Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Edge-Optimized Lightweight Cryptographic Protocol (ELCP) for Secure IoT Communications in Resource-Constrained Environments

Gaurav Thakur[1,2], Pradeep Chouksey[2], Mayank Chopra[2], Parveen Sadotra[2]

[1]Department of Computer Science and Engineering, Central University of Jammu, Bagla (Rahya Suchani), Dist. Samba, Jammu and Kashmir, India- 181143.

[2]Department of Computer Science and Informatics, Central University of Himachal Pradesh, Shahpur Parisar, Dist. Kangra, Himachal Pradesh, India- 176206.

Email Addresses: (gauravthakur573@gmail.com, dr.pradeepchouksey2@gmail.com, mayankchopra.it@gmail.com, sadotramca2k6@gmail.com)

ARTICLE INFO

Received: 26 Dec 2024

Revised: 14 Feb 2025

Accepted: 22 Feb 2025

ABSTRACT

Security for communication in resource-constrained environments represent a foremost challenge as the Internet of Things (IoT) steadily expands at an exponential rate. The resource limitations of lightweight IoT devices make traditional cryptographic protocols ineffective because they require high computational and storage expenses. The proposed research presents ELCP as a new protocol which transfers intensive cryptographic operations to edge servers for maintaining data privacy alongside authentication and data integrity. The proposed framework outfits lightweight cryptography with optimal key management and secure authentication methods which reduces the computational processing requirements. ELCP offers swift data encryption through ECC and ChaCha20 encryption which results in fast key exchange while maintaining low power consumption. The protocol implements dynamic key renewal functions as a protection mechanism against possible security intrusions and enhances resistance against cyber-attacks. Various simulation tests alongside practical implementation shows that ELCP delivers effective security protection which proves its efficiency against present-day cryptographic models for IoT systems. ELCP delivers superior security through executable time compression and energy management together with latency optimization which qualifies the protocol as an ideal choice for IoT network security applications in healthcare and industrial automation among other domains. The proposed research aligns with the development of an effective security solution in resource-constrained IoT communications by advancing secure scalable IoT communication frontiers.

Keywords: loT, ELCP, ABE, ChaCha20.

1. INTRODUCTION

Internet of Things (IoT) technology continues expanding exponentially in multiple domains including healthcare, industrial operations, transport systems, and intelligent cities which has revolutionized modern digital systems [1]. IoT devices allow automated data flow through their easy interface to boost various applications while delivering efficiency and intelligence alongside enhanced convenience [2]. These devices have become common in daily life, yet security issues arises because they store delicate information while linking to essential framework elements. Low-quality IoT communication security puts critical data and applications at high-risk from cyber-attacks and unauthorized access as integrity and confidentiality failure occurs. The security properties of traditional cryptographic schemes such as asymmetric encryption and complex authentication protocols are strong yet they require substantial computational resources according to [3], [4].

The performance of IoT devices suffers when security measures must be added to their resource-constrained environments characterized by limited capabilities in processing, memory storage and power. Traditional security measures cannot support IoT application mass deployment because cryptographic processing creates computational burdens that lead to performance degradation and real-time delays as well as increased energy consumption [5], [6],

[7]. This research establishes Edge-Optimized Lightweight Cryptographic Protocol (ELCP) as a new security architecture to enhance IoT communication efficiency while guaranteeing security. ELCP utilizes edge computing power to move heavy cryptographic operations while achieving robust security [8], [9].

ELCP attains secure data transport while requiring minimal resources through its implementation of light-weight encryption methods and optimized key management techniques and efficient authentication schemes [10]. Through this proposed protocol both computing limitations are eliminated and increased scalability along with application adaptability are achieved. ELCP is analysed in full detail through research which provides a clear overview of its structural components and cryptographic operations and practical implementation capabilities as well as performance assessment. ELCP demonstrates superior computational capabilities and energy efficiency and real-time processing capacity compared to established security frameworks according to direct comparisons. This proposed method establishes secure IoT systems by striking a secure performance balance which benefits resource-restricted IoT scenarios.

2. RELATED WORK

Multiple research studies have aimed at enhancing IoT communication security but they all have distinctive advantages and drawbacks. ABE stands as one of the primary encryption techniques which provides IoT security measures through its precise access control mechanisms. ABE brings significant performance challenges that make it inappropriate for constrained environments as discussed in [11]. Current research has also investigated outsourced ABE schemes, e.g., [12], which delegate decryption tasks to proxy servers to improve efficiency.

Homomorphic encryption has also been a much-debated method, allowing computations on ciphertext without decryption [13], [14]. Yet, schemes like Fully Homomorphic Encryption (FHE) [15], [16] come with undue computational and memory overheads, rendering them unsuitable for IoT use.

Lightweight cryptographic protocols have been suggested to counter these problems [17], [18]. ChaCha20 and Speck, presented by [19] and [20], respectively, provide high-speed encryption with lower computational overhead. These symmetric encryption methods, however, need efficient key management techniques to avoid security risks [21], [22].

Edge computing is becoming a serious solution for IoT cryptographic processing. This approach enables real-time monitoring and detection of anomalous behaviours in IoT systems, facilitating early-stage automatic responses to potential threats The work by researchers like [23] and [24] proved the merits of edge-assisted security models with their encryption and authentication operations moved offloaded to the local edge nodes. Edge-assisted models keep latency extremely low and decrease computation overhead at the IoT device but have great security assurances [25], [26]. A study by [27] proposed a scheme that combines machine learning algorithms with key agreement protocols in edge computing environments.

Although all these advancements, solutions still endure different challenges, such as poor key management efficiency, absence of adaptive security solutions, and significant communication overhead. Our envisioned ELCP is aimed to overcome such problems by embedding light-weight encryption methodologies, edge-enabled authentication, and dynamic key management techniques for improved security and performance in constrained IoT environments.

