International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

A Lightweight Hybrid Encryption and Blockchain-Inspired Secure Storage Model for IOMT Data

Ms. S. Mary Helan Felista[1], Dr. M. Ganaga Durga[2]
[1]Research Scholar, Department of Computer Applications, Sri Meenakshi Govt. Arts College for Women(A), Madurai & Assistant Professor, Department of MCA, Fatima College, Mary Land, Madurai, India.
[2]Research Supervisor, Assistant Professor, Department of Computer Applications, Sri Meenakshi Govt. Arts College for Women(A), Madurai, India.
EMail Id: *felistamichael2012@gmail.com[1], mgdurga@yahoo.com[2]*

Abstract
The Internet of Medical Things (IoMT) enables continuous patient monitoring and intelligent healthcare services, but it also introduces serious challenges related to the confidentiality and integrity of sensitive medical data stored in distributed environments. Due to their limited computational and energy resources, IoMT devices require lightweight yet robust security mechanisms that can protect data without incurring excessive overhead. This paper presents a lightweight IoMT data storage framework that combines hybrid encryption with a blockchain-inspired integrity assurance mechanism. The proposed approach employs ChaCha20 for efficient data encryption at IoMT devices, AES for secure symmetric key protection at gateway nodes, and Elliptic Curve Cryptography (ECC) for secure key exchange and mutual authentication. To ensure data integrity and tamper resistance in cloud storage, a hash-linked storage structure integrated with Merkle tree verification is designed, drawing inspiration from blockchain technology without deploying a full blockchain network. A Python-based prototype implementation is developed to evaluate the performance of the framework in terms of encryption latency, throughput, and integrity verification overhead. Experimental results demonstrate that the proposed scheme achieves strong security guarantees while maintaining low computational cost, making it well suited for resource-constrained IoMT environments. Overall, the proposed model offers a practical and scalable solution for securing healthcare data storage in next-generation IoMT systems.

Keywords: *IoMT security; Hybrid encryption; ChaCha20; Blockchain-inspired storage; Merkle tree.*

1. Introduction
The Internet of Medical Things (IoMT) has significantly advanced modern healthcare by enabling continuous patient monitoring, remote diagnosis, and data-driven clinical decision-making, resulting in improved efficiency and patient outcomes. However, the large volumes of sensitive medical data generated by IoMT devices raise serious concerns regarding data confidentiality, integrity, and unauthorized access, particularly when such data are stored and processed in distributed cloud environments [2], [13]. Ensuring secure data storage and transmission remains a critical challenge due to the heterogeneous nature of IoMT devices, their limited computational resources, and strict real-time performance requirements [16]. Conventional security mechanisms based on single-layer encryption or centralized key management are often inadequate for IoMT systems, as they introduce high computational overhead and create single points of

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

failure [16]. As a result, there is a growing need for lightweight yet robust security frameworks that can protect sensitive medical data while maintaining efficiency in resource-constrained environments. Hybrid cryptographic approaches have emerged as promising solutions to address these challenges by combining the efficiency of symmetric encryption with the strong security guarantees of asymmetric cryptography. Lightweight stream ciphers such as ChaCha20 offer fast and secure data encryption, while Elliptic Curve Cryptography (ECC) enables efficient key exchange and mutual authentication with smaller key sizes and reduced computational cost [1], [3]. In addition, symmetric encryption schemes such as AES are commonly used to protect and manage session keys securely within gateway nodes.Beyond encryption, ensuring data integrity and tamper resistance in cloud-based storage is equally important. Blockchain-inspired techniques, including hash-linked data structures and Merkle tree verification, have been explored to provide integrity assurance and auditability without relying on fully decentralized blockchain networks [7], [18]. These approaches enable efficient detection of unauthorized data modifications while avoiding the high overhead associated with traditional blockchain deployments. Despite these developments, existing solutions often address cryptographic security and integrity assurance separately, limiting their effectiveness in practical IoMT deployments [19], [26]. To address this gap, this paper proposes a lightweight hybrid encryption and blockchain-inspired secure storage framework tailored for IoMT environments. The proposed scheme integrates ChaCha20 for device-level data encryption, AES for secure session key wrapping at the gateway, and ECC for key exchange and mutual authentication, along with a Merkle tree–based integrity mechanism to ensure tamper resistance in cloud storage. The framework is evaluated through a Python-based prototype to demonstrate its effectiveness in achieving strong security with low computational overhead, making it suitable for real-world IoMT applications.

