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

Enhancing Cloud Data Security with Modified RSA Encryption and Salt: A Comprehensive Review

Aarju
Mtech Scholar
SSGI Dinanagar

Harjinder Kaur
Assistant Professor
SSGI Dinanagar

Abstract:

Cloud computing has become increasingly popular, offering cost-effective resources on a pay-per-use basis. However, security concerns persist, as malicious users can disrupt cloud services. To address these issues, encryption mechanisms are widely employed. This review paper explores the use of Modified RSA with salt in cloud data encryption, enhancing security by introducing randomness. We summarize the current state of cloud security, various encryption techniques, and their applications. The primary problem addressed is the need to detect false positives in cloud data encryption using RSA without salt. Our study's objectives are to ensure secure data transmission in the cloud, employ RSA with salt for each data block, enhance cloud security, and evaluate this approach's performance. We discuss the methodology, which involves RSA with salt encryption for each data block, and potential future research directions. Ultimately, employing RSA with salt for block chaining enhances cloud security by providing reliable data storage and transmission.

Keywords: Cloud computing, security, encryption, RSA, salt, data transmission.

Introduction:

Cloud computing offers accessible and cost-effective resources on a pay-per-use basis. However, its widespread adoption has raised concerns about security, as malicious users can disrupt cloud services, affecting the entire user community. Security is a critical aspect of cloud computing, and one of the major concerns is ensuring the confidentiality and integrity of data in transit and at rest within the cloud infrastructure. Cloud computing has revolutionized the way businesses and individuals access and utilize computational resources. It offers the allure of virtually unlimited computing power, storage capacity, and software services on a pay-per-use basis, transforming traditional IT infrastructure into a dynamic, scalable, and cost-effective model. This paradigm shift has led to widespread adoption, making cloud computing a cornerstone of modern digital technology. However, this rapid growth and dependency on the cloud have also raised critical concerns, particularly in the realm of security.

Cloud computing operates on a simple premise – users can access computing resources and services hosted on remote servers, often referred to as data centers, via the internet. These resources encompass a spectrum of offerings, ranging from infrastructure-as-a-service (IaaS), where users can provision and manage virtualized hardware resources, to platform-as-a-service (PaaS), which provides an environment for developing and deploying applications, and software-as-a-service (SaaS), where users access software applications and services over the web. The ease of accessibility, scalability, and cost-efficiency have led to cloud services becoming integral to various industries, including e-commerce, finance, healthcare, and entertainment.

However, this very accessibility also exposes the cloud to a range of security challenges. As the cloud infrastructure hosts data and applications from multiple users and organizations, it becomes an attractive target for cyberattacks. Malicious actors constantly seek vulnerabilities and weaknesses in cloud systems to gain unauthorized access, steal sensitive data, disrupt services, or even launch large-scale attacks on cloud providers. The shared nature of cloud environments, where multiple tenants coexist on the same infrastructure, presents a unique set of security concerns. An attack on one tenant could potentially compromise the security and privacy of others. Consequently, ensuring robust security measures within the cloud is imperative to maintain user trust and the continued growth of cloud services.

This review paper delves into one of the fundamental aspects of cloud security - data encryption. Encryption plays a pivotal role in safeguarding data, both at rest and in transit, within the cloud. It transforms plaintext data into ciphertext using complex algorithms, rendering it unreadable without the corresponding decryption keys. Encryption provides a layer of security that is essential for preserving data confidentiality, integrity, and authenticity. While encryption is a well-established security practice, its application in the context of cloud computing comes with unique challenges and opportunities.

In this paper, we explore a modified RSA (Rivest-Shamir-Adleman) approach with the incorporation of salt (password-based encryption schemes) to enhance data encryption within the cloud. This modified RSA approach adds an extra layer of randomness and complexity to the encryption process, making it more resilient against brute-force attacks and other security threats. We delve into the intricacies of cloud computing, the layers within cloud architecture (IaaS, PaaS, and SaaS), and the major security issues that cloud services face. Additionally, we provide a comprehensive literature survey, examining various encryption techniques and their applications in cloud security. The subsequent chapters will focus on defining the problem, outlining the objectives, presenting the methodology, and concluding with insights into the future scope of this research. Ultimately, the aim of this paper is to contribute to the ongoing efforts to fortify cloud security and provide users with confidence in the integrity of their data stored and processed in the cloud.