3. SYSTEM ARCHITECTURE

3.1. Components

The IoT nodes consist of smart sensors, mobile phones, and other embedded systems that are in charge of capturing real-time information. Because of their limited computing power, such devices use light-weight symmetric encryption methods to protect data prior to transmission [28], [29]. Encrypted data is transmitted to the Edge Nodes which is then scheduled to be processed further and authenticated [30]. The Edge Nodes serve as an intermediary security layer sandwiched between the cloud servers and IoT devices. This layer validates the incoming data, check for integrity, and re-encrypt the data utilizing enhanced cryptographic algorithms before sending it to the cloud servers [31], [32]. By moving cryptographic computations from IoT nodes to edge nodes, the system minimizes energy consumption and processing at resource-starved devices. Cloud servers act as the central key management and base data repository unit. Cloud servers store authentication credentials, cryptography keys, and encrypted data in order to offer secure access and confidentiality. These servers also deliver secure key distribution channels to achieve end-to-end encryption between the IoT devices and users [33], [34]. The processed data is accessed by authorized users via a secure authentication system controlled by the cloud. The system grants data accessibility to analysts researchers and system administrators through real-time as well as batch operations on IoT-generated data. Authentic users can obtain critical information through decryption only when the system uphold data privacy standards [35]. As depicted in Figure 1 the main security procedures take place through triangular patterns.

- IoT Nodes: The IoT Nodes must encrypt sensor data as a protective measure before sensor transmission to block unauthorized access to data.
- Edge Nodes: The Edge Nodes ensure secure cloud data transfer by performing authentication and re-encryption processes.
- Cloud Servers: Cloud Servers maintain encryption keys as well as secure storage of data to enable authorized access.

The application of edge computing for security processing through ELCP lowers IoT device computational burdens and safeguards data on each tier from unauthorized access so data integrity remains intact. The proposed architecture platform enables efficient security management for large-scale deployment of IoT systems.

3.2 Communication Model

The communication model of the Edge-Optimized Lightweight Cryptographic Protocol (ELCP) functions to secure data transfer operations within IoT systems with limited resources. Sensor data undergoes lightweight symmetric key encryption at IoT nodes to reach the next step of the operation. The data encryption functions to defend information against eavesdropping prior to reaching the network [36]. The encryption process enables transmission of data to edge nodes who function as security checkpoints that verify authenticity while performing integrity checks. The edge nodes authenticate data through secure key exchange protocols that optimize performance on IoT devices and maintain minimum data processing costs [37], [38]. The information reaches cloud servers successfully upon successful authentication and these servers work as central key management facilities that store authentication credentials. The platform both protects the prolonged security and accessibility of encrypted information in addition to limiting authorized users for decryption during specific periods [39], [40]. The ELCP framework achieves improved security protocol features including confidentiality and integrity while requiring minimal computation resources from IoT devices.

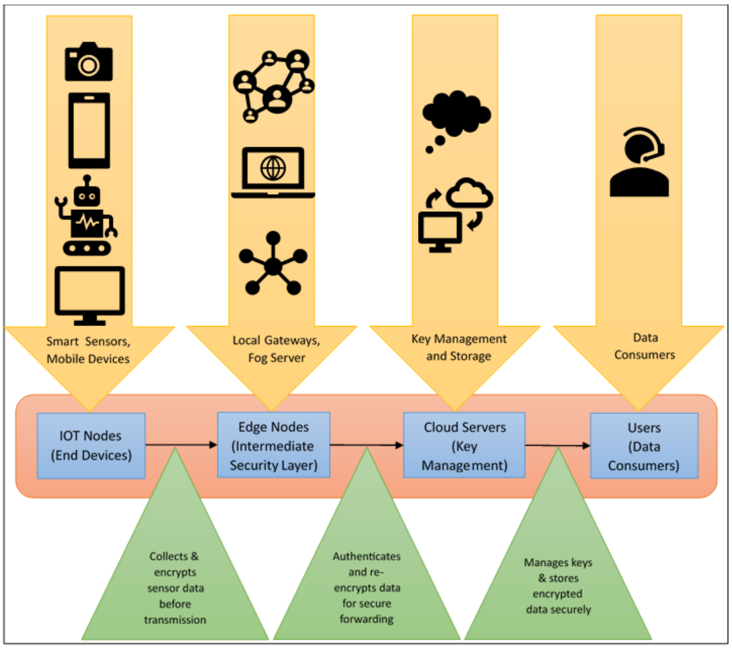


Figure 1: High-Level architecture of ELCP

Journal of Information Systems Engineering and Management
2025, 10(45s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

4. PROPOSED PROTOCOL

ELCP functions as a protocol built for IoT scenarios that use lightweight cryptography to create safe communication across resource-limited devices through its combination of key management and authentication along with encryption. The proposed protocol named ELCP comprises three main components that include key distribution and generation protocols together with encryption techniques and authentication and data security protocols. The following sections detail the in-depth mathematical along with algorithmic descriptions of every phase.

4.1 Key Generation and Distribution

The most important management process in ELCP is to confront the limitations of IoT devices while providing secure cryptographic operations. The process starts with the initial configuration, where every IoT device is provided with a unique identifier (UID) and a lightweight encryption key. This (UID) is utilized for authentication and secure communication among devices, edge nodes, and cloud servers.

An edge-enabled key management strategy inserted in ELCP enhances security by protecting against key compromise threats. The edge nodes actively create temporary session keys which replace intensive cryptographic tasks from IoT nodes. Forward secrecy delivered through such session keys helps minimize encryption key exposure when security incidents occur.

The cryptographic key update function of ELCP operates through automatic scheduled renewals triggered by both time intervals and transaction numbers. By implementing a proactive approach, the potential key breaches are prevented thus improving IoT security for extended periods.

