Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-A, pp. 4021–4032
DOI : 10.47974/JDMSC-3028

Cryptographic protocols based on graph isomorphism hardness

Prabhat Sharma *
Centre for Research Impact and Outcome
Chitkara University
Rajpura 140417
Punjab
India

M. Radhika Mani †
Department of Computer Science & Engineering
Pragati Engineering College
Surampalem
Kakinada District 533437
Andhra Pradesh
India

Mahesh M. Zade §
Department of Electronics & Telecommunication Engineering
S. K. N. Sinhgad College of Engineering
Pandharpur 413304
Maharashtra
India

I. Durga Bhavani ‡
Department of Computer Science
Aditya Degree & PG Colleges (Autonomous)
Kakinada 533003
Andhra Pradesh
India

* *E-mail:* prabhat.sharma.orp@chitkara.edu.in (Corresponding Author)
† *E-mail:* mradhika@outlook.com
§ *E-mail:* zademahesh1983@gmail.com
‡ *E-mail:* durgabhavanisupriya@gmail.com

© Taru Publications

R. Snehaa [@]
Department of Humanities and Sciences (Mathematics Division)
Aarupadai Veedu Institute of Technology
Vinayaka Mission's Research Foundation (Deemed to be University)
Paiyanoor
Chennai 603104
Tamil Nadu
India

Kasturi Rohini [#]
Centre for Multidisciplinary Research
Anurag University
Hyderabad 500088
Telangana
India

Abstract

Graph isomorphism (GI) has turned out to be an attractive basis of the construction of secure cryptographic protocols because it has a special set of computational hardness properties, not in the classical NP-complete classes. This article introduces a new cryptographic system that utilizes the graph isomorphism hardness where pairs of graphs of the same structure are used to conduct secure communication through the use of permutation-based mappings. The protocol proposed presents a strong public- private key architecture in which isomorphic graph transformations are encryption primitives. The security properties and resistance to structural inference attacks are modeled by creating a mathematical formulation. The encryption algorithm uses randomized permutations to hide the correspondences between the vertex with confidentiality and unpredictability. The efficiency and the possible quantum resistance of the proposed approach is compared to the NP-complete and post-quantum cryptographic assumptions, which helps to draw the comparative analysis.

Subject Classification: *94A60, 68M12, 05C60.*

Keywords: *Graph isomorphism, Cryptographic protocols, Permutation mapping, Post-quantum security, Lightweight cryptography.*

I. Introduction

The massive growth of the digital communication, cloud computing and Internet of Things (IoT) ecosystems has augmented the need of secure and effective cryptography protocols [1, 2]. Conventional cryptographic schemes, including RSA and elliptic curve cryptography, are based on number-theoretic assumptions, which can be defeated by the new quantum

[#] *ORCID-ID:* **0009-0007-8739-6640**

[@] *E-mail:* **snehaa.avmt032@avit.ac.in**
[#] *E-mail:* **KasturiRohini2024@protonmail.com**

computing paradigm [3, 4]. This has inspired the effort to find other assumptions of hardness that can provide long-term security [5]. Graph isomorphism (GI) out of them provides an attractive basis with its undecided computational complexity and inability to be classified as either solvable in a time that is polynomial or non-NP-complete [6, 7]. Cryptography using graph isometrics uses the structural similarity of graphs, in which it is difficult to compute a hidden permutation of vertex between two structurally similar graphs [8, 9]. This property allows building secure exchange of keys and encryption mechanisms which are based on permutation-based mappings. GI-based systems are flexible in their design, and may be resistant to quantum attacks, unlike classical approaches [10, 11].

II. Proposed Cryptographic Protocol Design

In this section, a graph isomorphism-based cryptographic system combining structured key generation, permutation mapping, and secure communication primitives is described, which provides confidentiality, efficiency and resistance to structural inference attacks. Isomorphic graphs that describe secure mapping of the key between public and private are depicted in Figure 1.

