EasyChair Preprint

№ 12755

Hybrid Cryptographic Model to Enhance the Security in the Cloud

S.L.V.S.Likhita Majji, Vaishnavi Gunapati, Sai Teja Allu, Ravindranath Kurmapu and Vipul Dabhi

March 27, 2024

Hybrid Cryptography Model To Enhance The Security In Cloud

M S.L V S Likhita
Department of Computer Science Engineering and Technology
Parul Institute of Engineering and Technology
Vadodara, India
200303124337@paruluniversity.ac.in

Gunapati Vaishnavi
Department of Computer Science Engineering and Technology
Parul Institute of Engineering and Technology
Vadodara, India
200303124236@paruluniversity.ac.in

Allu Sai Teja
Department of Computer Science Engineering and Technology
Parul Institute of Engineering and Technology
Vadodara, India
200303124109@paruluniversity.ac.in

K Ravindranath
Department of Computer Science Engineering and Technology
Parul Institute of Engineering and Technology
Vadodara, India
200303124318@paruluniversity.ac.in

Prof. Dr.Vipul Dabhi
Department of Computer Science Engineering and Technology
Parul Institute of Engineering and Technology
Vadodara, India
vipulkumar.dabhi23496@paruluniversity.ac.in

Abstract*— The world is developing into a highly technological place where technology wants to optimally improve space and cost. the way for this growth cloud is the acquisition of their richness in technology by the way they render the services. The cloud is the demand of the hour, which needs to be protected. Each data entity residing in the cloud must be authenticated so that there is no escape from critical and sensitive information. One of the primary requirements that are challenging in the cloud sector is the factor of security for the humongous data in the cloud. The cloud network is not a constrained homogeneous one, but a very dynamic and heterogeneous which makes it fall to threats. Data encryption is a significant part of security that has to be concentrated on improving security mechanisms. This research paper combines various cryptographic algorithms such as RSA with a digital signature, SHA-3, and Brotli. This is a hybrid approach where RSA with a digital signature is used for confidentiality and authentication, hashing is done by SHA-3 and Brotli will be used for compression. This paper also provides an analysis of how the various algorithms are selected depending on various factors and the bound through which will provide a better hybrid approach. In today's rapidly evolving technological landscape, the cloud plays a pivotal role in driving innovation and efficiency by offering scalable and cost-effective services. However, with the increasing reliance on cloud computing, ensuring the security of data becomes paramount. The dynamic and heterogeneous nature of cloud networks presents unique challenges, making it susceptible to various threats. Therefore, enhancing security mechanisms, particularly through data encryption, is imperative to safeguard critical and sensitive information stored in the cloud. This research paper proposes a hybrid approach that combines multiple cryptographic algorithms, including RSA with a digital signature for confidentiality and authentication, SHA-3 for hashing, and Brotli for compression. By leveraging the strengths of these algorithms and analyzing their suitability based on different factors, this approach aims to enhance the overall security of data in the cloud while optimizing space and cost efficiencies.

Keywords—Encryption Algorithms, Cloud Security, Data Security, Information Security

I. INTRODUCTION

Cloud computing is an example that reduces services in several forms to cloud users. The ease of its usage, access, and scalability are the significant features of the cloud. Cloud computing is a massive arena that offers numerous benefits to organizations, but these profits can become a drawback if appropriate information security and privacy are not ensured. The breach of safety in cloud services can result in high costs of failure. There is an important part of the cloud where the security is supposed to be realistic data storage. Cloud computing has revolutionized the way services are delivered to users by providing access to resources over the Internet, thereby reducing the need for on-premises infrastructure and management. The cloud offers unparalleled ease of usage, allowing users to access services and data from anywhere with an internet connection. Scalability is another key advantage, as cloud resources can be easily scaled up or down to meet changing demands, providing organizations with flexibility and cost-efficiency.

Despite its numerous benefits, the widespread adoption of cloud computing also introduces new challenges, particularly in terms of information security and privacy. Organizations entrust sensitive data and critical operations to cloud service providers, making it imperative to ensure robust security measures are in place to protect against threats such as data breaches and unauthorized access. Failure to adequately address security concerns can lead to significant financial losses, damage to reputation, and legal liabilities.

One of the critical aspects of cloud security revolves around data storage. As organizations increasingly rely on cloud storage solutions to store vast amounts of data, ensuring the confidentiality, integrity, and availability of this data becomes paramount. Realistic data storage solutions in the cloud must not only provide secure storage environments but also implement robust encryption, access controls, and monitoring mechanisms to safeguard against potential threats. Additionally, compliance with regulatory requirements and industry standards adds another layer of complexity to cloud data storage, necessitating careful planning and implementation of security measures. Overall, achieving realistic data storage in the cloud requires a comprehensive approach that addresses both technical and organizational aspects of security.

