Lightweight Compression and Chaos-Based Encryption for Secure IoT Healthcare Data Storage on Blockchain

S. Mubeena
Department of Electronics and Communication Engineering, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India
mubeena_ece_july2022@cescent.education

P. K. Jawahar
Department of Electronics and Communication Engineering, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, India
jawahar@crescent.education (corresponding author)

Received: 23 June 2025 | Revised: 6 August 2025, 29 August 2025, 10 September 2025, and 12 September 2025 | Accepted: 15 September 2025

 DOI: https://doi.org/10.48084/etasr.12888

ABSTRACT

The increasing incidence of cyberattacks on healthcare infrastructure has highlighted the critical vulnerability of sensitive patient data, necessitating the implementation of advanced security measures. Although blockchain technology offers a promising solution for ensuring data integrity and confidentiality, its integration into resource-constrained medical devices, especially low-power embedded systems, presents significant challenges. This study addresses these challenges by proposing two novel frameworks: Zlib Hardware Accelerator with Adaptive Dictionary Encoding (ZHA-ADE) for efficient data compression, and Chaotic Hybrid Asymmetric and Symmetric Encryption (CHASE) for lightweight and secure encryption. ZHA-ADE enhances traditional Zlib compression with adaptive dictionary encoding, optimizing biomedical data throughput and reducing the computational load on ARM Cortex-A microcontrollers while maintaining compatibility with blockchain. Simultaneously, CHASE combines chaotic key generation with Advanced Encryption Standard (AES) and Elliptic Curve Cryptography (ECC) techniques to provide high-entropy outputs and strong defense against cryptographic attacks while using minimal processing power, making it ideal for real-time applications in the healthcare industry. The proposed system was evaluated across key metrics, including compression ratio, processing time, energy efficiency, and implementation cost. Results demonstrate that the hardware-optimized blockchain–Internet of Things (IoT) framework significantly improves healthcare data integrity. Compression was accelerated by 98%, enabling the processing of large datasets in 35 ms. Meanwhile, the encryption model achieved outstanding performance, recording the lowest encryption time of 2.8 ms and the highest ciphertext entropy of 8.0 bits per byte. These results establish the proposed architecture as a highly viable solution for future decentralized and real-time healthcare systems, enhancing both the security and accessibility of critical patient data in resource-limited environments.

Keywords-blockchain; healthcare; Internet of Things (IoT); ARM Cortex; data compression; chaotic encryption; Zlib; patient data security

I. INTRODUCTION

The rapid expansion of healthcare information, due to the widespread use of wearable technology and remote monitoring equipment, highlights the urgent need for secure, scalable, and efficient data management in modern e-Health systems. Internet of Things (IoT)-based healthcare generates vast streams of sensitive patient information through biomedical sensors embedded in wearables, home-based equipment, and other monitoring infrastructures [1]. These real-time data streams are essential for diagnostics, chronic disease management, and personalized care. However, traditional centralized systems struggle to provide the necessary levels of security, privacy, and integrity, leaving patient records vulnerable to breaches, unauthorized access, and manipulation [2]. High-profile incidents, such as the data breach in India's CoWIN portal that exposed sensitive patient information [3], underscore these risks. The convergence of IoT with intelligent systems also plays a critical role in secure data management across sectors, including healthcare and manufacturing. Initiatives like "Made in China 2025" stress the importance of enhancing the security of data in the Manufacturing Internet of

Things (MIoT) ecosystem [4]. However, the vast amounts of sensitive data generated by IoT ecosystems remain exposed to security threats, including spoofing attacks and tampering.

Blockchain has emerged as a promising solution for decentralized healthcare data management by providing immutability, traceability, and eliminating third-party intermediaries [5]. It enables secure authentication via smart contracts and enhances privacy, yet challenges persist, especially in resource-constrained environments. Issues such as limited storage capacity, low transaction throughput, block size optimization [6], and increased storage burdens as the blockchain grows hinder real-time processing and energy efficiency. Additionally, integrating blockchain with embedded microcontrollers introduces thermal stress and heightened energy consumption [7].

Extensive research has explored blockchain's potential to enhance the security, integrity, and interoperability of healthcare systems. Recent frameworks have integrated hybrid encryption mechanisms tailored for smart healthcare, aiming to secure IoT-generated medical data and enable privacy-preserving data exchange [8]. For instance, the Blockchain-Assisted Improved Puma Edge Computing Network (BA-IPEN) utilizes the Twofish algorithm for secure, tamper-proof data storage in edge computing environments [9]. Similarly, blockchain-based Sensor-Cloud architectures have been proposed to safeguard sensitive patient data [10], whereas advanced compression schemes like Zstandard have been adopted to alleviate network congestion and reduce storage overhead [11].

Cutting-edge approaches like BlockMedCare [12] use blockchain and IoT technology to provide remote patient monitoring, especially for chronic care. These frameworks frequently use hashed blockchain storage in conjunction with proxy re-encryption methods, which allow for fine-grained access control and guarantee data integrity. They usually rely on traditional encryption techniques, which might not withstand new quantum threats like Shor's algorithm. This emphasizes the urgent need for lightweight, quantum-resilient cryptographic solutions, particularly in IoT situations with limited resources. To address this, we have developed the Chaotic Hybrid Asymmetric and Symmetric Encryption (CHASE) hybrid framework, which combines hardware-level compression, chaotic key generation, and blockchain immutability to offer a real-time, secure, and energy-efficient encryption solution designed for scalable IoT-based healthcare data storage.

Several researchers have proposed blockchain–InterPlanetary File System (IPFS) hybrid models for managing Electronic Health Records (EHRs), leveraging SHA-256 for integrity, IPFS for off-chain storage, and robust authentication mechanisms to prevent unauthorized access. The integration of blockchain, IPFS, fog computing, and the Internet of Medical Things (IoMT) further enhances decentralized, secure smart healthcare ecosystems [13]. By processing data near IoMT devices via fog nodes, these models reduce latency, alleviate network congestion, and ensure privacy and availability through immutable blockchain records and off-chain IPFS storage.

Building on these advancements, this study presents a hardware-optimized, blockchain-based healthcare data management framework tailored for embedded systems. Two major contributions define the proposed model.

- Zlib Hardware Accelerator with Adaptive Dictionary Encoding (ZHA-ADE), designed to improve compression efficiency for biomedical data on resource-limited devices.
- CHASE, combining Advanced Encryption Standard (AES), Elliptic Curve Cryptography (ECC), chaotic key generation, and Hash-Based Message Authentication Code (HMAC) to ensure robust cryptographic security with minimal overhead. CHASE is implemented on a low-power ARM Cortex-A platform with real biomedical sensors.
- Employing Numba Just-In-Time (JIT) enables the integration of a high-dimensional chaotic system with software-level acceleration, greatly lowering encryption latency while maintaining high entropy and diffusion conditions.

The proposed framework demonstrates substantial improvements in compression, encryption, energy efficiency, and blockchain scalability through a Remote Patient Monitoring (RPM) use case. This work contributes to the development of decentralized, real-time e-Health systems that are secure, energy-efficient, and suitable for low-resource environments.