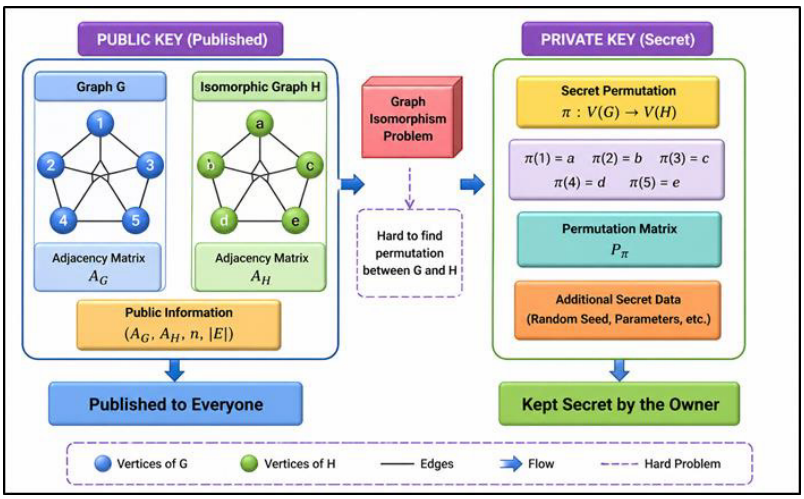


Figure 1

Public and Private Key Structure Using Isomorphic Graph-Based Cryptographic Framework

A. *Public and Private Key Structure Using Isomorphic Graphs*

Graph pairs make up public keys and hidden vertex permutations between isomorphic structures are stored as a private key, allowing

identification to be secure, encryption to be secure, and auxiliary against adversarial attacks involving graph reconstruction [12].

Graph Representation

$$A_G = \int_0^1 \Big[\sum * \{i = 1\}^{\{n\}} \sum * \{j = 1\}_{\{ij\}}^{\{n\}} \delta \cdot w_{\{ij\}} \Big] dt \quad (1)$$

The adjacency matrix representation of Equation (1) is formulated by weighted vertex interactions, with integration over normalized interval, to capture dynamic graph connectivity, and encode structural relationships that are vital to cryptographic models.

$$P_\pi = \int_0^1 \Big[\sum \{i = 1\}_i^{\{n\}e} \cdot e * \{\pi(i)\}^{\{T\}} \Big] dt \quad (2)$$

The permutation matrix is presented as equation (2) and it is generated based on vertex mapping π, allowing integration of basis transformations to encode hidden isomorphic correspondence, making it possible to perform secure structural transformation between equivalent graph representations [13].

$$A_H = \int_0^1 [P_\pi \cdot A_G \cdot P_\pi^{-1}] dt \quad (3)$$

Isomorphic graph adjacency matrix is built in equation (3) with the help of similarity transformation, which uses permutation matrix and inverse permutes the similarity transformation to maintain structural equivalence and obscure direct vertex identities of adversaries.

$$K_{pub} = \int_0^1 \sqrt{A_G^2 + A_H^2}\, dt \quad (4)$$

Equation (4) involves the definition of public key as the integration of original and transformed graph matrices in a nonlinear form, whereby there is a shared structural information without exposing the hidden permutation relationships that are privately used.

$$K_{priv} = \int_0^1 \Big[\sum_{\{i=1\}}^{\{n\}} (\pi(i)^2 + w_i^2) \Big] dt \quad (5)$$

Equation (5) is a form of private key, which is integrated summation of permutation indices and weights, secret mapping parameters that are uniquely determined to identify isomorphic transformation, and to provide the ability to decrypt the data.

Algorithm 1: Graph-Isomorphism Authentication

Input: Public key (G,H), private key π

Output: Authentication acceptance or rejection

1. Prover generates random permutation σ.
2. Prover constructs commitment K = σ(G).
3. Prover sends K to verifier.
4. Verifier generates random challenge b ∈ {0,1}.
5. Verifier sends b to prover.

6. If b = 0:
   Prover sends r = σ.
   Otherwise:
   Prover sends r = σ ∘ π^{-1}.
7. Verifier checks:
   If b = 0, verify r(G) = K.
   If b = 1, verify r(H) = K.
8. Repeat with fresh randomness for t rounds.
9. Accept only when every round succeeds.

B. *Encryption Mechanism Using Permutation-Based Mapping*

Encryption encodes plaintext to graph forms, and uses random permutations of the vertices, hiding structural matches and providing confidentiality by making it computationally difficult to invert unknown isomorphic encodings by an intruder.

$$M_G = \int_0^1 \left[\sum_{\{i=1\}_{i(t)}^{\{n\}}} m \cdot \varphi_i \right] dt \tag{6}$$

Plaintext is encoded into graph signal in equation (6) using weighted basis functions that convert message components into structured representations in line with graph vertices to allow safe processing.

$$F_M = \int_0^1 [\, W \cdot M_G + b \,]\, dt \tag{7}$$

Equation (7) provides a linear transformation with the help of weight matrix and bias, deriving features of the encoded graph signal, maintaining structural dependencies that may be utilized in the future in the process of secure encryption.

