

An ASCII-Based Matrix Technique for Enhancing Data Confidentiality in Cloud Computing

Dr. B Fathimamary^{1*}, P Sophiya²

Department of Commerce Computer Applications, St. Joseph's College (Autonomous), Tiruchirappalli, India^{1*}

Department of Information Technology, St. Joseph's College (Autonomous), Tiruchirappalli, India²

Abstract: Cloud computing has evolved into a widely adopted technology for storing, processing and sharing large volumes of data owing to its scalability, flexibility, and cost-effectiveness. However, the challenge of guaranteeing the confidentiality of data during storage and transmission persists due to threats like unauthorized access, cyberattacks, and data breaches. This paper introduces an ASCII-Based Matrix Technique to tackle these security concerns and improve data confidentiality in cloud environments. The proposed technique converts plaintext into ASCII values and applies a secret prime key to convert the data into encrypted characters. The corresponding ciphertext is then written out column-wise in a matrix format, adding an extra layer of complexity and security. In decryption the reverse operations are done using the same secret key to correctly recover the original plaintext. The proposed technique is evaluated by using text files of different sizes varying from 5 MB to 25 MB. Experimental results show that as file size increased, encryption time increased from 61 ms to 84 ms, and decryption time increased from 38 ms to 62 ms. The results show that proposed techniques offers better data confidentiality with an acceptable processing time and is appropriate for secure cloud storage and data transmission. The matrix based arrangement and secret key mechanism greatly enhance the resistance to unauthorized access and cryptanalysis, while keeping the efficient performance.

Keywords: Cloud Computing, Data Security, Encryption, ASCII, Matrix Encryption, Data Confidentiality, Cloud Storage.

1.INTRODUCTION

Cloud computing is one of the widely adopted technologies. It provides cost-effectiveness, scalability and flexibility for information processing, sharing and storage[1][2]. Both individuals and enterprises are storing large amounts of sensitive data, such as financial and medical data, educational materials, and business papers, on the cloud[3][4]. Despite the numerous benefits offered by cloud computing, protecting data during storage and transmission remains a significant concern[5][6]. In cloud environments, encryption is an effective method to safeguard critical data. Traditional encryption methods like Rivest-Shamir-Adleman (RSA), Data Encryption Standard (DES), and Advanced Encryption Standard (AES). Despite its robust security, AES may introduce computational overhead when handling large datasets and require effective key-management mechanisms[7][8]. Several lightweight encryption techniques have been developed by academics to boost computing performance while preserving a decent level of security[9][10].

By rearranging encrypted data into various matrix forms, matrix-based encryption techniques offer additional diffusion and confusion, which has garnered a lot of attention[11][12]. Similarly, plaintext is converted into integers using ASCII-based encryption techniques before being calculated using secret keys[13][14]. Nevertheless, a lot of existing approaches either employ predefined transformation patterns that restrict the amount of randomness in the resulting ciphertext or simple replacement techniques that are vulnerable to cryptanalysis[15][16].

To address these challenges, the present study introduces an ASCII-Driven Matrix Approach to boost data confidentiality in cloud environments. The proposed technique converts the plaintext into ASCII values, performs character transformation using the secret prime key and arranges the encrypted data in a matrix column-wise to generate the final ciphertext. The matrix structure provides an additional complexity level to analyze or reconstruct the original plaintext for unauthorized users[17]. When decrypting, The reverse transformations are applied using the same secret key so that they can correctly retrieve the original data. Arranging the processed data in a matrix enhances its complexity, thereby making unauthorized interpretation or reconstruction of the plaintext more difficult. The original data is subsequently recovered by applying the reverse procedure with the corresponding secret key.

2. MOTIVATION

Cloud encompasses a broad range of issues such as network security, data security, access control, and regulatory compliance[18]. As pointed out by Halevi et al. [19], although cloud computing provides fast and convenient computing and storage services, it has serious security issues that must be solved to ensure the successful application of cloud technologies. Thabit et al. [20] reported that data loss, data breaches, account hijacking, insecure APIs, and malicious insiders are among the major security concerns in cloud computing. These are compounded by the shared responsibility model of cloud computing, where both the cloud service provider and the consumer are responsible for maintaining security.

A key challenge is maintaining data security, as data owners have limited direct control over the infrastructure in which their information is stored. Fathimamary [21] reported that users are generally unaware of where their data is physically stored and which computing systems are responsible for processing it. Cloud data centers are infrastructure that enables organizations to transfer their data across the globe[22]. Several key characteristics contribute to the credibility of cloud computing systems, making careful assessment of their reliability essential [23]. The overall trust in cloud computing depends on the adoption of diverse methods and strategies for effective data protection and security [24].