II. PROPOSED METHODOLOGY

A. System Architecture

The proposed architecture presents a secure and efficient data management framework for healthcare applications by employing blockchain technology with IoT devices. As the system workflow depicts, patient health data are first generated through wearable biomedical sensors. These IoT devices transmit raw physiological signals to an edge processor, typically a low-power ARM Cortex-A microcontroller, for real-time processing. At the edge layer, the data undergo filtering to remove noise and irrelevant information. The cleaned data are then compressed using the proposed ZHA-ADE framework, which leverages dictionary-based compression and hardware acceleration to enhance storage efficiency and reduce latency. Following compression, the data are encrypted using the CHASE scheme, which integrates the AES and ECC encryption methods with chaotic key generation based on nonlinear dynamic systems. Figure 1 shows the overall system architecture using a block diagram.

The implemented model of the proposed work is shown in Figure 2(a). Patient details and sensor data are obtained through the web interface, as illustrated in Figures 2(b) and 2(d). The complete patient records are logged and presented in Figure 2(c). This hybrid encryption approach ensures high entropy, low computational cost, and strong resistance to cryptographic attacks, making it suitable for healthcare settings with limited resources.

The encrypted and compressed data are then securely transmitted to the blockchain network, where a cryptographically signed transaction is created and recorded.

Figures 3(a) and 3(b) show the compression results obtained by the proposed method for each patient data set and the corresponding transaction recorded in Blockscout, which is used to maintain scalability. Essentially, the hash is kept on-chain to ensure data integrity and inviolability, whereas the entire information is kept off-chain via the IPFS. The blockchain explorer facilitates transparent access to transaction logs and metadata, whereas authorized stakeholders can retrieve and analyze health data from IPFS. This architecture ensures data privacy, integrity, and auditability and supports real-time access and remote patient monitoring.

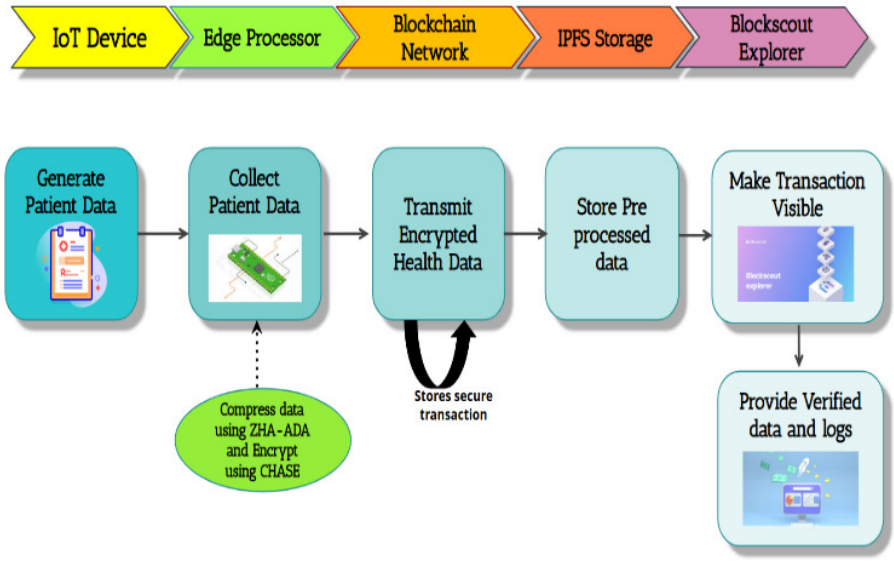

Fig. 1. Block diagram of the proposed healthcare data management framework.

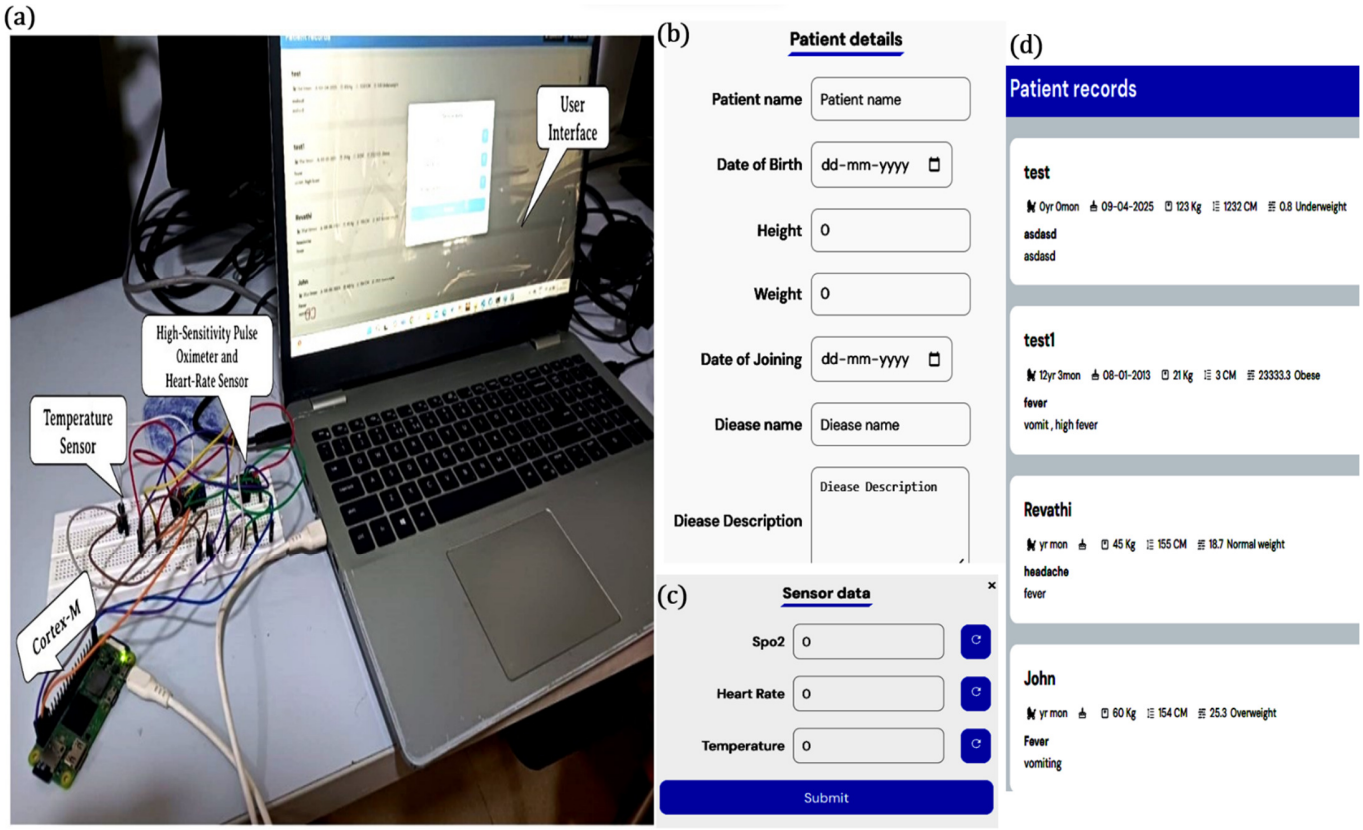

Fig. 2. (a) Implemented model of the proposed framework, (b) obtaining patient details via the web interface, (c) patient records, (d) obtaining sensor data via the web interface.