II. LITERATURE SURVEY

Security has been a serious approach in the cloud, which prevents confidentiality, mechanisms of integrity, and irrefutability. The three most popular algorithms used for protection are AES, Blowfish, and RSA. [1] The comparison is made according to the Java implementation The programming language and the RAM used played an important role in determining the performance solution for comparison. Data security has been a major problem in the cloud as several algorithms set up to ensure adequate security. Especially the Blowfish, RSA, and AES algorithms compared to their purpose. Protection, integrity, and authentication are

provided by the cloud's elliptic curve encryption technology storage Cloud storage security is also a major concern when it comes to data access from unauthorized users and data modification is a major problem. The hybrid security algorithm is it has been proposed to protect data in the cloud because it can be accessed anywhere, anytime. [4] ECDSA and SHA-256 are used to obtain a hybrid method for encryption and decryption to protect data in the cloud. Identity-based hybrid encryption is proposed by combining ECC with RSA to improve the security of outsourced data. Algorithms used in cloud security were reviewed and found that the biggest concern of the cloud is security and privacy. Security in the cloud is a critical aspect that ensures confidentiality, integrity, and irrefutability of data. Various algorithms are employed to achieve these goals, with AES, Blowfish, and RSA being among the most popular choices. The performance of these algorithms can vary based on factors such as the programming language used for implementation and the amount of RAM utilized. Data security remains a significant challenge in the cloud environment, necessitating robust protection mechanisms. AES, Blowfish, and RSA algorithms are often compared concerning their effectiveness in providing security for different purposes. Each algorithm has its strengths and weaknesses, making it essential to choose the most suitable one based on specific security requirements. Cloud storage security is another critical concern, particularly regarding unauthorized access and data modification. To address these issues, hybrid security algorithms have been proposed.

These algorithms combine different encryption techniques, such as ECDSA and SHA-256, to enhance data protection in the cloud. Hybrid methods offer flexibility and accessibility, allowing data to be securely accessed from anywhere and at any time. Identity-based hybrid encryption is a promising approach that combines elliptic curve cryptography (ECC) with RSA to further improve data security in the cloud. By leveraging the strengths of both ECC and RSA, this method enhances the confidentiality and integrity of outsourced data. Overall, the security and privacy of data in the cloud remain paramount concerns. Continued research and development in encryption algorithms and security protocols are essential to address evolving threats and ensure the protection of sensitive information stored in cloud environments. As cloud computing continues to proliferate and become increasingly integrated into various aspects of modern life, the need for robust security measures becomes more pressing than ever. The dynamic nature of the cloud, coupled with the vast amounts of data stored and processed within it, creates a fertile ground for potential security breaches and vulnerabilities.

Therefore, cloud service providers and organizations alike must prioritize security and implement comprehensive strategies to safeguard sensitive information. In addition to the encryption algorithms mentioned, other advanced security techniques such as access control, multi-factor authentication, and intrusion detection systems are also essential components of a holistic cloud security framework. Access control mechanisms ensure that only authorized users have access to specific resources and data, thereby minimizing the risk of unauthorized access and data breaches. Multi-factor authentication adds an extra layer of security by requiring users to provide multiple forms of identification, such as a password and a unique token or biometric verification. Furthermore, the implementation of robust intrusion detection and prevention systems helps detect and mitigate potential security threats in real time, thereby minimizing the impact of security incidents on cloud infrastructure and data. Regular security audits and compliance assessments are also crucial to ensure that cloud environments adhere to industry standards and regulatory requirements, thereby mitigating legal and regulatory risks. Ultimately, achieving robust security in the cloud requires a collaborative effort between cloud service providers, organizations, and cybersecurity experts. By adopting a proactive and holistic approach to security, leveraging advanced encryption algorithms, and implementing comprehensive security measures, organizations can mitigate the risks associated with cloud computing and ensure the confidentiality, integrity, and availability of their data in the cloud.

Additionally, as the landscape of cybersecurity evolves, organizations need to stay abreast of emerging threats and vulnerabilities specific to cloud environments. Threat intelligence gathering and analysis play a crucial role in identifying potential risks and vulnerabilities, enabling organizations to preemptively address security issues before they escalate into full-fledged breaches. Moreover, ongoing employee training and awareness programs are vital for promoting a security-conscious culture within organizations. Human error remains one of the leading causes of security breaches, making it imperative for employees to understand their role in maintaining the security of cloud-based systems and data. By educating employees about best practices for data handling, password management, and recognizing phishing attempts, organizations can significantly reduce the likelihood of security incidents stemming from human error. Collaboration and information-sharing within the cybersecurity community also play a critical role in bolstering cloud security. By participating in industry forums, sharing threat intelligence, and collaborating with peers and security experts, organizations can leverage collective knowledge and insights to enhance their security posture and stay ahead of evolving threats.