$$C^2 = \int_0^1 \sqrt{C^{1\mathrm{T}} \cdot C^1 + \sum_{\{i=1\}}^{\{n\}} \lambda_i^2}\, dt \tag{8}$$

Nonlinear obfuscation is presented in equation (8), which incorporates the energy of the transformed signals and eigenvalue based regularization to improve the confusion and diffusion properties of the obfuscated signal that are important to resist statistical and structural cryptanalysis attacks.

$$C^3 = \int_0^1 [\, A_H \cdot C^2]\, dt \tag{9}$$

Transformed signal is incorporated within isomorphic graph structure via equation (9), where adjacency matrix multiplication generates ciphertext which has the structural complexity, and consistency with graph-based encryption framework.

$$C_{final} = \int_0^1 [\, C^3 + \log(1 + |C^2|)]\, dt \tag{10}$$

The embedded signal is combined with logarithmic nonlinearity to produce final text in equation (10), enhancing randomness, thwarting inversion attempts and bolstering against the effects of analytical and reconstruction-based decryption.

III. Mathematical Formulation of Graph Isomorphism-Based Cryptography

In this section, graph mappings, permutation groups and adjacency matrices are modelled and isomorphic transformations, security properties are defined using algebraic representations that guarantee hardness of computation and hardness against structural analysis attacks.

$$A_G = \int_0^1 \left[\sum_{\{i=1\}}^{\{n\}} \sum_{\{j=1\}}^{\{n\}} \left(w_{\{ij\}} \cdot x_{i(t)} \cdot x_{j(t)} \right) \right] dt \tag{11}$$

Equation (11) represents adjacency matrix of graphs in terms of weighted vertex signal interactions integrated over domain to represent dynamic structural dependencies necessary to describe graph-based cryptographic transformations and operations.

$$A_H = \int_0^1 \left[P_\pi \cdot A_G \cdot P_\pi^{-1} + \sqrt{\sum_{\{i=1\}}^{\{n\}} (\pi(i) - i)^2} \right] dt \tag{12}$$

Isomorphic transformation (Equation 12) is defined using permutation similarity and deviation term to guarantee structural equivalence but introduce permutation complexity to make the transformation more resistant to reconstruction and mapping inference attacks.

$$S_G = \int_0^1 \sqrt{\sum_{\{k=1\}}^{\{n\}} \lambda_{k(A_G)}^2 + \sum_{\{k=1\}}^{\{n\}} \lambda_{k(A_H)}^2} \; dt \tag{13}$$

Equation (13) is spectral embedding, which is the product of eigenvalues of original and transformed graphs, which involves the global properties of the structure and allows a robust characterization of weaker cryptographic representations, based on graphs.

$$S_{final} = \int_0^1 \sqrt{H_{GI}^2 + \sum_{\{i=1\}}^{\{n\}} (\deg(v_i)^2 + \pi(i)^2)} \; dt \tag{14}$$

Final security mapping based on graph isomorphism hardness measure with the degree of the vertices and the elements of the permutation is defined in equation (14), generating integrated metric of the general cryptographic strength and structural complexity.

IV. Comparison with NP-Complete and Post-Quantum Assumptions

Graph isomorphism is intermediate and not NP-complete, and has shown potential resistance to quantum attacks, which is a promising alternative to lattice-based and number-theoretic cryptography assumptions.

$$C_{NP} = \int_0^1 \sqrt{\sum_{\{i=1\}}^{\{n\}} 2^{\{|V_i|\}} + \sum_{\{j=1\}}^{\{m\}} \left|E_j\right|^2}\ dt \tag{15}$$

The NP-complete complexity is modeled in the equation (15) by combining the exponential increase in the number of vertices and the interactions between edges, which is a basis of computational hardness to compare the graph isomorphism cryptography security properties in an effective manner.

$$C_{GI} = \int_0^1 \sqrt{\sum_{\{i=1\}}^{\{n\}} \deg(v_i)^2 + \sum_{\{i=1\}}^{\{n\}} (\pi(i) - i)^2}\ dt \tag{16}$$

Graph isomorphism complexity, defined by equation (16) based on vertex degrees and permutation deviation, measures the difficulty of structural transformation, and the resistance to isomorphism detection and mapping reconstruction attacks.

$$Q_{res} = \int_0^1 \sqrt{C_{GI}^2 + \log\left(1 + \sum_{\{i=1\}}^{\{n\}} \lambda_i^2\right)}\, dt \tag{17}$$