(a)

Metrics	Value
Original Size	241.67
Compressed Size	98.15

(b) **Blocks**

Ads: Pay4Bit – Accept payments from anywhere in the world. Create merchant

All | Forked | Uncles | Network utilization (last 50 blocks): 5.35% | Block countdown

Block	Size, bytes	Miner	Txn	Gas used	Reward ETH	Burnt fees ETH	Base fee
scanning new blocks...							
30 4d ago	1,277	0x00...0000	1	258,864 3.85% | -92.3%	0.00000000	0.00000744 2.79%	0.0287 Gwei
29 4d ago	1,277	0x00...0000	1	258,972 3.85% | -92.29%	0.00000000	0.00000841 3.15%	0.0325 Gwei
28 4d ago	1,277	0x00...0000	1	258,912 3.85% | -92.3%	0.00000000	0.00000951 3.54%	0.0367 Gwei
27 4d ago	1,277	0x00...0000	1	258,936 3.85% | -92.3%	0.00000000	0.00001075 3.99%	0.0415 Gwei
26 4d ago	1,277	0x00...0000	1	258,912 3.85% | -92.3%	0.00000000	0.00001215 4.48%	0.0469 Gwei

Fig. 3. (a) Compression metrics for a single block, (b) recorded transaction in Blockscout after adaptive dictionary Zlib compression.

To further validate the scalability and responsiveness of the proposed architecture under high-load conditions, a simulation was conducted using 1,000 virtual IoT nodes interacting with the blockchain backend via a secure tunneling mechanism. This emulates a realistic healthcare environment involving simultaneous transmissions from distributed patient monitoring devices. The transactions were executed using cryptographically signed payloads that simulate encrypted and compressed health records.

The simulation results yielded a total of 1,386 successfully recorded transactions, with an average latency of 4.12 s per transaction and a network throughput of 0.78 transactions per second. The complete simulation was executed in approximately 29.5 min (1772.5 s). These metrics demonstrate the system's ability to handle concurrent device transmissions and provide timely data recording to the blockchain, even under high concurrency. This validates the architectural resilience and supports its suitability for deployment in scalable, real-time healthcare infrastructures.

B. Adaptive Dictionary Zlib Compression

Zlib hardware accelerators are specialized components designed to improve the speed and efficiency of data compression and decompression, particularly for the widely used Zlib algorithm based on DEFLATE. Common in applications like PNG, Gzip, and ZIP, Zlib reduces data size by eliminating redundancies, making it ideal for storage and transmission. While effective, software-based Zlib compression can be computationally intensive, especially for large datasets, due to operations like hash table lookups and pattern matching. Hardware accelerators address these challenges by offloading compression tasks from the CPU, reducing memory usage, network bandwidth, and processing overhead [14].

Adaptive techniques during encoding adjust to changing data patterns, unlike static dictionary techniques (e.g., Lempel–Ziv–Welch (LZW) or DEFLATE), which rely on an initially created or predetermined dictionary. Text, logs, sensor data, and multimedia are a few instances of real-world datasets in which the frequency and complexity of repeating patterns change over time. Adaptive dictionary encoding performs better than static methods, as it considers these variations.

This is further explained using the following mathematical formulation. Consider $S = \{s_1, s_2 \ldots . s_n\}$ as an input data sequence of symbols (bytes, characters), and $\mathcal{D}_i$ as a dictionary at time step t for mapping phrases to shortened codes. The frequency of the windowed phrase w_i in the past data is denoted as $f(w_i)$ at the frequency threshold θ, which includes the phrases in the dictionary. A phrase w_i of length k to extract substrings starting at position i can be expressed as:

$$w_i = s_i, s_{i+1} \ldots \ldots, s_{i+k+1}, \text{for } i = 1 \; to \; n - k - 1 \quad (1)$$

These substrings are called n-grams.

- Step 1: For each w_i, the frequency over the input string S is computed as:

$$f(w_i) = \sum_{j=1}^{n-k+1} \mathbb{I}(w_j = w_i) \quad (2)$$

where $\mathbb{I}(.)$ is the indicator function:

$$\mathbb{I}(w_j = w_i) = \begin{cases} 1, & \text{if } w_j = w_i \\ 0, & \text{otherwise} \end{cases}$$

This gives a frequency distribution of all repeating patterns for a defined frequency threshold. For a frequent substring, the initial dictionary $\mathcal{D}_0$ is defined as:

$$\mathcal{D}_0 = \{w_i \mid f(w_i) \geq \theta\} \quad (3)$$

- Step 2: At each step t, the dictionary is updated based on the frequency in the recent sliding window:

$$\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{w \in W_t | f_t(w) \geq \theta\} \quad (4)$$

where W_t is the set of new phrases in the window at time t, and $f_t(w)$ is the updated local frequency of phrase w.

- Step 3: Each matching phrase $w = \mathcal{D}_t$ is replaced by its dictionary code $c(w)$, which has a shorter representation. Let us consider $L(w)$ as the original length of w and $L(c(w))$ as the length of the encoded phrase. Then the compression gain for each replacement is:

$$G(w) = L(w) - L\big(c(w)\big) \quad (5)$$

The total gain over all replacements is:

$$G_{total} = \sum_{w \, \epsilon \, \mathcal{D}_t} f_t(w).\, G(w) \quad (6)$$

- Step 4: Let the size of the data after step 1 be S_{lz77}. Then, the compressed size after adaptive dictionary encoding is:

$$S_{dict} = S_{lz77} - G_{total} \quad (7)$$

Again, it can be computed as:

$$S_{dict} = S_{lz77} \,.(1 - D_{dict}) \quad (8)$$

where $D_{dict} \in [0,1]$ represents the dictionary gain or the proportional improvement in compression due to adaptive dictionary encoding.

The corresponding pseudocode for this mathematical formulation is presented below.

```
#Pseudocode for ZHA-ADE
#Get original size
S_orig ← GetSize(S)
#Apply LZ77 compression
lz77_output ← LZ77_Compress(S)
S_lz77 ← GetSize (lz77_output)
#Construct initial dictionary
if S_lz77 > 0 {
  D_0 ← BuildInitialDictionary(lz77_output,
   0) #using (3)
  #Apply adaptive dictionary encoding
  D_t ← replacePatternsWithDictionary(lz77_
  output, D_0) #using (4)
  S_dict ← GetSize(D_t) #using (7)
  D_dict ← 1 − (S_dict / S_lz77)
}
else {
   S_dict ← S_lz77
   D_dict ← 0
}
return S_orig, S_lz77, S_dict, D_dict
```

C. Chaotic Hybrid Asymmetric and Symmetric Encryption

In IoT-enabled blockchain healthcare systems, achieving both high performance and strong security is challenging due to resource constraints. Symmetric algorithms like AES are efficient for real-time transmission but lack secure key distribution, whereas asymmetric methods like RSA and ECC ensure secure key exchange but demand significant computational power [15]. Software-based AES implementations on embedded processors consume high energy and processing time, making them unsuitable for latency-sensitive healthcare IoT applications [16, 17]. Hardware acceleration of AES via Instruction Set Architecture (ISA) extensions improves efficiency. To overcome these challenges, the CHASE scheme combines RSA for secure key exchange, AES for data encryption, chaotic maps for dynamic key updates, and HMAC for integrity, specifically tailored for securing healthcare data in blockchain systems.