Lastly, as the regulatory landscape governing data privacy and security continues to evolve, organizations must ensure compliance with relevant regulations and standards governing the protection of data stored and processed in the cloud. Failure to comply with regulatory requirements not only exposes organizations to legal and financial liabilities but also undermines trust and credibility among customers and stakeholders. In conclusion, while the adoption of cloud computing offers numerous benefits in terms of scalability, flexibility, and cost-efficiency, it also introduces inherent security challenges that must be addressed effectively. By adopting a multi-layered approach to security, leveraging advanced encryption algorithms, implementing robust security measures, staying informed about emerging threats, fostering a security-conscious culture, and ensuring regulatory compliance, organizations can mitigate the risks associated with cloud computing and safeguard their sensitive data effectively. And distribution, rotation, and disposal of cryptographic keys used for encryption and decryption. Key management standards, such as those outlined by NIST (National Institute of Standards and Technology) or ISO (International Organization for Standardization), provide guidelines and best practices for managing cryptographic keys effectively. These standards address key generation techniques, key storage mechanisms, access control policies, key rotation schedules, and key disposal procedures to minimize the risk of key compromise and unauthorized access to encrypted data. Furthermore, service providers should implement robust access controls and authentication mechanisms to restrict access to cryptographic keys based on the principle of least privilege. Only authorized personnel with a legitimate need should be granted access to encryption keys, and strong authentication methods, such as multi-factor authentication, should be enforced to prevent unauthorized key access. This involves regularly assessing the effectiveness of existing security measures, identifying areas for improvement, and promptly implementing necessary changes to mitigate emerging threats.

III. SOFTWARE AND ALGORITHMS USED

1. PAAS

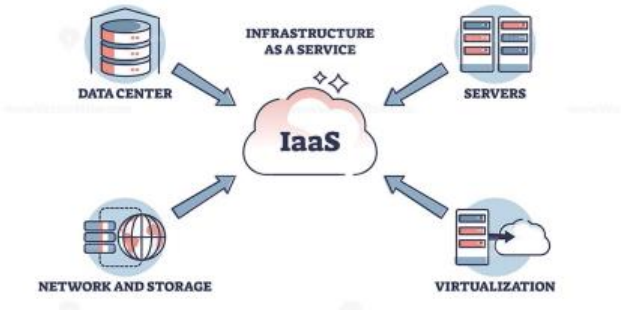
In a service offering multiple operating systems, web server technologies, and execution environments, ensuring the security and privacy of customer data becomes even more challenging due to the distributed nature of the infrastructure. Customers may not always be aware of where their data is stored or processed, making it imperative for the service provider to implement robust encryption practices and adhere to stringent key management standards. Proper encryption of data at rest and in transit is fundamental to protecting sensitive information from unauthorized access and interception. Advanced encryption algorithms, such as AES (Advanced Encryption Standard) with strong key lengths, should be employed to encrypt data stored in databases, file systems, and other storage repositories. Additionally, encryption should be applied to data transmitted over networks using secure protocols such as TLS (Transport Layer Security) to prevent eavesdropping and man-in-the-middle attacks. However, encryption alone is not sufficient to guarantee the security of customer data in a cloud environment. Effective key management is equally critical to ensure the confidentiality and integrity of encrypted data. Key management involves the secure generation, storage, distribution, rotation, and disposal of cryptographic keys used for encryption and decryption.

Key management standards, such as those outlined by NIST (National Institute of Standards and Technology) or ISO (International Organization for Standardization), provide guidelines and best practices for managing cryptographic keys effectively. These standards address key generation techniques, key storage mechanisms, access control policies, key rotation schedules, and key disposal procedures to minimize the risk of key compromise and unauthorized access to encrypted data. This causes electrons to move through the tag's antenna and subsequently powers the chip. The chip then responds by sending its stored information back to the reader in the form of another radio signal. This is called aback scatter. The reader detects and interprets this backscatter and sends the data to a computer or microcontroller.

2. IAAS

In Infrastructure as a Service (IaaS) models, where customers are provided with software stacks, middleware, and applications, the responsibility for securing the infrastructure and data often falls on both the service provider and the customer. Given the shared responsibility model of cloud computing, both parties need to collaborate and adapt their security policies to address the unique challenges and requirements of the cloud environment. Standard encryption mechanisms play a crucial role in ensuring the security and confidentiality of data stored within the IaaS infrastructure and during communication between users and the cloud environment. Data encryption helps protect sensitive information from unauthorized access, whether it's at rest within storage systems or in transit across networks. By encrypting data using robust encryption algorithms and ensuring proper key management practices, organizations can mitigate the risk of data breaches and unauthorized disclosures.