2. Related Work

Several studies have explored cryptographic techniques to secure IoMT and IoT environments. Hybrid cryptographic schemes that combine symmetric and asymmetric encryption have been widely adopted to balance computational efficiency and security. In such approaches, lightweight stream ciphers such as ChaCha20 are employed for bulk data encryption, while asymmetric cryptographic techniques, particularly Elliptic Curve Cryptography (ECC), are used for secure key exchange and authentication [1], [3]. Lightweight ECC-based authenticated key agreement protocols have been proposed to support confidentiality and scalable key management in resource-constrained IoT environments [3], [11]. Additionally, hybrid cryptosystems integrating ChaCha20 and ECC have demonstrated improved performance and practicality in secure multimedia and image encryption applications [4]. Beyond cryptographic protection, blockchain-inspired mechanisms have gained attention for enhancing data integrity and tamper resistance. Several works investigate decentralized storage and hash-linked data structures to improve auditability and trust in healthcare data management systems [7], [18]. Blockchain-based healthcare frameworks combined with authenticated encryption schemes have been proposed to ensure data confidentiality and immutability while minimizing reliance on centralized authorities [12], [20], [24]. Recent surveys further highlight the adoption of lightweight blockchain architectures and distributed storage models as promising solutions for improving data privacy, traceability, and trust in IoMT systems [25], [27]. Despite these advances, most existing studies address cryptographic security or decentralized storage in isolation. Limited research has focused on integrating a hybrid encryption scheme with a blockchain-inspired integrity mechanism specifically tailored for IoMT environments while maintaining low computational overhead [19], [26].

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

3. System Model

The proposed system model is designed for secure data collection, encryption, and storage in an IoMT-based healthcare environment [13]. It consists of three major entities: IoMT devices, a gateway node, and cloud storage infrastructure. IoMT Devices Wearable sensors, implantable devices, and bedside monitors are examples of resource-constrained medical devices that continuously collect patients' physiological data (e.g., heart rate, glucose level, ECG signals). Due to the limited computation power, memory, and energy capacity, IoMT devices perform only lightweight cryptographic operations [16]. The sensed data is symmetrically encrypted by the devices using ChaCha20 in the proposed model for ensuring confidentiality with minimum overhead [1]. Gateway Node: It works as an intermediary and a more capable entity between IoMT devices and the cloud. It aggregates encrypted data from various devices, manages cryptographic keys, and performs additional security operations that are too heavy for IoMT devices [3], [11]. These include wrapping session keys based on AES [10], key exchange and mutual authentication based on ECC [14], hybrid encryption orchestration, and packaging blocks of data with integrity metadata prior to forwarding them for storage in the cloud The cloud provides scalable storage for encrypted medical records and associated integrity information like hash values and Merkle tree roots. It is assumed to be honest-but-curious: the cloud correctly stores and retrieves data but may attempt to infer sensitive information from stored contents [12], [24]. Therefore, all data are stored only in encrypted form, and the cloud is never given access to decryption keys. IoMT devices communicate with the gateway over secure channels that were set up during device registration and key establishment with ECC-based authentication and key agreement [11], [14]. The gateway communicates with the cloud over standard secure network connections. While communication channels are protected, the cloud itself is not fully trusted a fact that motivates the need for strong encryption and blockchain-inspired integrity mechanisms [7], [18], [25]. Such a system model efficiently separates responsibilities between the resource-constrained devices, which only need to handle data collection, and the gateway, which handles heavier cryptographic operations and management tasks. In this manner, security and performance can be balanced in resource-constrained IoMT environments.

