DLKS-MQTT: A Lightweight Key Sharing Protocol for Secure IoT Communications

Sharadadevi Kaganurmath
Department of Computer Science & Engineering, Global Academy of Technology, Bengaluru, India
sharadask@gmail.com (corresponding author)

Nagaraj G. Cholli
Department of Information Science & Engineering, RV College of Engineering, Bengaluru, India
nagaraj.cholli@rvce.edu.in

M. R. Anala
Department of Information Science & Engineering, RV College of Engineering, Bengaluru, India
analamr@rvce.edu.in

Received: 12 January 2025 | Revised: 29 January 2025 and 9 February 2025 | Accepted: 10 February 2025

 DOI: https://doi.org/10.48084/etasr.10216

ABSTRACT

The increasing reliance on Message Queuing Telemetry Transport (MQTT) as a lightweight messaging protocol for Internet of Things (IoT) applications requires robust security mechanisms that address resource constraints while ensuring data integrity, confidentiality, and authenticity. This paper proposes the Dynamic Lightweight Key Sharing for MQTT (DLKS-MQTT) mechanism, a novel approach that integrates ephemeral key generation, streamlined authentication, and lightweight cryptographic operations to enhance the security of MQTT-based IoT communications. The mechanism employs a 128-bit key generated using a Linear Congruential Generator (LCG), providing robust resistance to brute-force and cryptanalytic attacks while maintaining computational and energy efficiency. Through extensive performance evaluations, DLKS-MQTT demonstrates significant improvements: reducing CPU energy consumption to 0.000002 mJ, achieving an execution time of 0.40 s, and minimizing communication overhead to 60 bytes, outperforming existing methods such as Dynamic Lightweight Authentication for MQTT (DLA-MQTT), Improved Ciphertext Policy-Attribute-Based Encryption (ICP-ABE), and Secure MQTT (SMQTT). The use of ephemeral session keys and nonces ensures protection against replay and Man-in-the-Middle (MitM) attacks, whereas lightweight hashing guarantees message integrity without burdening resource-constrained devices. This work establishes DLKS-MQTT as a practical, scalable, and secure solution for modern IoT networks, offering a balance between performance and security.

Keywords-Internet of Things (IoT); Message Queuing Telemetry Transport (MQTT); lightweight cryptography; resource-constrained devices; security; pseudo-random number generator; ephemeral key generation

I. INTRODUCTION

Internet of Things (IoT) has impacted several industries, including healthcare, smart homes, automotive and smart cars, smart factories, smart grids, and smart cities [1]. While these environments are great examples of interconnectivity, they also pose massive security threats, such as the Mirai botnet attack, which targeted insecure IoT devices to launch DDoS attacks. These vulnerabilities result in losses in terms of computational capacity, memory, and power consumption. Message Queuing Telemetry Transport (MQTT) is the most widely used lightweight messaging protocol for IoT applications but it is insecure by default, making it vulnerable to eavesdropping, message forging, and Man-in-the-Middle (MitM) attacks [2]. Conventional cryptographic techniques, though efficient, present challenges of increased computational and power intensity that modern IoT devices cannot support. These challenges have only recently been addressed by incorporating lightweight cryptographic protocols and authentication mechanisms. The concepts of security solutions for IoT need to be optimized, as they do not provide the flexibility required due to the frequently changing device states, network topologies, or threats. State-of-the-art techniques such as post-quantum cryptography and low-overhead authentication need to be sufficiently explored [3]. In this study, the Dynamic Lightweight Key Sharing for MQTT (DLKS-MQTT) mechanism is proposed, which effectively utilizes ephemeral key generation, minimalistic handshake protocols, and lightweight cryptographic hashes. Since DLKS-MQTT integrates security features by extending the MQTT protocols,

the integration does not add additional communication overhead. The objectives of this work include:

- Develop a dynamic key-sharing mechanism that generates ephemeral session keys with minimal computational and energy requirements, ensuring suitability.
- Uncover and mitigate vulnerabilities in MQTT-based IoT networks by evaluating the protocol's robustness against threats such as eavesdropping, replay attacks, and MitM attacks.
- Evaluate the protocol's computational efficiency, scalability, and security performance under diverse IoT conditions.

The techniques used to achieve these objectives are:

1. Dynamic authentication for resource-constrained IoT devices: Dynamic key generation and lightweight cryptographic operations are exploited in the proposed DLKS-MQTT mechanism to achieve flexible, robust, and scalable security suitable for the MQTT-based IoT networks.
2. Resource efficiency and computational overhead: The proposed work balances resource efficiency by minimizing computational overhead while ensuring high security. DLKS-MQTT employs a lightweight handshake protocol, an ephemeral key generation based on an optimized Linear Congruential Generator (LCG), and lightweight hashing.
3. Integration of advanced cryptographic techniques: This work is a bridge between lightweight cryptographic mechanisms and advanced security techniques. It integrates resource-aware and scalable cryptographic techniques such as 128-bit ephemeral keys and lightweight hashing with MQTT-based IoT networks.