The security of RSA relies on the mathematical challenge of factoring large composite numbers generated from two large primes, making private key retrieval from the public key practically infeasible [18]. RSA enables two distinct parties to securely establish a common secret key even in the absence of a pre-existing secure communication channel. The publicly accessible key can be freely shared without jeopardizing the confidentiality of either the private key or the session key it is used to protect. Furthermore, RSA supports authentication through the use of digital certificates. When a server provides the public key contained in a certificate that has been authenticated and issued by an authorized Certificate Authority (CA), the client can validate the server's identity. This process not only facilitates secure key exchange but also introduces an essential layer of trust and authentication within the communication protocol. The RSA key generation process can be described mathematically by choosing two large numbers p and q to generate the modulus:

$$n = p.q \quad (9)$$

The Euler totient function is then computed as:

$$\phi(n) = (p-1)(q-1) \quad (10)$$

Select the public key exponent e shush that it is coprime with $\phi(n)$ and satisfies $1 < e < \phi(n)$, i.e., $\gcd\big(e, \phi(n)\big) = 1$. Then, calculate the modular multiplicative inverse of e to obtain the private key d:

$$d \equiv e^{-1} \mod \phi(n) \quad (11)$$

This yields:

$$d.e \equiv 1 \mod \phi(n) \quad (12)$$

The public key is (e, n) and the private key is (d, n). Let K_{AES} denote the AES key, which is converted into an integer m. Encryption is performed as:

$$C = m^e \mod n \quad (13)$$

This ensures that only one user with the private key d can decrypt the AES key. The receiver uses private key d to retrieve the original AES key:

$$m = C^d \mod n \quad (14)$$

Data secrecy is ensured via the popular symmetric-key encryption method AES, which encrypts and decrypts data using the same key. Due to its strong resilience to cryptographic attacks, AES, which is renowned for its speed and security, is ideal for encrypting private medical information in blockchain systems. AES incorporates functions like SubBytes, ShiftRows, MixColumns, and AddRoundKey.

To enhance key security, CHASE employs the chaotic logistic map, a nonlinear iterative formula that is defined as:

$$x_{n+1} = r.x_n.(1 - x_n) \quad (15)$$

where x_n represents the value at the n^{th} iteration, typically within the range (0,1). This value can contribute to a part of the encryption key. The next iteration is x_{n+1} and r is the control parameter, which acts as a part of the secret key.

In dynamic key alteration schemes, the logistic map is employed to enhance the security of symmetric encryption by generating key components that vary with each data block. During the initialization phase, both the sender and the recipient concur upon a secret seed value x_0 and a control parameter r, which form part of the shared secret key. For each block of data to be encrypted, the logistic map is iterated a predefined number of times. The output values from specific iterations x_n are then utilized to derive dynamic components of the encryption key for that block. This derivation may involve directly using x_n , quantizing it into a binary sequence, combining multiple iterations to construct a more complex key segment, or applying transformations such as indexing a lookup table. The dynamically derived key component is subsequently used to modify the primary symmetric encryption key K_{sym}, typically through a bitwise XOR operation:

$$K'_{block} = K_{sym} \oplus derived\_key(x_n) \quad (16)$$

where K'_{block} is the modified symmetric key for the current data block and $derived\_key(x_n)$ represents the key components derived from the logistic map output. After generating the encryption key for a data block, the logistic map updates x_n to x_{n+1}, which serves as the starting point for deriving the next block's key. This iterative process ensures that each data block is encrypted with a unique, dynamically generated key, enhancing overall security.

To further protect data integrity and authenticity, HMAC integrates a cryptographic hash function with a private key to certify the sender's legitimacy and ensure that the message has not been changed.

III. EXPERIMENTAL SETUP AND RESULTS

A. Hardware and Software Description

The proposed patient-centric data storage system was implemented using an ARM Cortex-A series microcontroller, chosen for its balance of low power consumption, high performance, and real-time processing capabilities, which are essential for healthcare IoT applications. Its efficient architecture and integrated peripherals enable seamless interfacing with biomedical sensors such as the LM35 (temperature) and MAX30102 (heart rate and oxygen saturation). The microcontroller's deterministic interrupt handling ensures low-latency and real-time data acquisition in critical environments.

A lightweight Proof-of-Stake (PoS) consensus mechanism was embedded for decentralized, secure data validation and storage. Its energy-efficient processing reduces the overall energy footprint, making it suitable for long-term, battery-powered use. Wireless communication via Wi-Fi was used to enable encrypted data transmission to the blockchain. The hardware was tested in simulated scenarios for latency, throughput, energy efficiency, and reliability, confirming its effectiveness for real-time, patient-centric IoT systems.

The system was implemented on an ARM Cortex-A microcontroller, with development and simulation performed on an AMD Ryzen 2.6 GHz, 16 GB RAM host. Docker [17] was used to enable lightweight simulation of distributed blockchain environments, and Blockscout [18] provided a web-based interface for monitoring blockchain transactions, contracts, and account activity.

B. Results of Zlib Hardware Accelerator with Adaptive Dictionary Encoding

The study investigates different data compression algorithms by analyzing hardware-accelerated compression metrics. Figure 4 compares the compression performance between various text compression algorithms against data variance. The compression algorithms evaluated include standard Zlib, Gzip, Lempel–Ziv–Markov Chain Algorithm (LZMA), Zstandard, and the proposed ZHA-ADE.

The DEFLATE algorithm, implemented in Zlib, combines the LZ77 algorithm for finding and replacing duplicate strings with Huffman coding for entropy encoding [19]. Gzip is a lossless compression algorithm and file format based on the DEFLATE algorithm (the same as that used by Zlib) but adds a header, a CRC-32 checksum for error detection, and the original file length [20]. LZMA is a lossless data compression algorithm offering a high compression ratio [21]. It is the primary compression method used in the 7z archive format and employs a dictionary compression scheme. Zstandard is a relatively new lossless compression algorithm designed for high speed and good compression ratios [22].

