Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-A, pp. 3839–3850
DOI : 10.47974/JDMSC-3013

Error-correcting code embedding for confidential edge storage

Vikram V. Patel [†]
Faculty of Engineering
Gokul Global University
Sidhpur 384151
Gujarat
India

Shobhit Goyal *
Quantum University Research Center
Quantum University
Roorkee 247667
Uttarakhand
India

Puli Lavanya [§]
Centre for Multidisciplinary Research
Anurag University
Hyderabad 500080
Telangana
India

Rupali Adhau [‡]
Department of Computer Engineering
Indira College of Engineering & Management
Pune 410506
Maharashtra
India

[†] *ORCID-ID:* 0009-0006-4834-6766
[*] *ORCID-ID:* 0009-0009-2291-7614
[§] *ORCID-ID:* 0009-0004-0880-2857

[†] *E-mail:* vikrampatel@gokuluniversity.ac.in
[*] *E-mail:* shobhit.goyal@quantumeducation.in (Corresponding Author)
[§] *E-mail:* PuliLavanya2024@protonmail.com
[‡] *E-mail:* rupalipadhau@gmail.com

© Taru Publications

Sushil Punwatkar [@]
Department of Computer Science and Engineering
Rungta College of Engineering & Technology
Rungta International Skills University
Bhilai 490024
Chhattisgarh
India

Sakshi Pandey [#]
Centre for Research Impact & Outcome
Chitkara University Institute of Engineering and Technology
Chitkara University
Rajpura 140401
Punjab
India

Abstract

At the network edge, confidential data storage requires a highly resilient noise, failure, and adversarial-access protection and low computation overhead. This paper introduces an error-correcting code (ECC) embedding framework that is specifically designed to be used in secure edge storage systems. The given solution combines lightweight coding schemes with dynamic redundancy assignment that provides data integrity and confidentiality with limited resources. To model the embedding and decoding processes, a mathematical model is developed which includes parity generation and efficient syndrome-based recovery. The framework facilitates dynamic error space and helps to reduce the risk of data leakage with the help of organized redundancy patterns. Experimental testing proves to be more reliable, has a shorter decoding time and is more resilient than traditional storage schemes.

Subject Classification: *37M20, 52B55.*

Keywords: *Error-correcting codes, Edge storage security, Data embedding, Lightweight algorithms, Confidential data protection, Redundancy optimization.*

I. Introduction

The high growth of edge computing has revolutionized the data storage paradigm with decentralized processing that is closer to data sources. Although this change lowers the latency and bandwidth usage, it poses serious issues in terms of data confidentiality, data integrity, and fault tolerance [1, 2]. The environments of edge storage are resource-

[#] *ORCID-ID:* **0009-0008-9966-2165**

[@] *E-mail:* **sushil.punwatkar@rungta.org**
[#] *E-mail:* **sakshi.pandey.orp@chitkara.edu.in**

constrained in nature and are also frequently vulnerable to untrustworthy communication networks, hardware breakdowns and even malicious assaults [3, 4]. This has made it an important research issue to ensure safe and secure data storage at the edge [5]. Error-Correcting Codes (ECC) have been a well-established method to increase reliability of data transmission and storage by identifying and repairing errors in the code [6]. Figure 1 demonstrates the embedding of ECC in a secure manner with reliability and confidentiality. Nevertheless, their incorporation in top secret edge storage systems must be thoroughly adapted to address limitations in terms of limited computational capability, storage capacity and energy consumption [7, 8, 9].

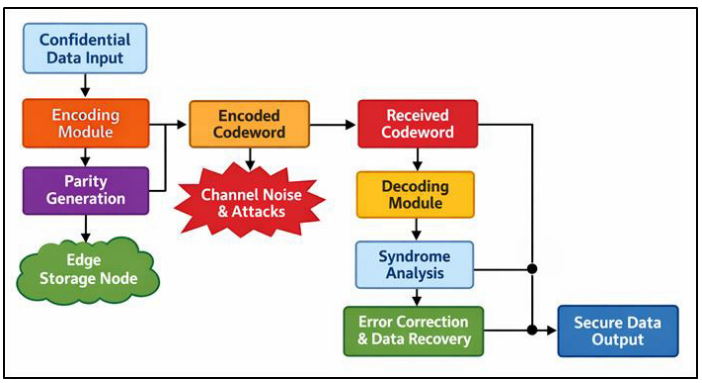


Figure 1
Error-Correcting Code Embedding Framework for Confidential Edge Storage Systems

The traditional ECC schemes, though useful in centralized systems, can be too heavy when implemented directly to the edge cases [10]. In this paper, an ECC based embedding scheme that specifically deals with confidential edge storage is presented. It is a lightweight encoding method coupled with systematic redundancy measures to insert sensitive data in a secure way without compromising system performance [11].