Moreover, in addition to encryption, implementing access controls, network segmentation, intrusion detection systems, and regular security audits are essential components of a comprehensive security strategy for IaaS environments. Access controls help enforce least privilege principles, ensuring that only authorized users and applications have access to specific resources and data. Network segmentation isolates sensitive workloads and data, reducing the potential attack surface and limiting the impact of security incidents. Intrusion detection systems help detect and respond to suspicious activities or anomalies in real-time, while regular security audits ensure compliance with industry regulations and best practices, as well as identifying and remedying potential security gaps. By integrating standard encryption mechanisms with a holistic security approach encompassing access controls, network segmentation, intrusion detection, and ongoing security assessments, organizations can enhance the security posture of their IaaS deployments and effectively protect their data and infrastructure in the cloud. This collaborative effort between cloud service providers and customers is essential for maintaining trust, mitigating risks, and ensuring the confidentiality, integrity, and availability of data in the cloud environment.



3. SAAS

Ensuring confidentiality, integrity, and availability (CIA) of data is paramount for any cloud service provider offering software-linked applications to users. To achieve this, clients must be empowered to select encryption standards that align with their specific security requirements and compliance obligations. This flexibility enables clients to tailor their security measures to the sensitivity of their data and the regulatory environment in which they operate. Data encryption standards applied to data at rest and in motion play a crucial role in safeguarding data against unauthorized access and interception. Encryption ensures that even if data is compromised, it remains unreadable to unauthorized parties without the corresponding decryption keys. By allowing clients to choose encryption standards, the cloud service provider enables them to implement encryption protocols that meet their security and compliance needs.

Moreover, clients must be made aware of the diverse nature of their data, understanding that different types of data may require different levels of protection and handling. Sensitivity classifications, such as personally identifiable information (PII), intellectual property, or proprietary business data, guide clients in determining appropriate security measures. By being cognizant of the nuances within their data landscape, clients can prioritize security resources effectively and apply appropriate encryption and access controls accordingly. In addition to encryption, clients should also consider other security measures, such as data loss prevention (DLP), intrusion detection systems (IDS),

and access controls, to enhance the overall security posture of their cloud-based applications. Regular security audits and compliance assessments help ensure that security measures remain effective and aligned with evolving threats and regulatory requirements. By empowering clients to select encryption standards, raising awareness of data sensitivity, and promoting best practices in data security, cloud service providers can support their clients in achieving the desired levels of confidentiality, integrity, and availability for their software-linked applications. This collaborative approach fosters trust between providers and clients and demonstrates a commitment to protecting sensitive data in the cloud environment.

4. BROTLI ALGORITHM

This algorithm is a combination of Huffman coding, second-order context modeling, and LZ77. Unlike most general-purpose compression algorithms, Brotli has a high compression ratio and uses a predefined 120 KB dictionary containing 13,000 common words, phrases, and other substrings derived from large corpus texts and HTML documents in addition to a dynamically filled/sliding window dictionary Here we use the Brotli algorithm to compress our data because it is much faster compared to other compression algorithms and also after compression significantly reduces the size of the raw data. That's why we pack our data here first with the Brotli algorithm and then encrypt it. Brotli algorithm for compression of our data because it is much faster compared to other compression algorithms and after compression, it also greatly reduces the original data size. This helps to reduce space and also helps in fast information serving. The decision to use the Brotli compression algorithm for data compression before encryption aligns with best practices in data security and efficiency. By compressing data before encryption, organizations can significantly reduce the size of the data being transmitted or stored, resulting in faster transfer speeds and optimized use of storage resources. Brotli's high compression ratio, combined with its relatively fast processing speed, makes it particularly well-suited for this purpose.

Furthermore, the use of a predefined dictionary containing common words, phrases, and substrings, along with a dynamically filled/sliding window dictionary, enhances the compression efficiency of Brotli. This dictionary-based approach allows Brotli to achieve superior compression ratios by identifying and encoding repetitive patterns within the data, thereby further reducing the size of the compressed output. Encrypting data after compression adds layer of security, ensuring that even if the compressed data were intercepted, it would remain unreadable without the decryption key. By employing encryption alongside compression, organizations can protect sensitive information from unauthorized access and mitigate the risk of data breaches or leaks. Overall, the combination of Brotli compression and encryption offers a powerful solution for optimizing data storage and transmission while maintaining robust security. By leveraging the speed and efficiency of Brotli compression and the protective measures of encryption, organizations can effectively balance the need for space optimization and data security in their information management strategies. This approach not only facilitates faster information serving but also enhances data protection and confidentiality, contributing to overall data integrity and security posture.