The experimental setup enabled an effective comparison of the variability and central tendency across algorithms, demonstrating superior compression efficiency for the proposed method. Input files ranging from 100 to 500 bytes were tested, with the proposed method achieving up to 60% size reduction, compressing 100 KB files to around 0.056 KB, as shown in Figure 4, outperforming Zlib, Gzip, LZMA, and Zstandard. Statistical evaluation using the Shapiro–Wilk test showed that most algorithms lacked normal distribution ($p < 0.05$), except Zstandard. Consequently, non-parametric tests, like the Kruskal–Wallis H test, confirmed significant

differences (statistic = 44.93, $p < 0.0001$), and Mann–Whitney U tests consistently validated the proposed method's superiority ($p = 0.0001$).

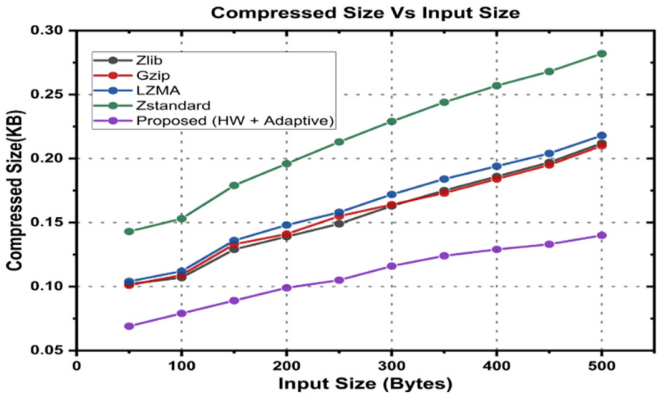

Fig. 4. Data compression analysis for all algorithms.

The compression time, depicted in Figure 5, demonstrates that the hardware-accelerated approach of the proposed ZHA-ADE performs approximately 72% faster than traditional methods. For a 500-byte input, the compression is completed in as little as 2.85 ms. In contrast, conventional algorithms such as LZMA require over 2500 ms to compress the same input size, emphasizing the significant reduction in processing latency achieved by the proposed system.

The energy consumption analysis demonstrates the most compelling results. As shown in Figure 6, the proposed method consumes only 0.0036 mJ on average, nearly 64% less than the highest-consuming standard, the Zstandard algorithm. The suggested method achieves substantial energy savings, consuming only 9 mJ at maximum input size, approximately 70% less than existing methods.

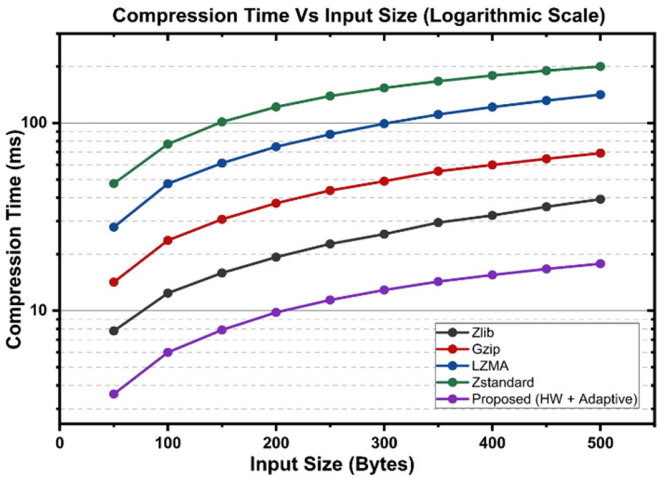

Fig. 5. Compression time depicted on a logarithmic scale for all algorithms.

Furthermore, the blockchain gas cost, shown in Figure 6, reflects the storage cost based on compressed size at 0.02 gas units per KB. The proposed method reduces gas consumption by 75%, resulting in significantly more cost-efficient data storage. The suggested method demonstrates an improvement over traditional Zlib compression by applying an aggressive compression ratio, illustrating the benefits of hardware acceleration combined with adaptive dictionary optimization techniques.

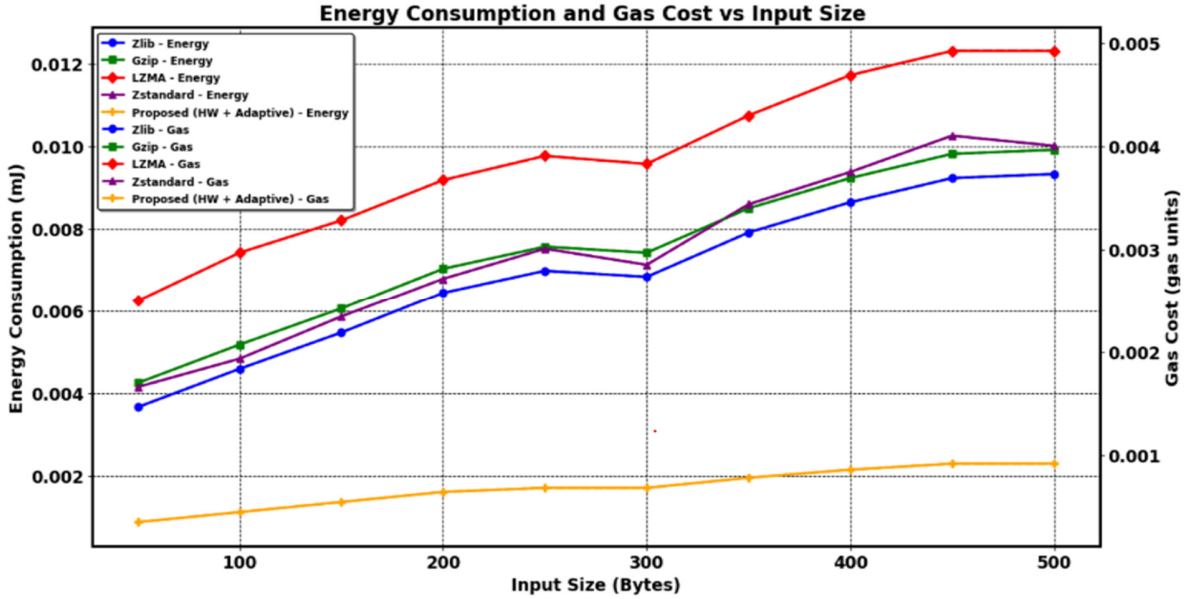

Fig. 6. Energy consumption graph and gas cost graph in terms of gas units for all algorithms.

C. Results of Chaotic Hybrid Asymmetric and Symmetric Encryption

To assess the efficiency of the proposed CHASE Hybrid encryption algorithm, its performance was compared with established standards, including ChaCha20, Blowfish, AES, and Data Encryption Standard (DES) [23]. ChaCha20, a modern stream cipher based on Addition, Rotation, XOR (ARX) operations, offers high-speed encryption with a 256-bit key and 96-bit nonce, making it ideal for platforms without AES acceleration. Blowfish, a symmetric block cipher with variable key lengths up to 448 bits, uses a 16-round Feistel structure but suffers from slow key setup [24]. AES, the widely adopted NIST standard, encrypts 128-bit blocks using keys of 128, 192, or 256 bits through multiple rounds of substitution and permutation. DES, once popular for 64-bit block encryption with a 56-bit key, is now obsolete due to its susceptibility to brute-force attacks [25].