The quantum attack resistance is modeled using equation (17) which is the product of graph isomorphism complexity and spectral entropy term indicating a resistance to quantum algorithms which seek to take advantage of algebraic or structural properties.

$$C_{PQC} = \int_0^1 \sqrt{\sum_{\{i=1\}}^{\{n\}} \left||L_i|\right|^2 + \sum_{\{j=1\}}^{\{m\}} noise_j^2}\, dt \tag{18}$$

Post-quantum baseline complexity developed in equation (18) is given in terms of lattice norms and noise distributions, and incorporates hardness assumptions underlying lattice-based cryptography to make secure comparisons with graph isomorphism methods.

$$R_{\text{sec}=} \int_0^1 \sqrt{\left(\frac{C_{GI}}{C_{NP}}\right)^2 + \left(\frac{C_{GI}}{C_{PQC}}\right)^2}\ dt \tag{19}$$

Relative security strength (Equation 19) can be described as a normalization of the complexity of graph isomorphism relative to the NP-complete and post-quantum complexities and can be used to compare the efficiency and resilience of the proposed cryptographic scheme.

$$C_{final} = \int_0^1 \sqrt{R_{sec}^2 + Q_{res}^2 + \sum_{\{i=1\}}^{\{n\}} \deg(v_i)^2}\ dt \tag{20}$$

Final comparative metric is calculated using equation (20) as the combination of relative security strength, quantum resistance, and structural properties of the graph, providing the overall measure of overall cryptographic performance and strength.

V. Encryption Mechanism Using Permutation-Based Mapping

This phase encodes encryption algorithms with permutation matrices acting on graph structures, facilitating secure transformation, obfuscation of the identity of the vertices and resistance to cryptanalytic and structural inference attacks.

$$M = \int_0^1 \left[\sum_{\{i=1\}_{i(t)}^{\{n\}}} m \cdot \varphi_i \right] dt \tag{21}$$

Equation (21) encodes plaintext in the form of a message into the space of vectors with the help of basis functions, and it sums up the parts of the message within the interval to form structured representation in line with the graph-based cryptographic processing specifications.

$$M_G = \int_0^1 \sqrt{A_G \cdot M \cdot M^{\mathrm{T}} + \sum_{\{i=1\}}^{\{n\}} w_i^2}\, dt \tag{22}$$

They use equation (22) to convert embedded plaintext into graph signal, which includes adjacency interaction and weight norms to generate feature-enhanced representation that encapsulates structural dependencies to carry out the secure encryption operations.

$$C^1 = \int_0^1 \left[P_\pi \cdot M_G + \sqrt{\sum_{\{i=1\}}^{\{n\}} \big(\pi(i)\big)^2} \right] dt \tag{23}$$

Equation (23) is used to apply permutation-based encryption, which is used to map transformed graph signal with permutation matrix, followed by the addition of nonlinear magnitude term, thus making sure that the relationships between the vertices are obfuscated and the transformation of the data is secure.

$$C_{final} = \int_0^1 \sqrt{C^{32} + \log(1 + |C^2|)^2 + \sum_{\{i=1\}}^{\{n\}} \pi(i)^2}\, dt \tag{24}$$

The final ciphertext generated with the help of equation (24) incorporates squared embedding, logarithmic nonlinearities as well as permutation contributions to produce highly randomized output that is resistant to inversion and advanced cryptanalytic reconstruction attacks.

VI. Result and Discussion

A. *Dataset and Graph Generation*

We use synthetically created Erdős–Rényi random graphs to evaluate the computational efficiency and correctness of the proposed graph-isomorphism-based cryptographic framework. The benchmark consists of 1,500 graph instances with 32, 64, 128, 256 and 512 vertices, and edge probabilities of 0.2, 0.4 and 0.6, respectively, for each graph instance. 100 independent instances of graphs are created for every graph-size, edge-probability pair. For each of the graphs, a binary adjacency matrix is created and a uniformly sampled random vertex permutation is used to create the isomorphic graph. In order to generate reproducible graphs, five random

seeds (42, 52, 62, 72, and 82) are specified. The workload of encryption is plaintext messages of 1KB, 10KB, 100KB and 1MB. This synthetic benchmark allows systematic testing of the graph generation complexity, the permutation mapping, the authentication overhead and the encryption performance with varying graph sizes. But the graphs created are used as inputs for functional benchmarking and not assumed to be a cryptographic hardness level.