II. PREVIOUS WORK

The application of the IoT in various industries has led to remarkable changes in the way information is collected and used. Among the communication protocols for IoT, MQTT is a widely used protocol due to its lightweight nature for resource-constrained IoT devices [4]. MQTT was originally designed for low-bandwidth and resource-constrained devices using the publish/subscribe model, which is efficient but has many security issues, including eavesdropping, message forgery, and MitM attacks [5]. This protocol does not have built-in security, so research has been initiated on various security enhancements, with an emphasis on authentication, encryption, and integrity [6]. The IoT-specific characteristics such as limited processing power, limited memory, and limited power sources paved the way for research on the feasibility of lightweight cryptographic techniques [7, 8]. Authors in [3] emphasized the importance of lightweight security protocols and focused on protecting the IoT networks without overloading resources. Several recent publications have proposed various lightweight encryption and authentication schemes. For example, authors in [9] proposed a lightweight encryption scheme that can be applied to secure MQTT while addressing the computational overhead. In the same logic, authors in [10] proposed a lightweight mutual authentication mechanism that protects and enhances device integrity and privacy while having low system overhead. Authors in [11] present the latest versions of lightweight cryptography to address the security challenges in IoT devices without compromising their performance. They are computationally less complex and suitable for MQTT-based communication.

Key management and distribution are another important requirement for IoT security. To address this issue, authors in [12] developed a dynamic key management system that takes into account the dynamic characteristics of IoT environments. Authors in [13] proposed an innovative authentication framework for MQTT that uses blockchain for device authenticity and data integrity in IoT communication. The integration of lightweight Public Key Infrastructures (PKIs) and certificate management systems has been explored as a solution to achieve manageable, yet secure, IoT device authentication [14]. Lightweight PKIs for IoT show promise in increasing MQTT security without consuming too many resources. The greedy heuristic method is used by the authors in [15] to effectively create dominating sets for improving security services in IoT networks. Authors in [16] proposed the Enhanced Wireless Intrusion Detection System (EW-IDS) method to provide stable security in IoT against cyber threats.

To boost the generation and preloading of keys in the sensor nodes for the multistage IoT networks, authors in [17] suggested an adaptive and resilient POK scheme which reduces the communication overhead and completely dispenses with time synchronization. It provides energy efficiency and is immune to attacks due to its self-healing property. Authors in [18] proposed a novel MQTT lightweight, secure secret key-sharing system with a (k, n) threshold secret-sharing mechanism. Their approach is faster than the public-key based systems, which involve key-sharing latency and computational complexity. Authors in [19] suggested a symmetric cryptographic Key-Generating, Renewing, and Distributing (KGRD) system, utilizing the primary TPM 2.0 hardware module for IoT nodes. The system provides secure key management and data exchange, and the communication medium between KGRD system nodes is MQTT. Authors in [20] proposed an enhanced security framework for IoT communication by integrating the MQTT protocol with the ARIA chipper 256 algorithm cryptography and mbedTLS library. Authors in [21] used adversarial training with Deep Q-Networks (DQN) to preserve semantic communication accuracy in both encrypted and unencrypted modes. This addresses important privacy concerns while striking a balance between secrecy and semantic communication correctness. Authors in [22] proposed the Fuzzy Mutual Information-based Feature Selection (Fuzzy-MIFS) technique, which combines fuzzy logic and Gaussian membership functions to increase the efficiency and accuracy of intrusion detection systems in IoT.

A. Research Gap

The literature highlights the high vulnerability of MQTT in IoT communication, which requires improved security features. Much research has been done on lightweight cryptographic solutions with success, but further research is crucial to respond to the dynamic security threats in the context of IoT.

Investigating security improvements for MQTT-type of IoT communication, especially for low-power devices, has revealed an existing research gap that is in line with the objective of this paper. Further research is needed on the IoT domain-specific adaptive lightweight cryptographic solutions and authentication protocols. The research should aim at optimizing IoT services in the context of enhanced security features. This paper contributes to this research gap by proposing a new lightweight dynamic MQTT-based IoT authentication approach that emphasizes security robustness while respecting the limited resources of IoT devices.