5. SHA(Secure Hashing Algorithm)

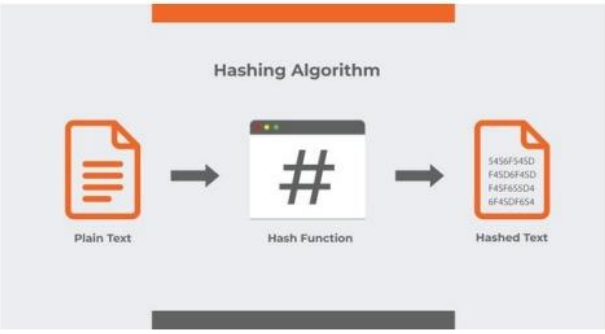
SHA-3 employs a mushroom structure, where the input size varies, and the output is consistently fixed. After receiving the input, it undergoes XOR operations before being entirely transformed by the permutation function. The outcome is retrieved from a specific subset in an alternating manner with the mode conversion function. SHA-3 is chosen for compression due to its quicker hardware implementation and increased resilience against breaking compared to alternative algorithms. Introduced on August 5, 2015, SHA-3, standing for Secure Hash Algorithm 3, is the latest addition to the family of Secure Hash Algorithm standards. Despite belonging to the same set of standards, SHA-3 distinguishes itself internally from structures like MD5, SHA-1, and SHA-2. SHA-3 is based on the mushroom function instead of Merkle-Damgrd, so it should not be vulnerable to the same attacks as previous SHA algorithms. Specifically, it is not vulnerable to extension-of-length attacks that affect all Merkle-Damgrd (MD) hashes such as MD5, SHA-1, and SHA-2.SHA-3, also known as Keccak, was designed by Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Its design process involved an open competition initiated by the National Institute of Standards and Technology (NIST) to develop a new cryptographic hash function. The competition attracted a significant amount of attention from the cryptographic community, and Keccak emerged as the winner.One of the key differences between SHA-3 and its predecessors lies in its underlying structure. While MD5, SHA-1, and SHA-2 are based on the Merkle-Damgård construction, SHA-3 employs the sponge construction. This structural variance fundamentally changes the behavior of the hash function and provides resistance against certain types of attacks.

The sponge construction used in SHA-3 consists of two main components: the absorbing phase and the squeezing phase. During the absorbing phase, the input message is XORed with the internal state of the sponge and then transformed using a permutation function. This process is repeated until the entire message has been absorbed. In the squeezing phase, the output hash is obtained by repeatedly applying the permutation function and extracting portions of the internal state. The use of a permutation function in SHA-3 ensures that the output is determined solely by the input message and not by any intermediate state. This property makes SHA-3 resistant to length extension attacks, which have been a concern for hash functions based on the Merkle-Damgård construction.

Furthermore, the sponge construction allows for flexibility in the choice of hash output length, making SHA-3 suitable for a wide range of applications. It also enables efficient hardware implementation, contributing to its speed and versatility.

Overall, SHA-3 represents a significant advancement in cryptographic hash functions, offering improved security

properties and performance characteristics compared to its predecessors. Its resistance to known attacks and its adaptability make it a valuable tool for ensuring data integrity and authenticity in various security-sensitive applications.



6. RSA(RIVEST SHAMIR ADLEMAN)

Formulated in 1977 by Rivest, Shamir, and Adelman, the RSA algorithm employs a dual-key system. It involves a public key, utilized by the sender for encryption, and a private key, employed by the recipient for decrypting the message. RSA entails mathematical operations to derive both the encryption and decryption keys (E and D). Subsequently, the ciphertext and plaintext can be computed through a straightforward formula.

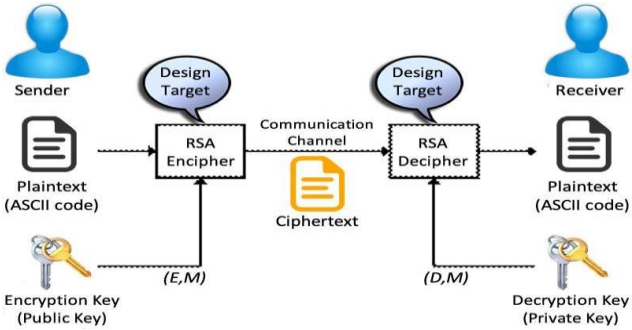
• C=MEmod(n)

• P=MDmod(n)(1)(2)

RSA operates on the principles of asymmetric cryptography, relying on prime numbers in its algorithm. The private key generated by RSA is constructed using these prime numbers, and the level of encryption strength is directly tied to the key size. The complexity of RSA encryption poses a challenge to unauthorized access, particularly for those lacking a deep understanding of prime numbers. Over the years, attempts to crack RSA have involved the generation and testing of

various large prime numbers. RSA encryption, with its robust mathematical foundation, has been a staple in secure communication since its inception. Its principles are rooted in the strength of prime numbers and the difficulty of factoring large integers, which underpins its security.