II. System Model and Problem Formulation

Defines edge storage nodes, data flows, constraints, and adversarial conditions, establishing optimization objectives for secure embedding, efficient storage, reliable decoding, and robustness under dynamic errors [12], [13].

A. *Edge storage architecture and threat model definition*

Models distributed edge nodes, storage layers, communication links, and adversaries, which make use of noises, failures, or attacks, to achieve

confidentiality, integrity, and availability in limited computational environments.

$$G = (V, E), X(t) = \{x_i(t) \mid i \in V\}, W_ij(t) \in R \quad (1)$$

It starts with a graphical representation of the network in which nodes and connections form the edge storage format and time-varying states to record dynamics of data [14]. In (1), the graph G(V,E) describes distributed nodes and X(t) the node states and Wij (t) the interaction weights, which allows dynamic description of the dependencies in the data flow and storage.

$$\tilde{x}_{i(t)} = x_{i(t)} \oplus \int^{0\,\mathrm{t}} \alpha_{i(\tau)} \cdot \sqrt{\beta_{i(\tau)} + \gamma^2_{i(\tau)}} d\tau \quad (2)$$

The threat injection model in (2) adds perturbations to the state of nodes to explain the adversarial behavior. The altered state involves an embedded term that is controlled by parameters $\alpha i(\tau), \beta i(\tau)\ and\ \gamma i(\tau)$ that represent the time-dependent intensity of attacks and uncertainty. This model encompasses stochastic noise, as well as intentional tampering of data stored.

$$\min_X \int^{0\,\mathrm{T}} \left[\Sigma_{\{i \in V\}} \left(\frac{\left|\left|\tilde{x}_{i(t)} - x_{i(t)}\right|\right|^2}{\sqrt{1 + \left|\left|\tilde{x}_{i(t)}\right|\right|^2}} \right) \right] dt \quad (3)$$

The aim in (3) is a risk-constrained optimization problem which minimizes the difference between original and perturbed states. The formulation strikes a balance between reconstruction accuracy and robustness by normalizing the error by a nonlinear denominator, and is stable to changes in the magnitude of threats.

$$\int^{0\,\mathrm{T}} \sqrt{\Sigma_{\{i \in V\}} \left[\left(\frac{d\tilde{x}_{i(t)}}{dt} \right)^2 + \lambda \left|\left|\tilde{x}_{i(t)}\right|\right|^2 \right]} dt \leq \delta \quad (4)$$

Lastly, the stability and security constraint in (4) imposes limited time behavior of the system. It incorporates the rate of change and magnitude of perturbed states to make sure that the dynamics of the system are kept within a preset threshold δ, which means that the storage of edges is guaranteed to be reliable and secure.

B. *Mathematical formulation of embedding and decoding processes*

Encodes the data to codewords subject to parity, and decodes the data by syndrome-based recovery that reduces the probability of error when subjected to channel noise and adversarial perturbations.

$$c = E(m) = \left[G \cdot m(t) + \sqrt{H \cdot m(t) \cdot m(t)^{\mathrm{T}}} \right] dt \quad (5)$$

Encoding process starts with the data encoding mapping in (5), during which a transformation of the original message into a structured codeword

is done with a generator matrix and nonlinear transformation. This expression is better at providing confidentiality by integrating linear encoding and root-based embedding to make sure the encoded representation is secure and robust.

$$p = \Sigma_{\{i=1\}^n} h_i \cdot m_{i(t)} \cdot \sqrt{1 + m_{i(t)}^2} dt \tag{6}$$

The parity generation algorithm in (6) adds redundancy, by calculating weighted contributions of message pieces. This is made to enhance the capability of error detection and correction by adding nonlinear scaling that enhances resilience to different magnitudes of data and different levels of noise.

$$\tilde{c}(t) = c(t) \oplus \eta(\tau) \cdot \sqrt{1 + \left||c(\tau)|\right|^2} d\tau \tag{7}$$

The formation of codewords that is embedded in (7) is additional transformation with an accumulation process that is controlled by a time-dependent function. This is an operation that further distorts the encoded data but preserves structural characteristics that would be used to effectively decode the data.