Literature Survey:

Numerous encryption techniques have been proposed to address cloud security concerns. Researchers have explored methods such as counting bloom filters, block chaining, and DNA-based encryption. These approaches aim to enhance data security during transmission and storage in the cloud. Notable research includes the use of format-preserving encryption, DNA encryption, bitwise encryption, and RSA-based cryptographic techniques. These methods have shown varying degrees of success in improving cloud security, including data integrity and confidentiality.

Counting Bloom Filter for Cloud Security

In their work, Changsong et al. (2020) proposed a novel approach to tackle security concerns within cloud computing. Their focus was on data migration, a critical process where data is moved from its original source to a destination machine. During this migration, the security of the data is paramount. Changsong et al. introduced a counting bloom filter to address these concerns. The counting bloom filter serves as a data structure that allows efficient testing of whether an element belongs to a set or not. It is particularly useful for minimizing false positives during data migration. The authors evaluated their approach by considering key size and the speed at which data transmission occurs. This research sheds light on the importance of efficient data filtering mechanisms to maintain the integrity of data during cloud operations.

Enterprise Cloud Security and Block Chaining

Chandel, Ni, and Yang (2018) delved into the security challenges specific to enterprise cloud environments. Enterprise clouds often cater to a vast and diverse user community, making it challenging to predict the intentions and behaviors of users. The unpredictability of user actions poses a significant security challenge. To address this, Chandel et al. emphasized the importance of block chaining as an approach to detect and mitigate security threats. Block chaining involves linking together blocks of data, ensuring data integrity and security through cryptographic techniques. In their study, the authors considered parameters such as security and key size, highlighting the need for tailored security measures in enterprise cloud settings.

Format Preserving Encryption for Cloud Security

Gupta, Jain, and Agarwal (2018) proposed a format preserving encryption mechanism as a means to enhance cloud security. This encryption method involves altering data bits at specific positions, providing a fast and reliable approach to address security concerns within cloud computing. Format preserving encryption ensures that the format and structure of the data are preserved even after encryption, making it suitable for various applications and data types. This research demonstrates the importance of encryption techniques that not only provide security but also maintain data usability and format.

DNA Encryption for High Security Data Transmission

Hossain (2018) introduced a unique approach to achieve high security during data transmission within the cloud—DNA encryption. This method involves encrypting data using a pattern inspired by human DNA. The complexity and uniqueness of human DNA make it challenging for hackers to decrypt the transmitted data. By incorporating the principles of DNA into encryption, Hossain achieved both speed and reliability in cloud security. This research showcases the potential for unconventional encryption techniques to provide robust security solutions in cloud environments.

Parity-Based Secure Data Transmission

Cherillath Sukumaran and Mohammed (2018) explored a secure mechanism for data transmission within the cloud. Their approach involved converting plain text into ciphertext using a parity checker mechanism. The decryption process relied on even and odd parity checks, adding an extra layer of security during data transmission. This method offers a high degree of security and demonstrates the significance of innovative encryption strategies in protecting data within the cloud.

DNA-Based Binary Encryption for Multi-Level Security

Sohal and Sharma (2018) proposed a DNA-based binary encryption mechanism to enhance data security within the cloud. This approach employs multiple levels of security to ensure the safe transmission of data from source to destination. The research presented results in terms of key size and reliability, emphasizing the importance of evaluating encryption techniques based on multiple security parameters. The use of DNA as an inspiration for encryption highlights the potential for unconventional methods in cloud security.

Bitwise Encryption for Cloud Security

Diao et al. (2017) discussed a security concern within cloud computing and addressed it using bitwise encryption mechanisms. Bitwise encryption involves performing bitwise operations on data to enhance security. This research evaluated the results in terms of key size and reliability, underscoring the need for encryption strategies that effectively protect data integrity within the cloud.