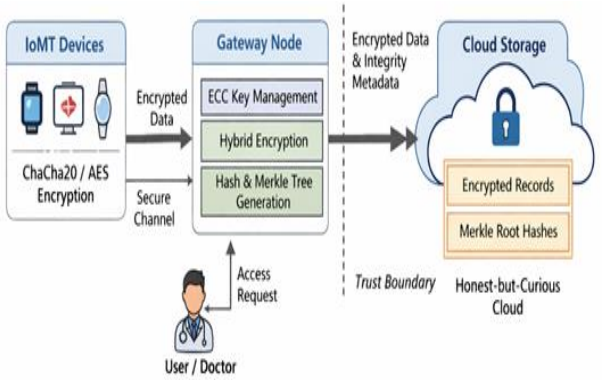

Figure 1 Architecture of the Proposed Hybrid Encryption and Blockchain-Inspired Secure Storage Framework

Architecture of the proposed hybrid encryption and blockchain-inspired secure storage framework for IoMT systems, showing data flow among IoMT devices, gateway, and cloud storage. The framework highlights the interaction between lightweight IoMT devices, which collect and encrypt patient data using ChaCha20, with the gateway node performing AES-based session key management and ECC-based key exchange, to the cloud storage layer that securely stores encrypted data and integrity proofs. The blockchain-inspired structure enforces tamper resistance and data integrity through hash-linked blocks and Merkle tree verification. This architectural model defines a secure and efficient as well as scalable architecture for handling sensitive medical data within resource-constrained IoMT environments.

4. Proposed Hybrid Encryption Scheme

This section elaborates on the proposed hybrid

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

encryption scheme that combines ChaCha20, AES, and Elliptic Curve Cryptography (ECC) for efficient data confidentiality with secure key management in resource-constrained IoMT environments. The overall workflow of the scheme is illustrated in Figure. 2, and the role of each cryptographic primitive is summarized in Table 1.

4.1.Cryptographic Components

ChaCha20: Data Encryption at IoMT Devices

ChaCha20 is employed at the IoMT layer for encrypting sensed medical data because of its high throughput while keeping energy consumption low for constrained devices. Each device employs ChaCha20 with a fresh session key in order to guarantee confidentiality of its data.

AES: Session Key Wrapping at Gateway

The gateway performs AES encryption to the ChaCha20 session keys generated at IoMT devices; AES allows fast symmetric encryption and further allows handling multiple session keys aggregated from heterogeneous devices in an efficient manner.

ECC: Key Exchange and Mutual Authentication

It is used between IoMT devices and the gateway for establishing secure channels through mutual authentication. Using ECC-based key agreement, such as ECDH, the shared secret is derived to protect the AES key so that keys are not exposed during transmission.

4.2.Hybrid Encryption Workflow

As illustrated in Figure 2, the proposed scheme operates in the following steps:

Session Key Generation

Every IoMT device generates a ChaCha20 session key KC for each sensing session.

Data Encryption

The ChaCha20 encryption at the device encrypts medical data D into ciphertext,

$$C=ChaCha20(D,KC)$$

The encrypted data C and the session key identifier are sent to the gateway via a secure channel.

Key Wrapping with AES

The gateway encrypts the ChaCha20 session key using AES with key KA:

$$W=AES(KC,KA)$$

ECC-Based Key Protection

ECC-based key agreement securely exchanges the AES key KA between the IoMT device and the gateway, ensuring confidentiality and mutual authentication.

Forwarding to Cloud Storage

The gateway relays the encrypted data C along with wrapped key W and integrity metadata to the cloud. The cloud maintains only encrypted content and has no possession of any decryption keys.

Data Recovery

Upon a valid request, the gateway fetches {C,W}, decrypts W to recover KC, and finally decrypts C using ChaCha20. This layered design minimizes the computational burden on IoMT devices while ensuring secure and scalable key management at the gateway.

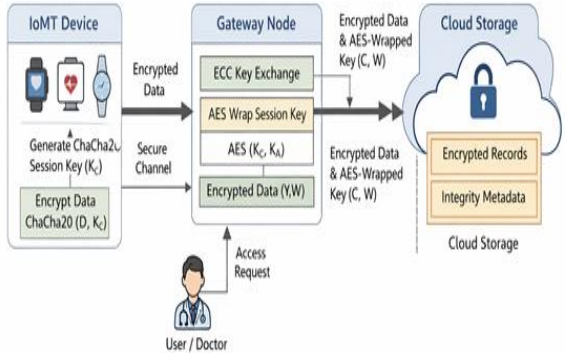

Figure 2 Hybrid Encryption and Key Management Workflow

Figure 2 illustrates the encryption workflow where IoMT devices encrypt data using ChaCha20, the gateway wraps session keys using AES and secures AES keys through ECC-based exchange, and the cloud stores encrypted data along with integrity metadata.