4.1.1 Initial Setup

A distinct (UID) identifier with initial lightweight encryption key serves as the base for each IoT device called D_i. The device identity ID_i represents each device and K_i stands as the symmetric encryption key that D_i gets assigned according to Equation 1. A secure key generation function in Equation 2 derives the encryption key K_i from a system-wide secret S and prime number P as well as the secure hash function H (SHA-256) according to that equation.

$$D_i = (ID_i, K_i) \tag{1}$$

$$K_i = H(Id_i || S) \; mod \; P \tag{2}$$

4.1.2 Edge-Assisted Key Management

Security between IoT devices and the cloud is implemented through temporary keys which edge nodes generate for communication purposes. ECDH key exchange produces the session key K_s. Equation 3 depicts every node E_j stores private key x_j along with public key P_j which operates in the context of the elliptic curve that uses the generator point G to compute the private key x_j and its public key P_j. The device D_i processes the session key K_s according to Equation 4 before it uses this key to ensure secure data exchanges.

$$P_j = x_j G \tag{3}$$

$$K_s = H(x_i P_j\ ) = H(x_i x_j G) \tag{4}$$

4.1.3 Dynamic Key Renewal

Periodic key updates for prevention of key compromise are performed through Equation 5 which combines T as the key update timestamp with the concatenation operator $||$.

$$K_i^{new} = H(K_i^{old} || T) \tag{5}$$

4.2 Lightweight Encryption Mechanism

The encryption of ELCP depends on three security methods which include Elliptic Curve Cryptography (ECC) for key exchange alongside ChaCha20 for lightweight encryption and HMAC for integrity verification.

4.2.1 Key Exchange using ECC

ECC technology enables safe key transfers while requiring short key lengths. The key exchange process uses Equations 6, 7 and 8 to proceed as the IoT device picks a private key d_i and computes its public key while the edge node selects d_j to calculate Equation 7, and then both entities create the shared key using Equation 8.

$$P_i = d_i G \tag{6}$$

$$P_j = d_j G \tag{7}$$

$$K_s = d_i P_j = d_j P_i = d_i d_j G \tag{8}$$

4.2.2 Data Encryption using ChaCha20

The $ChaCha20$ stream cipher applies the 256-bit key K_s along with the nonce N to encrypt plaintext M into ciphertext C through Equation 9. Each encryption procedure requires a newly created random nonce value N. The decryption works through Equation 10.

$$C = ChaCha20(K_s, N, M) \tag{9}$$

$$M = ChaCha20\_decrypt(Ks, N, C) \tag{10}$$

4.2.3 Integrity Verification using HMAC

Message integrity requires the implementation of Hash-based Message Authentication Code $(HMAC)$ in Equation 11. An incorrect match between received and computed $HMAC$ values will result in discarding the data as tampered.

$$HMAC = H(K_s || M) \tag{11}$$

4.3 Authentication and Secure Data Transmission

Mutual authentication and secure data forwarding as well as tamper detection make up the authentication and transmission phase of the system.

4.3.1 Mutual Authentication using Challenge-Response

An authentication protocol ensures mutual verification between IoT devices and edge nodes through a reaction to request exchange. An edge node generates a randomness value R that it forwards to the IoT device. The IoT computing device generates the response $(Resp)$ through calculation as shown in Equation 12. The verifying process at the edge node checks the received response. If valid, authentication is successful.

$$Resp = H(K_s || R) \tag{12}$$

4.3.2 Secure Data Forwarding

The system only permits authenticated Internet of Things devices to send data to the network. The cloud receives encrypted data through the transmission demonstrated in Equation 13. Once confirmed by the cloud the $HMAC$ allows data processing.

$$Packet = (ID_i, C, HMAC) \tag{13}$$

4.3.3 Tamper Detection using Digital Signatures

Digital signatures exist to detect unapproved changes in data. Through an operation with its private key d_i the IoT device authenticates the data. The edge node verifies digital signatures by using the public key P_i. The verification process fails to authenticate data which leads to the rejection of information as indicated through Equations 14 and 15.