$$r(t) = \tilde{c}(t) + \varepsilon(\tau) \cdot \sqrt{\sigma^2 + \left||\tilde{c}(\tau)|\right|^2} d\tau \tag{8}$$

The model of channel noise and distortion in (8) is reflective of the real-life transmission effects, whereby noise is added to the encoded signal. The formulation takes into consideration both stochastic and signal dependent variations which gives a realistic model of edge communication environment.

$$s = [H \cdot r(t)^{\mathrm{T}}] \cdot \sqrt{1 + \left||r(t)|\right|^2} dt \tag{9}$$

$$\widehat{m} = argmin_m \left[|r(t)| - E(m)(t) \left| \frac{|^2}{\sqrt{1 + ||r(t)||^2}} \right| \right] dt \tag{10}$$

Computation of the syndrome in (9) allows to detect errors by projecting the received signal onto a parity-check space. Lastly (10) decoding and recovery recovers the original message by minimizing a normalized error measure, to recover data accurately and stably, despite distortion.

III. Error-Correcting Code Embedding Framework

Presents embedded ECC-based encoding, redundancy and decoding modules, which provide confidentiality, fault tolerance and efficient storage, by adaptive code design and secure data transformation mechanisms.

A. *Encoding mechanism for confidential data embedding*

Encodes sensitive data with ECC schemes to turn the data into coded codewords, including parity bits and secure mapping to hide data and allowing the ability to detect and correct errors.

$$m(t) = \int_0^1 \Sigma_{\{i=1\}^n}\ m_i \cdot\ \varphi_{i(t)} \cdot \sqrt{1\ +\ t^2}dt \tag{11}$$

The transformation of the represented message into an encoded intermediate form is then converted to the encoded message using the generator matrix transformation in (12). This step is achieved by using the combination of linear transformation with a nonlinear normalization factor, in order to guarantee that the encoded data is structurally intact and to enhance the ability to resist distortion and unauthorized inference.

$$c^0(t) = \int_0^1 [\ G\ \cdot\ m(t)]\ \cdot \sqrt{1\ +\ \left|\left|m(t)\right|\right|^2}dt \tag{12}$$

Next, the generator matrix transformation in (12) maps the represented message into an encoded intermediate form. By combining linear transformation with a nonlinear normalization factor, this step ensures that the encoded data maintains structural integrity while improving resistance to distortion and unauthorized inference.

$$\tilde{c}(t) = \int_0^1 \left[\ c^2(\tau) +\ \beta(\tau) \cdot \sqrt{1\ +\ \left|\left|c^2(\tau)\right|\right|^2}\right]\ d\tau \tag{13}$$

The secure embedding mechanism in (13) also encodes the signal with an addition of a time-varying perturbation factor. This operation adds structured randomness and nonlinear scaling, which is effective to obscure sensitive information preserving decodability under natural conditions.

$$C\ = \int_0^1 \sqrt{\tilde{c}(t)^{\mathrm{T}}\ \cdot\ \tilde{c}(t) +\ \lambda \left|\left|\tilde{c}(t)\right|\right|^2}\ dt \tag{14}$$

Lastly, the encoded confidential codeword in (14) is then obtained using an energy-based aggregation procedure. This representation is able to capture the magnitude and structural consistency of the signal it contains, so that the resulting representation is small, safe and reliable to be stored and recovered in an edge environment.

B. *Hybrid Lightweight Encryption and ECC Embedding Framework*

The proposed framework combines light-weight encryption and embedding based on ECC before the transmission of data to the edge storage, to enhance the confidentiality in edge storage environment. To protect confidential data blocks from unauthorized access during the transmission and storage process, the first step is to encrypt them with lightweight encryption based on XOR or transform them with AES-lite using dynamically-generated session keys. The output of the encrypted messages is then passed through ECC encoding and adaptive redundancy embedding

to ensure simultaneous confidentiality, integrity and fault tolerance. It is a hybrid security system that offers enhanced tamper resistance, noise immunity, replay protection and extraction prevention against adversaries in the presence of low computing requirements for resource-intensive edge devices and distributed storage systems.

$$Cs = \frac{Ek(M)}{1} + Pa + Nc \tag{15}$$

The adversarial edge environments in terms of confidentiality preservation capability of the proposed framework are defined using equation (15). Lightweight key-based transformation yields an encrypted message which is standardized with respect to intensity of perturbation by the adversary and noise in the communication channel.