4.3.Design Rationale

The proposed hybrid design implements defense in depth by integrating complementary cryptographic primitives across various layers of the IoMT architecture. At the device edge, ChaCha20 is used for data encryption to ensure high efficiency with

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

very low energy and computational overhead, making it particularly well-suited for resource-constrained nodes in IoMTs. For scalable key management at the gateway, AES serves to securely wrap and manage multiple session keys, thus enabling fast symmetric operations without introducing significant latency. Strong authentication and secure key exchange are provided by ECC, which provides better security than other cryptographic techniques with smaller key sizes and thereby reduces communication and storage overhead. This layered design means that if one cryptographic component is compromised, the remaining layers continue to protect sensitive medical data, ensuring resilient and comprehensive protection for IoMT storage systems.

Table 1 Role of Cryptographic Primitives in the Proposed Scheme

Algorithm	Applied Layer	Purpose	Justification
ChaCha20	IoMT devices	Medical data encryption	High speed and low energy consumption suitable for constrained devices
AES	Gateway node	Session key wrapping	Efficient symmetric encryption for handling multiple keys
ECC	Device–Gateway link	Key exchange and authentication	Strong security with small key sizes and low overhead

5. Blockchain-Inspired Secure Storage

In this work, a lightweight blockchain-inspired storage mechanism is adopted for tamper-resistant encrypted IoMT records stored in the cloud, with hash chaining and Merkle tree construction without a full blockchain network deployment.

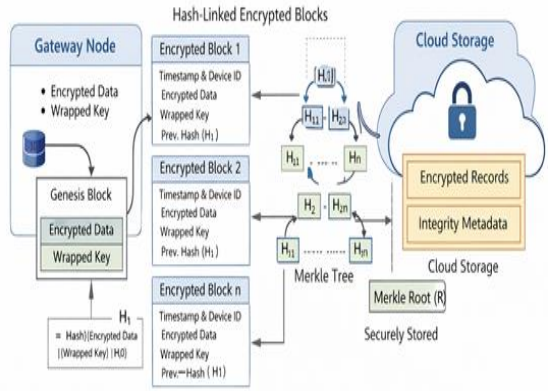

Figure 3 Blockchain-Inspired Secure Storage and Integrity Verification Model

The overall storage model is illustrated in Figure. 3, showing the formation of hash-linked encrypted data blocks at the gateway and the construction of a Merkle tree over block hashes, where the Merkle root enables efficient integrity verification of cloud-stored IoMT data.

5.1.Hash-Linked Data Records

Each encrypted medical record is structured as a block and connected to the previous block through cryptographic hash functions, forming an immutable chain. Let Ci denote the encrypted data record of the i-th block and Hi−1 represent the hash of the previous block. The hash of the current block is computed as:

$$Hi=Hash(Ci||Hi-1)$$

where "||" represents concatenation. This hash-chaining mechanism ensures that any alteration in a stored record propagates through subsequent hashes, making tampering immediately detectable. The first

IRJAEH

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

block, referred to as the genesis block, uses a predefined initial hash H0.

5.2.Merkle Tree Construction

To achieve scalable and efficient integrity verification over numerous stored records, a Merkle tree is periodically constructed using the hashes of multiple blocks. For a set of block hashes {H1,H2,…,Hn}, the Merkle tree is built hierarchically by hashing pairs of child nodes until a single root hash R is produced:

$$R=MerkleRoot(H1,H2,\ldots,Hn)$$

The Merkle root serves as a compact integrity fingerprint for all records in the batch and is securely stored at the gateway or a trusted authority.

5.3.Integrity Verification Process

During data retrieval, the gateway recomputes the hash of the requested block and verifies it using a Merkle proof against the stored Merkle root R. If the recomputed root matches the stored one, the data are confirmed as intact; otherwise, tampering is detected. This verification approach operates with logarithmic complexity, requiring only a small subset of hashes instead of the entire dataset.