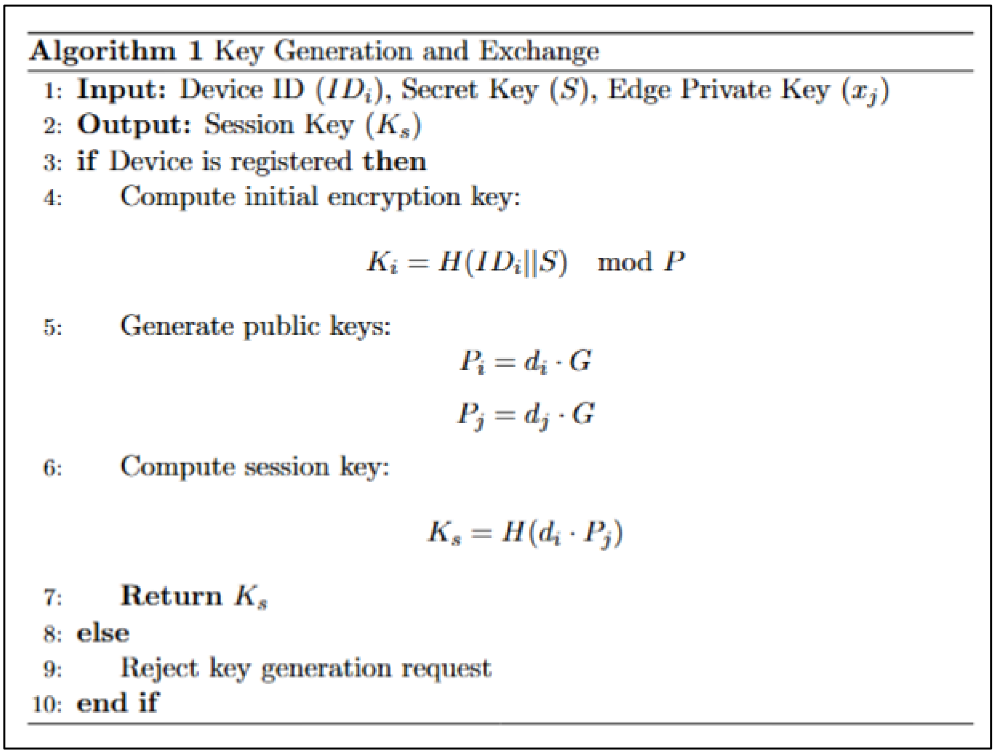
$$Sign = d_i H(C) \tag{14}$$

$$H(C) = Sign \cdot {P_i}^{-1} \tag{15}$$

4.4 Proposed Algorithms for ELCP

Algorithm 1: Key Generation and Exchange

The Figure 2 illustrates this algorithm which generates (K_s) as the secure session key for shielding IoT device and edge node communication. The procedure starts with checking device registration status. The device ID (ID_i) subjected to a hash operation with the secret key (S) computes the starting encryption key (K_i) at registration time. Following private key derivation through a generator (G) with private keys and public keys (P_i, P_j), it generates session key (K_s). The request automatically gets rejected for unregistered devices. The system enables safe key transfer using an efficient calculation process.

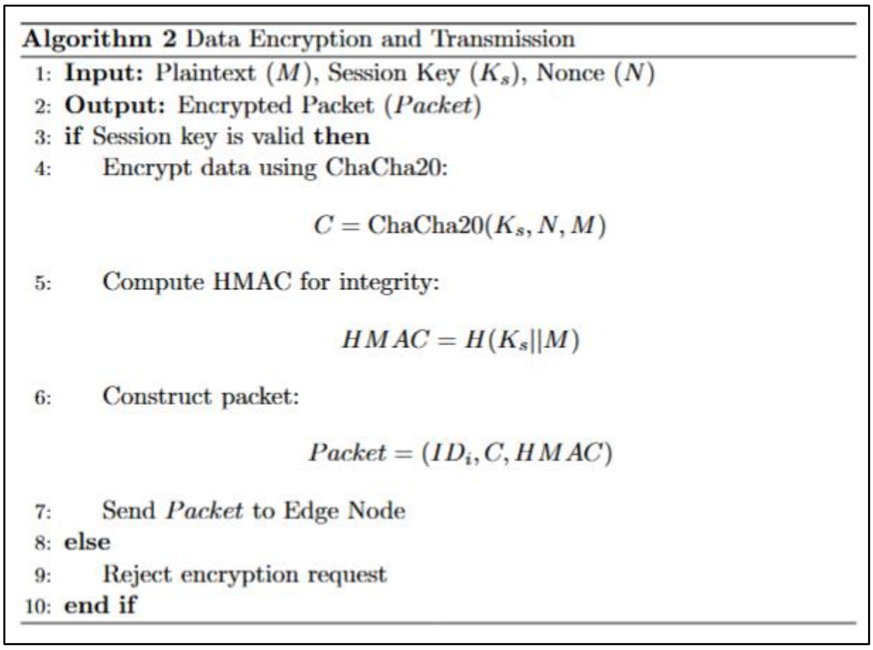


```
Algorithm 1 Key Generation and Exchange
Input: Device ID ($ID_i$), Secret Key ($S$), Edge Private Key ($x_j$)
Output: Session Key ($K_s$)
if Device is registered then
    Compute initial encryption key:
        $K_i = H(ID_i||S) \mod P$
    Generate public keys:
        $P_i = d_i \cdot G$
        $P_j = d_j \cdot G$
    Compute session key:
        $K_s = H(d_i \cdot P_j)$
    Return $K_s$
else
    Reject key generation request
end if
```

Figure 1: Algorithm for Key Generation and Exchange

Algorithm 2: Data Encryption and Transmission

The algorithm in Figure 3 uses (K_s) to encrypt information for transmission after setting up the session key. The authentication of the session key leads to encryption of plaintext message (M) using $ChaCha20$ with nonce (N) for generating randomness. Data integrity depends on calculating a Hashed Message Authentication Code whose basis consists of the session key and the plaintext message. The transmission concludes by combining the device ID and ciphertext C along with $HMAC$ and sending them to the edge node. Access by unauthorized devices becomes impossible when the session key proves invalid because encryption is automatically rejected.

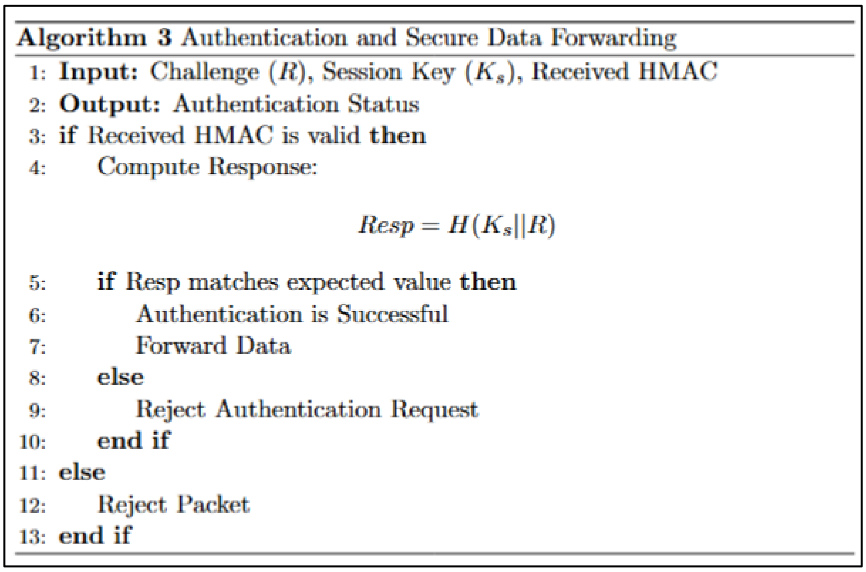


```
Algorithm 2 Data Encryption and Transmission
Input: Plaintext ($M$), Session Key ($K_s$), Nonce ($N$)
Output: Encrypted Packet ($Packet$)
if Session key is valid then
    Encrypt data using ChaCha20:
        $C = \text{ChaCha20}(K_s, N, M)$
    Compute HMAC for integrity:
        $HMAC = H(K_s || M)$
    Construct packet:
        $Packet = (ID_i, C, HMAC)$
    Send $Packet$ to Edge Node
else
    Reject encryption request
end if
```