Two-Level Security Mechanism for Data Transmission

Hammami, Brahmi, and Brahmi (2017) presented a comprehensive security mechanism for data transmission within the cloud. They considered security at both the source and destination ends, emphasizing the importance of end-to-end security. This two-level approach aimed to ensure encryption and reliability, further highlighting the significance of comprehensive security measures within cloud operations.

RSA Encryption and Modified ElGamal Algorithm for Data Sharing

A Dharini (2014) explored a secure data sharing scheme in cloud computing that utilized RSA encryption and a modified ElGamal algorithm. While this approach provided a level of security, the research noted that the security associated with the cloud was not particularly strong. This study highlighted the need for continuous improvement in cloud security measures.

Security Challenges and Solutions in Cloud Computing

BM Shereek (2014) investigated the security challenges and solutions in cloud computing, with a specific focus on the use of RSA encryption for secure data transmission. The research evaluated results based on classification accuracy and execution time. It emphasized the importance of efficient and effective encryption techniques to address cloud security concerns.

Privacy-Preserving Public Auditing Scheme for Cloud Storage

SR Lenka (2014) presented a privacy-preserving public auditing scheme for cloud storage, leveraging RSA-based cryptographic techniques. This approach provided a high level of security, particularly in terms of data privacy and integrity. The research highlighted the potential for cryptographic techniques to enhance cloud security.

RSA-Based Secure Data Sharing in Cloud Storage

V Masthanamma (2015) proposed an RSA-based secure data sharing scheme for cloud storage, with a focus on confidentiality and integrity. While this approach offered security, it was noted to be more suitable for smaller datasets. This research emphasized the need for scalability and efficiency in cloud security solutions.

Secure and Efficient Data Sharing Scheme for Cloud Storage

N Katende (2017) introduced a secure and efficient data sharing scheme for cloud storage, incorporating RSA encryption for secure key exchange. The implemented trust model provided benefits to cloud service providers in terms of execution time. This study underscored the importance of trust mechanisms in cloud security.

Efficient Cloud Data Sharing Scheme with RSA Encryption

S Gunasekaran (2015) proposed an efficient cloud data sharing scheme using RSA encryption and attribute-based access control. This approach combined RSA and AES-based algorithms for data transmission, providing a comprehensive security solution for cloud data sharing. The research highlighted the significance of access control in cloud security.

Homomorphic Encryption for Enhanced Cloud Security

Bhardwaj et al. (2016) presented a secure data sharing scheme for cloud storage that combined RSA encryption with homomorphic encryption techniques. This approach not only discussed the concept of security within the cloud but also provided safeguards against other attacks, such as distributed denial of service attacks. The research emphasized the need for holistic security measures within cloud computing.

Dynamic Hash Table for Cloud Data Storage Security

S Pallikonda (2017) proposed a cloud data storage security scheme using RSA encryption and a dynamic hash table for efficient and secure data access. This approach highlighted the need for security within cloud operations, emphasizing the importance of total throughput achieved as a performance metric.

The comparative table is given in table 1

Paper Title	**Encryption Technique**	**Security Focus**	**Notable Findings**
Changsong et al., 2020	Counting Bloom Filter	Data migration security	Efficient filtering to minimize false positives
Chandel, Ni, and Yang, 2018	Block Chaining	Enterprise cloud security	Emphasis on block chaining and key size
Gupta, Jain, and Agarwal, 2018	Format Preserving Encryption	Data bit alteration for security	Maintaining data format while enhancing security
Hossain, 2018	DNA Encryption	Data transmission security	Utilizing DNA-inspired encryption for high security
Cherillath Sukumaran and Mohammed, 2018	Parity Checker Mechanism	Data transmission security	Secure transmission using parity checks
Sohal and Sharma, 2018	DNA-Based Binary Encryption	Multi-level security	DNA-based encryption for enhanced security
Diao et al., 2017	Bitwise Encryption	Cloud security concerns	Bitwise operations for improved security