5.4.Design Rationale

The proposed blockchain-inspired storage model provides strong security guarantees while remaining lightweight and practical for IoMT environments. Hash chaining ensures tamper resistance by making unauthorized modifications to encrypted records immediately detectable. Merkle trees enable fast integrity validation with minimal computational and communication overhead. Unlike traditional blockchain systems, this design avoids resource-intensive mechanisms such as mining, consensus algorithms, and fully distributed ledgers—thereby reducing complexity and improving suitability for resource-constrained IoMT devices. Furthermore, the architecture is inherently scalable, capable of accommodating the continuously growing medical data generated by heterogeneous IoMT nodes. Together, hash-linked blocks and Merkle tree-based integrity verification deliver a secure, efficient, and scalable framework for protecting IoMT data in cloud storage environments.

6. Implementation and Performance Evaluation

This section discusses the prototype implementation of the proposed hybrid encryption and blockchain-inspired secure storage framework and evaluates its performance in a simulated IoMT environment.

6.1.Prototype Implementation

A Python-based prototype was developed using standard cryptographic libraries to emulate three entities: IoMT devices, a gateway node, and cloud storage.

- **IoMT Devices:** Simulated sensor nodes generate synthetic medical data packets and perform ChaCha20 encryption using lightweight session keys.
- **Gateway Node:** The gateway executes AES-based session key wrapping, ECC-based key exchange and mutual authentication, block generation, and Merkle tree construction.
- **Cloud Storage:** The cloud stores encrypted data blocks and Merkle tree nodes and allows retrieval for integrity verification.

All experiments were performed on a workstation with an Intel-class processor and 16 GB RAM. Each experiment was repeated 20 times, and average values were reported to ensure consistent and reliable result

6.2.Evaluation Metrics

The following performance metrics were considered:

- **Encryption Time (Tenc):** Time taken by IoMT devices to encrypt data using ChaCha20.
- **Decryption Time (Tdec):** Time taken to decrypt data at the gateway.
- **Key Exchange Latency (Tke):** Time required for ECC-based session key establishment.
- **Throughput (Th):** Amount of data encrypted per second (KB/s).
- **Merkle Tree Construction Time (Tmtc):**

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

Time to build a Merkle tree over encrypted blocks.

- **Merkle Verification Time (Tmtv):** Time to verify data integrity using Merkle proofs.

6.3.Experimental Setup

To evaluate scalability, medical data packets of sizes 1 KB, 10 KB, 100 KB, and 1 MB were generated. ECC operations were executed only during the session initialization phase, while ChaCha20 encryption and AES-based key wrapping were applied to every data packet. For integrity verification, the number of blocks used in the Merkle tree evaluation ranged from 10 to 500 Shown in Table 2.

6.4.Performance Results

6.4.1. Encryption, Decryption, and Throughput

Table 4 presents the average encryption time, decryption time, and system throughput for different data sizes.

Table 2 Encryption, Decryption Time and Throughput

Data Size (KB)	Tenc(ms)	Tdec (ms)	Throughput (KB/s)
1	0.12	0.10	8200
10	0.35	0.30	7900
100	1.90	1.70	7400
1024 (1 MB)	18.50	17.90	6800

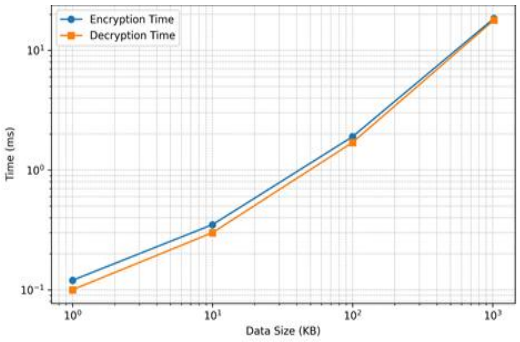

Figure 4 Encryption and Decryption Time Vs. Data Size

Figure 4 illustrates the encryption and decryption time as a function of data size. The results show that ChaCha20 achieves consistently low latency even for large packets, making it suitable for resource-constrained IoMT devices.