The process of RSA encryption begins with the selection of two distinct large prime numbers, p and q. These primes are multiplied to obtain the modulus n, which serves as the backbone for both the public and private keys. The totient function (n), derived from p and q, is then calculated. Once the parameters are established, an encryption exponent e is chosen, typically a small prime number such as 65537, owing to its efficient computation and good cryptographic properties. This exponent, along with the modulus n, forms the public key, allowing anyone to encrypt messages intended for the recipient. The private key is derived from the chosen primes and the encryption exponent. Specifically, it involves finding the decryption exponent d, which satisfies the equation. $d \times e \equiv 1 \bmod(n)$. This process is typically accomplished using the extended Euclidean algorithm. With the public and private keys in place, RSA encryption proceeds by raising the plaintext message M to the power of the encryption exponent e modulo n. The resulting ciphertext C is then transmitted to the recipient, who utilizes their private key to decrypt the message. Decryption involves raising the ciphertext to the power of the decryption exponent d modulo n, yielding the original plaintext. The security of RSA encryption hinges on the challenge of factoring the modulus n into its constituent prime factors p and q. As long as the primes are sufficiently large and chosen randomly, the computational effort required to factor n becomes prohibitive, safeguarding the confidentiality of the encrypted messages. In practical applications, RSA encryption serves as a cornerstone of secure communication protocols, enabling the exchange of sensitive information over insecure channels with confidence. Its reliance on asymmetric keys and the mathematical properties of prime numbers ensures robust protection against unauthorized access and interception, fulfilling the objectives of minimizing overhead, enhancing performance, and ensuring privacy in digital communication.

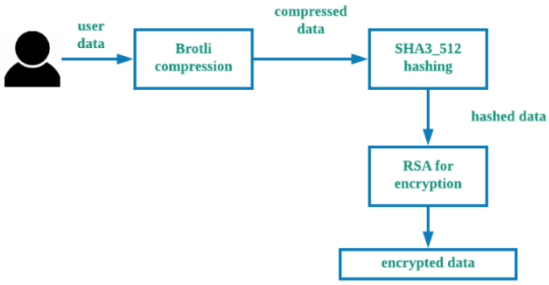


IV. IMPLEMENTATION

Initially, we installed the RSA package for the terminal. Subsequently, we initiated the process by implementing two auxiliary methods designed for the generation of public and private keys. These keys form a pair comprising both public and private components, and following their generation, they are then saved to respective files. Once the key pair generation process is complete, the public and private keys are typically saved to separate files for future use. This ensures that the keys can be easily retrieved and utilized by the encryption and decryption functions as needed. Additionally, securely storing the private key file is crucial, as it should only be accessible to authorized users to maintain the confidentiality and integrity of encrypted data. After the keys have been generated and saved, they can be used for encrypting and decrypting data using the RSA algorithm. This enables secure communication and data transmission over insecure channels, as only the intended recipient possessing the private key can decrypt the encrypted messages. Overall, the installation of the RSA package, generation of key pairs, and subsequent saving of keys are essential steps in implementing RSA encryption and decryption functionality within a system.

In cryptography, encryption is the process of converting plaintext data into ciphertext, which is unintelligible without the appropriate decryption key. The encryption method typically involves manipulating the plaintext using a specific algorithm and a secret key. In the context of RSA encryption, the encryption method takes the plaintext message and the recipient's public key as input, producing ciphertext that can only be decrypted by the corresponding private key. To create an encryption method, we first need to convert the plaintext message into a format suitable for encryption. In many cases, plaintext messages are represented as strings of characters, which can be encoded into a numerical format, such as ASCII (American Standard Code for Information Interchange). This

encoding scheme assigns a unique numerical value to each character in the message, allowing it to be processed by cryptographic algorithms. Implementing an encryption method requires careful consideration of security measures, such as padding schemes to ensure the confidentiality and integrity of the encrypted data. Additionally, it's essential to handle errors and edge cases gracefully to prevent vulnerabilities and ensure the robustness of the encryption process. Overall, the encryption method plays a crucial role in secure communication systems, enabling parties to exchange sensitive information confidentially over insecure channels. By leveraging cryptographic algorithms and encryption keys, plaintext messages can be transformed into ciphertext, providing confidentiality and privacy protection in digital communication.



In cryptography, decryption is the process of converting ciphertext back into its original plaintext form using the appropriate decryption key. In the context of RSA encryption, decryption is typically performed using the recipient's private key, which corresponds to the public key used for encryption. The decryption method takes the encrypted message and the recipient's private key as input, and it applies the inverse operation of the encryption process to retrieve the original plaintext message. To create a decryption method, we need to reverse the encryption process applied to the ciphertext.

In RSA encryption, this involves raising the encrypted message to the power of the decryption exponent modulo the modulus n derived from the recipient's private key. The resulting numerical representation is then decoded back into the original plaintext format, typically using ASCII or another encoding scheme, to obtain the decrypted message. Implementing a decryption method requires attention to security considerations, such as verifying the integrity of the ciphertext and ensuring that only authorized parties with access to the correct private key can decrypt the message. It's crucial to handle errors and edge cases effectively to prevent vulnerabilities, such as padding oracle attacks or chosen ciphertext attacks, which could compromise the confidentiality of the decrypted data. Overall, the decryption method complements the encryption process, enabling secure communication systems to securely transmit and receive sensitive information. By leveraging cryptographic algorithms and decryption keys, ciphertext can be transformed back into plaintext, providing confidentiality and privacy protection for digital communication. Moreover, the encryption-decryption process enables secure communication over insecure channels, such as the Internet, where data may be susceptible to interception or eavesdropping. By encrypting messages before transmission and decrypting them upon receipt, individuals and organizations can safeguard their communications against potential threats and attacks, thereby preserving the privacy and integrity of their data.