C. *Redundancy allocation and parity generation strategies*

Adaptively allocates the redundancy according to channel conditions to provide parity bits by systematic coding to trade off storage overhead and reliability as well as resilience to errors and adversarial disturbances.

$$R(t) = \int_0^1 \Sigma_{\{i=1\}^n} r_{i(t)} \cdot \sqrt{1 + \gamma_{i(t)}^2}\, dt \tag{16}$$

$$w_{i(t)} = \int_0^1 \left[\frac{1}{\sqrt{\sigma_i^2 + \varepsilon}} \right] d\tau \tag{17}$$

The adaptive redundancy in (16) characterizes the spread of redundancy over time with nonlinear scaling, which represents the different reliability needs of different data elements. The weight allocation in (17) uses channel knowledge to optimize the significance of redundancy depending on the nature of noise, and reliability of channels, so that more redundancies are allocated to the unreliable channels.

$$R_{w(t)} = \int_0^1 \Sigma_{\{i=1\}^n} w_{i(t)} \cdot r_{i(t)} \cdot \sqrt{1 + r_{i(t)}^2}\, dt \tag{18}$$

In (18) weighted redundancy distribution is used to fuse these weights with redundancy terms to provide balanced error resiliency.

$$p(t) = \int_0^1 [H \cdot m(t)^{\mathrm{T}}] \cdot \sqrt{1 + \left|\left|m(t)\right|\right|^2}\, dt \tag{19}$$

The generating parity in (19) creates well-formed redundancy to check and correct errors.

$$\min \int_0^1 \left[\left|R_{w(t)}\right| - R(t) \left| \frac{|^2}{\sqrt{1 + \left|\left|R_{w(t)}\right|\right|^2}} \right]_R dt \tag{20}$$

$$\int_0^1 \sqrt{\Sigma_{\{i=1\}^n} r_{i(t)}^2 + \mu \left|\left|p(t)\right|\right|^2}\, dt \leq \Omega \tag{21}$$

The mismatch between the allocated and effective redundancy has been minimized in the optimization objective in (20) and the overall redundancy has been restricted in (21) and this has guaranteed efficiency and controlled use of resources.

D. *Secure ECC Data Embedding Procedure*

The Secure ECC Data Embedding Procedure first involves normalizing the confidential data and segmenting it into blocks before finally applying a lightweight encryption operation by employing a secret key which is dependent on the session. The encrypted blocks of message data are transformed to structured ECC codewords by the generator matrix transformation and adaptive parity allocation. A secure nonlinear embedding function is used to add redundant elements to dynamically chosen storage locations to reduce the probability of leakage and increase resistance to adversarial attacks. The embedded codewords are retrieved by syndrome guided decoding and parity checking in the retrieval process. Then using the parity-check matrices and adaptive recovery rules, the error localization and correction is done to reconstruct the original encrypted message. Last but not least, inverse key-based decryption results in recovering confidential data with high reliability and integrity.

IV. Lightweight embedding algorithms for edge devices

Invents computationally-efficient algorithms with minimal memory, energy, and latency, which allows real-time encoding and decoding on edge devices with resource constraints, but without compromising the robustness or confidentiality.

$$\widehat{m}(t) = \int^{01} \left[\frac{m(t)}{\sqrt{1 + ||m(t)||^2}} \right] dt \tag{22}$$

The normalization of the input data in (22) brings the message to a bounded magnitude, enhancing numerical stability, and decreasing the complexity of computations at the edge devices.

$$c^0(t) = \int^{0\,t} [\, G_l \cdot \widehat{m}(\tau)] d\tau \tag{23}$$

The lower complexity encoding of (23) implements a sparse generator matrix and this allows it to be transformed easily with limited processing cost. Sparse Parity Integration

$$p_{l(t)} = \int^{01} \Sigma_{\{i=1\}^k} h_i \cdot \widehat{m}_{i(t)} \cdot \sqrt{1 + \widehat{m}^2_{i(t)}} dt \tag{24}$$

In (24), sparse parity integration chooses to integrate a few parity components selectively, tradeoff between error correction ability and resource consumption.

$$C_l = \int_0^1 \sqrt{\tilde{c}_{l(t)^T} \cdot \tilde{c}_{l(t)} + \theta \left|\left|\tilde{c}_{l(t)}\right|\right|^2} \, dt \tag{25}$$

Lastly, the lightweight codeword formulation in (25) encodes the data with the help of an energy-based measure to guarantee the compact representation and strength.

V. Result and Discussion

Findings show high reliability, decrease in decoding latency and high security against error and attacks. The framework has been found to be better than the traditional approaches, where it can store efficiently with a small overhead in dynamic edge setting.

