Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 28 (2025), No. 5-B, pp. 2137–2149
DOI : 10.47974/JDMSC-2430

Securing the cloud : Comparative analysis of NCS, ERSA, and EHS cryptographic methods

Anjani Kumar Singha [‡]
Department of School of Computer Science and Engineering
Galgotias University
Greater Noida
Uttar Pradesh
India

Pradeep Kumar Tiwari [†]
Minakshee Bari [§]
Department of Computer Science and Applications
Dr. Vishwanath Karad MIT World Peace University
Pune 411038
Maharashtra
India

Rakesh Kumar *
Department of Mechanical Engineering
Manipal University Jaipur
Jaipur 303007
Rajasthan
India

Mandeep [@]
Department of School of Computer Science and Engineering
Galgotias University
Greater Noida
Uttar Pradesh
India

[‡] *E-mail:* anjani348@gmail.com
[†] *E-mail:* Pradeeptiwari.mca@gmail.com
[§] *E-mail:* minaksheebari96@gmail.com
* *E-mail:* rakesh.kumar@jaipur.manipal.edu (Corresponding Author)
[@] *E-mail:* mandeepmehra97@gmail.com

Abstract

The paradigm change from locally located data centers toward the cloud is necessary due to the recent boom in data generated through different social networking platforms, e-commerce websites, and other enterprises. Cryptography is the most effective method for safeguarding the privacy and safety of cloud data. Researchers have created numerous algorithms for cryptography. However, they all have long execution times that are predictable, linear, and memory-intensive. The performance of three cryptographic schemes—Enhanced Homomorphic Scheme (EHS), Enhanced RSA (ERSA) and is Non-Deterministic Cryptographic Scheme (NCS), evaluated using a CPU, focusing on CPU usage, throughput efficiency, and memory consumption. The experiment's outcomes showed that EHS and NCS produced non-deterministic and non-linear run times. Once more, for text and numeric data types, EHS and NCS yielded the lowest throughput and memory consumption when processing data sizes of $7n*102$ (KB ($\in$1, 2, 4, 10, 20, 40)). On the other hand, deterministic run-time trends, linear, and ERSA generated predictable results.

Subject Classification: *Primary 93A30, Secondary 49K15.*

Keywords: *Cryptography, Encryption, Throughput, Non-deterministic, Execution time, Decryption.*

1. Introduction

Data security is becoming increasingly important due to the complexity of communication brought about by increasing individual activities [1]. It has become necessary to shift the standard model of keeping data beyond local data centers toward the clouds as a way to protect the massive volumes of data generated by multiple platforms for social media [2]. An analysis conducted in 2017 by statistical firm Cyence found that S&P 500 companies lost $150 million due to a four-hour Amazon cloud computing outage [3]. Additionally, it was anticipated by Apica, a network traffic tracking company that transactions on the 54 most popular e-commerce sites would decline by at least 20% [4]. Furthermore, Maeser projected that even while the Internet of things will produce a lot of information [5], cloud computing would become more and more necessary between 2013 and 2020—roughly 266% more in total values [6]. The cloud computing industry will lead to a rise in demand of 85% or additional information [7], which is in line with RightScale's projections [8]. The benefits associated with cloud computing [9], the creation of work and lower operating costs through the efficient use of pay-as-you-use services [10], keep going to make it more and more popular than traditional on-site data centers [11]. Unlike in the past, this has forced computer giants to invest enormous sums of money in computing infrastructure to provide cloud services [12]. These advantages have prompted businesses to use

cloud service models to move their data into the cloud [13]. The use of cloud computing creates several safety concerns [14], such as the confidentiality and privacy of data [15]. It has demonstrated that using encryption algorithms is a practical and efficient and safety of cloud data [16]. In these methods, such as Enhanced Homophism scheme, Nondeterministic Cryptographic Scheme (NCS) and Enhanced RSA (ERSA). This research examines throughput, the processing times [17], times of three algorithms—Enhanced RSA (ERSA) [18], Nondeterministic Cryptographic Scheme (NCS) [19], and Enhanced Homomorphic Scheme (EHS) [20]—to identify which one is best for guaranteeing the privacy and confidentiality of cloud data.

Contribution of Search

- By comparing NCS, ERSA, and EHS, it provides a structured approach for assessing the efficacy of different cryptographic methodologies in cloud environments.
- The research offers an in-depth comparative analysis of NCS, ERSA, and EHS cryptographic solutions.
- One of the significant contributions of this study is the identification of runtime trends in cryptographic solutions for cloud security.

