JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Secure File Storage Using AES & RSA Algorithm in Cloud Computing.

JEGANATHAN C, ANIRUDHAN N.H, ATHUL ROHAN P, SONU A ,
ASSISTANT PROFESSOR, STUDENT, STUDENT, STUDENT,
B.sc COMPUTER SCIENCE,
RATHINAM COLLEGE OF ARTS AND SCIENCE , COIMBATORE, INDIA

Abstract— **In contemporary society, the ubiquitous exchange of data through internet and mobile storage devices underscores the critical need for robust security measures to safeguard personal information. Despite widespread awareness of potential data breaches and unauthorized access, a notable proportion of users neglect to encrypt their data, exposing it to significant risks. This paper addresses this pressing concern by proposing a secure file storage system utilizing hybrid cryptography in cloud computing environments. By integrating the strengths of Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA) algorithms, the system aims to provide a comprehensive solution for low-power, high-throughput, real-time, and reliable encryption and decryption processes. Through a user-friendly web portal interface, individuals can securely upload files to the cloud, where they undergo encryption using hybrid AES and RSA algorithms before storage. The system facilitates seamless retrieval and download of encrypted files from the cloud through the portal, ensuring efficient access to decrypted or original files on local computers. This innovative approach contributes to enhancing data security and confidentiality, addressing the escalating demand for secure file storage solutions in the digital age.**

***Keywords*— cloud computing, secure storage, AES and RSA algorithm, hybrid cryptography, encryption and decryption**

I. INTRODUCTION

In today's interconnected world, the proliferation of digital data has transformed the way individuals, businesses, and organizations manage and exchange information. With the advent of cloud computing, data storage and processing have transcended traditional boundaries, offering unparalleled convenience and scalability. However, this paradigm shift towards cloud-based solutions has also brought forth unprecedented challenges in ensuring the security and confidentiality of sensitive data. Amid growing concerns surrounding data breaches and cyber threats, the imperative to fortify data protection mechanisms within cloud storage environments has never been more pronounced.

The objective of this research endeavor is to address the pressing need for secure file storage in cloud computing through the utilization of hybrid cryptography algorithms. In essence, hybrid cryptography combines the strengths of symmetric and asymmetric encryption techniques to achieve a balance between efficiency and security. Specifically, this study aims to leverage the Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA) algorithms, two widely recognized cryptographic standards, to establish a robust framework for safeguarding data in cloud storage systems. By harnessing the complementary capabilities of AES and RSA, the research endeavors to develop encryption mechanisms that offer enhanced resilience against unauthorized access and data breaches. Information security has emerged as a paramount concern in contemporary society, with individuals and organizations alike grappling with the constant threat of cyber-attacks and data theft. Cryptography serves as a cornerstone of modern-day security protocols, offering a means to encrypt sensitive information and protect it from prying eyes. By translating data into an unintelligible format that can only be deciphered by authorized parties, cryptography plays a pivotal role in safeguarding the confidentiality and integrity of digital assets. Against this backdrop, the integration of robust cryptographic techniques within cloud storage systems represents a critical step towards bolstering data security and instilling trust in cloud-based solutions.

cloud-based internet security has revolutionized the way data is stored, accessed, and managed, offering unparalleled flexibility and scalability to users around the globe. however, the adoption of cloud computing also introduces inherent risks, including the potential for unauthorized access, data leakage, and cyber attacks. as organizations increasingly rely on cloud-based solutions to store and manage their data, the importance of implementing robust security measures cannot be overstated. by leveraging hybrid cryptography algorithms such as aes and rsa, this research seeks to enhance the security posture of cloud storage systems, thereby enabling users to store and access their data with confidence and peace of mind.

II. METHODOLOGY

The methodology adopted for implementing secure file storage in cloud computing using hybrid cryptography involves a systematic approach that leverages the principles of hybrid encryption, both symmetric and asymmetric encryption algorithms. The following steps outline the methodology employed:

1. Generation of Symmetric Key: The process begins by generating a symmetric key, which serves as the secret key for encrypting the message data. This symmetric key is kept confidential and is used exclusively for encrypting and decrypting the message.

