2018/08/12/finding-large-prime-numbers-and-rsa-miller-rabin-test/))​​ ([ASecuritySite](https://asecuritysite.com/primes/rabin" \t "_blank))​​ ([Rosetta Code](https://rosettacode.org/wiki/Miller%E2%80%93Rabin_primality_test))​.

# 4.6Encryption

In encryption, the plain text content represented in the form of an image is converted to cypher text using another image as a secrete key. The picture can likewise be changed over to scrambled structure utilizing the random image as key and the resultant image after encryption. The scrambled picture is then sent over an insecure channel to the receiver. At the receiver end, the scrambled picture is decoded using the private key of the receiver. The proposed image encryption technique is used for providing better security of information. Subsequent, to encoding information, objective- scrambled information was decoded with assistance of association called as unscrambling. The resultant encrypted image is generated by applying the RSA algorithm between information image and the key image i.e., the binary equivalent of information image undergoes encryption with each pixel corresponding binary equivalent of key image to produce the encrypted image.



**Fig 3:Encryption**

# 4.7Decryption

Decryption is the process of transforming data that has been rendered unreadable through encryption back to its original form. In decryption, the system extracts and converts the garbled data and transforms it to texts and images that are easily understandable not only by the reader but also by the system. Decryption may be accomplished manually or automatically. It may also be performed with a set of keys or passwords.



**Fig 10:Decryption**

**5.PERFORMANCE ANALYSIS**

Choose two numbers p and q that are prime and distinct in nature.

For high security purposes, the prime numbers p and q should be randomly taken and must have the same bit-length.

Compute n=pq, where n is used for modulus of both the public and private keys. Its size is expressed in bits which is known as key length.

Compute ϕ(n)=ϕ(p)ϕ(q)=(p−1) (q−1) =n−(p+q−1), where ϕis Euler’s totient function.

Choose an integer e such that 1 <e<ϕ(n) and gcd(e,ϕ(n)) =1, i.e., e and

ϕ(n) are coprime.

Determine das d≡e−1(mod ϕ(n));

i.e., d is the multiplicative inverse of e (modulo ϕ(n)).

Solve d given d·e≡1(mod ϕ(n)).

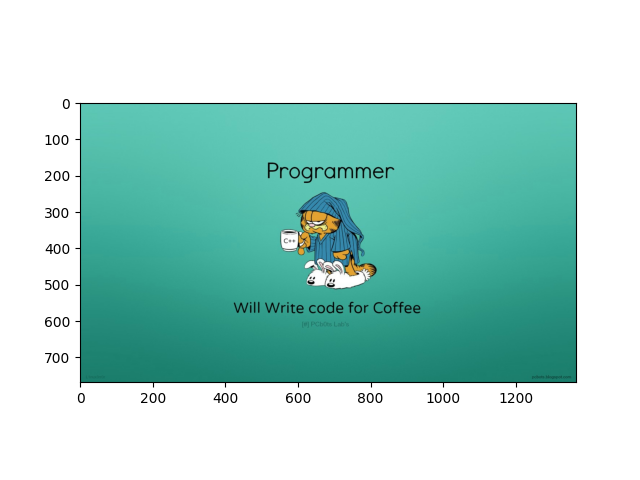
**5.1 Practical Work**

As mentioned earlier, there are two different primes P and Q used to generate n. Write code to generate them. As the length increases, it takes longer to generate the relative primes e and d. The following table shows the relationship between the numbers chosen for P and Q and the time allotted to generate them.

**Table 1: Prime Number Time Establishment**

|  |  |  |
| --- | --- | --- |
| **Chosen Numbers** | | **Time Established**  **(seconds)** |
| **P** | **Q** |
| **7** | **5** | **0.010387** |
| **11** | **13** | **0.040969** |
| **17** | **23** | **0.124453** |
| **29** | **53** | **0.964863** |
| **47** | **59** | **3.504031** |
| **113** | **71** | **18.675820** |
| **239** | **173** | **590.455964** |