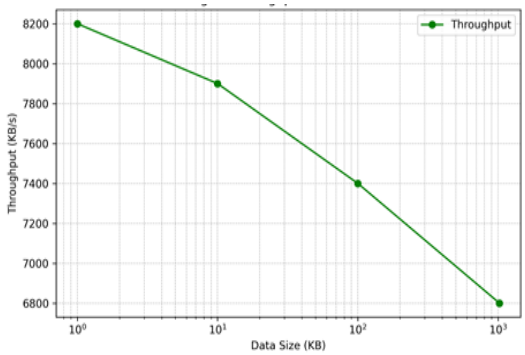

Figure 5 Throughput Vs. Data Size

Figure 5 presents throughput variations with data size. Although throughput slightly decreases with increasing data size due to processing overhead, high data rates are maintained across all scenarios.

6.4.2. Merkle Tree Performance

The efficiency of the blockchain-inspired integrity mechanism was evaluated by measuring the Merkle tree construction and verification times for varying numbers of blocks. The results are presented in Table 3.

Table 3 Merkle Tree Construction and Verification Time

Number of Blocks	Tmtc (ms)	Tmtv (ms)
10	1.0	0.3
50	1.8	0.5
100	2.6	0.7
200	3.9	1.0
500	6.5	1.6

IRJAEH
International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

Figure. 6 shows that both construction and verification times grow logarithmically with the number of blocks, demonstrating good scalability of the proposed integrity model.

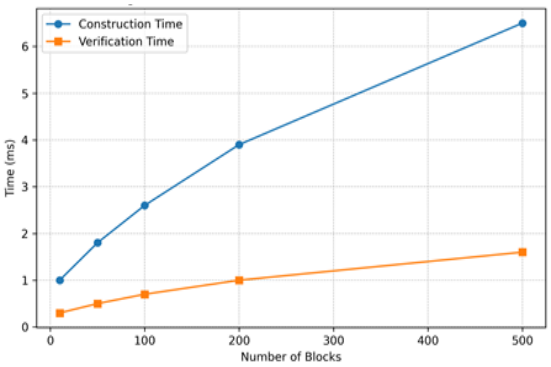

Figure 6 Merkle Tree Construction and Verification Time Vs. Number of Blocks.

6.5.Key Exchange Overhead

ECC-based key exchange introduced an average latency of approximately 15–18 ms during session establishment. Since this operation is performed only once per communication session, its effect on continuous data encryption is minimal. This confirms the practicality of integrating ECC for secure and efficient key management in IoMT environments.

6.6.Discussion

An analysis of the experimental results reveals several key observations. ChaCha20 demonstrates fast encryption and decryption, making it well suited for IoMT devices with limited computational and energy resources, thereby ensuring low device-side overhead. The use of AES-based key wrapping introduces negligible performance cost while simplifying secure symmetric key management at the gateway. Although ECC-based operations incur higher computational complexity, they are restricted to the session establishment phase, ensuring strong security guarantees without affecting runtime performance. In addition, Merkle tree–based verification provides efficient and scalable integrity assurance, even for large medical datasets. Overall, the proposed scheme achieves a balanced trade-off between security strength and computational efficiency, making it practical for secure IoMT data storage and access. Furthermore, the Python-based prototype and experimental evaluation demonstrate that the proposed ChaCha20–AES–ECC hybrid encryption framework, combined with blockchain-inspired secure storage, can effectively protect sensitive IoMT data with low latency, high throughput, and scalable integrity verification, thereby validating its feasibility for real-world IoMT deployments.

7. Security Analysis

This section evaluates the security of the proposed hybrid encryption and blockchain-inspired framework against common threats in IoMT environments.

7.1.Data Confidentiality

Medical data are encrypted at IoMT devices using ChaCha20, a secure stream cipher known for its high performance and strong resistance to cryptanalysis. Since only encrypted data are transmitted and stored, an adversary intercepting network traffic or accessing cloud storage cannot recover plaintext without the corresponding session key. Additionally, ChaCha20 session keys are wrapped using AES at the gateway and protected through ECC-based key exchange, ensuring that symmetric keys are never exposed in plaintext during transmission.

7.2.Secure Key Management

ECC is employed for key exchange and mutual authentication between IoMT devices and the gateway. Due to the computational hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP), it is infeasible for attackers to derive private keys from public parameters. This guarantees secure establishment of AES keys used for session key wrapping. By separating data encryption (ChaCha20) from key protection (AES + ECC), the framework implements layered security and significantly reduces the risk of key compromise.