One of the key advantages of cloud computing is its capability to support data sharing across multiple organizations. This advantage has multiple benefits but also a potential threat to data security. Traditional information systems cannot address complex data security issues in cloud computing. The data security issues include threats, data loss, service disruption, vulnerabilities, military issues, and malicious attacks.

3. AN ASCII-BASED MATRIX TECHNIQUE

The encryption procedure starts with the plaintext, representing the original information that requires protection. Each plaintext character is mapped to its corresponding ASCII (American Standard Code for Information Interchange) value to facilitate numerical processing during encryption. The ASCII codes for uppercase alphabetic characters range from 65 to 90, whereas lowercase characters are represented by values from 97 to 122. After converting the plaintext into ASCII values, a secret key is incorporated to transform the data into ciphertext. During the decryption stage, the encrypted values are processed using the same key to reconstruct the original plaintext. As an additional security measure, a matrix-based encryption mechanism is incorporated, in which the data elements are organized and processed column-wise. This arrangement introduces an additional level of complexity, thereby improving the protection of the stored information against unauthorized access.

Encryption

The proposed method performs file encryption and decryption by integrating ASCII values, a secret key, and matrix multiplication. The use of matrix multiplication within the encryption procedure introduces an additional security mechanism for protecting the data. To validate the correctness and functionality of the proposed approach, an initial experiment was conducted using the single word 'HELLO.' The encryption and subsequent decryption results were examined to confirm that the original data could be successfully recovered.

Step 1: The plaintext (original text) is "**HELLO.**"

Step 2: Every plaintext character is converted into its corresponding ASCII code.

Step 3: The ASCII representations of the plaintext 'HELLO' are presented below.

H	E	L	L	O
72	69	76	76	79

Step 4: A secret key is selected to encrypt the plaintext.

Step 5: For example, the prime number 5 is selected as the secret key.

Step 6: The secret key (5) is added to the ASCII value of each character in the plaintext.

Step 7: The secret key (5) is added to the ASCII value of each plaintext character as shown below.

- **H:** $72 + 5 = 77$
- **E:** $69 + 5 = 74$
- **L:** $76 + 5 = 81$
- **L:** $76 + 5 = 81$
- **O:** $79 + 5 = 84$

Step 8: After adding the secret key (5) to the ASCII value of each character, the encrypted characters obtained are:

77 → M, 74 → J, 81 → Q, 81 → Q, 84 → T

Hence, the resulting ciphertext is **MJQQT**.

Step 8: The plaintext "HELLO" is encrypted as the ciphertext "MJQQT."

Step 9: The ciphertext is arranged in a 3×3 matrix in a column-wise manner.

Step 10: The ciphertext "MJQQT" is arranged in a 3×3 matrix in a column-wise manner. Since the ciphertext contains only five characters, the remaining position is filled with the padding character '@' to complete the matrix.

The resulting matrix is shown below:

Column 1	Column 2
M	Q
J	T
Q	@

Step 11: After arranging the ciphertext in the 3×3 column-wise matrix, the final encrypted text is "MQJTQ@".

Step 12: By reversing the encryption process, the ciphertext is decrypted to obtain the original plaintext.

The preceding steps illustrate the encryption and decryption of a single word using ASCII values and a column-based matrix structure. In the proposed approach, the encryption technique is extended to text files containing a substantial volume of data. Since larger files involve more data elements, their encryption and decryption require comparatively greater processing time than a single word. A secret key is incorporated during encryption to strengthen data confidentiality and prevent unauthorized access. In this approach, a prime number is selected as the secret key to increase the difficulty of key prediction and unauthorized recovery. In addition, the encrypted data is organized column-wise within a matrix structure, introducing greater complexity to the ciphertext and thereby providing an additional layer of protection. During decryption, the corresponding operations are performed in reverse order to reconstruct the original plaintext.

Decryption:

Decryption involves applying the corresponding inverse operations to the ciphertext to reconstruct the original plaintext. The proposed encryption scheme is designed to strengthen data protection during both encryption and decryption. To evaluate its performance, the method is applied to text files containing different volumes of data. As the amount of data increases, the processing time required for encryption and decryption is expected to be greater than that required for a single word. A secret key is incorporated into the encryption procedure to improve data confidentiality and authentication. In the proposed approach, a prime number is selected as the secret key, increasing the difficulty of unauthorized key prediction or compromise. Furthermore, the encrypted values are organized in a column-wise matrix structure, which increases the complexity of the ciphertext and provides an additional layer of protection. During decryption, the corresponding operations are performed in reverse order to retrieve the original plaintext.