Paper Title	**Encryption Technique**	**Security Focus**	**Notable Findings**
Hammami, Brahmi, and Brahmi, 2017	Two-Level Security	Data transmission security	End-to-end security approach
A Dharini, 2014	RSA Encryption and Modified ElGamal Algorithm	Data sharing security	Emphasizing the need for stronger cloud security
BM Shereek, 2014	RSA Encryption	Cloud security challenges	Evaluating security based on accuracy and execution time
SR Lenka, 2014	RSA-Based Cryptographic Techniques	Data privacy and integrity	Privacy-preserving public auditing scheme
V Masthanamma, 2015	RSA-Based Encryption	Data sharing in cloud storage	Suitable for smaller datasets
N Katende, 2017	RSA Encryption	Secure key exchange	Trust model benefits for cloud service providers
S Gunasekaran, 2015	RSA and AES Encryption	Cloud data sharing	Comprehensive security solution for data sharing
Bhardwaj et al., 2016	RSA and Homomorphic Encryption	Secure data sharing	Safeguards against various attacks, including DDoS
S Pallikonda, 2017	RSA Encryption and Dynamic Hash Table	Cloud data storage	Emphasis on total throughput in cloud data storage

Table 1: Comparative Analysis

Problem Definition

The central problem addressed in this research is the security of data within cloud computing environments. While cloud computing offers unparalleled advantages, including scalability and cost-effectiveness, it also introduces significant security challenges. The primary concern is ensuring the confidentiality, integrity, and authenticity of data as it traverses the cloud infrastructure.

One specific aspect of this problem is the vulnerability of data during migration between cloud resources or from on-premises systems to the cloud. During these transitions, data is exposed to potential threats, such as unauthorized access or data breaches. Ensuring the security of data during these critical processes is paramount.

Another facet of the problem lies in the shared nature of cloud environments. Multiple users and organizations coexist on the same infrastructure, creating a potential risk for data leakage or unauthorized access. Protecting the data of one tenant from another is a significant challenge.

Furthermore, the diverse range of users with varying intentions and behaviors accessing cloud services raises uncertainty about security. Malicious actors may attempt to exploit vulnerabilities in the cloud infrastructure to compromise data.

To address these challenges, this research focuses on enhancing data security within the cloud through a modified RSA approach with the inclusion of salt (password-based encryption schemes). This approach aims to add an extra layer of randomness and complexity to encryption, making it more resilient against brute-force attacks and security threats, ultimately ensuring the safety of data within cloud computing environments.

Proposed Methodology

The flowchart showing the process of generating RSA keys. RSA is a public-key cryptography algorithm that uses two keys to encrypt and decrypt data: a public key and a private key. The public key is shared with everyone, while the private key is kept secret.

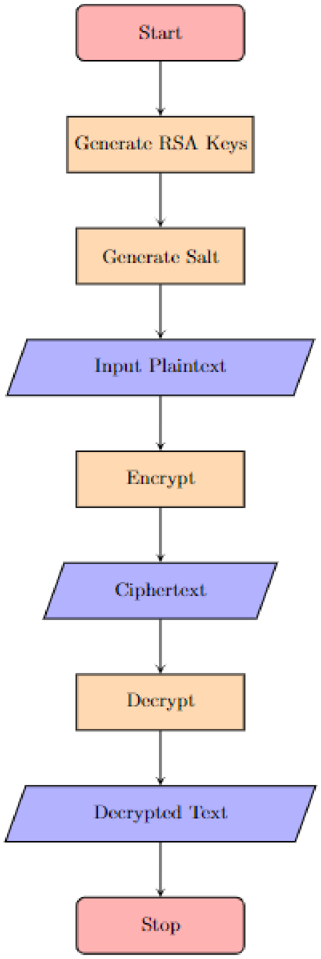

Figure 1: proposed Methodology

The flowchart shows the following steps:

Generate RSA keys: This step generates the public and private keys. This is done by generating two large prime numbers, p and q, and multiplying them together to create the modulus n. The public key is the modulus n and a public exponent e, which is typically set to 65537. The private key is the modular inverse of e modulo (p-1)(q-1).

Generate salt: This step generates a random salt value. A salt is a random string of characters that is added to the plaintext before it is encrypted. This makes it more difficult to crack the encryption, even if the attacker knows the algorithm used to encrypt the data.