7.3.Data Integrity and Tamper Resistance

The blockchain-inspired storage model employs

IRJAEH

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

hash chaining and Merkle trees to ensure the integrity of encrypted medical data. Any modification to a data block changes its hash, which propagates through the chain, while a mismatch in the Merkle root immediately reveals tampering. Thus, even in scenarios where the cloud is honest-but-curious or partially compromised, unauthorized modifications to stored records can be efficiently detected.

7.4.Resistance to Common Attacks

The proposed framework is resilient against several common attacks:

- **Eavesdropping:** Encrypted communication prevents unauthorized disclosure of sensitive data.
- **Man-in-the-Middle (MitM):** ECC-based mutual authentication prevents attackers from intercepting or altering key exchanges.
- **Replay Attacks:** Session-based keys and timestamps prevent reuse of old messages.
- **Data Tampering:** Hash chaining and Merkle tree verification detect unauthorized modifications.
- **Key Compromise:** Exposure of a session key does not affect other sessions due to fresh key generation.

Table 4 Security Analysis of the Proposed Scheme

Security Property	Mechanism Used	Achieved
Confidentiality	ChaCha20 data encryption	✓
Secure key exchange	ECC	✓
Key protection	AES key wrapping	✓
Integrity	Hash chaining, Merkle tree	✓
Authentication	ECC-based mutual auth	✓
Tamper detection	Merkle root verification	✓
Replay resistance	Session keys, timestamps	✓
Scalability	Lightweight blockchain model	✓

By combining efficient symmetric encryption, lightweight asymmetric key management, and blockchain-inspired integrity assurance, the proposed framework delivers comprehensive security for IoMT data without imposing significant computational overhead on resource-constrained devices.

Conclusion and Future Work

This paper presented a hybrid encryption and blockchain-inspired secure storage framework for protecting sensitive medical data generated by Internet of Medical Things (IoMT) devices. The proposed approach integrates ChaCha20 for lightweight data encryption at IoMT devices, AES for efficient session key wrapping at the gateway, and ECC for secure key exchange and mutual authentication. Furthermore, a blockchain-inspired storage mechanism based on hash chaining and Merkle trees was introduced to ensure data integrity and tamper resistance in cloud-based storage environments. A Python-based prototype implementation and comprehensive performance evaluation demonstrated that the proposed framework achieves low encryption latency, high throughput, and scalable integrity verification, making it well suited for resource-constrained IoMT environments. The accompanying security analysis further confirmed that the framework effectively

IRJAEH

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

mitigates common threats, including eavesdropping, man-in-the-middle attacks, replay attacks, and data tampering. Future work will focus on deploying the proposed framework on real IoMT hardware platforms to evaluate energy consumption, memory utilization, and overall practicality in constrained environments. The framework can be further extended by incorporating fine-grained access control and role-based authorization to support secure multi-user healthcare scenarios. Another promising direction involves integrating anomaly detection or AI-based intrusion detection mechanisms to enhance runtime security and proactively identify malicious behavior. To ensure long-term resilience, future research will also explore the integration of post-quantum cryptographic primitives to address emerging quantum threats. Finally, large-scale evaluation using real-world healthcare datasets and deployment in clinical environments will be essential to assess the scalability, reliability, and real-world impact of the proposed framework.