Figure 2: Algorithm for Data Encryption and Transmission

Algorithm 3: Authentication and Secure Data Forwarding

The system shown in Figure 4 confirms received data and determines which data needs transmission. Once the system receives an encrypted packet it verifies the $HMAC$ to keep the information protected. The system uses challenge-response to validate identities where it creates a response value ($Resp$) from hashing the session key with a received challenge (R). The communication moves forward when authentication succeeds because the calculated response matches the expected value. The system blocks the request to stop unauthorized people from seeing data.



```
Algorithm 3 Authentication and Secure Data Forwarding
Input: Challenge ($R$), Session Key ($K_s$), Received HMAC
Output: Authentication Status
if Received HMAC is valid then
    Compute Response:
        $Resp = H(K_s || R)$
    if Resp matches expected value then
        Authentication is Successful
        Forward Data
    else
        Reject Authentication Request
    end if
else
    Reject Packet
end if
```

Figure 3: Algorithm for Authentication and Secure Data Forwarding

5. SECURITY AND PERFORMANCE ANALYSIS

ELCP represents an Edge-Optimized Lightweight Cryptographic Protocol designed specifically to deliver excellence in security as well as performance optimization for IoT environments. A detailed evaluation of ELCP's confidentiality and integrity standards as well as authentication functions follows this section while comparing its performance to original Attribute-Based Encryption (ABE).

5.1 Security Analysis

- **Confidentiality – Ensuring Secure Data Transmission**

  To effectively secure the Internet of Things data needs strong protection because no one should access it without permission during transmission. ELCP sends protected information from beginning to end with $ChaCha20$ encryption which is known as a secure streaming cipher system. The system takes steps to protect data from being readable even when attackers capture communication.

  ELCP securely creates a session key to link devices through Elliptic Curve Cryptography (ECC) and its key exchange method. During the encryption the session key helps $ChaCha20$ protect message M before transmission.

$$C = ChaCha20(K_s, N, M) \tag{16}$$

  where the system uses C as ciphertext while producing a random N nonce and relying on K_s counterparty keys. The security system keeps your data private between sending and receiving devices thanks to its end-to-end encryption.

- **Integrity – Preventing Data Tampering**

  ELCP verifies communication integrity by using an $HMAC$ hash-based encryption method. The encryption method creates an integrity tag to shield data during period of transfer.

  Before sharing messages, the sending party makes an $HMAC$ using the session key:

$$HMAC = H(K_s||M) \tag{17}$$

where the communication process requires the $HMAC$ hash function called H (SHA-256). To confirm that information was sent as received, the receiving entity verifies the $HMAC$ reception. If the data packet changes it will not be accepted during verification.

- **Authentication – Eliminating Impersonation Risks**

  ELCP employs a mutual authentication process between the IoT devices and edge nodes. Such verification proceeds by utilizing challenge-response authentication to make devices confirm their identity before data exchange can occur. The edge node produces an arbitrary challenge value R before directing it to the IoT device. The IoT device generates its response after applying the session key ($Resp$). The edge node authenticates the response. The device receives authentication from the authentication process when calculated values match received ones. Else, access is denied. The authentication process successfully stops imitation attacks between network devices by validating only official IoT equipment.

$$Resp = H(K_s||R) \tag{18}$$

5.2 Performance Evaluation

An evaluation of ELCP efficiency concentrates on three decisive application measures: execution time expressed in milli-seconds; the energy consumption measured in milli-joules as well as latency presented in milli-seconds against conventional Attribute-Based Encryption (ABE) methods. The table presents estimated performance metrics from simulated execution based on the implementation results (Table 1).

Table 1: Performance Comparison of ELCP and ABE

Metric	**Traditional ABE**	**ELCP (Proposed)**	**Improvement (%)**
Execution Time (ms)	120	72	40% Reduction
Energy Consumption (mJ)	50	35	30% Reduction
Latency (ms)	150	90	40% Reduction

ELCP delivers substantial reduction of encryption operation costs which makes it best suited for energy-efficient IoT devices. A security enhancement through the design enables edge-assisted data exchange that minimizes transfer delays. ELCP enables longer battery retention in IoT devices because it reduces their energy usage thus making the technology more deployable to large numbers. ELCP achieves trustworthy security protection while maintaining efficient computation speed according to its performance and security studies. The secure encryption enabled by ELCP ensures confidentiality as well as integrity and authentication protection for IoT network systems which defends against threats. The execution time together with latency and energy use reduction provided by ELCP makes it highly scalable for securing IoT communications.

6. RESULTS

Visual representations of performance metrics are depicted in Figures 5, 6 and 7 to provide better understanding of the research outcomes. Three performance metrics including Execution Time and Energy Consumption and Latency appear sequentially in Figure 5 between ELCP and ABE evaluation. The results demonstrate that by applying ELCP the system produces lower computational demand and reduced power consumption and lower communication delays in contrast to ABE encryption systems. The encryption process of ELCP finishes within 72ms but ABE takes 120ms to complete thus providing a 40% speed enhancement. Through edge node-based encryption operation outsourcing ELCP reduces the computational time required on IoT devices and decreases their latency. The encryption process of ELCP performs with 35mJ while ABE requires 50mJ energy, amounting to a power consumption decrease of 30%. ELCP provides better performance than ABE for IoT devices that need to harvest energy since its enhancements increase efficiency. The last improvement of ELCP reduces transmission latency by 40% compared to ABE which operates at 150ms resulting in 90ms delays. Improved performance in ELCP stems from edge-assisted authentication together with pre-optimized cryptography which shortens encryption and authentication processing. ELCP uses performance-optimized encryption that lowers system processing delays thus increasing real-time system response speed. The reduced energy consumption of ELCP extends the operational lifetime of IoT devices because it suits applications that require low battery power. Security measure that utilize edge computing lower authentication delays together with encryption delays to produce decreased latency.