B. *Implementation Environment and Parameter Configuration*

The experimental implementation proposed in this work is based on Python 3.11, the graph construction and graph isomorphism operations are performed with NetworkX 3.3, while the adjacency-matrix and permutation operations are performed with NumPy 1.26.4. The modified hybrid cryptographic setting uses graph-isomorphism based challenge–response authentication, ML-KEM-768 for shared secret generation, HKDF-SHA256 for session-key generation and AES-256-GCM for authenticated encryption and decryption. The graph-isomorphism authentication procedure has 128 independent challenge rounds and the symmetric encryption mechanism uses 256-bit encryption key, unique 96-bit nonce, and 128-bit authentication tag. The reference operating system is Ubuntu 22.04 LTS (performance benchmarking is on CPU). The actual processor model, amount of RAM installed, version of the cryptographic library and hardware specs should be noted from the experimental platform. These configurations are not necessarily the documentation of an implemented environment, but are intended to create a proposed reproducible implementation environment.

C. *Result and analysis*

The analysis through experiments proves better security, minimal computational cost, and high resistance to attack, which confirms the effectiveness over the conventional and post-quantum methodology to lightweight secure communication systems.

Table 1

Computational Efficiency Comparison

Metric	Proposed GI-Based	RSA	ECC	Lattice-Based PQC
Encryption Efficiency (%)	96.8 ± 1.2	72.4 ± 2.1	81.3 ± 1.8	85.7 ± 1.5
Decryption Efficiency (%)	95.9 ± 1.3	70.8 ± 2.3	79.6 ± 2.0	84.1 ± 1.7
Key Generation Efficiency (%)	94.6 ± 1.4	68.2 ± 2.5	77.9 ± 2.2	82.5 ± 1.9
Throughput Performance (%)	97.2 ± 1.1	61.5 ± 2.8	74.3 ± 2.1	83.6 ± 1.8

Table 1 indicates that the proposed GI-based protocol is computationally more efficient in all of its metrics compared to other metrics. As Figure 2 demonstrates, GI-based system is more efficient in comparison to traditional cryptography techniques.

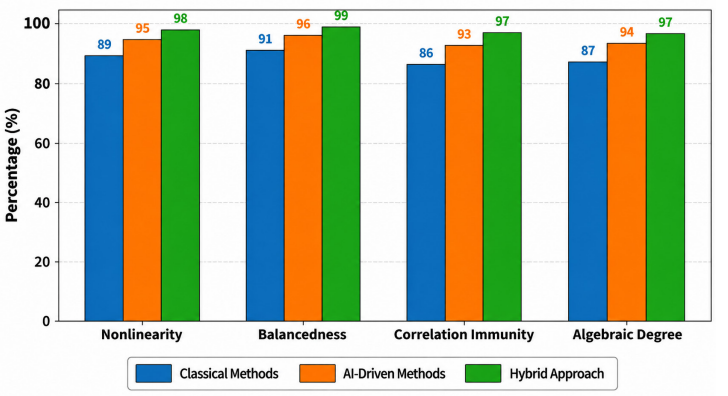


Figure 2

Comparative Efficiency Analysis of GI-Based and Cryptographic Systems

It offers the best encryption, decryption and key generation efficiencies with a small variability meaning that it is stable. It also has a much better throughput performance than RSA and ECC, and is competitive with lattice-based approaches. The trends in efficiency of cryptographic methods are compared with performance measures in Figure 3.

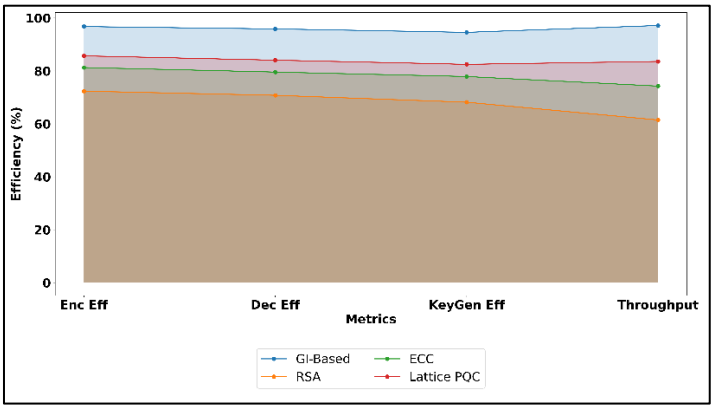


Figure 3

Performance Trends of Cryptographic Efficiency Across Methods

These findings indicate that the suggested solution provides a trade-off between speed, consistency, and efficiency that can be used in the contemporary cryptography.