4. RESULTS AND DISCUSSION

Table 4.1 displays the encryption and decryption times for files of various sizes. The findings indicate that the encryption and decryption times grow as the file size does. The time required to encrypt a 5 MB file is 61 ms and the time required to decrypt it is 38 ms. The decryption and encryption timings of 10 MB file are 68 ms and 41 ms respectively. Similarly a 15 MB file takes 73 ms to encrypt and 52 ms to decrypt. For a file of 20 MB, the encryption time increases to 79 ms and the decryption time increases to 57 ms. Finally, for a file of size 25 MB the encryption time is 84 ms and decryption time is 62 ms. The findings demonstrate that the suggested encryption method's processing time increases as file size increases. But the increment is proportional to the amount of data processed. This shows that the proposed method is efficient in handling files of different size with secure decryption and encryption. Figure 1 illustrates the graphical representation of the proposed technique.

Table 4.1. Analysis of various files

File Size(MB)	Encryption(MS)	Decryption(MS)
5	61	38
10	68	41
15	73	52
20	79	57
25	84	62

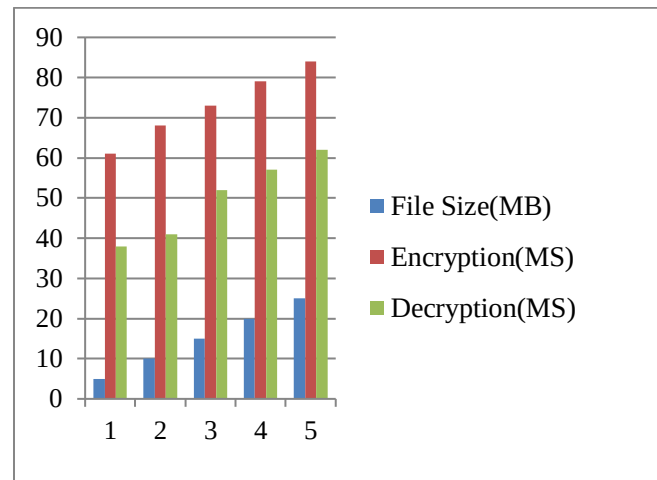


Figure 1. Comparison of various files

4. CONCLUSION

Protection of data in a cloud during transmission to the authorized users is a challenging issue. There are many methods to prevent illegal access to information, but hackers are still able to manipulate data. To tackle this issue, the proposed technique enhances the security of the data during the decryption and encryption process. The proposed method employs a secret key assigned to the data and ASCII values for decryption and encryption of data. Then the encrypted data is formatted as a matrix column wise. This adds another level of complexity. The matrix method increases data confidentiality by making the data difficult to decode by unauthorized users. The proposed work implies that data shared in private cloud environment is more secure. The security is based on the use of a secret key. The key is like a strong encryption key that is hard to break. The technique has been implemented and evaluated on file data, showing improved security in private cloud storage and transmission. The decryption and encryption time of the file has also been calculated and analyzed. In order to minimize execution time while maintaining a high level of security, the technique can be expanded in subsequent work to handle larger files more effectively.

REFERENCES

- [1] Juare, S, "Survey on Data Security and Integrity Issues in Cloud Computing", IOSR Journal of Computer Engineering, 2016, pp. 59-62.
- [2] Mushtaq, M. F., Jamel, S., Disina, A. H., Pindar, Z. A., Shakir, N. S. A., & Deris, M. M, "A Survey on the cryptographic encryption algorithms", International Journal of Advanced Computer Science and Applications, 2017, pp. 333-344.
- [3] Manisha Kumari, Kirubanad V.B, "Data encryption and decryption using graph plotting. International Journal of Civil Engineering and Technology", 2018, pp. 36-46.
- [4] Kalpana, G., Kumar, P. V., Aljawarneh, S., & Krishnaiah, R. V, "Shifted adaption homomorphism encryption for mobile and cloud learning", Computers & Electrical Engineering, 2018, pp. 178-195.
- [5] Elhoseny, M., Shankar, K., Lakshmanaprabu, S. K., Maseleno, A., & Arunkumar, N, "Hybrid optimization with cryptography encryption for medical image security in Internet of Things", Neural computing and applications, 2018, pp. 1-15.
- [6] Yang, L., Han, Z., Huang, Z., & Ma, J, "A remotely keyed file encryption scheme under mobile cloud computing. Journal of Network and Computer Applications", 2018, pp. 90-99.