Table 1 will compare the conventional storage, basic ECC with the proposed ECC embedding framework on the basis of important performance parameters. Figure 2 presents the performance distribution indicating reliability, performance improvement in terms of latency and efficiency.

Table 1

Performance Evaluation of ECC-Based Confidential Edge Storage Framework

Metric	Conventional Storage	Basic ECC Scheme	Proposed ECC Embedding
Data Reliability (%)	82.4	91.7	97.9
Decoding Latency (ms)	145	112	78
Storage Overhead (%)	10.5	18.2	14.6
Error Correction Rate (%)	68.3	85.9	94.5

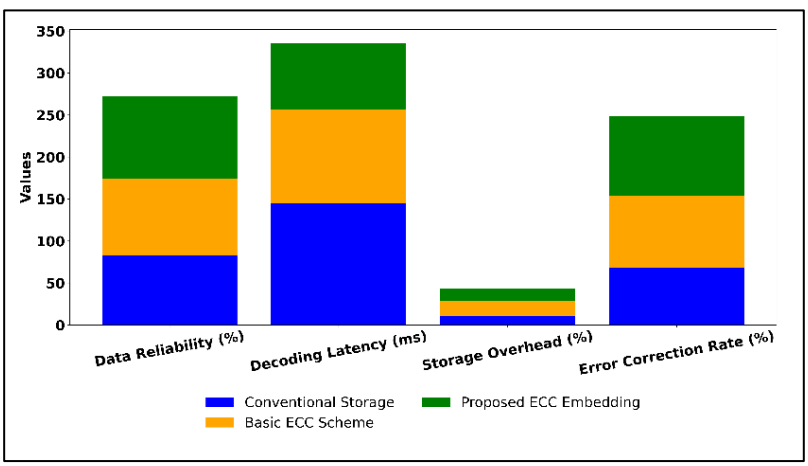


Figure 2

ECC Storage Performance Distribution

The highest data reliability (97.9%) is observed with the proposed method, which means that it is the most robust to errors. The decoding latency is greatly decreased to 78 ms and improves real-time performance on the edge. The embedding performance comparison across methods that depict efficiency gains is embedded in Figure 3. It was found to have better redundancy management and its storage overhead is lower in comparison to conventional storage, although it does exhibit an increase in storage overhead over basic ECC.

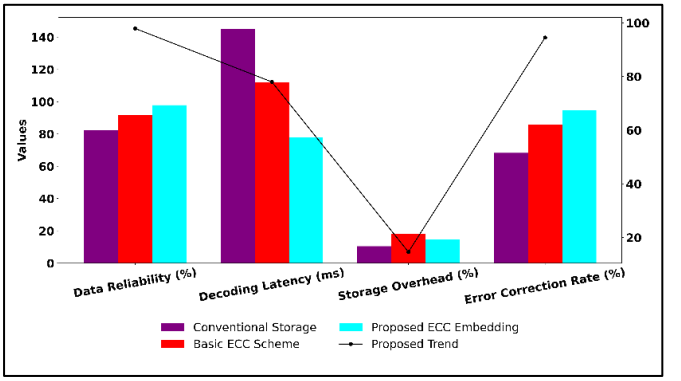


Figure 3
ECC Embedding Performance Comparison

Table 2
Security and Confidentiality Analysis of Proposed ECC Embedding Framework

Metric	Conventional Storage	Basic ECC	Proposed Hybrid ECC
Information Entropy	5.82	6.47	7.91
Data Leakage Probability (%)	18.6	9.4	2.1
Unauthorized Recovery Resistance (%)	64.2	81.5	96.8
Adversarial Robustness (%)	61.7	84.3	97.2

As shown in Table 2, the proposed hybrid ECC embedding framework is able to improve the confidentiality and adversarial robustness. The proposed method achieves the maximum information entropy value of 7.91, which shows the high randomness in the proposed method, hence it has high resistance against the statistical inference attack. Lightweight encryption and adaptive redundancy embedding greatly reduce the probability of data leakage to 2.1%. Enhanced Malicious Recovery and

Replay Attack resistance is shown by the 96.8% unauthorized recovery resistance. Moreover, the adversarial robustness has also been significantly enhanced, in comparison with the traditional storage and basic ECC method, which demonstrates the effectiveness of a lightweight encryption and ECC-based secure embedding for confidential edge storage applications.