III. PROPOSED METHOD

To overcome the challenges associated with the existing lightweight authentication mechanism for MQTT-based IoT communication, this paper presents an alternative approach, referred to as the DLKS-MQT, which aims to provide advanced security and computational savings by incorporating a more lightweight random key generation method and a new handshake process. DLKS-MQTT also takes into account the characteristics of IoT environments, including limited resources and varying network conditions.

A. Conceptual Framework

The proposed DLKS-MQTT framework is compatible with MQTT, thus providing secure data exchange while moderately increasing the computational and energy overhead. DLKS-MQTT separates the MQTT broker from the IoT devices that connect to it, is responsible for message exchange, and controls the authentication process and secure communication. Each device has an ephemeral key generation module that generates session keys by using an LCG. These keys are transient in nature, providing security for each communication session. The IoT device uses the key in the payload of the "connect" message in the efficient handshake process. The payload is authenticated by the MQTT broker.

The DLKS-MQTT mechanism provides solid performance, efficiency, and simplicity of computation, ideal for the resource-constrained IoT devices. Authentication is built into the established message exchange design of MQTT, making it secure. This framework ensures that session keys are used to eliminate long-term vulnerability, and all transmitted messages are encrypted to prevent eavesdropping. Integration with DLKS-MQTT is always lightweight, secure, and relevant to various IoT scenarios.

1) Ephemeral Key Generation

Key generation in DLKS-MQTT mechanism is temporary or transient in each session to improve security while optimizing computational complexity. This process uses LCG, which is a simple and efficient random number generator specifically designed for limited IoT nodes

2) Minimalistic Handshake Protocol

The minimalistic handshake protocol in DLKS-MQTT helps to securely establish a connection between an IoT device and the MQTT broker. By integrating the authentication procedures into the standard "connect" and "connack" messages of the MQTT protocol, the need for additional overhead is avoided and the lightweight nature of MQTT is maintained. The procedure is as follows:

- Device-initiated connection: The IoT device sends an MQTT "connect" message containing the ephemeral session key K_{sess}, a nonce N, and its unique identifier ID_{dev}.These elements are embedded in the payload, hashed for integrity, and encrypted for confidentiality.
- Broker verification: The MQTT broker decrypts and verifies the payload by recomputing the hash and comparing it to the received hash. It validates the nonce to prevent replay attacks and authenticates the device.
- Session establishment: Upon successful verification, the broker responds with a "connack" message, completing the handshake. A secure session is then established using the ephemeral session key.

3) Lightweight Cryptographic Hashes

In the DLKS-MQTT mechanism, lightweight cryptographic hashing ensures message integrity and device authentication while maintaining computational efficiency. The approach leverages lightweight hash functions such as BLAKE2s or SipHash, which are tailored for resource-constrained IoT environments. These hashes provide strong security guarantees against common attacks, such as data tampering and forgery without computational overhead.

B. Integration with MQTT Protocol

The DLKS-MQTT mechanism seamlessly integrates with the MQTT protocol to enhance its security without compromising its lightweight nature by embedding authentication and encryption into MQTT's existing "connect" and "connack" messages, DLKS-MQTT achieves robust security with minimal modifications to the protocol's workflow. This ensures compatibility with existing MQTT-based IoT infrastructures while enhancing security.

1) Modifications to the "Connect" Message

The "connect" message of MQTT is retained in DLKS-MQTT, but with additional security features. The payload consists of an ephemeral session key that secures the communication between the device and the MQTT broker. To prevent replay attacks, the payload includes the nonce, which is a random number that guarantees that each connection request is new, ensuring integrity and that the message is not altered in transit. It is then encrypted with the broker's public key before being sent. The MQTT "connect" message, typically used to initiate a connection between an IoT device and the MQTT broker, is extended to include the following fields:

1. Ephemeral session key (K_{sess}): A temporary session key generated by the LCG to secure communication.
2. Device identifier (ID_{dev}): A unique device identifier.
3. Nonce (N): A randomly generated value to prevent attacks.
4. Payload hash (H_P): A cryptographic hash of the payload.

The payload of the enhanced "connect" message is constructed as follows:

$$P_{connect} = Encrypt_{K_{pub}}(K_{sess} \parallel ID_{dev} \parallel N \parallel H_P) \quad (1)$$

where K_{pub} is the broker's public key used for encryption.