Input plaintext: This step inputs the plaintext message that is to be encrypted.

Encrypt: This step encrypts the plaintext message using the public key. The ciphertext is then sent to the recipient.

Decrypt: This step decrypts the ciphertext using the private key. The decrypted plaintext message is then retrieved.

The flowchart also shows that the salt is decrypted before the plaintext message is retrieved. This is because the salt is used to generate a key that is used to decrypt the plaintext message.

RSA is a very secure encryption algorithm, and it is widely used in applications such as HTTPS, SSH, and PGP.

Here is a more detailed explanation of each step in the flowchart:

Generate RSA keys

To generate the RSA keys, we first need to generate two large prime numbers, p and q. These numbers should be at least 1024 bits long, and preferably longer. Once we have generated p and q, we can calculate the modulus n as follows:

n = p * q

The public key is then the modulus n and a public exponent e. The public exponent is typically set to 65537, but it can be any value that is greater than 1 and relatively prime to (p-1)(q-1).

The private key is the modular inverse of e modulo (p-1)(q-1). This means that it is the value d such that:

d * e = 1 mod (p-1)(q-1)

The modular inverse can be calculated using the extended Euclidean algorithm.

Generate salt

The salt is a random string of characters that is added to the plaintext before it is encrypted. This makes it more difficult to crack the encryption, even if the attacker knows the algorithm used to encrypt the data.

The salt can be generated using a random number generator. It is important to use a strong random number generator, such as a cryptographically secure pseudorandom number generator (CSPRNG).

Input plaintext

The plaintext message is the message that is to be encrypted. It can be any type of data, such as text, binary data, or images.

Encrypt

To encrypt the plaintext message, we first convert it to a number. This can be done by using a padding scheme, such as PKCS#1 v1.5 padding.

Once the plaintext message has been converted to a number, we can encrypt it using the public key. The ciphertext is then calculated as follows:

ciphertext = plaintext ^ e mod n

Decrypt

To decrypt the ciphertext, we first need to convert it to a number. This is done using the same padding scheme that was used to encrypt the plaintext message.

Once the ciphertext has been converted to a number, we can decrypt it using the private key. The plaintext message is then retrieved as follows:

plaintext = ciphertext ^ d mod n

Decrypt salt

The salt is decrypted before the plaintext message is retrieved. This is because the salt is used to generate a key that is used to decrypt the plaintext message.

The salt is decrypted using the private key. The decryption process is the same as the decryption process for the plaintext message.

Once the salt has been decrypted, it is used to generate a key that is used to decrypt the plaintext message. The plaintext message is then retrieved using the key.

RSA is a very secure encryption algorithm, and it is widely used in applications such as HTTPS, SSH, and PGP.

Conclusion

In this review paper, we have explored various approaches to enhancing the security of data within cloud computing environments. Cloud computing offers immense benefits in terms of resource scalability and cost-efficiency, but it also introduces significant security challenges. The primary concern revolves around safeguarding data during its lifecycle within the cloud, especially during migration, sharing, and storage.

To address these challenges, researchers have proposed innovative encryption techniques and security mechanisms. These approaches include Counting Bloom Filters, Block Chaining, Format Preserving Encryption, DNA-based Encryption, Parity Checker Mechanisms, Bitwise Encryption, and various implementations of RSA encryption.

The Modified RSA approach with the inclusion of salt emerges as a promising solution to strengthen cloud data security. By adding randomness and complexity to encryption, it becomes more resilient against brute-force attacks and security threats. This approach ensures the confidentiality and integrity of data during transmission and storage in the cloud.

As cloud computing continues to play a pivotal role in modern IT infrastructure, securing sensitive data within the cloud is of paramount importance. The various techniques discussed in this literature survey contribute to advancing the field of cloud security, each with its unique strengths and applications.

The future of cloud security lies in the continuous exploration of innovative encryption techniques, robust authentication mechanisms, and proactive threat detection systems. As cloud technology evolves, so too must our security measures to protect against emerging threats and vulnerabilities. The pursuit of secure cloud computing remains a dynamic and evolving field, vital for maintaining the trust and integrity of digital ecosystems.