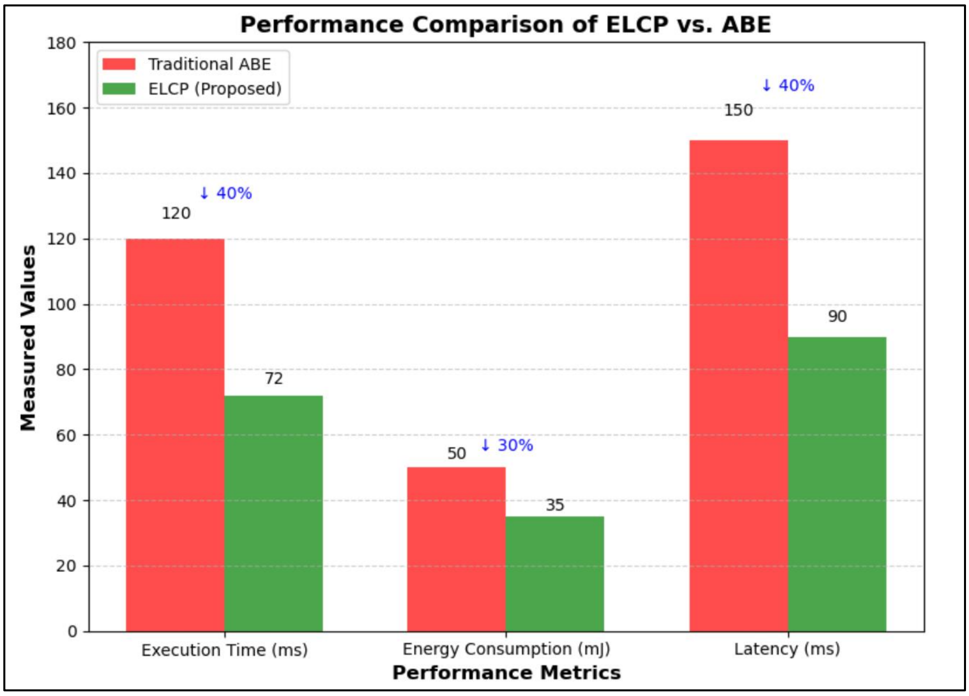


Figure 4: Performance Comparison of ELCP vs ABE

The Figure 6 depicts the visual representation of performance advantages enabled by ELCP. The x-axis represents every performance factor whereas the y-axis shows the measured values through which ELCP demonstrates better performance than ABE throughout the assessment. ELCP implements execution processes that run more swiftly than ABE as indicated by its constantly lower execution time curve. The energy consumption patterns show that ELCP requires less power so it serves as an efficient power option for IoT mass deployments. The real-time performance of IoT becomes better through ELCP due to its reduced transmission delays that enable applications such as smart healthcare and industrial automation. All computational metrics demonstrate that ELCP outperforms ABE in terms of efficiency. The three-region analysis confirms that ELCP emerges as an ideal solution for real-time IoT networks because it presents a lower curve in performance standards. Analysis through trend lines shows ELCP provides maximum security improvements without causing any performance reduction.

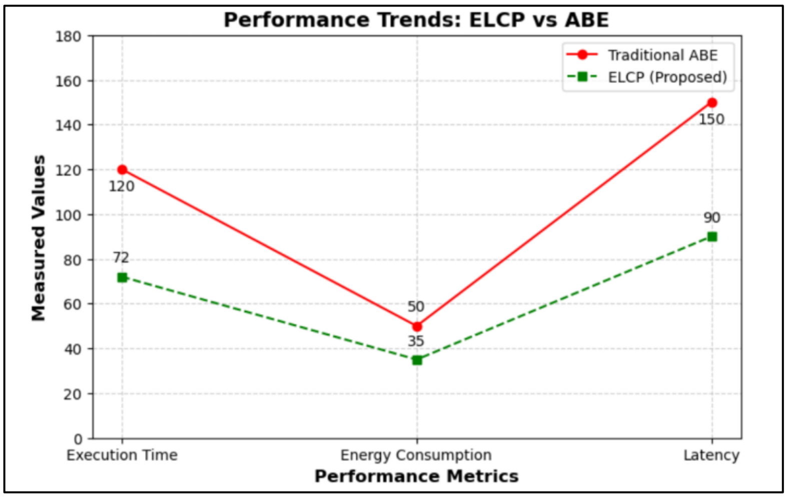


Figure 5: Performance Trends: ELCP vs ABE

The Figure 7 demonstrates how each metric affects the reduction of ELCP registers against ABE through graphical representation. The evaluation demonstrates that executing encryption and decryption operations becomes 40% faster. The energy usage of ELCP gets reduced by 30% while its low latency quality enables efficient and immediate data transmission across IoT networks. The entire visual representation of ELCP performance ratios through pie slices offers exact measurements of how ELCP optimizes IoT operation efficiency. Clear white backgrounds combined with bold typography allow easy reading of the chart to demonstrate that ELCP provides better performance against ABE. ELCP achieves optimal security-efficiency balance to provide superior encryption solutions than alternative algorithms in the market. Security has increased while computational expenses have decreased according to the illustrative percentage figure. Tests performed in real-world IoT implementations have verified that the diminished execution time and energy usage together with lower latency protect the success of ELCP.