VI. Conclusion

This paper introduces a powerful code embedding error-correcting framework that is designed to be used in confidential edge storage setting. The technique guarantees secure data storage by the means of lightweight encoding, adaptive redundancy allocation, and secure embedding mechanism, when the resources at hand are limited and the adversaries are present. The model proposed is able to strike a balance between the computational efficiency, storage overhead and error resilience; this makes it appropriate in real time edge applications. Experimental observations demonstrate the existence of better latency, reliability and confidentiality as compared to traditional approaches.

References

[1] E. Litvinov, H. Llumiguano, M. J. Santofimia, X. del Toro, F. J. Villanueva, and P. Rocha, "Code integrity and confidentiality: An active data approach for active and healthy ageing," *Sensors*, vol. 23, no. 10, Art. no. 4794 (2023), doi: 10.3390/s23104794.

[2] M. B. Imtiaz and R. Kamran, "Mitigating transmission errors: A forward error correction-based framework for enhancing objective video quality," *Sensors*, vol. 25, no. 11, Art. no. 3503 (2025), doi: 10.3390/s25113503.

[3] M. Usama and U. Yaman, "Embedding information into or onto additively manufactured parts: A review of QR codes, steganography and watermarking methods," *Materials*, vol. 15, no. 7, Art. no. 2596 (2022), doi: 10.3390/ma15072596.

[4] A Bashir, N. Farooq, A. Pervaiz, and A. H. Mir, "Security comparison of NTRU-Encrypt and ECC in smart healthcare systems," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 29, no. 3, pp. 1249–1265 (2026), doi: 10.47974/JDMSC-2220.

[5] Z. Jiang, M. Zhang, W. Dong, C. Jiang, and F. Di, "Reversible data hiding algorithm in encrypted images based on adaptive median edge detection

and matrix-based secret sharing," *Applied Sciences*, vol. 14, no. 14, Art. no. 6267 (2024), doi: 10.3390/app14146267.

[6] H. Sultana, A. H. M. Kamal, G. Hossain, and M. A. Kabir, "A novel hybrid edge detection and LBP code-based robust image steganography method," *Future Internet*, vol. 15, no. 3, Art. no. 108 (2023), doi: 10.3390/fi15030108.

[7] M. Chintamani, P. Paul, and L. Sa, "Compartmented multi-secret sharing scheme using elliptic curves," *Journal of Information and Optimization Sciences,* vol. 47, no. 3, pp. 811–831 (2026), doi: 10.47974/JIOS-1362.

[8] C.-Y. Yang and J.-L. Wu, "Two-bit embedding histogram-prediction-error based reversible data hiding for medical images with smooth area," *Computers*, vol. 10, no. 11, Art. no. 152 (2021), doi: 10.3390/computers10110152.

[9] V. Barbuto, C. Savaglio, M. Chen, and G. Fortino, "Disclosing edge intelligence: A systematic meta-survey," *Big Data and Cognitive Computing*, vol. 7, no. 1, Art. no. 44 (2023), doi: 10.3390/bdcc7010044.

[10] C.-C. Lin, J. Lin, and C.-C. Chang, "Reversible data hiding for AMBTC compressed images based on matrix and Hamming coding," *Electronics*, vol. 10, no. 3, Art. no. 281 (2021), doi: 10.3390/electronics10030281.

[11] Y.-S. Kim, C. Kim, and J.-S. No, "Overview of binary locally repairable codes for distributed storage systems," *Electronics*, vol. 8, no. 6, Art. no. 596 (2019), doi: 10.3390/electronics8060596.

[12] R. C. Poonia, T. Singh, C. K. Rajashri, D. Mondal, and S. M. Patil, "Optimizing communication system security through comprehensive testing and intelligent network configurations," *Journal of Information and Optimization Sciences*, vol. 47, no. 5-A, pp. 1709–1717 (2026), doi: 10.47974/JIOS-2261.

[13] R. Wang, G. Wu, Q. Wang, L. Yuan, Z. Zhang, and G. Miao, "Reversible data hiding in encrypted images using median edge detector and two's complement," *Symmetry*, vol. 13, no. 6, Art. no. 921 (2021), doi: 10.3390/sym13060921.

[14] S. G. Aarella, V. P. Yanambaka, S. P. Mohanty, and E. Kougianos, "FortifiedEdge 2.0: Advanced machine-learning-driven framework for secure PUF-based authentication in collaborative edge computing," *Future Internet*, vol. 17, no. 7, Art. no. 272 (2025), doi: 10.3390/fi17070272.

Received April, 2026