- [7] Al Badawi, A. Q. A., Polyakov, Y., Aung, K. M. M., Veeravalli, B., & Rohloff, K. "Implementation and performance evaluation of RNS variants of the BFV homomorphic encryption scheme", IEEE Transactions on Emerging Topics in Computing, 2019.
- [8] Stergiou, C., Psannis, K. E., Kim, B. G., & Gupta, B, "Secure integration of IOT and cloud computing, Future Generation Computer Systems", 2018, pp.964-975.
- [9] Shanthni, K. K., Kaviya, K., Sujithra, M," A Survey on Cloud Computing: Data Security Challenges and Their Defensive Mechanisms", International Journal of Recent Science, 2018, pp.26497-26500.
- [10] Dr.D.I.George Amalarethnam, B.Fathimamary, "DMUCE- A Confidentiality Enabled Technique To Improve Cloud User Security", International Journal of Applied Engineering Research,2015, pp.34103-34108.
- [11] Dr.D.I.George Amalarethnam, B.Fathimamary, "eDSSuMRT- Ensured Data Security Strategy using Matrix Random Traversal In Cloud Storage Environment", International Journal of Applied Engineering Research,2015,pp.272-279.
- [12] Dr.D.I.George Amalarethnam, B.Fathimamary, Confidentiality Technique for Enhancing Data Security in Public Cloud Storage using Data Obfuscation, International journal of control theory and application,2016,pp.1-11.
- [13] Dr.D.I.George Amalarethnam, B.Fathimamary, Data Security Enhancement in Public Cloud Storage using Data Obfuscation and Steganography, IEEE Explore,2016, pp.181-184.
- [14] B.Fathimamary, Dr.D.I George Amalarethnam, Data security enhancement in public cloud storage using data obfuscation and steganography, world congress on computing and communication technologies (WCCCT),2016, pp.181-184.
- [15].Dr.D.I.George Amalarethnam, B.Fathimamary, Data Security Enhancement in Public Cloud Storage, Journal of Computing and Intelligent Systems, 2017, pp.1-5.
- [16].Dr.D.I.George Amalarethnam, B.Fathimamary, Security Enhancement for Public Cloud Storage with Minimum Cost, International Journal of Pure and Applied Mathematics,2018, pp.1-9.
- [17].Dr.D.I.George Amalarethnam, B.Fathimamary, "Comparative Analysis of Obfuscation Techniques in Public Cloud", American International Journal of Research in Science, Technology, Engineering & Mathematics,2019,pp.309-312.
- [18]Halevi, S., Polyakov, Y, Shoup V, "An improved RNS variant of the BFV homomorphic encryption scheme in Cryptographers", Springer, pp. 83-105.
- [19] Thabit, F., Can, O., Alhomdy, S., Al-Gaphari, G. H., & Jagtap, S, "A novel effective lightweight homomorphic cryptographic algorithm for data security in cloud computing", International Journal of Intelligent Networks, 2022, pp.16–30. <https://doi.org/10.1016/j.ijin.2022.04.001>
- [20]Fathimamary B, "Data Security Enhancement In Cloud Using Magic Square Obfuscation", International Multidisciplinary Research Journal Reviews (IMRJR), vol.2, 2025, pp.66-70. <https://DOI: 10.17148/IMRJR.2025.020908>
- [21] Fathimamary B, "A Comprehensive Study on Cryptographic Approaches for Securing Data in Cloud Storage", International Multidisciplinary Research Journal Reviews (IMRJR), vol.2, Issue 11, 2025, pp.66-72. <https://DOI: 10.17148/IMRJR.2025.021107>
- [22] Fathimamary B. A novel magical encryption technique for secure and efficient public cloud storage. Advancement of Computer Technology and Its Applications. 2026;9(2):45-50. <https://doi.org/10.5281/zenodo.18397099>
- [23] Fathimamary B, Baby Nirmala M, Nasreen M. Enhanced Positional Vigenère (EPV): A confidentiality-enabled encryption technique for secure cloud storage. The Scientific Temper.2026;17(4):6005-6009. <https://doi.org/10.58414/SCIENTIFICTEMPER.2026.17.4.09>
- [24] Fathimamary B, Remy J, Geetha R. (2026) A Novel Three-Layered Security Framework for Enhancing Data Confidentiality in Cloud Storage Environments. Indian Journal of Science and Technology. 19(31): 2197-2205. <https://doi.org/10.17485/IJST/v19i31.1093>