## Now we use digital images as data to be encrypted.



**Fig 5: The image used for encryption**

## Fig 6: After Encrypting image.

**5.2Screenshots Of Implementation**

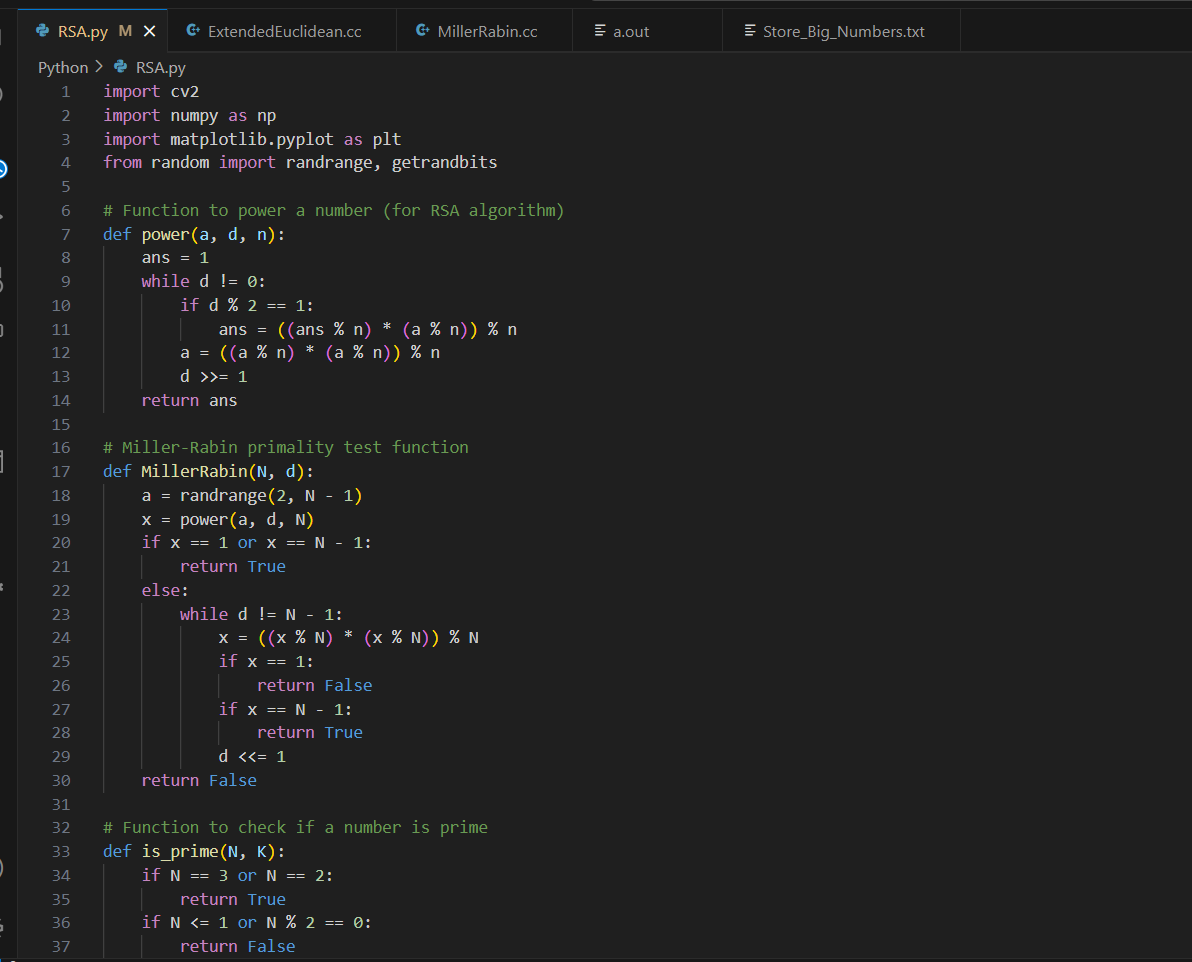


Fig-7

## 

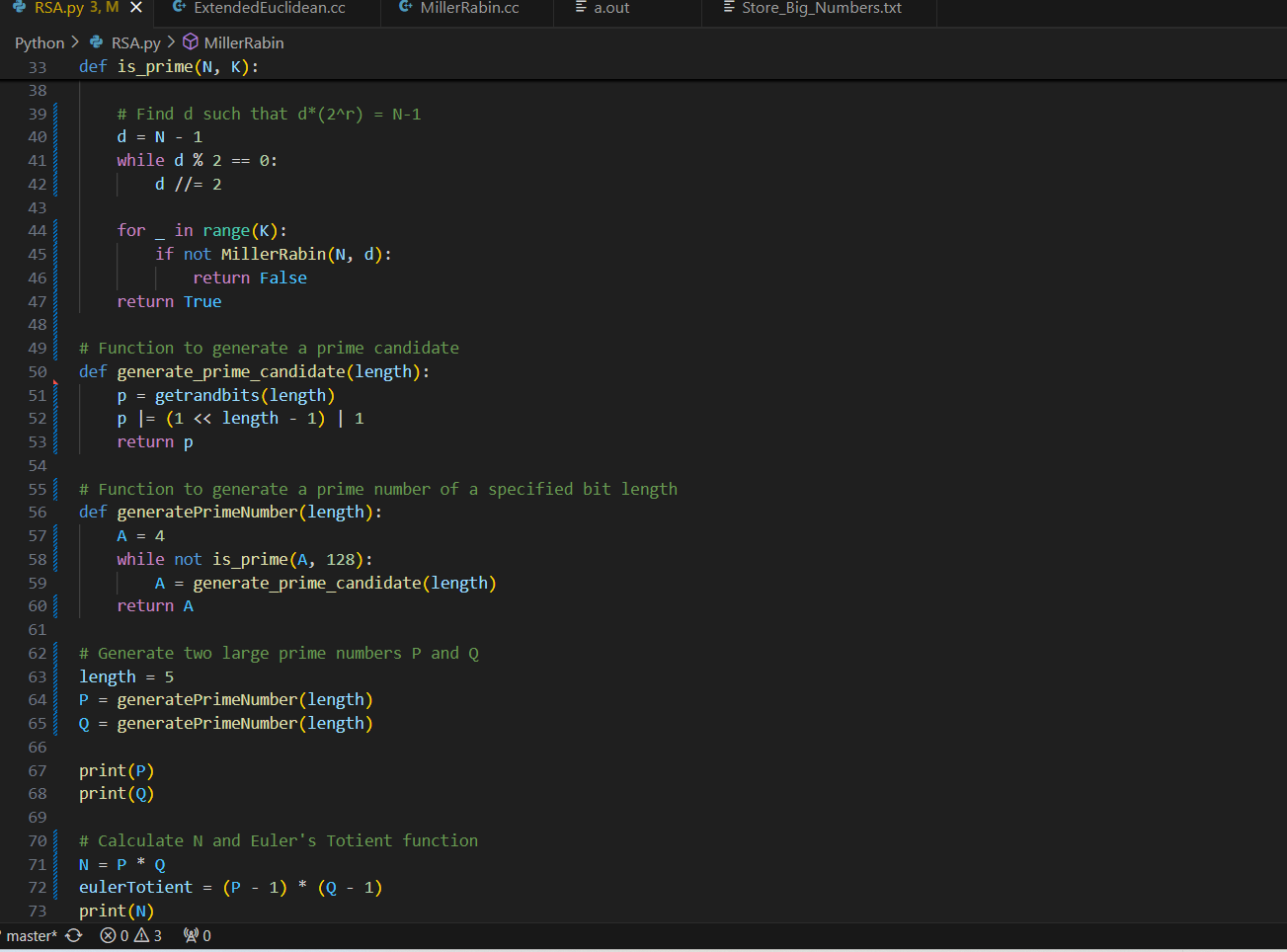


Fig-8

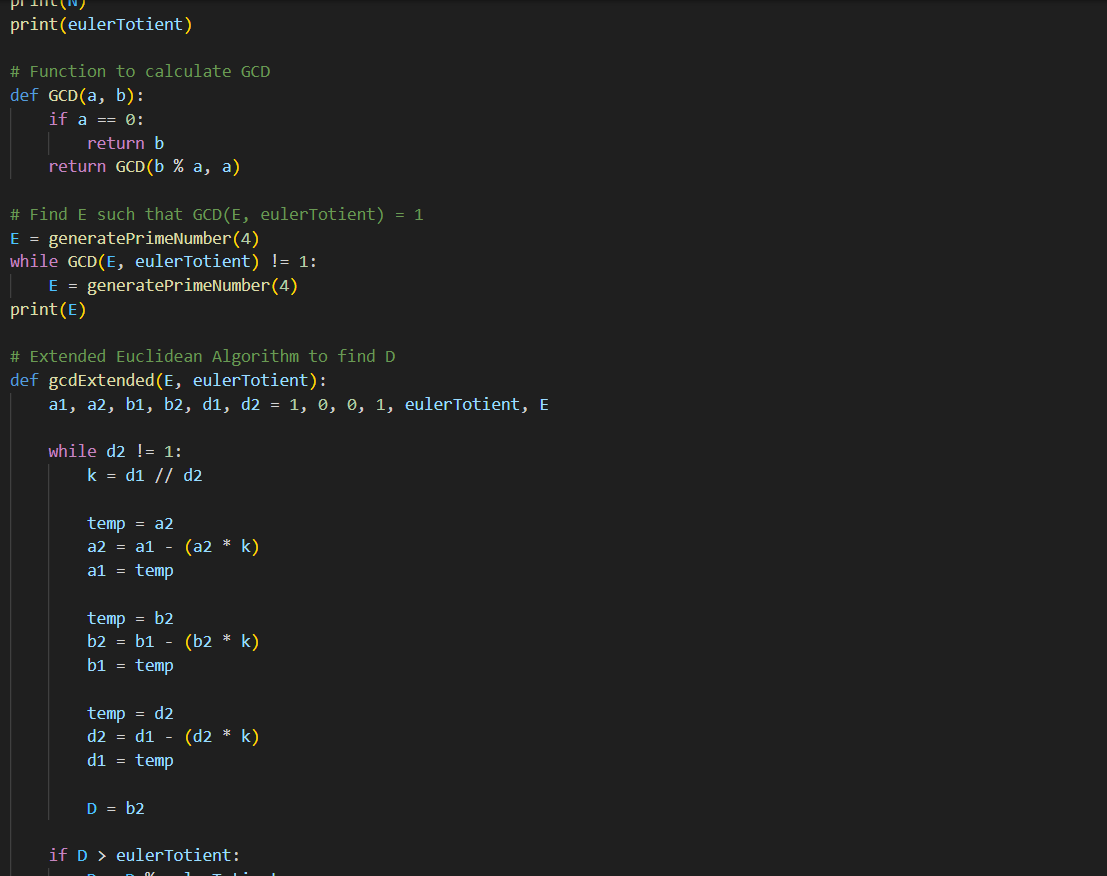


Fig-9

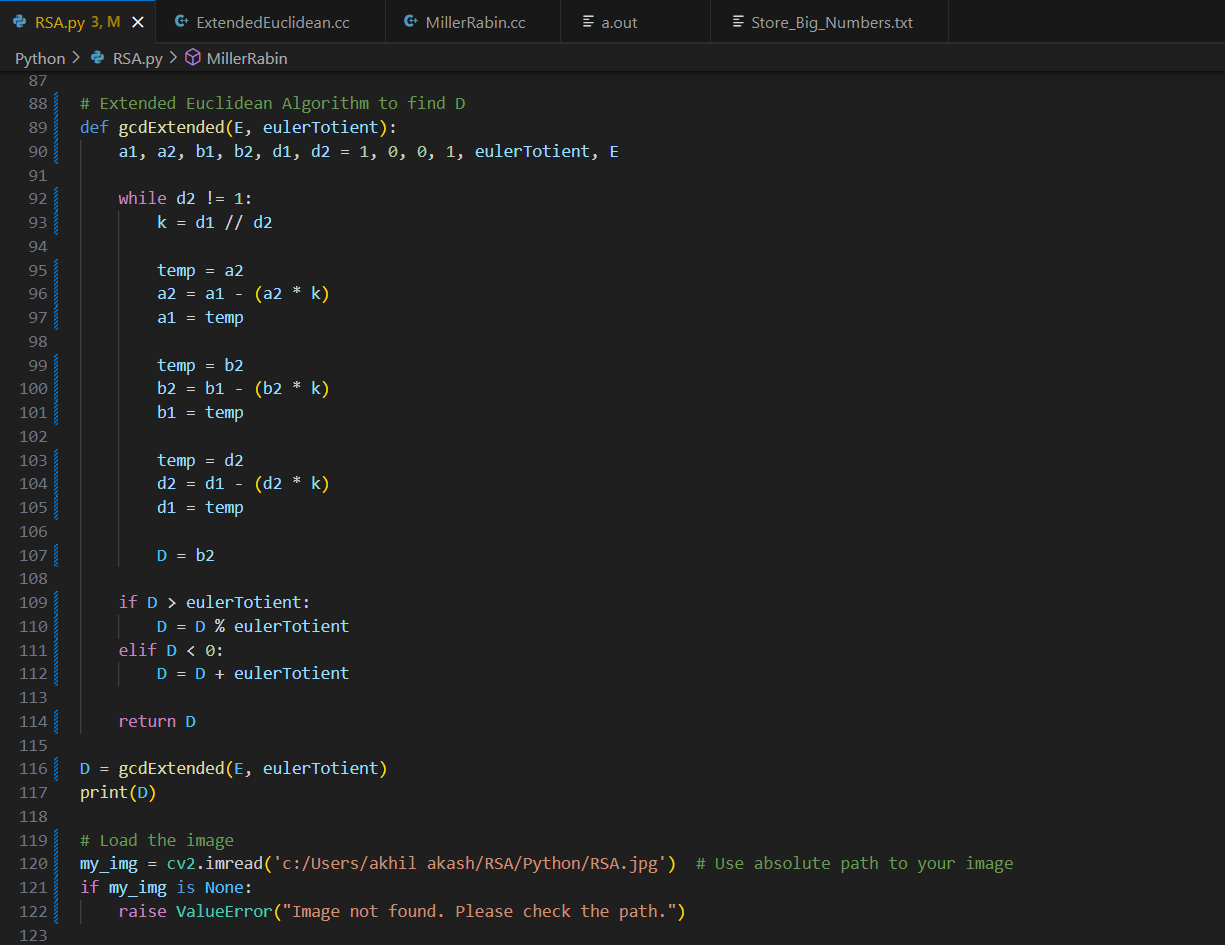


Fig-10

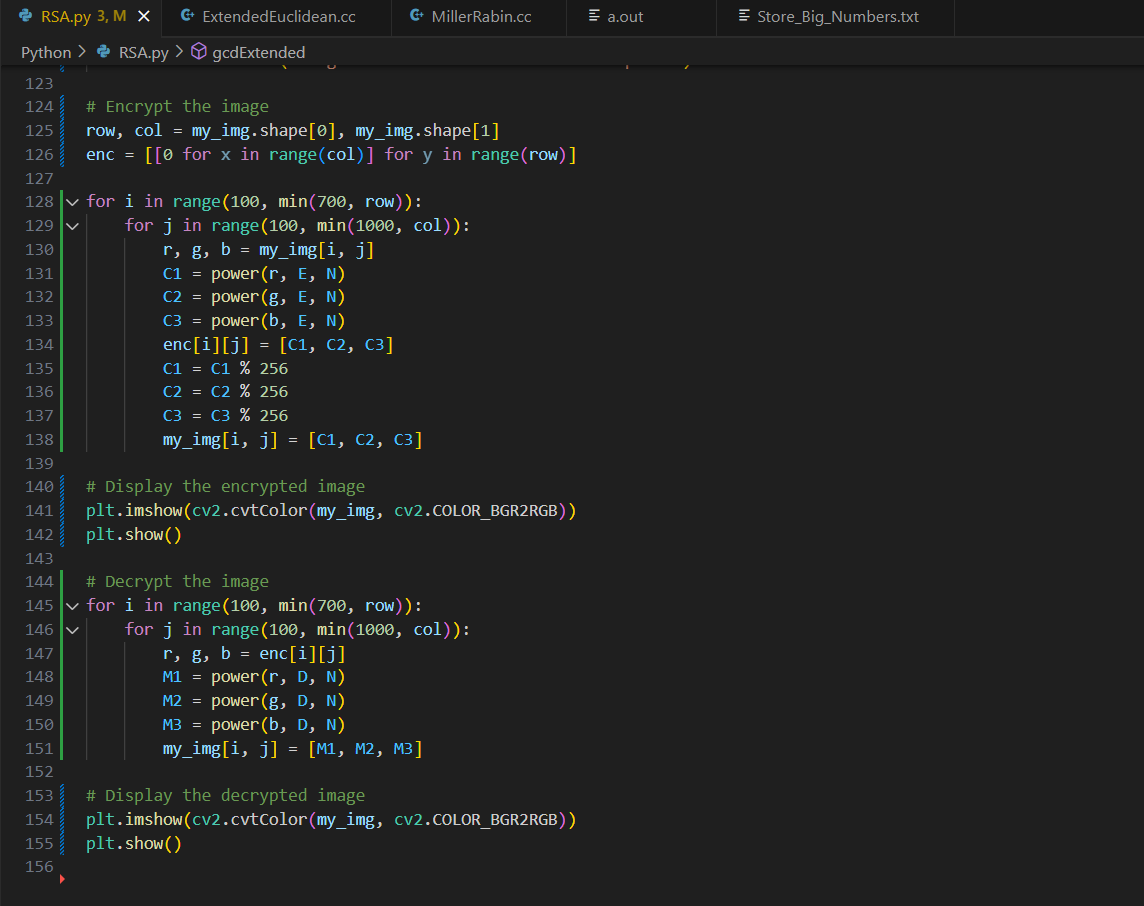


Fig-11

**6.CONCLUSION**

In the computerized world, the security of pictures has gotten more significant as the correspondence has expanded quickly. Every one of the methods is in an ongoing picture encryption that could just track down to a lowest degree of safety. Here, the