The study examined four crucial cryptographic parameters, including encryption time, ciphertext entropy, re-keying overhead, and avalanche effect, to evaluate encryption effectiveness for real-time, IoT-based healthcare systems integrated with blockchain. As shown in Figure 7, the CHASE Hybrid encryption achieved the lowest encryption time of 2.8 ms, outperforming ChaCha20 (3.5 ms), Blowfish (4.0 ms), AES (4.8 ms), and DES (5.1 ms). This improvement is due to its lightweight architecture, combining AES with chaotic key generation, minimizing computational overhead for real-time healthcare needs.

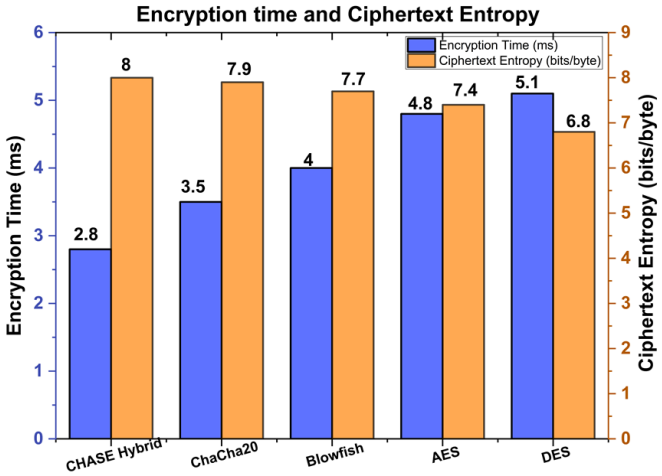

Fig. 7. Encryption time and ciphertext entropy graph for all algorithms.

Ciphertext entropy, shown in Figure 7, indicates the randomness of encrypted data, with CHASE Hybrid reaching the ideal value of 8.0 bits/byte, surpassing ChaCha20 (7.9), AES (7.7), Blowfish (7.4), and DES (6.8). This enhanced entropy results from the chaotic dynamics introduced during key generation, making the ciphertext highly unpredictable and resistant to statistical attacks.

An empirical entropy study of the resulting ciphertext was conducted in accordance with NIST SP 800-90B recommendations to further support the robustness of the CHASE Hybrid encryption. To determine the encrypted output byte stream's unpredictability and resistance to statistical inference, the Shannon entropy and min-entropy were computed, revealing that CHASE Hybrid achieved a high Shannon entropy of 6.88 bits/byte and a min-entropy of 5.49 bits/byte, outperforming the usual thresholds needed for cryptographic-quality randomness in secure applications. These results empirically confirm that the proposed encryption yields sufficiently random output, reducing the risk of entropy-based attacks and supporting its suitability for secure blockchain-integrated IoT healthcare data storage.

Furthermore, Figure 8 highlights that CHASE Hybrid achieved the lowest re-keying overhead of 1.5%, ensuring minimal computational load during frequent key update, which is essential for secure communication in IoT environments with limited resources. This is a significant improvement over ChaCha20 (2.0%), Blowfish (3.0%), AES (10.0%), and DES (15.0%). Finally, the avalanche effect, shown in Figure 8, reflects the encryption's sensitivity to input changes, with CHASE Hybrid achieving 94%, indicating strong diffusion properties. This makes even minor changes in the input produce drastically different ciphertexts, protecting against differential cryptanalysis. Together, these results demonstrate CHASE Hybrid's superiority in balancing security and efficiency for secure, real-time IoT healthcare systems.

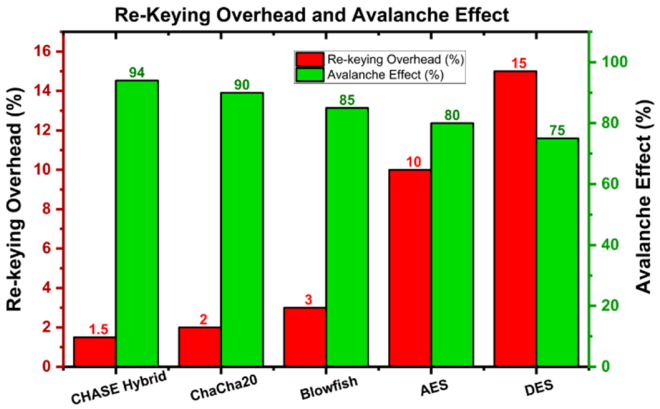

Fig. 8. Re-keying overhead graph in percentage and avalanche effect for all algorithms.

Further tests under simulated attack scenarios empirically validated the resilience of the CHASE Hybrid encryption scheme. The phase-space reconstruction plot, depicted in Figure 9(a), demonstrates a highly unpredictable trajectory, exhibiting robust resistance to phase-space reconstruction attacks and confirming the inherent non-linear, non-repetitive behavior of the underlying chaotic system. This unpredictability is essential for ensuring that encrypted data cannot be reverse-engineered. Additionally, Figure 9(b) presents the execution-time distribution obtained from timing-based side-channel analysis, which remains steady and consistent, indicating the absence of timing leakage. These findings, together with high entropy and avalanche effect measurements, support the robustness and appropriateness of CHASE Hybrid for safe implementation in real-time IoT healthcare systems.

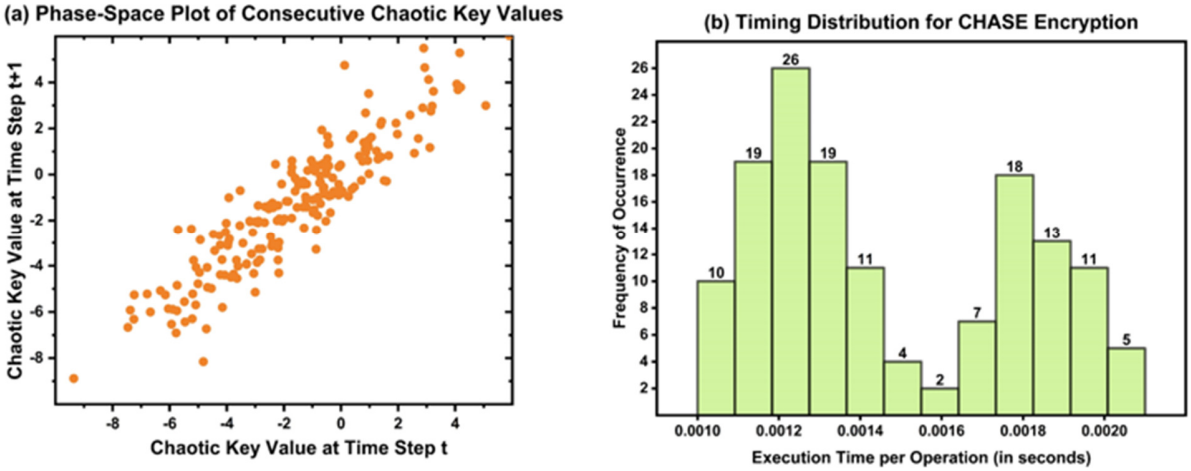

Fig. 9. (a) Phase-space reconstruction plot, (b) timing-based side-channel resistance.