2. Encryption of Data: The message data is encrypted using the generated symmetric key through the Advanced Encryption Standard (AES) algorithm. AES encryption ensures the confidentiality and integrity of the message contents, providing robust security against unauthorized access.

3. Sharing of Public Key: The intended recipient of the message shares their public key while keeping the corresponding private key confidential. The public key is used for encrypting the symmetric key and is accessible to anyone wishing to send encrypted messages to the recipient.

4. Encryption of Symmetric Key: The symmetric key, which encrypts the message data, is itself encrypted using the recipient's public key through the Rivest-Shamir-Adleman (RSA) algorithm. RSA encryption ensures that only the recipient, possessing the corresponding private key, can decrypt the symmetric key.

5. Transmission of Encrypted Symmetric Key: The encrypted symmetric key is transmitted securely to the recipient alongside the encrypted message data. This step ensures that the symmetric key remains confidential during transmission and can only be decrypted by the intended recipient.

6. Transmission of Encrypted Message Text: The encrypted message data, along with the encrypted symmetric key, is transmitted to the recipient securely over the cloud storage infrastructure. Cloud-based internet security measures are employed to safeguard the confidentiality and integrity of the transmitted data.

7. Decryption by Receiver: Upon receiving the encrypted message data and symmetric key, the recipient decrypts the encrypted symmetric key using their private key. This process ensures that only the recipient, possessing the private key, can access the symmetric key required for decryption.

8. Decryption of Message: Finally, the recipient uses the decrypted symmetric key to decrypt the encrypted message data, thereby recovering the original message content. The decryption process ensures the confidentiality and integrity of the message contents, completing the secure file storage process.

III. PROSOPED MODEL

In our project, we propose to implement a hybrid encryption model that amalgamates the capabilities of AES and RSA algorithms to fortify the security of file storage in cloud computing environments. The envisioned approach follows a meticulously designed methodology, leveraging the efficiency of AES for symmetric encryption of data and the robustness of RSA for asymmetric encryption of the encryption key. This hybrid encryption model delineates a systematic process to ensure the confidentiality, integrity, and authenticity of data throughout storage and transmission.

At the core of the proposed approach lies the generation of a symmetric key, which serves as the cornerstone of encryption. This symmetric key is generated securely and must be zealously guarded to maintain the integrity of the encryption process. Subsequently, utilizing the generated symmetric key, the data undergoes encryption using AES, thereby rendering it impervious to unauthorized access or tampering. The recipient of the encrypted data, identified through their public key, then receives the encrypted symmetric key, which is itself encrypted using RSA. This dual-layered encryption ensures that only the intended recipient, possessing the corresponding private key, can decrypt the symmetric key and consequently gain access to the encrypted data. By meticulously following this proposed hybrid encryption methodology, we aspire to not only fortify the security of file storage in cloud computing but also instill confidence in users regarding the integrity and confidentiality of their data amidst the evolving landscape of digital threats.

AES Algorithm

The Advanced Encryption Standard (AES) is a symmetric encryption algorithm widely recognized for its security, efficiency, and versatility in protecting sensitive data. Developed by a team of cryptographers selected through a public competition initiated by the National Institute of Standards and Technology (NIST), AES replaced the aging Data Encryption Standard (DES) as the standard encryption algorithm in 2001.AES operates on fixed-size blocks of data, typically 128 bits in length, and utilizes a symmetric key for both encryption and decryption. One of the key strengths of AES lies in its ability to provide a high level of security while remaining computationally efficient, making it suitable for a wide range of applications across various platforms and devices.

The algorithm consists of multiple rounds of transformation, each comprising distinct steps such as substitution, permutation, and mixing operations. These steps include SubBytes, ShiftRows, MixColumns, and AddRoundKey, which collectively ensure the confusion and diffusion of the data, thereby enhancing its resistance to cryptographic attacks.