V. RESULT

In this process, after encrypting and decrypting the message using RSA encryption and decryption, respectively, the next step involves determining the original size of the data and measuring the time it takes to compress the message using the Brotli compression algorithm. This step provides insight into the efficiency of the compression process and the resulting reduction in data size. Once the compression process is completed, the original data size and the duration of compression are recorded. The output size after compression is also noted, providing a comparison between the original data size and the compressed data size. This information helps evaluate the effectiveness of the compression algorithm in reducing the size of the encrypted message while maintaining data integrity. Following compression, the decompression process is initiated to restore the message to its original form. The original data size and the time taken for decompression are measured and recorded. This step ensures that the decompressed message matches the original plaintext message, verifying the integrity of the compression and decompression processes. In summary, this process involves encrypting and decrypting a message using RSA encryption, followed by measuring the original data size, compressing the message using Brotli compression, recording the compression time and resulting output size, decompressing the message, and measuring the decompression time. Through these steps, the effectiveness of the compression algorithm in reducing the size of encrypted messages while maintaining data integrity is assessed.

CONCLUSION

The developed hybrid model enhances the security of extensive data stored in the cloud by implementing a hybrid algorithm. Compression is applied to RSA, resulting in a more efficient process where only text, not the entire message, is compressed. This optimization leads to quicker and more reliable access, heightening the security level and expediting authentication in the RSA algorithm. In this proposed approach, an additional layer of security is introduced by hashing the original message, ensuring the integrity of the message. The efficacy of the hybrid algorithm lies in the synergy of combined algorithms, capitalizing on their significant complexity for an added layer of security. This hybrid system facilitates secure data transmission between a mobile phone and the cloud, mitigating the risks associated with data attacks. Through empirical evidence, it is demonstrated that the incorporation of hybrid algorithms not only elevates the encryption level for mobile data but also reduces the time required for both encryption and decryption processes.

REFERENCES

[1] Saini, Reena, and Nachiket Sainis. ”Cryptographic Hybrid Model-An Advancement in Cloud Computing Security: A survey.” Vol. 11 Issue 06, June-2022

[2] Parms, Jason. ”Symmetric vs. Asymmetric Encryption-What are differences?.” SSL2BUY Wiki-Get Solution for SSL Certificate Queries (2020).

[3] Cardoso Dos Santos, Luan. Design, Cryptanalysis, and Protection of Symmetric Encryption Algorithms. Diss. University of Luxembourg, Luxembourg, 2022.

[4] Ng, Ruth Ii-Yung. Symmetric Cryptography: New Definitions and

Schemes. University of California, San Diego, 2021.

[5] Alenezi, Mohammed N., Haneen Alabdulrazzaq, and Nada Q. Mohammad. ”Symmetric encryption algorithms: Review and evaluation study.” International Journal of Communication Networks and Information Security 12.2 (2020): 256-272.

[6] Adee, Rose, and Haralambos Mouratidis. ”A dynamic four-step data security model for data in cloud computing based on cryptography and steganography.” Sensors 22.3 (2022): 1109.

[7] Yang, Pan, Naixue Xiong, and Jingli Ren. ”Data security and privacy protection for cloud storage: A survey.” IEEE Access 8 (2020): 131723-131740.

[8] R. S. Cordova, R. L. R. Maata, A. S. Halibas, and R. Al-Razavi, “Comparative Analysis on the Performance of Selected Security Algorithms in Cloud Computing,” pp. 4–7, 2017.

[9] O. Cinar, R. H. Guncer, and A. Yazici, “Database Security in Private Database Clouds,” ICISS 2016 - 2016 Int. Conf. Inf. Sci. Secur., 2017.

[10] S. Sawant, “Towards Privacy-Preserving for Dynamic Data in Cloud Storage,” 2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput., pp. 296–299, 2017.

[11] D. Boneh, G. Di Crescenz, R. Ostrovsky and G. Persiano, ”Public key encryption with the keyword search”, Proc. EUROCRYPT, vol. 3027, pp. 506-522, 2004.

[12] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachene, ”TFHE: Fast fully homomorphic encryption over the torus”, J. Cryptol., vol. 33, no. 1, pp. 34-91, Jan. 2020.

[13] V. K. Soman, “An Enhanced Hybrid Data Security Algorithm for Cloud,” no. July, pp. 421–424,2017.

[14] A. Azougaghe, Z. Kartit, M. Hedaboui, M. Belkasmi, and M. E. L. Marraki, “An efficient algorithm for data security in cloud storage.”