2. Related Works

Many cryptographic methods are currently developed to safeguard cloud data's privacy and secrecy [21]. The second part provides a summary of the evaluation of previous research. Almarwani et al. [22] introduced termed Tagging of Outsourcing Data to protect the confidentiality of data. Their plan enabled data to be stored in the cloud by using verification. Mobile phones and tablets may find significant use for the algorithm they employ due to its low run time. Their approach was linear, no matter how creative it seemed. To help Almarwani et al. attain data privacy, Tahir et al. suggested the CryptoGa genetic algorithms in their study [23]. Their algorithms operated more quickly than modern algorithms. The process of nonlinear, though. Shen et al. [24] suggested using ORAM (Oblivious Random Access Memory) with proxy re-encryption to ensure cloud data confidentiality and privacy. Their solution was created to enable cloud data sharing across several users. Member can maintain the confidentiality of information by controlling access. Adee and Mouratidis [25] work

provided evidence to support their findings in the fields of steganography and cryptography. Despite the algorithm's seemingly promised efficiency, its running time was linear. Cloud data privacy and secrecy were secured using Thabit et al. [26] method for a lightweight encryption algorithm. They increased the complexity of the encryption by using the Feistel and substitution methods. Their method was incredibly run-time efficient while having a linear execution time. Still, they were predictable in that the execution times were inversely correlated with the volume of data handled.

3. Methodology

3.1 *Algorithms used in this work*

3.1.1. Enhanced RSA

Enhanced RSA raised the bar on standard RSA security to a fourth level by fusing it with the Gaussian interpolation formula[15]. The message's characters in ASCII are first encoded using Gaussian Moving forward interpolation through the standard RSA; encryption and decoding are used for the second and third levels[16]. Gaussian First Backward interpolation is used in the final stage in show Figure 1[17]. The integration makes overcoming the well-known RSA factorization challenge easier [18].

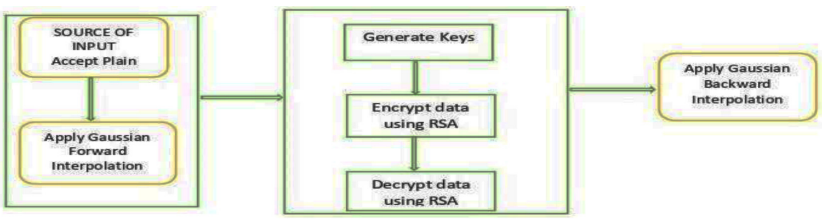

Figure 1

Work flow diagram of enhanced RSA [18].

3.1.2 Non-deterministic cryptographic scheme (NCS)

NCS uses three stages: critical production, information decryption data. It is three stages aim to produce hidden improve algorithm security [19]. These consist of XORing the output to plaintext the algorithms. Eleven numbers are created in NCS once the Fixed Sliding Window [19] is used, as demonstrated in Algorithm 1. It computes a sub-array of (m(b[i])/ 4) on these twelve values.

Algorithm 1 The proposed NCS algorithm

Process NCS
Calculate K=A*B
Y_x=H(Y_x -1) +l mod m, such that (m>0,0<h<m,0m, Calculate the CLG)
(Y_{y-1})=K
(Y_x€1,2,3,-------1000,000)
For (i=0,i<13,i++) do
{b[i]=Rand(1,1000000)}
End for
On 12 arrays, apply sub-array of 3.
$_{bx}= b_{i=} b_i+ b_{i+1+b+2}$
$_{bx1}= b_{i+3}+ b_{i+4+b+5}$
$_{bx2}= b_{i+6}+ b_{i+7+b+8}$
$_{bx3}= b_{i+9+}+ b_{i+10+b+11}$
S_k=max(b_x, b_{x1}, b_{x2}, b_{x3})
$X_i=Y_i \oplus S_i$ mod((m[b_i]/4))
$Y_i=X_i \oplus S_i$ mod((m[b_i]/4))
End process.