References

A Dharini, R.S.D.I.C. (2014) ‘Data security for cloud computing using RSA with magic square algorithm’, Int J Innov Sci Res, 11(2), pp. 439–444.

Bhardwaj, A. et al. (2016) ‘Security algorithms for cloud computing’, Procedia Comput Sci, 85, pp. 535–542. Available at: https://doi.org/10.1016/j.procs.2016.05.215.

BM Shereek, Z.M.S.Y. (2014) ‘Improve cloud computing security using RSA algorithm with Fermat’s little theorem’, IOSR J Eng, 4(2), pp. 1–8. Available at: https://doi.org/10.9790/3021-04260108.

Chandel, S., Ni, T.Y. and Yang, G. (2018) ‘Enterprise cloud: Its growth & security challenges in china’, Proceedings - 5th IEEE International Conference on Cyber Security and Cloud Computing and 4th IEEE International Conference on Edge Computing and Scalable Cloud, CSCloud/EdgeCom 2018, pp. 144–152. Available at: https://doi.org/10.1109/CSCloud/EdgeCom.2018.00034.

Changsong, Y. et al. (2020) ‘Secure data transfer and deletion from counting bloom filter in cloud computing’, Chinese Journal of Electronics, 29(2), pp. 273–280. Available at: https://doi.org/10.1049/cje.2020.02.015.

Cherillath Sukumaran, S. and Mohammed, M. (2018) ‘DNA Cryptography for Secure Data Storage in Cloud’, International Journal of Network Security, 20(3), pp. 447–454. Available at: https://doi.org/10.6633/IJNS.201805.20(3).06.

Diao, Z. et al. (2017) ‘Study on Data Security Policy Based on Cloud Storage’, Proceedings - 3rd IEEE International Conference on Big Data Security on Cloud, BigDataSecurity 2017, 3rd IEEE International Conference on High Performance and Smart Computing, HPSC 2017 and 2nd IEEE International Conference on Intelligent Data and Securit, pp. 145–149. Available at: https://doi.org/10.1109/BigDataSecurity.2017.12.

Gupta, S., Jain, S. and Agarwal, M. (2018) ‘Ensuring Data Security in Databases Using Format Preserving Encryption’, Proceedings of the 8th International Conference Confluence 2018 on Cloud Computing, Data Science and Engineering, Confluence 2018, pp. 214–218. Available at: https://doi.org/10.1109/CONFLUENCE.2018.8442626.

Hammami, H., Brahmi, H. and Brahmi, I. (2017) ‘A Security Approach for Data Migration in Cloud Computing Based on Human Genetics’, IEEE Access, pp. 384–396. Available at: https://doi.org/10.1007/978-3-319-65930-5.

Hossain, F. (2018) 'Enhanced Secure Cloud Computing with DNA Encryption based Strong Authentication Methodology for Resources in ICT World', IEEE ACCESS [Preprint], (January 2015).

V Masthanamma, G.L.P. (2015) 'An efficient data security in cloud computing using the RSA encryption process algorithm', Int J Innov Res Sci Eng Technol, 4(3), pp. 1441–1445.

N Katende, C.W.A.K. (2017) 'Enhancing trust in cloud computing using MD5 hashing algorithm and RSA encryption standard', Int J Sci Eng Res, 8(3), pp. 550–564.

S Gunasekaran, M.L. (2015) 'A review on enhancing data security in cloud computing using RSA and AES algorithms', Int J Adv Eng Res, 9(4), pp. 1–7.

Sohal, M. and Sharma, S. (2018) 'BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing', Journal of King Saud University - Computer and Information Sciences [Preprint]. Available at: https://doi.org/10.1016/j.jksuci.2018.09.024.

S Pallikonda, S.Y.R. (2017) 'Securing cloud data using encryption algorithms', Int J Adv Res Sci Eng, 6(11), pp. 1188–1193.

SR Lenka, B.N. (2014) 'Enhancing data security in cloud computing using RSA encryption and MD5 algorithm', Int J Comput Sci Trends Technol, 2(3), pp. 60–64.

JETIR