[15] A. Bansal and A. Agrawal, “Providing security, integrity, and authentication using ECC algorithm in cloud storage,” 2017 Int. Conf. Comput. Commun. Informatics, ICI 2017, 2017

[16] P. Semwal and M. K. Sharma, “Comparative study of different cryptographic algorithms for data security in cloud computing,” 2017 3rd Int. Conf. Adv. Comput. Autom., pp. 1–7, 2017

[17] Susmitha, Chivukula, et al. ”Hybrid Cryptography for Secure File Storage.” 2023 7th International Conference on Computing Methodologies and Communication (ICCMC). IEEE, 2023.

[18] Deb, Moumita, and Abantika Choudhury. ”Hybrid cloud: A new paradigm in cloud computing.” Machine Learning Techniques and Analytics for Cloud Security (2021): 1-23.

[19] Kusbeci, Polathan. ”HYBRID CLOUD: A NEW PARADIGM IN CLOUD COMPUTING.” ¨(2023).

[20] Anuj Kumar et al., A New Approach for Security in Cloud Data Storage for IOT Applications Using Hybrid Cryptography Technique, IEEE, pp. 514-517, 2020.

[21] A. Grover, “A Framework for Cloud Data Security,” pp. 1199–1203, 2016.

[22] Mohd. Akbar et al., ”Study and improved data storage in cloud computing using cryptography”, IRJASH, pp. 94-99, 2021.

[23] Y. Shin, D. Koo, J. Yun, and J. Hur, “SEED: Enabling serverless and efficient encrypted deduplication for cloud storage,” Proc. Int. Conf. Cloud Comput. Technol. Sci. CloudCom, pp. 482–487, 2017.

[24] Sahil, S. K. Sood, S. Mehmi, and S. Dogra, “Designing and analysis of user profiling system for cloud computing security using fuzzy guided genetic algorithm,” 2016 Int. Conf. Comput. Commun. Autom., pp. 724–731, 201

[25] Randa Mohamed, Abdel Haleem et al., ”Enhancing the Integrity of Cloud Computing by Comparison between Blowfish and RSA Cryptography Algorithms”, IJERT, pp. 125-128, 2022.

[26] Vishal Agrahari, ”Data Security in Cloud Computing Using Cryptography Algorithms”, IJSDR, pp. 257-260, 2020.

[27] Ming Zeng, Haifeng Qian, Jie Chen et al., Forward Secure Public Key Encryption with Keyword Search for Outsourced Cloud Storage, IEEE, pp. 426-438, 2022.

[28] S. K. Prashanth, N. S. Rao, and C. S. Kumar, “Hybrid Cuckoo search - ABC algorithm based vulnerabilities mapping and security in clouds,” Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016, pp. 2569–2572, 2016.

[29] Yingying et al., Similarity Search for Encrypted Images in Secure Cloud Computing, IEEE, pp. 1142-1155, 2022.

[30] Yuan Zhang et al., Blockchain-Assisted Public-Key Encryption with Keyword Search Against Keyword Guessing Attacks for Cloud Storage, IEEE, pp. 1335-1348, 2021.

[31] Kwangsu Lee, “Comments on” Secure Data Sharing in Cloud Computing Using RevocableStorage Identity-Based Encryption” IEEE, pp. 1299-130, 2020.

[32] K. V Raipurkar and A. V Deorankar, “Improve data security in a cloud environment by using LDAP and two-way encryption algorithm,” Colossal Data Anal. Netw. (CDAN), Symp., pp.1–4, 2016.

[33] Y. Shin, D. Koo, J. Yun, and J. Hur, “SEED: Enabling serverless and efficient encrypted deduplication for cloud storage,” Proc. Int. Conf. Cloud Comput. Technol. Sci. CloudCom, pp. 482–487, 2017.

[34] 34. based proxy re-encryption for efficient data sharing”, Inf. Sci., vol. 511, pp. 94-113, Feb. 2020.

[35] Hossein Abroshan, “A Hybrid Encryption Solution to Improve Cloud Computing Security using Symmetric and Asymmetric Cryptography Algorithms”, International Journal of Advanced Computer Science and Applications, Vol. 12, No. 6, 2021 pp. 31-37

[36] Pravin Soni, Rahul Malik (2021) “A Hybrid Cloud Security Model for Securing Data on Cloud” WCNC-2021: Workshop on Computer Networks & Communications, Chennai, India. pp. 118-125

[37] G. Viswanath and P. V. Krishna, (2020) “Hybrid encryption framework for securing big data storage in a multi-cloud environment,” Evol. Intell., no. 0123456789,doi: 10.1007/s12065-020-00404-w.

[38] Pothukuchi, Ameya Shastri, Lakshmi Vasuda Kota, and Vinay Mallikarjunaradhya. ”A Critical Analysis of the Challenges and Opportunities to Optimize Storage Costs for Big Data in the Cloud.” (2021).

[39] Chauhan, Nidhi, and K. Sampath Kumar. ”Cyber Attacks Detection and Prevention using Cryptography Algorithms for Industrial Automation.” 2023 2nd International Conference for Innovation in Technology (INOCON). IEEE, 2023.