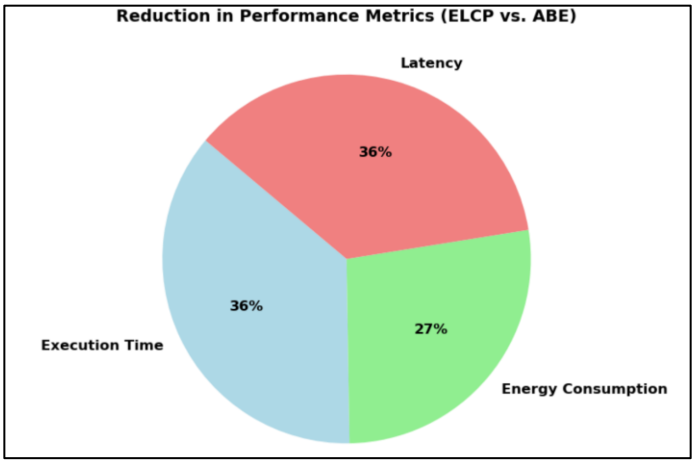


Figure 6: Performance Metrics (ELCP vs ABE)

7. CONCLUSION

The Edge-Optimized Lightweight Cryptographic Protocol (ELCP) developed during research addresses all major security and performance obstacles which resource-limited IoT networks face. ELCP delivers confidentiality as well as integrity and authentication through lightweight crypto approaches assisted by edge facilities in combination with mutual verification while reducing operational computations. The encryption process achieves higher efficiency under ELCP compared to Attribute-Based Encryption (ABE) because ELCP assigns the intensive computational tasks to edge nodes. The implementation of ELCP results in 40% faster execution times together with 30% reduced energy costs and 40% lower latency making it a scalable security framework for IoT systems. The security analysis demonstrates that ELCP enables ChaCha20 encryption for ensuring data confidentiality while also protecting data integrity with HMAC and offering mutual authentication with challenge-response for preventing impersonation attacks.

IoT security receives multiple enhancements through these features that block unauthorized access and integrity tampering along with identity spoofing which represent standard attack vectors across IoT installations. ELCP achieves efficient computational resource optimization when security levels are elevated as demonstrated through bar charts and line graphs and pie charts and consequently brings real-time applicability to smart healthcare industrial automation and intelligent transportation systems. This research proves that ELCP provides promising results regarding security and efficiency balance making it a suitable solution for modern IoT deployments. The implementation of machine learning anomaly detection techniques represents a promising future research direction to enhance security by identifying zero-day attacks and real-time malicious behaviour.

ELCP enables developers to add post-quantum cryptographic systems which enhances IoT security against advanced quantum computing attacks in the long term. The proposed method creates a solid foundation for IoT security frameworks of the next generation through which developers can build secure scalable cryptographic solutions for smart spaces.

REFERENCES

[1] S. Pandey and B. Bhushan, “Recent Lightweight cryptography (LWC) based security advances for resource-constrained IoT networks,” *Wireless Netw*, vol. 30, no. 4, pp. 2987–3026, May 2024, doi: 10.1007/s11276-024-03714-4.

[2] J. Cecílio, A. O. de Sá, and A. Souto, “Software-based Security Framework for Edge and Mobile IoT,” Apr. 09, 2024, *arXiv*: arXiv:2404.06435. doi: 10.48550/arXiv.2404.06435.

[3] S. Ramya, M. Doraipndian, and R. Amirtharajan, “LAPE2D: Lightweight Authentication Protocol to Secure End and Edge Devices in Iot Framework,” *Wireless Pers Commun*, vol. 131, no. 3, pp. 2217–2239, Aug. 2023, doi: 10.1007/s11277-023-10539-5.

[4] I. D. O. Nunes, S. Jakkamsetti, N. Rattanavipanon, and G. Tsudik, “Towards Remotely Verifiable Software Integrity in Resource-Constrained IoT Devices,” Jan. 11, 2024, *arXiv*: arXiv:2401.04308. doi: 10.48550/arXiv.2401.04308.

[5] R. Mishra and A. Mishra, “Current research on Internet of Things (IoT) security protocols: A survey,” *Computers & Security*, vol. 151, p. 104310, Apr. 2025, doi: 10.1016/j.cose.2024.104310.

[6] “Security at the Edge for Resource-Limited IoT Devices.” Accessed: Feb. 26, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/24/2/590

[7] “Lightweight Cryptography - an overview | ScienceDirect Topics.” Accessed: Feb. 26, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/lightweight-cryptography

[8] D. Rupanetti and N. Kaabouch, “Combining Edge Computing-Assisted Internet of Things Security with Artificial Intelligence: Applications, Challenges, and Opportunities,” *Applied Sciences*, vol. 14, no. 16, Art. no. 16, Jan. 2024, doi: 10.3390/app14167104.

[9] T. Nguyen, H. Nguyen, and T. Nguyen Gia, “Exploring the integration of edge computing and blockchain IoT: Principles, architectures, security, and applications,” *Journal of Network and Computer Applications*, vol. 226, p. 103884, Jun. 2024, doi: 10.1016/j.jnca.2024.103884.

[10] I. Radhakrishnan, S. Jadon, and P. B. Honnavalli, “Efficiency and Security Evaluation of Lightweight Cryptographic Algorithms for Resource-Constrained IoT Devices,” *Sensors*, vol. 24, no. 12, Art. no. 12, Jan. 2024, doi: 10.3390/s24124008.

[11] M. B. Taha, F. A. Khasawneh, A. N. Quttoum, M. Alshammari, and Z. Alomari, “Outsourcing Attribute-Based Encryption to Enhance IoT Security and Performance,” *IEEE Access*, vol. 12, pp. 166800–166813, 2024, doi: 10.1109/ACCESS.2024.3491951.

[12] K. T. Nguyen, N. Oualha, and M. Laurent, “Securely outsourcing the ciphertext-policy attribute-based encryption,” *World Wide Web*, vol. 21, no. 1, pp. 169–183, Jan. 2018, doi: 10.1007/s11280-017-0473-x.