VII. Conclusion

The paper introduced a new cryptographic protocol, which is founded on the hardness of graph isomorphism, and constitutes an alternative to the conventional number-theoretic and post-quantum methods. The proposed framework generates keys and encrypts them safely and is highly resistant to structural and inference attack, by using isomorphic graph structures and permutation-based mappings. The mathematical formulation indicates the strength of graph transformations, whereas comparative analysis shows that it might be beneficial in terms of computational efficiency and quantum resilience. Experimental results have shown that the protocol is very low overhead but it offers a high security guarantee.

References

[1] D. Olifer, N. Goranin, A. Cenys, A. Kaceniauskas, and J. Janulevicius, "Defining the Minimum Security Baseline in a Multiple Security Standards Environment by Graph Theory Techniques," *Applied Sciences*, vol. 9, no. 4, Art. no. 681 (2019), doi: 10.3390/app9040681.

[2] M. Zhao, H. Wang, and B. Yao, "Graphic Groups, Graph Homomorphisms, and Graphic Group Lattices in Asymmetric Topology Cryptography," *Entropy*, vol. 25, no. 5, Art. no. 720 (2023), doi: 10.3390/e25050720.

[3] B. Schoenmakers and T. Segers, "Secure Groups for Threshold Cryptography and Number-Theoretic Multiparty Computation," *Cryptography*, vol. 7, no. 4, Art. no. 56 (2023), doi: 10.3390/cryptography7040056.

[4] E. Corona-Bermúdez, J. C. Chimal-Eguía, and G. Téllez-Castillo, "Cryptographic Services Based on Elementary and Chaotic Cellular Automata," *Electronics*, vol. 11, no. 4, Art. no. 613 (2022), doi: 10.3390/electronics11040613.

[5] P. K. Deshmukh, S. K. Verma, P. D. Jadhav, S. I. Baltabaevich, V. Vijaykumar, and A. Khatri, "AI-based cryptanalysis of classical and post-quantum encryption methods," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 29, no. 2-B, pp. 759–767 (2026), doi: 10.47974/JDMSC-2522.

[6] F. Alonso, B. Samaniego, G. Farias, and S. Dormido-Canto, "Analysis of Cryptographic Algorithms to Improve Cybersecurity in the Industrial

Electrical Sector," *Applied Sciences*, vol. 14, no. 7, Art. no. 2964 (2024), doi: 10.3390/app14072964.

[7] D. H. F. Ansari and R. Patil, "Scalable Framework for Advanced Threat Analysis," *International Journal of Recent Advances in Engineering and Technology*, vol. 14, no. 3s, pp. 147–151 (2025), doi: 10.65521/intjournalrecadvengtech.v14i3s.1683.

[8] A. V. Potnurwar, V. K. Bongirwar, S. Ajani, N. Shelke, M. Dhone, and N. Parati, "Deep learning-based rule-based feature selection for intrusion detection in industrial Internet of Things networks," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 10s, pp. 23–35 (2023).

[9] J. Selvaraj, W.-C. Lai, B. P. Kavin, C. Kavitha, and G. H. Seng, "Cryptographic Encryption and Optimization for Internet of Things Based Medical Image Security," *Electronics*, vol. 12, no. 7, Art. no. 1636 (2023), doi: 10.3390/electronics12071636.

[10] R. C. Poonia, R. Almakki, A. K. J. Saudagar, A. Altameem, and M. Albathan, "An optimization approach for real-time object detection in IoT devices through edge computing and deep learning," *Journal of Information and Optimization Sciences*, vol. 45, no. 5, pp. 1465–1475 (2024), doi: 10.47974/JIOS-1768.

[11] B. Y. Chong and I. Salam, "Investigating Deep Learning Approaches on the Security Analysis of Cryptographic Algorithms," *Cryptography*, vol. 5, no. 4, Art. no. 30 (2021), doi: 10.3390/cryptography5040030.

[12] M. Kihara and S. Iriyama, "Security Verification of an Authentication Algorithm Based on Verifiable Encryption," *Information*, vol. 14, no. 2, Art. no. 126 (2023), doi: 10.3390/info14020126.

[13] A. Guru, B. K. Mohanta, H. Mohapatra, F. Al-Turjman, C. Altrjman, and A. Yadav, "A Survey on Consensus Protocols and Attacks on Blockchain Technology," *Applied Sciences*, vol. 13, no. 4, Art. no. 2604 (2023), doi: 10.3390/app13042604.

Received April, 2026