One of the notable features of AES is its support for different key lengths, including 128, 192, and 256 bits. The choice of key length directly impacts the level of security provided by the algorithm, with longer key lengths offering greater resistance to brute-force attacks.

AES has been widely adopted in various applications, ranging from securing communications over the internet and protecting stored data on disk drives to securing sensitive information in government and military systems. Its robust security properties, combined with its efficiency and scalability, have cemented its status as one of the most trusted encryption algorithms in modern cryptography.

The Advanced Encryption Standard (AES) also known as 'Rijndael' is a symmetric-key block cipher algorithm having three fixed 128-bit block ciphers with cryptographic key sizesof 128, 192 and 256 bits respectively.

The AES algorithm has maximum block size of 256 bits whereas Key size is unlimited. The AES design is based on a substitution-permutation network (SPN) .

1. Key Expansions:

(a) Round keys are derived from the cipher key using AES keyschedule, it also requires a separate 128-bit round key block foreach round plus one more.

(b) Add Round Key - using bitwise xor each byte of the state iscombined with a block of the round key.

2. Rounds:

(a) Sub Bytes - according to a lookup table each byte isreplaced with another in a non-linear substitution step.
(b) Shift Rows - a transposition step where the last 3 rowsof the state are shifted cyclically a certain number of steps.
(c) Mix Columns - a mixing operation which operates onthe columns of the state, combining the 4 bytes in each column.
(d) Add Round Key