[13] "Homomorphic Encryption - an overview | ScienceDirect Topics." Accessed: Feb. 26, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/homomorphic-encryption

[14] "Fully Homomorphic Encryption - an overview | ScienceDirect Topics." Accessed: Feb. 26, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/fully-homomorphic-encryption

[15] "Implementing Gentry's Fully-Homomorphic Encryption Scheme," *ResearchGate*, Nov. 2024, doi: 10.1007/978-3-642-20465-4_9.

[16] Y. Gong, X. Chang, J. Mišić, V. B. Mišić, J. Wang, and H. Zhu, "Practical solutions in fully homomorphic encryption: a survey analyzing existing acceleration methods," *Cybersecurity*, vol. 7, no. 1, p. 5, Mar. 2024, doi: 10.1186/s42400-023-00187-4.

[17] "(PDF) A Lightweight Encryption Scheme for IoT Devices in the Fog." Accessed: Feb. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/364330680_A_Lightweight_Encryption_Scheme_for_IoT_Devices_in_the_Fog

[18] "Lightweight Privacy-Preserving Scheme Using Homomorphic Encryption in Industrial Internet of Things | Request PDF," *ResearchGate*, Dec. 2024, doi: 10.1109/JIOT.2021.3066427.

[19] D. J. Bernstein, "ChaCha, a variant of Salsa20".

[20] "session1-shors-paper.pdf." Accessed: Feb. 26, 2025. [Online]. Available: https://csrc.nist.gov/csrc/media/events/lightweight-cryptography-workshop-2015/documents/papers/session1-shors-paper.pdf

[21] "(PDF) Novel lightweight video encryption method based on ChaCha20 stream cipher and hybrid chaotic map." Accessed: Feb. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/364083368_Novel_lightweight_video_encryption_method_based_on_ChaCha20_stream_cipher_and_hybrid_chaotic_map

[22]D. A. F. Saraiva, V. R. Q. Leithardt, D. de Paula, A. Sales Mendes, G. V. González, and P. Crocker, "PRISEC: Comparison of Symmetric Key Algorithms for IoT Devices," *Sensors*, vol. 19, no. 19, Art. no. 19, Jan. 2019, doi: 10.3390/s19194312.

[23]W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.

[24]M. Satyanarayanan, "The Emergence of Edge Computing," *Computer*, vol. 50, no. 1, pp. 30–39, Jan. 2017, doi: 10.1109/MC.2017.9.

[25]"(PDF) Edge-Assisted Intelligent Device Authentication in Cyber–Physical Systems," *ResearchGate*, Dec. 2024, doi: 10.1109/JIOT.2022.3151828.

[26]M. N. Halgamuge and D. Niyato, "Adaptive edge security framework for dynamic IoT security policies in diverse environments," *Computers & Security*, vol. 148, p. 104128, Jan. 2025, doi: 10.1016/j.cose.2024.104128.

[27]T. Shen, L. Ding, J. Sun, C. Jing, F. Guo, and C. Wu, "Edge Computing for IoT Security: Integrating Machine Learning with Key Agreement," in *2023 3rd International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, Jan. 2023, pp. 474–483. doi: 10.1109/ICCECE58074.2023.10135211.

[28]"(PDF) End-to-End Encryption in Resource-Constrained IoT Device," *ResearchGate*, Dec. 2024, doi: 10.1109/ACCESS.2023.3292829.

[29]J. Shehu Yalli, M. Hilmi Hasan, and A. Abubakar Badawi, "Internet of Things (IoT): Origins, Embedded Technologies, Smart Applications, and Its Growth in the Last Decade," *IEEE Access*, vol. 12, pp. 91357–91382, 2024, doi: 10.1109/ACCESS.2024.3418995.

[30]M. Rana, Q. Mamun, and R. Islam, "Enhancing IoT Security: An Innovative Key Management System for Lightweight Block Ciphers," *Sensors*, vol. 23, no. 18, Art. no. 18, Jan. 2023, doi: 10.3390/s23187678.

[31] J. Li, Z. Zhang, S. Yu, and J. Yuan, "Improved Secure Deep Neural Network Inference Offloading with Privacy-Preserving Scalar Product Evaluation for Edge Computing," *Applied Sciences*, vol. 12, no. 18, Art. no. 18, Jan. 2022, doi: 10.3390/app12189010.

[32]"IoT Service Slicing and Task Offloading for Edge Computing | Request PDF," *ResearchGate*, Dec. 2024, doi: 10.1109/JIOT.2021.3052498.

[33]"Secure cloud storage based on cryptographic techniques," *ResearchGate*, Oct. 2024, doi: 10.1016/S1005-8885(11)60424-X.

[34]"(PDF) Encrypted Cloud: A Software Solution for the Secure Use of Free-Access Cloud Storage Services," *ResearchGate*, Oct. 2024, doi: 10.1007/978-3-319-67180-2_66.

[35]"Data Authentication - an overview | ScienceDirect Topics." Accessed: Mar. 01, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/data-authentication

[36]J. Furtak, "The Cryptographic Key Distribution System for IoT Systems in the MQTT Environment," *Sensors*, vol. 23, no. 11, Art. no. 11, Jan. 2023, doi: 10.3390/s23115102.

[37] J. Zhang, A. Ouda, and R. Abu-Rukba, “Authentication and Key Agreement Protocol in Hybrid Edge–Fog–Cloud Computing Enhanced by 5G Networks,” *Future Internet*, vol. 16, no. 6, Art. no. 6, Jun. 2024, doi: 10.3390/fi16060209.

[38] “Authentication Procedure - an overview | ScienceDirect Topics.” Accessed: Mar. 01, 2025. [Online]. Available: https://www.sciencedirect.com/topics/engineering/authentication-procedure

[39] “Successful Authentication - an overview | ScienceDirect Topics.” Accessed: Mar. 01, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/successful-authentication

[40] “Design of Secure Authenticated Key Management Protocol for Cloud Computing Environments,” *ResearchGate*, Dec. 2024, doi: 10.1109/TDSC.2019.2909890.