3.1.3 Improved homomorphism algorithm

As organizations increasingly rely on cloud services to handle large volumes of confidential information, ensuring data confidentiality, integrity, and security—without compromising performance—remains a top priority. Traditional cryptographic methods such as Non-Deterministic Cryptographic Scheme (NCS), Enhanced RSA (ERSA), and Enhanced Homomorphic Scheme (EHS) have attempted to address these concerns with varying degrees of success. However, limitations in terms of computational efficiency, restricted operation support, and excessive ciphertext size hinder their scalability and applicability in real-time cloud environments. As seen in Equation (1) for data encryption [20], the second value is Mj, the first value is si, the fourth value is Ml, the third value is M_i, and with N the plaintext.

$$D_I = N_t + M_k * M_i + M_j * M_r \quad (1)$$

Algorithm 2 The proposed EHS algorithm

Process EHS
Calculate K=A*B , such that K,A,B € Prime Number.
Y_x=H(Y_x -1) +l mod m, such that (m>0,0<h<m,0m, Calculate the CLG)
(Y_{y-1})=K
(Y_x€1,2,3,-------1000,000)
For (i=0,i<13,i++) do
{a[i]=Rand(1,1000000)}
End for
On 12 arrays, to the sub-array of 3.

$$S_k = \Sigma_{m=0}^{3} b_{x^m}$$

$$S_i = \Sigma_{m=3}^{5} b_{x^m}$$

$$S_i = \Sigma_{m=6}^{8} b_{x^m}$$

$$S_r = \Sigma_{m=9}^{11} b_{x^m}$$

D_I=N_t + S_k* S_i+ S_j* S_r
D_m=D_m+D_I
D_t= D_t+Di
N_d= D_t mod S_j
End

4. Experimentation

To analyze CPU performance, each algorithm's CPU cycles were monitored during operation execution. IHA consumed fewer CPU cycles than EHS due to its noise control mechanism and computational batching strategy. In terms of memory usage, IHA also proved more efficient, exhibiting controlled memory allocation even under larger datasets, whereas NCS and ERSA displayed higher memory consumption due to repeated key and data expansion processes. Throughput, defined as the number of successful encrypted operations processed per second, was significantly higher for IHA, especially during batch operations in parallel processing scenarios.

4.1 *An explanation of the dataset used in this project*

The Kaggle database provided the dataset used in this investigation [27]. The dataset offers a textual and numerical translation from the

English language to French, enabling the effectiveness of the methods to be assessed using an efficiency trend. When just numbers were discussed, their decryption time tendency was linear. The proposed techniques were evaluated on $5m * 10^2$ (KB ($\in$ 1, 2, 4, 10, 20, 40)) data sets.

5. Results and Discussion

5.1 *Architectural Design for the Proposed Approach*

The four phases of the suggested RAsys are implemented in this part, and Figure 2 offers a visual representation of sharing information and storing. Memory is made available through computing in the cloud as a component of infrastructure-as-. This allows the sharing and saving of information to a remote server using any electronic device's web browser.

5.2 *The Proposed Framework of the System*

An overview of the upgraded homomorphism plan, improved RSA, and non-deterministic cryptographic system comparison is provided in this section. Figure 3 displays the diagrammatic representation (RAsys) of the framework. The five stages of the structures are essential creation, throughput, memory usage, decryption, and encryption. Figure 3 illustrates how a user contracts with a cloud service provider. After encrypting the plaintext with EHS, NCS, or ERSA, the registered client uploads the Ciphertext to the web, and then timings. The throughput, memory storage, and decryption times are calculated.

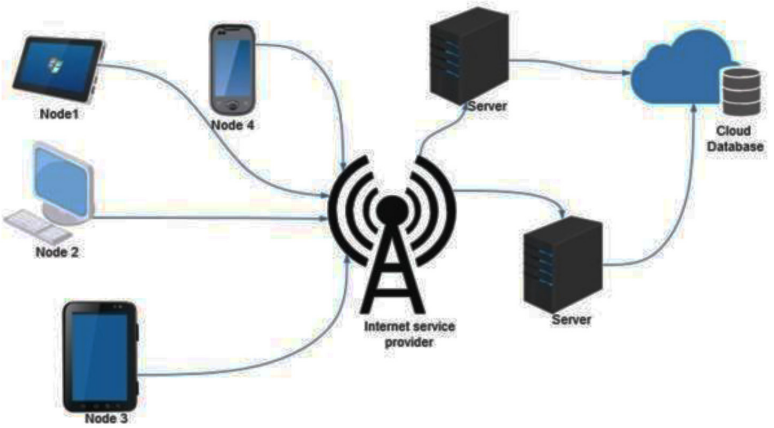

Figure 2

Design of the proposed RAsys' architecture.