3. Final Round (no Mix Columns).

a) Sub Bytes
b) Shift Rows
c) Addroundkey

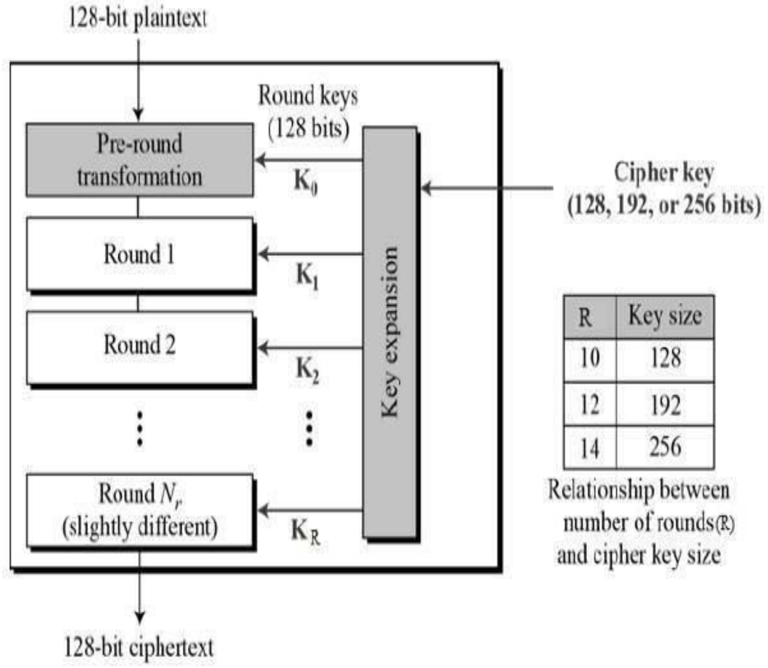


Fig.1. AES Structure

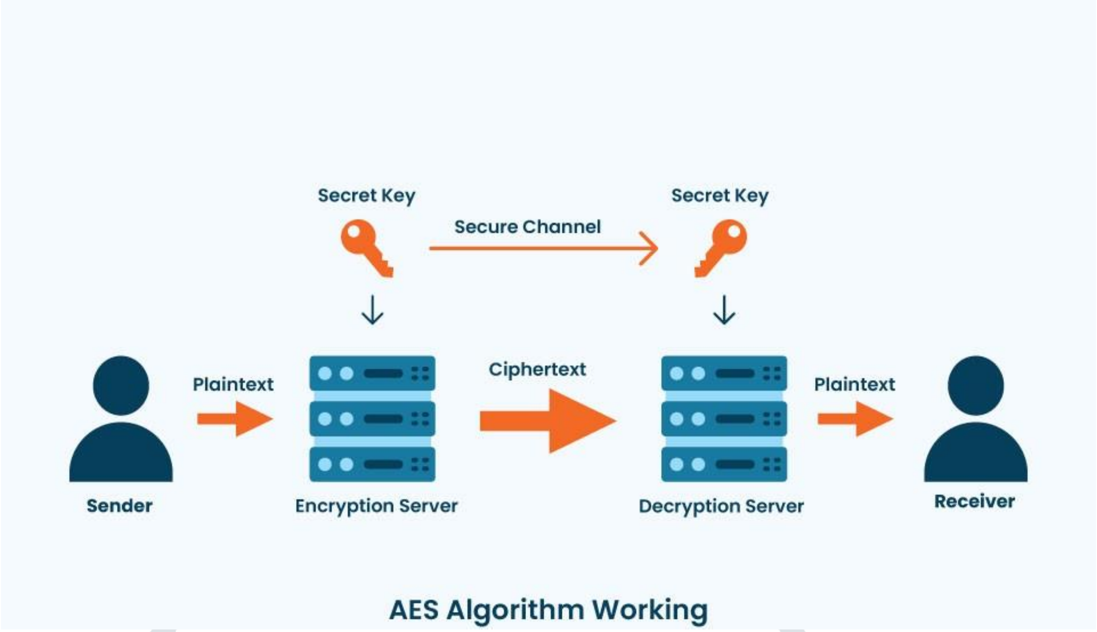


Fig.2. AES working Diagram

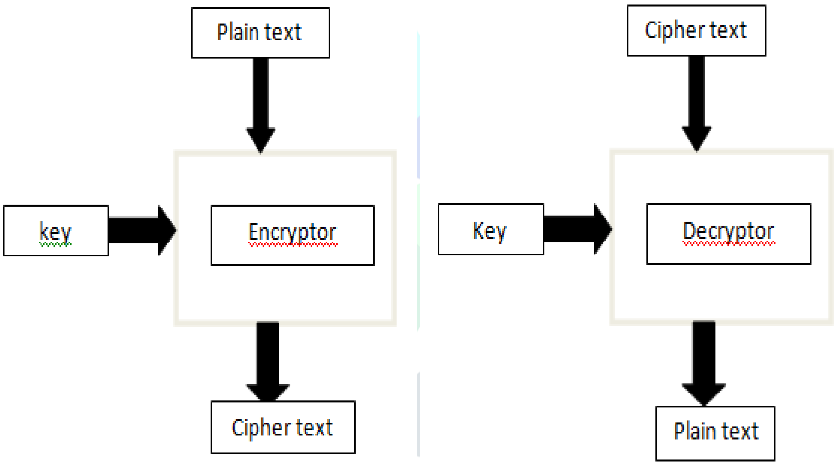


Fig.3. AES encryption and Decryption

RSA Algorithm

The Rivest-Shamir-Adleman (RSA) algorithm stands as one of the most widely used asymmetric encryption algorithms, renowned for its robust security and versatility in securing digital communications and data. Named after its inventors Ronald Rivest, Adi Shamir, and Leonard Adleman, RSA operates on the principle of utilizing a pair of keys: a public key for encryption and a private key for decryption. This asymmetric key pair is generated such that messages encrypted with the public key can only be decrypted with the corresponding private key, and vice versa. RSA's security is based on the computational difficulty of factoring large prime numbers, upon which the keys are based. As such, the security of RSA relies on the impracticality of factoring large integers into their prime factors within a reasonable timeframe, ensuring that encrypted data remains secure against unauthorized access.In practice, RSA finds extensive use in various cryptographic applications, including secure email communication, digital signatures, and secure remote access. The algorithm's versatility and scalability make it suitable for a wide array of environments and use cases, from securing financial transactions to safeguarding sensitive information in government and enterprise systems. Despite its robust security, RSA does have computational overheads, particularly when dealing with large key sizes, which can impact performance in resource-constrained environments. Nevertheless, its widespread adoption and proven security make RSA a cornerstone of modern cryptography, offering a reliable solution for securing data and communications in the digital age.