2) Modifications to "Connack" Message

DLKS-MQTT makes modifications to the MQTT "connack" message to confirm authentication and establish secure communication. In the modified "connack" message, an authentication status field is used to indicate that the device has successfully authenticated. This results in immediate feedback to the device with the result of the connection. To avoid staleness, a lightweight timestamp generated by the broker is added to prevent replay attacks and verify that the communication is up to date. The handshake is completed when the broker acknowledges the ephemeral session key sent in the "connect" message. Both parties then use the session key to encrypt and decrypt the payload traffic to further enhance the security. The "connack" message that the broker sends to confirm the connection is modified to include:

1. Authentication status (S_{auth}): Indicates whether or not the authentication was successful.
2. Session key acknowledgement: Confirms the use of the session key for subsequent communications.
3. Timestamp (T_{broker}): A timestamp generated by the broker to validate the current state of the session.

The enhanced "connack" message payload is constructed as follows:

$$P_{connack} = Encrypt_{K_{sess}}(S_{auth} \parallel T_{broker}) \quad (2)$$

3) Session Key Integration

After the handshake process, the ephemeral session key is integrated into DLKS-MQTT to ensure secure communication. The session key is generated, exchanged, and authenticated during the "connect" and "connack" process, and it is used to encrypt all subsequent data transfers. The content can only be decrypted by the authorized parties. This process ensures that the communication is valid and legitimate, minimizing the exposure of the data to third parties. The session key used in DLKS-MQTT is ephemeral and unique, providing robust protection with the lightweight simplicity of the MQTT protocol. After the successful handshake:

1. The ephemeral session key (K_{sess}) is used to encrypt all messages exchanged between the device and the broker.
2. Each message payload M is encrypted using $M_{encrypted} = Encrypt_{K_{sess}}(M)$, where M represents the actual data to be transmitted.
3. Upon receipt, the broker decrypts the message using $M = Decrypt_{K_{sess}}(M_{encrypted})$.

C. Algorithm Design and Development

The DLKS-MQTT mechanism begins with ephemeral session key generation using an LCG, which ensures unique keys for each session. The authentication payload, which includes the session key, device identifier, and nonce, is hashed for integrity and encrypted using the broker's public key. This payload is transmitted via the "connect" message, initiating the handshake. The broker decrypts and validates the payload, verifying its integrity and authenticity. Once authenticated, the broker confirms the session key via an encrypted "connack" message. This algorithm integrates ephemeral key generation, hashing, and secure and lightweight encryption into the MQTT protocol. DLKS-MQTT provides a balance between encryption and authentication methods that minimizes computational and storage costs and optimizes resource utilization for devices with limited resources. As a result, efficiency and security are prioritized.

IV. RESULTS AND DISCUSSION

This section evaluates the performance and security of the proposed DLKS-MQTT mechanism. The analysis is based on simulations performed in the Cooja simulator using Contiki OS with realistic IoT network configurations. The DLKS-MQTT mechanism is shown to be efficient, scalable, and robust in various scenarios.

A. Experimental Setup

The simulation environment consisted of a mesh network topology with 31 IoT nodes, including 1 MQTT broker and 30 client nodes. The devices operated in a 200 m × 200 m area, with a transmission range of 50 m per node. MitM attack scenarios were simulated using two attacker nodes. The payload size was set to 128 bytes, incorporating DLKS-MQTT overhead. The system utilized a 128-bit key and BLAKE2s for hashing, ensuring robust security. Evaluation metrics included network latency, computational overhead, energy consumption, and security effectiveness. Table I presents the simulation environment specifications.

TABLE I. SIMULATION SCENARIO

Parameter	Specified values
Simulator	Cooja (Contiki OS)
Device models	Sky motes
Network topology	Mesh
Number of devices	31 IoT nodes (30 client nodes and 1 broker node)
Attacker nodes	2 nodes simulating various attack vectors (MitM attacks)
Transmission range	50 m per node
Simulation area	200 m × 200 m square area
MQTT broker	Internal simulated broker with real-world connection capabilities
Communication frequency	1 message per 2 minutes
Payload size	128 bytes (with DLKS-MQTT overhead included)
Security parameters	Key size: 128 bits Hash function: BLAKE2s – 128-bit
Evaluation metrics	Network latency: < 500ms Computational overhead: CPU cycles Energy consumption: measured in mJ Security effectiveness: against eavesdropping and MitM attacks

B. Comparative Analysis

Figure 1 illustrates the average energy consumption of various algorithms, including DLKS-MQTT, Dynamic

Lightweight Authentication for MQTT (DLA-MQTT), Improved Ciphertext Policy-Attribute-Based Encryption (ICP-ABE), Secure MQTT (SMQTT), and Key Schedule Algorithm (KSA)-PRESENT. The results highlight that DLKS-MQTT consumes the least energy at 0.0014 mJ, closely followed by DLA-MQTT at 0.0015 mJ. On the other hand, SMQTT exhibits the highest average energy consumption of 0.00177 mJ, whereas ICP-ABE and KSA-PRESENT demonstrate moderate energy consumption of 0.00155 mJ and 0.0016 mJ, respectively.