IV. DISCUSSION

To substantiate the practical feasibility and performance optimization of the proposed CHASE scheme in real-time IoT-based healthcare systems, an additional experiment was conducted using a 5-dimensional input-coupled fractional-order hyperchaotic system integrated with Numba JIT compilation. To simulate a real-world medical scenario, a 256 × 256 grayscale chest X-ray picture from the publicly accessible COVID-19 Radiography Dataset was utilized [26]. To ensure high input sensitivity and uniqueness in key stream production, the system's initial conditions were dynamically constructed from the image's statistical properties, namely mean, standard deviation, and entropy.

Incorporating fractional memory dynamics increased chaotic complexity and introduced long-term reliance. Both JIT-accelerated and standard Python versions of the encryption procedure were benchmarked. The JIT-optimized version was acceptable for resource-constrained IoT edge devices, as it significantly increased execution speed by reducing runtime from 8020.14 ms to 786.23 ms (a 10.2× speedup), as depicted in Figure 10. With a significant avalanche effect (98.56%) and higher entropy (5.5353), the encrypted image demonstrated outstanding diffusion and resilience against differential attacks, demonstrating that security was not compromised.

Through the acceleration of sophisticated chaotic calculations without compromising security, this study demonstrates how JIT compilation provides real-time cryptographic performance in resource-constrained IoT healthcare systems. Along with offering a scalable and secure healthcare data architecture, it also establishes the foundation for future hardware acceleration employing FPGAs [27].

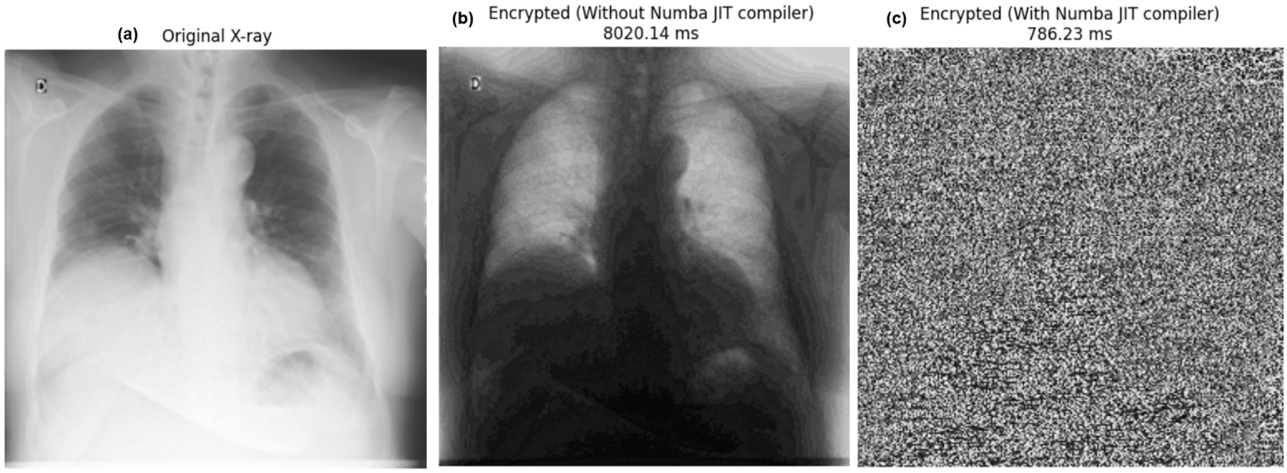

Fig. 10. (a) Original X-ray image, (b) encrypted image without Numba JIT compilation, (c) encrypted image with Numba JIT compilation.

V. CONCLUSION

This work presents a real-time, secure, and energy-efficient patient data management framework integrating Internet of Things (IoT)-based healthcare monitoring with blockchain technology. Biomedical sensor data were collected via IoT devices, processed on ARM Cortex-A microcontrollers, compressed using the Zlib Hardware Accelerator with Adaptive Dictionary Encoding (ZHA-ADE), and encrypted with using the Chaotic Hybrid Asymmetric and Symmetric

Encryption (CHASE) scheme before secure anchoring on a blockchain, with complete data stored in the InterPlanetary File System (IPFS). A complete hardware–software prototype was developed, with blockchain transactions simulated via Blockscout, and key performance metrics evaluated. Results showed ZHA-ADE improved compression efficiency by 60%, reduced processing time by 72% (2.85 ms for 500 bytes), achieved 70% energy savings, and decreased blockchain gas costs by 75%. The CHASE scheme demonstrated robust encryption performance, achieving 2.8 ms execution time, 8.0 bits/byte ciphertext entropy, 94% avalanche effect, and only 1.5% re-keying overhead, outperforming ChaCha20, Blowfish, Advanced Encryption Standard (AES), and Data Encryption Standard (DES). The integrated framework enhances scalability, reduces latency, strengthens security, and improves overall efficiency, making it suitable for practical, decentralized IoT healthcare deployments.

VI. FUTURE WORK

Future advancements to the proposed IoT–blockchain healthcare framework include building a fully hardware-accelerated encryption and compression pipeline on an FPGA to further improve latency and energy efficiency at the edge. The encryption module is being enhanced with post-quantum key exchange and adaptive re-keying for stronger security. The system is being scaled to handle multimodal medical data by integrating Artificial Intelligence (AI)-based anomaly detection at the edge. In our ongoing work, we are integrating synthetic and anonymized large-scale medical datasets, including high-resolution imaging and structured multi-gigabyte EHR records, to comprehensively validate the performance of our pipeline using metrics such as compression throughput per GB, encryption key regeneration overhead, and blockchain anchoring behavior for real-time archival on the IPFS. Additionally, deployment on a permissioned blockchain testbed using real hospital datasets is underway to assess scalability, interoperability, and compliance with healthcare standards.

REFERENCES

[1] S. Dhingra, R. Raut, K. Naik, and K. Muduli, "Blockchain Technology Applications in Healthcare Supply Chains—A Review," *IEEE Access*, vol. 12, pp. 11230–11257, 2024, https://doi.org/10.1109/ACCESS.2023.3348813.

[2] R. Ramani, A. Rosline Mary, S. Edwin Raja, and D. Arun Shunmugam, "Optimized data management and secured federated learning in the Internet of Medical Things (IoMT) with blockchain technology," *Biomedical Signal Processing and Control*, vol. 93, Jul. 2024, Art. no. 106213, https://doi.org/10.1016/j.bspc.2024.106213.

[3] "India health data faces rising risk of breaches, cyberattacks." ETHealthworld.com. https://health.economictimes.indiatimes.com/news/industry/india-health-data-faces-rising-risk-of-breaches-cyberattacks/102068648.

[4] J. Tan, J. Shi, J. Wan, H.-N. Dai, J. Jin, and R. Zhang, "Blockchain-Based Data Security and Sharing for Resource-Constrained Devices in Manufacturing IoT," *IEEE Internet of Things Journal*, vol. 11, no. 15, pp. 25558–25567, Aug. 2024, https://doi.org/10.1109/JIOT.2024.3363013.