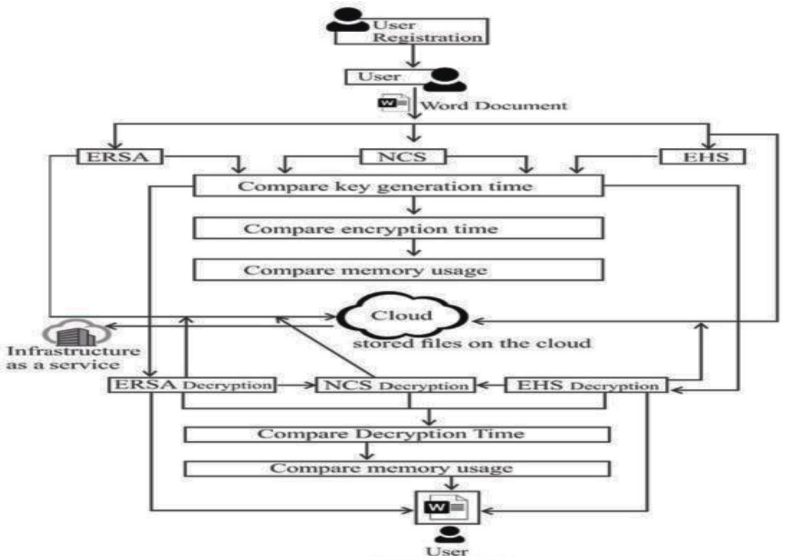

Figure 3

The proposed RAsys framework.

5.3 *Computational Results*

The outcomes of executing the algorithms fifty times and figuring out their averages are displayed in Tables 1 and 2.

5.3.1 Encryption time

In Table 1 show, Encryption time is a crucial performance indicator in evaluating the efficiency of cryptographic algorithms, particularly in cloud computing environments where rapid encryption of large datasets is essential. To assess the comparative performance of Enhanced Homomorphic Scheme (EHS), Non-Deterministic Cryptographic Scheme (NCS), and Enhanced RSA (ERSA), a series of experiments were conducted on datasets ranging from 500 KB to 20,000 KB. The results highlight noticeable differences in encryption speed among the three methods.

5.3.2. Decryption time

In the Table 2 show, Decryption time is a critical performance metric, especially in cloud-based systems where data retrieval and real-time responsiveness are vital. To evaluate the practical applicability of Enhanced Homomorphic Scheme (EHS), Non-Deterministic Cryptographic Scheme

(NCS), and Enhanced RSA (ERSA), average decryption times were measured across multiple data sizes ranging from 500 KB to 20,000 KB. The observed results reveal significant differences in decryption efficiency among the three cryptographic schemes. Table no. 3 shows the encryption throughput and table 4 decryption throughput

Table 1

Comparing the average encryption time of ERSA, NCS, and EHS

EHS (ms) [20]	NCS (ms) [19]	ERSA (ms) [18]	Data Size (KB)
480	159	833	20000
767	407	790	10000
669	806	754	5000
1000	693	702	2000
258	439	686	1000
382	615	649	500

Table 2

ERSA, NCS, and EHS average decryption times compared

EHS (ms) [20]	NCS (ms) [19]	ERSA (ms) [18]	Data Size (KB)
497	535	841	20000
767	726	769	10000
726	94	741	5000
70	9	694	2000
269	565	591	1000
393	647	564	500

Table 3

Encryption throughput (KB/ms)

EHS (ms) [20]	NCS (ms) [19]	ERSA (ms) [18]	Data Size (KB)
42.76	127.59	25.06	20000
14.06	25.64	13.9	10000
8.50	7.22	7.65	5000
3	3.90	3.86	2000
4.90	3.29	2.47	1000
2.32	.82	.75	500

Table 4
Decryption throughput (KB/ms)

EHS (ms) [20]	NCS (ms) [19]	ERSA (ms) [18]	Data Size (KB)
41.76	38.12	24.06	20000
13.06	14.80	14.9	10000
7.50	54.77	8.65	5000
29.12	2.45	3.87	2000
4.85	2.34	2.40	1000
2.00	.79	.90	500

The NCS decryption time grew to 726 , milliseconds before 10,010 KB of data was going through, however it dropped to 535 milliseconds after processing 2,010 KB of data.