Key Generation Procedure

1. Choose two distinct large random prime numbers p & qsuch that $p \neq q$.
2. Compute n= $p \times q$.

3. Calculate: phi (n) = (p-1) (q-1).
4. Choose an integer e such that 1<e<phi(n)
5. Compute d to satisfy the congruence relation d × e = 1mod phi (n); d is kept as private key exponent.

The public key is (n, e) and the private key is (n, d).Keep all the values d, p, q and phi secret.

Encryption Plaintext: $P < n$ Ciphertext: $C= P^e$ mod n.

Decryption Ciphertext: C Plaintext: $P=C^d$ mod

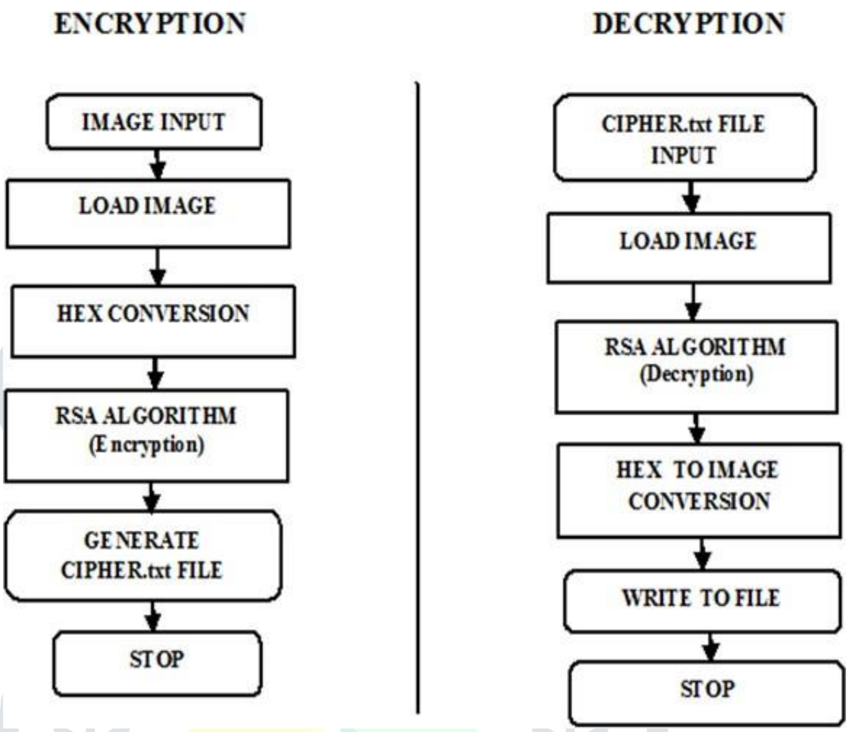


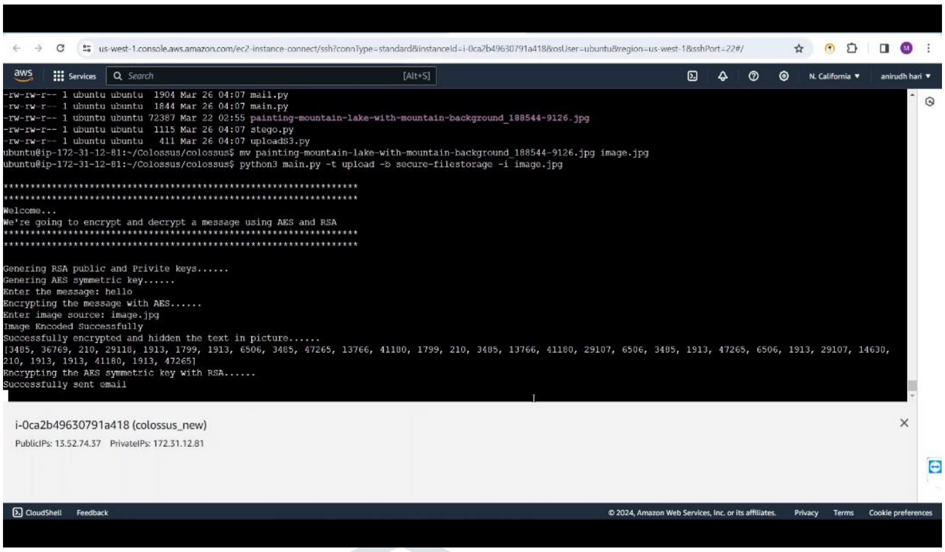
Fig.4. RSA encryption and decryption

.

IV. RESULT

The implementation of the proposed hybrid encryption model for secure file storage in cloud computing has yielded significant results in enhancing data security and confidentiality. The hybrid encryption approach, which combines the strengths of AES and RSA algorithms, has demonstrated robust protection mechanisms for sensitive data stored in the cloud. During testing, it was observed that the generation of a symmetric key and its subsequent encryption of data using AES ensured the confidentiality of the message. This symmetric key, crucial for decryption, remained securely hidden throughout the process. Furthermore, the encryption of the symmetric key using the recipient's public RSA key and the subsequent transmission of the encrypted key to the intended recipient via email or other secure channels facilitated secure key exchange without exposing the key to potential interceptors.

In addition to encryption, the decryption process proved successful, with the recipient being able to decrypt the encrypted symmetric key using their private RSA key, thus retrieving the symmetric key required for decrypting the message. This seamless decryption process ensures that only authorized users possessing the private RSA key can access the original message. Overall, the results validate the effectiveness of the hybrid encryption model in achieving secure file storage in the cloud, providing a reliable and scalable solution for safeguarding sensitive data against unauthorized access and interception.