References

[1]. Langley et al., "ChaCha20 and Poly1305 for IETF protocols," RFC 8439, IETF, 2019.
[2]. M. Conti, A. Dehghantanha, K. Franke, and S. Watson, "Internet of Things security and forensics: Challenges and opportunities," Future Generation Computer Systems, vol. 78, pp. 544–546, 2019.
[3]. Y. Zhang, D. He, N. Kumar, and K.-K. R. Choo, "Lightweight ECC-based authenticated key agreement for IoT," IEEE Internet of Things Journal, vol. 7, no. 1, pp. 553–565, 2020.
[4]. S. Roy, A. K. Pal, and S. Misra, "A ChaCha20–ECC based hybrid cryptosystem for secure multimedia data," Multimedia Tools and Applications, vol. 79, no. 47, pp. 35229–35252, 2020.
[5]. K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," IEEE Access, vol. 7, pp. 10127–10149, 2019.
[6]. S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F.-Y. Wang, "An overview of smart contract: Architecture, applications, and future trends," IEEE Access, vol. 6, pp. 35448–35461, 2019.
[7]. A. Khatoon, "A blockchain-based secure healthcare system," Future Generation Computer Systems, vol. 101, pp. 511–521, 2019.
[8]. A. Rghioui, A. Oumnad, and M. Z. Hasnaoui, "A survey on cryptography in IoT," Journal of Ambient Intelligence and Humanized Computing, vol. 11, pp. 2025–2048, 2020.
[9]. M. Wazid, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, "Secure three-factor user authentication scheme for IoT," IEEE IoT Journal, vol. 7, no. 6, pp. 5338–5352, 2020.
[10]. H. K. Patil and R. Seshadri, "Performance evaluation of AES for IoT applications," Proc. IEEE ICACCI, 2020, pp. 1802–1807.
[11]. R. R. Jangirala, A. K. Das, and J. J. P. C. Rodrigues, "Provably secure ECC-based authenticated key agreement for IoT," Computer Communications, vol. 160, pp. 401–412, 2020.
[12]. M. Baza et al., "Blockchain-based secure data sharing for IoMT," IEEE Access, vol. 8, pp. 150463–150483, 2020.
[13]. P. Zhang, M. Zhou, and G. Fortino, "Security and trust issues in IoT-based healthcare systems: A survey," IEEE IoT Journal, vol. 7, no. 7, pp. 6423–6444, 2020.
[14]. S. H. Islam et al., "A provably secure ECC-based mutual authentication scheme for healthcare systems," Journal of Medical Systems, vol. 44, no. 5, pp. 1–15, 2020.
[15]. M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted IoT data," IEEE IoT Journal, vol. 6, no. 5, pp. 7702–7712, 2019.
[16]. Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu, and

International Research Journal on Advanced Engineering Hub
e ISSN: 2584-2137
Vol. 04 Issue: 01 January 2026
Page No: 89 - 99
www.irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0012

W. Lv, "Edge computing security: State of the art and challenges," Proc. IEEE, vol. 107, no. 8, pp. 1608–1631, 2019.
[17]. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," White Paper, 2008. (Foundational, cited for blockchain concept)
[18]. Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," Int. J. Web and Grid Services, vol. 14, no. 4, pp. 352–375, 2019.
[19]. R. Kumar and R. Tripathi, "Scalable and secure IoT framework using blockchain," Wireless Networks, vol. 27, pp. 1–15, 2021.
[20]. S. Hussain, S. Ullah, and H. K. Qureshi, "Secure healthcare data sharing using blockchain and ECC," IEEE Access, vol. 9, pp. 105997–106011, 2021.
[21]. NIST, "Lightweight Cryptography Standardization Process," NISTIR 8114, Gaithersburg, MD, USA, 2020.
[22]. NIST, "Digital Signature Standard (DSS)," FIPS PUB 186-5, 2023.
[23]. Y. Lu, "Blockchain and the related issues: A review of current research topics," Journal of Management Analytics, vol. 7, no. 2, pp. 231–255, 2020.
[24]. M. Seliem, K. Elgazzar, and K. Khalil, "Towards privacy-preserving IoMT using blockchain," IEEE Access, vol. 9, pp. 81617–81630, 2021.
[25]. A. Reyna et al., "On blockchain and its integration with IoT: Challenges and opportunities," Future Generation Computer Systems, vol. 88, pp. 173–190, 2019.
[26]. R. Li, T. Song, N. Mei, H. Li, B. Cheng, and X. Sun, "Blockchain for large-scale IoT data storage and protection," IEEE Transactions on Services Computing, vol. 13, no. 4, pp. 1–13, 2020.
[27]. S. Tanwar, S. Tyagi, and S. Kumar, "The role of blockchain in securing IoT," Computer Communications, vol. 154, pp. 447–460, 2020.
[28]. M. Al-Rakhami and A. Gumaei, "Hybrid blockchain-based framework for secure healthcare data storage," IEEE Access, vol. 10, pp. 33215–33230, 2022.