picture that calculate encryption proposed productive and exceptionally secur- able with significant degree of safety and less calculation. The consequences of the recreation show that the RSA calculation enjoys benefits dependent on their methods which are applied on pictures. Henceforth, it is assumed that for picture encryption, the strategies are useful and give security in the open organization. In this digital world, as the communication increased rapidly, the security of image has become more important. It helps to share files with high security. Lost or stolen device can be protected using encryption.

All the techniques we used are in real-time image encryption that could find a low level of security.

•

•

Here, encryption using image as a key proposed highly secure with less computation.

•

Result of this research show that the algorithm gives advantages based on several techniques which are applied on data images.

•

There is no loss of data in the process as the decrypted image is the accurate of the original one.

•

The performance analysis of the proposed algorithm is increased by 40% as compared to text as a key.

•

Without knowing the private key, no one can determine the original image. Hence, it was concluded that various methods used here are good for encryption of image using image as a key, and it gives higher security in open network.

•

**7. REFERENCES**

<http://en.wikipedia.org/wiki/Encryption>

<http://en.wikipedia.org/wiki/Plaintext>

<http://en.wikipedia.org/wiki/Ciphertext>

<http://en.wikipedia.org/wiki/Decryption>

<http://en.wikipedia.org/wiki/Cryptography>

F. H. M. S. Al-Kadei, H. A. Mardan and N. A. Minas, "Speed Up Image Encryption by Using RSA Algorithm," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 1302-1307, doi: 10.1109/ICACCS48705.2020.9074430.

Kaur, M., Kumar, V. A Comprehensive Review on Image Encryption

Techniques. *Arch Computat Methods Eng* 27, 15–43 (2020).

https://doi.org/10.1007/s11831-018-9298-8.

Mai Helmy, El-Sayed M. El-Rabaie, Ibrahim Eldokany, F.E. Abd El-

Samie, Proposed Hybrid Encryption Algorithm for Robust 3D Image

Communication over Wireless Channels, Optik, 2022, 170205, ISSN

0030-4026, https://doi.org/10.1016/j.ijleo.2022.170205.

Sahoo, A., Mohanty, P., Sethi, P.C. (2022). Image Encryption Using

RSA Algorithm. In: Udgata, S.K., Sethi, S., Gao, XZ. (eds) Intelligent

Systems. Lecture Notes in Networks and Systems, vol 431. Springer,

Singapore. https://doi.org/10.1007/978-981-19-0901-6\_56.

https:/[/w](http://www.researchgate.net/publication/307545291_Digital_Image_E)w[w.researchgate.net/publication/307545291\_Digital\_Image\_E](http://www.researchgate.net/publication/307545291_Digital_Image_E)ncryption\_Based\_on\_RSA\_Algorit-hm.