6. Discussion

Each cryptographic solution - NCS, ERSA, and EHS - offers distinct advantages and trade-offs in enhancing cloud security. NCS provides network-level protection for data in transit, while ERSA offers a comprehensive risk-based security framework tailored to enterprise needs. EHS introduces innovative homomorphic encryption techniques to preserve data confidentiality in cloud environments [21]. The choice of cryptographic solution depends on various factors, including the specific security requirements, compliance mandates, performance considerations, and the level of trust in cloud service providers [22]. Organizations must carefully evaluate these factors to select the most suitable cryptographic solution that aligns with their security objectives and business goals in the cloud.

7. Conclusion

The objective of the present study is to achieve the fastest, both linear and non-linear execution times, by thoroughly analyzing three algorithms for cryptography. The performance metrics employed to compare the different algorithms were throughput, encryption time, and decryption time. The analysis showed that the ERSA-provided time-to-execution trends were deterministic, linear, patterning, and consistent. Once more, their execution durations were more extensive, and they needed more

RAM. The advantage of NCS and EHS, however, is the production of a non-linear, non-patterned, non-deterministic, and lowest execution time trend, making them resistant to side-channel attacks. This reduces hardware tearing and wear from excessive CUP use and eliminates the requirement for high bandwidth for transmitting and downloading vast amounts of data using NCS and EHS.

References

[1] A. K. Singha, A. Kumar, and P. K. Kushwaha, "Classification of brain tumors using deep Encoder along with regression techniques," *EPH-International Journal of Science and Engineering*, vol. 1, no. 1, pp. 444–449 (2018).

[2] I. A. Ibrahim and M. Bassiouni, "Improvement of job completion time in data-intensive cloud computing applications," *Journal of Cloud Computing,* vol. 9, no. 1 (Feb. 2020), doi: 10.1186/s13677-019-0139-6.

[3] A. K. Singha, G. Yadav, V. Hasanpuri, and R. R. Patel, "Meta-optimization for improved efficiency: an integrated methodology," *International Journal of Information Technology*, pp. 1–12 (2025).

[4] A. K. Singha, H. P. Singh, S. Kundu, L. S. Songare, and P. K. Tiwari, "Measuring network security in the cloud: A roadmap for proactive defense," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 27, no. 2-B, pp. 889–902 (2024). [Online]. Available: https://tarupublications.com/issues/issue-list/JDMSC/27/2-B.

[5] A. K. Singha, H. P. Singh, S. Kundu, A. S. Rajput, and P. K. Tiwari, "Estimating computer network security scenarios with association rules," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 1–10 (2024). [Online]. Available: https://tarupublications.com/doi/10.47974/JDMSC-1876.

[6] G. De Riso and M. Noce, "A unified description of cryptosystems: From classical to quantum protocols," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 26, no. 8, pp. 2285–2299 (2023).

[7] A. K. Singha, P. K. Tiwari, N. Shukla, and G. Yadav, "Transformative trends: analyzing the integration of blockchain in banking operations," unpublished (2024).

[8] Y. Al-Issa, M. A. Ottom, and A. Tamrawi, "eHealth Cloud Security Challenges: A Survey," *Journal of Healthcare Engineering*, vol. 2019, pp. 1–15 (Sep. 2019), doi: 10.1155/2019/7516035.

[9] A. K. Singha, M. Jena, S. Zubair, P. K. Tiwari, and A. P. S. Bhadauria, "Deep Neural Networks performance comparison for handwritten text recognition," in Proc. Int. Conf. Mobile Radio Communications & 5G Networks, Singapore, pp. 539–553 (Aug. 2023). Springer Nature Singapore.

[10] P. Kulkarni, R. Khanai, and G. Bindagi, "A Comparative Analysis of Hybrid Encryption Technique for Images in the Cloud Environment," in Proc. 2020 Int. Conf. Communication and Signal Processing (ICCSP) (Jul. 2020), doi: 10.1109/iccsp48568.2020.9182153.

[11] A. K. Singh, S. Zubair, A. Malibari, N. Pathak, S. Urooj, and N. Sharma, "Design of ANN based non-linear network using interconnection of parallel processor," *Computer Systems Science and Engineering*, vol. 46, no. 3, pp. 3491–3508 (2023).

[12] A. K. Singha and S. Zubair, "Machine learning for hypothesis space and inductive bias: a review," *AIJR Abstracts*, p. 70 (2022).