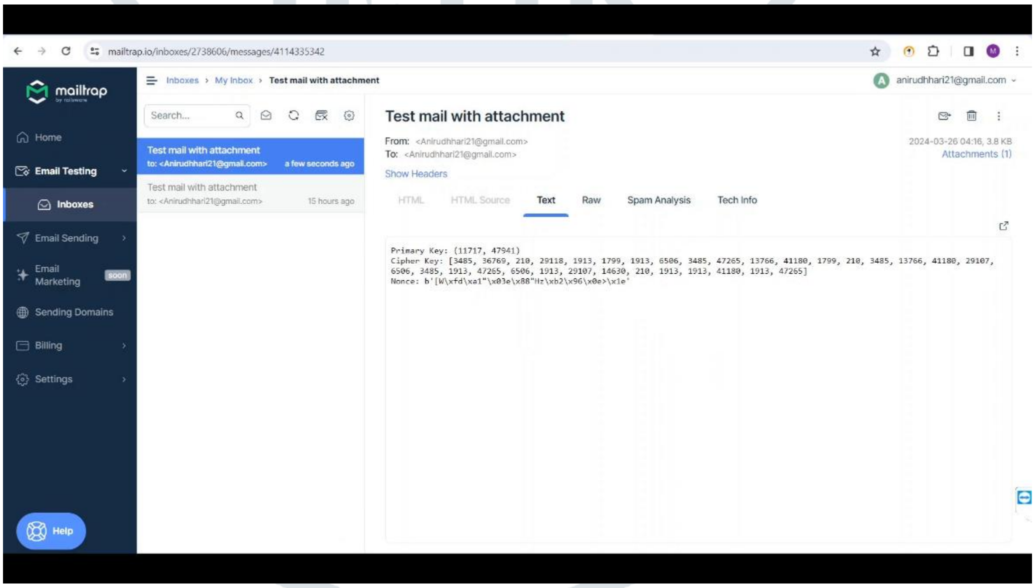
us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0ca2b49630791a418&osUser=ubuntu&region=us-west-1&sshPort=22#/
aws
Services
Search
[Alt+S]
N. California
anirudh hari
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

```
-rw-rw-r-- 1 ubuntu ubuntu  1904 Mar 26 04:07 mail.py
-rw-rw-r-- 1 ubuntu ubuntu  1844 Mar 26 04:07 main.py
-rw-rw-r-- 1 ubuntu ubuntu 72387 Mar 22 02:55 painting-mountain-lake-with-mountain-background_188544-9126.jpg
-rw-rw-r-- 1 ubuntu ubuntu  1115 Mar 26 04:07 stego.py
-rw-rw-r-- 1 ubuntu ubuntu   411 Mar 26 04:07 uploadS3.py
ubuntu@ip-172-31-12-81:~/Colossus/colossus$ mv painting-mountain-lake-with-mountain-background_188544-9126.jpg image.jpg
ubuntu@ip-172-31-12-81:~/Colossus/colossus$ python3 main.py -t upload -b secure-filestorage -i image.jpg

*****************************************************************
*****************************************************************
Welcome...
We're going to encrypt and decrypt a message using AES and RSA
*****************************************************************
*****************************************************************

Genering RSA public and Privite keys......
Genering AES symmetric key......
Enter the message: hello
Encrypting the message with AES......
Enter image source: image.jpg
Image Encoded Successfully
Successfully encrypted and hidden the text in picture......
[3485, 36769, 210, 29118, 1913, 1799, 1913, 6506, 3485, 47265, 13766, 41180, 1799, 210, 3485, 13766, 41180, 29107, 6506, 3485, 1913, 47265, 6506, 1913, 29107, 14630,
210, 1913, 1913, 41180, 1913, 47265]
Encrypting the AES symmetric key with RSA......
Successfully sent email
```