[5] S. Guan, Y. Cao, and Y. Zhang, "Blockchain-Enhanced Data Privacy Preservation and Secure Sharing Scheme for Healthcare IoT," *IEEE Internet of Things Journal*, vol. 12, no. 5, pp. 5600–5614, Mar. 2025, https://doi.org/10.1109/JIOT.2024.3487154.

[6] T. A. Alghamdi, R. Khalid, and N. Javaid, "A Survey of Blockchain Based Systems: Scalability Issues and Solutions, Applications and Future Challenges," *IEEE Access*, vol. 12, pp. 79626–79651, 2024, https://doi.org/10.1109/ACCESS.2024.3408868.

[7] O. Popoola, M. A. Rodrigues, J. Marchang, A. Shenfield, A. Ikpehai, and J. Popoola, "An optimized hybrid encryption framework for smart home healthcare: Ensuring data confidentiality and security," *Internet of Things*, vol. 27, Oct. 2024, Art. no. 101314, https://doi.org/10.1016/j.iot.2024.101314.

[8] G. Sarojini Karuppusamy and M. K. S, "TwoFish-Integrated Blockchain for Secure and Optimized Healthcare Data Processing in IoT-Edge-Cloud System," *Transactions on Emerging Telecommunications Technologies*, vol. 36, no. 3, Mar. 2025, Art. no. e70076, https://doi.org/10.1002/ett.70076.

[9] I. Masood, A. Daud, Y. Wang, A. Banjar, and R. Alharbey, "A blockchain-based system for patient data privacy and security," *Multimedia Tools and Applications*, vol. 83, no. 21, pp. 60443–60467, Jun. 2024, https://doi.org/10.1007/s11042-023-17941-y.

[10] B. Halak, Y. Yilmaz, and D. Shiu, "Comparative Analysis of Energy Costs of Asymmetric vs Symmetric Encryption-Based Security Applications," *IEEE Access*, vol. 10, pp. 76707–76719, 2022, https://doi.org/10.1109/ACCESS.2022.3192970.

[11] S. R. Mallick, R. K. Lenka, P. K. Tripathy, D. C. Rao, S. Sharma, and N. K. Ray, "A Lightweight, Secure, and Scalable Blockchain-Fog-IoMT Healthcare Framework with IPFS Data Storage for Healthcare 4.0," *SN Computer Science*, vol. 5, no. 1, Jan. 2024, Art. no. 198, https://doi.org/10.1007/s42979-023-02511-8.

[12] R. Gao, Z. Li, G. Tan, and X. Li, "BeeZip: Towards An Organized and Scalable Architecture for Data Compression," in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3*, La Jolla, CA, USA, 2024, pp. 133–148, https://doi.org/10.1145/3620666.3651323.

[13] Y. Li, Q. Wang, and S. Yu, "A novel hybrid scheme for chaotic image encryption," *Physica Scripta*, vol. 99, no. 4, Mar. 2024, Art. no. 045244, https://doi.org/10.1088/1402-4896/ad3171.

[14] T. M. Ignatius, T. Birjit Singha, and R. Paily Palathinkal, "Power Side-Channel Attacks on Crypto-Core Based on RISC-V ISA for High-Security Applications," *IEEE Access*, vol. 12, pp. 150230–150248, 2024, https://doi.org/10.1109/ACCESS.2024.3477961.

[15] M. A. Caraveo-Cacep, R. Vázquez-Medina, and A. Hernández Zavala, "A review on security implementations in soft-processors for IoT applications," *Computers & Security*, vol. 139, Apr. 2024, Art. no. 103677, https://doi.org/10.1016/j.cose.2023.103677.

[16] R. Ifrim, D. Loghin, and D. Popescu, "A Systematic Review of Fast, Scalable, and Efficient Hardware Implementations of Elliptic Curve Cryptography for Blockchain," *ACM Transactions on Reconfigurable Technology and Systems*, vol. 17, no. 4, Nov. 2024, Art. no. 62, https://doi.org/10.1145/3696422.

[17] "Docker Reference Documentation." Dockerdocs. https://docs.docker.com/reference/.

[18] "Blockscout Docs." Blockscout. https://docs.blockscout.com.

[19] C. Marcon, A. S. Mete, P. V. Gemmeren, and L. Carminati, "Optimizing ATLAS data storage: The impact of compression algorithms on ATLAS physics analysis data formats," *EPJ Web of Conferences*, vol. 295, May 2024, Art. no. 03027, https://doi.org/10.1051/epjconf/202429503027.

[20] M. Hema and S. P. Shyry, "Efficient Compression of Multimedia Data using Lempel–Ziv–Markov Chain Adaptive Block Compressive Sensing (LZMC-ABCS)," *Wireless Personal Communications*, May 2024, https://doi.org/10.1007/s11277-024-11187-z.

[21] F. Novanto, A. Nugraha, J. C. Kurniawan, and A. I. Prayogo, "Optimizing Digital Image Steganography through Hybridization of LSB and Zstandard Compression," *Sinkron : jurnal dan penelitian teknik informatika*, vol. 8, no. 1, pp. 75–82, Jan. 2024, https://doi.org/10.33395/sinkron.v9i1.13187.

[22] A. Soboń and S. Stachowiak, "ChaCha20 Cipher Cryptanalysis through SAT Problem Solving," in *2024 IEEE 17th International Scientific Conference on Informatics*, Poprad, Slovakia, 2024, pp. 355–361, https://doi.org/10.1109/Informatics62280.2024.10900867.

[23] A. Gupta, S. Namasudra, and P. Kumar, "A secure VM live migration technique in a cloud computing environment using blowfish and blockchain technology," *The Journal of Supercomputing*, vol. 80, no. 19, pp. 27370–27393, Dec. 2024, https://doi.org/10.1007/s11227-024-06461-7.

[24] Z. A. Mohammed, H. Q. Gheni, Z. J. Hussein, and A. K. M. Al-Qurabat, "Advancing Cloud Image Security via AES Algorithm Enhancement Techniques," *Engineering, Technology & Applied Science Research*, vol. 14, no. 1, pp. 12694–12701, Feb. 2024, https://doi.org/10.48084/etasr.6601.

[25] A. K. Singh, T. K. Jain, P. Pandey, and L. Rzayeva, "LVCMOS Based Low Power Implementation of DES Encryption Algorithm on 28nm FPGA," in *2024 3rd International conference on Power Electronics and IoT Applications in Renewable Energy and its Control*, Mathura, India, 2024, pp. 383–386, https://doi.org/10.1109/PARC59193.2024.10486400.

[26] "COVID-19 Radiography Database." Kaggle. [Online]. Available: https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database.

[27] K. Zourmba *et al.*, "Fractional order 1D memristive time-delay chaotic system with application to image encryption and FPGA implementation," *Mathematics and Computers in Simulation*, vol. 227, pp. 58–84, Jan. 2025, https://doi.org/10.1016/j.matcom.2024.07.035.