[13] A. K. Singha, A. Kumar, and P. K. Kushwaha, "Recognition of human layered structure using Gradient decent model," *EPH-International Journal of Science and Engineering*, vol. 1, no. 1, pp. 450–456 (2018).

[14] H. Aljader and R. Ajeena, "The optimized Diffie-Hellman key exchange using the graphical method," *Journal of Discrete Mathematical Sciences and Cryptography* (2022).

[15] A. K. Singha, N. Pathak, N. Sharma, P. K. Tiwari, and J. P. C. Joel, "COVID-19 disease classification model using deep dense convolutional neural networks," in Emerging Technologies in Data Mining and Information Security (IEMIS 2022), Vol. 2, Singapore: Springer Nature, pp. 671–682 (2022).

[16] S. Mittal et al., "Using Identity-Based Cryptography as a Foundation for an Effective and Secure Cloud Model for E-Health," *Computational Intelligence and Neuroscience,* vol. 2022, pp. 1–8 (Apr. 2022), doi: 10.1155/2022/7016554.

[17] A. K. Singha, N. Pathak, N. Sharma, P. K. Tiwari, and J. P. C. Joel, "Forecasting COVID-19 confirmed cases in China using an optimization method," in Emerging Technologies in Data Mining and Information Security (IEMIS 2022), Vol. 2, Singapore: Springer Nature, pp. 683–695 (2022).

[18] J. K. Dawson, F. Twum, J. B. H. Acquah, Y. M. Missah, and B. B. K. Ayawli, "An enhanced RSA algorithm using Gaussian interpolation formula," *International Journal of Computer Aided Engineering and Technology*, vol. 16, no. 4, p. 534 (2022), doi: 10.1504/ijcaet.2022.123996.

[19] J. K. Dawson, F. Twum, J. H. Acquah, and Y. M. Missah, "Ensuring confidentiality and privacy of cloud data using a non-deterministic cryptographic scheme," *PLOS ONE* (2022), doi: 10.1371/journal.pone.0274628.

[20] J. K. Dawson, F. Twum, J. H. Acquah, and Y. M. Missah, "Ensuring privacy and confidentiality of data on the cloud using an enhanced homomorphism scheme," *Informatica,* vol. 46, no. 8 (Nov. 2022), doi: 10.31449/inf.v46i8.4305.

[21] A. K. Singha and S. Zubair, "Enhancing the efficiency of the stochastic method by using non-smooth and non-convex optimization," *Journal of University of Shanghai for Science and Technology,* vol. 22, no. 10 (2020).

[22] R. Almarwani, N. Zhang, and J. Garside, "An effective, secure and efficient tagging method for integrity protection of outsourced data in a public cloud storage," PLOS ONE, vol. 15, no. 11, p. e0241236 (Nov. 2020), doi: 10.1371/journal.pone.0241236.

[23] M. Tahir, M. Sardaraz, Z. Mehmood, and S. Muhammad, "CryptoGA: a cryptosystem based on genetic algorithm for cloud data security," *Cluster Computing* (Jul. 2020), doi: 10.1007/s10586-020-03157-4.

[24] J. Shen, H. Yang, P. Vijayakumar, and N. Kumar, "A privacy-preserving and untraceable group data sharing scheme in cloud computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 4, pp. 2198–2210 (Jul.–Aug. 2022), doi: 10.1109/TDSC.2021.3050517.

[25] R. Adee and H. Mouratidis, "A dynamic four-step data security model for data in cloud computing based on cryptography and steganography," *Sensors*, vol. 22, no. 3, p. 1109 (Feb. 2022), doi: 10.3390/s22031109.

[26] F. Thabit, A. P. S. Alhomdy, A. H. A. Al-Ahdal, and P. D. S. Jagtap, "A new lightweight cryptographic algorithm for enhancing data security in cloud computing," *Global Transitions Proceedings*, vol. 2, no. 1, pp. 91–99 (Jun. 2021), doi: 10.1016/j.gltp.2021.01.013.

[27] S. T. Siddiqui, Z. Syed, S. G. Hashmi, J. Ahmad, A. K. Singha, and K. A. Qidwai, "A wearable EEG-based hybrid CNN-RNN framework for psychiatric disorder diagnosis: Leveraging XAI for enhanced clinical insights," in Proc. 2025 Devices for Integrated Circuit (DevIC), pp. 743–748. *IEEE* (Apr. 2025).

Received November, 2024