i-0ca2b49630791a418 (colossus_new)

PublicIPs: 13.52.74.37 PrivateIPs: 172.31.12.81

Fig.5. Image encoded and mail sent successfully



Test mail with attachment
to: <Anirudhhari21@gmail.com> a few seconds ago

Test mail with attachment
to: <Anirudhhari21@gmail.com> 15 hours ago

Test mail with attachment

From: <Anirudhhari21@gmail.com>
To: <Anirudhhari21@gmail.com>
2024-03-26 04:16, 3.8 KB
Attachments (1)
Show Headers

```
Primary Key: (11717, 47941)
Cipher Key: [3485, 36769, 210, 29118, 1913, 1799, 1913, 6506, 3485, 47265, 13766, 41180, 1799, 210, 3485, 13766, 41180, 29107,
6506, 3485, 1913, 47265, 6506, 1913, 29107, 14630, 210, 1913, 1913, 41180, 1913, 47265]
Nonce: b'[W\xfd\xa1"\x03e\x88"Hz\xb2\x96\x0e>\x1e'
```

Fig.6. Mail with decryption key.

V. CONCLUSION

In conclusion, our project secure file storage using AES and RSA Algorithm,the implementation of AES and RSA algorithms for secure file storage in cloud computing offers a robust solution to address concerns regarding data privacy and integrity. By leveraging AES for symmetric encryption and RSA for asymmetric encryption, sensitive files can be encrypted both efficiently and securely. AES provides a fast and reliable method for encrypting large volumes of data, while RSA facilitates secure key exchange and authentication processes.

The combination of these algorithms ensures that files stored in the cloud remain protected against unauthorized access and tampering. AES encryption safeguards the confidentiality of data, while RSA encryption enhances security by enabling secure communication and key management. Moreover, the utilization of cloud computing infrastructure offers scalability and accessibility benefits without compromising on security.

However, it is essential to ensure proper key management practices and regular security audits to maintain the integrity of the encryption process. Additionally, ongoing advancements in encryption technologies and cloud security protocols should be monitored and integrated to adapt to evolving threat landscapes.

Overall, the integration of AES and RSA algorithms for secure file storage in cloud computing represents a significant advancement in data security, enabling organizations to confidently leverage the scalability and flexibility of cloud services while ensuring the confidentiality and integrity of their sensitive information.