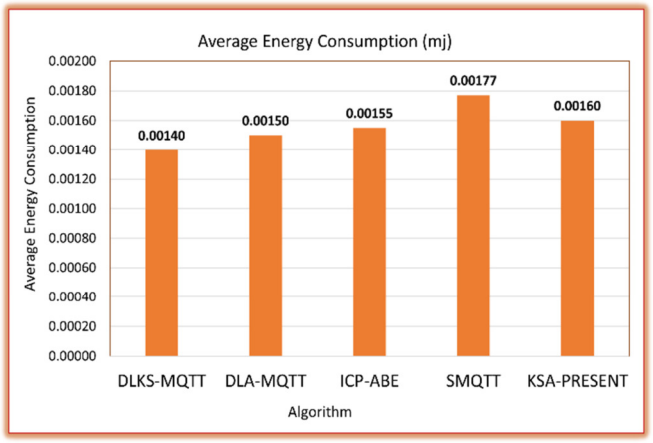

Fig. 1. Average energy consumption (mJ) comparison.

Figure 2 shows the CPU energy consumption of the algorithms, highlighting significant differences in energy efficiency. DLKS-MQTT exhibits the lowest CPU energy consumption at 0.000002 mJ, closely followed by DLA-MQTT with 0.000004 mJ. Both demonstrate exceptional energy efficiency, making them highly suitable for resource-constrained IoT environments. In contrast, SMQTT has the highest CPU energy consumption at 0.000472 mJ, significantly exceeding the consumption levels of all other algorithms. ICP-ABE and KSA-PRESENT show moderate CPU consumption values of 0.000041 mJ and 0.000047 mJ, respectively.

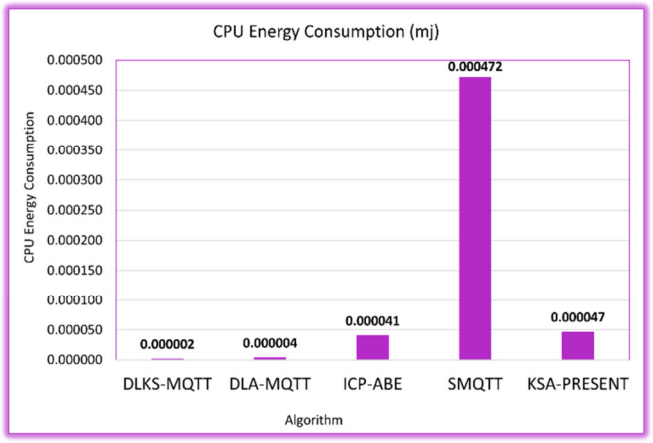

Fig. 2. CPU energy consumption (mJ) comparison.

Figure 3 presents the communication overhead of the algorithms in bytes, showcasing the impact of message size on network efficiency. Both DLKS-MQTT and DLA-MQTT demonstrate the lowest communication overhead, requiring only 60 bytes per message. Similarly, ICP-ABE exhibits a slightly higher overhead at 64 bytes, whereas KSA-PRESENT incurs an overhead of 80 bytes. On the other hand, SMQTT has the highest communication overhead at 128 bytes, more than double that of the most efficient algorithms.

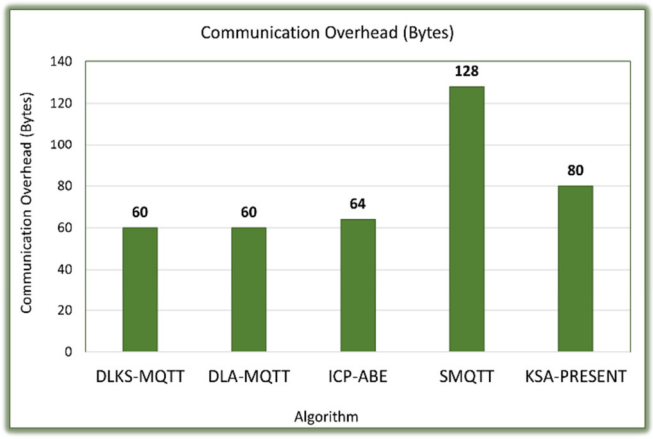

Fig. 3. Communication overhead (bytes) comparison.

Figure 4 illustrates the computational overhead of the algorithms, providing insight into their processing efficiency. Both DLKS-MQTT and DLA-MQTT have the lowest computational overhead at 280.00 units, reflecting their lightweight design. ICP-ABE incurs a slightly higher overhead of 305.49 units, whereas SMQTT has the highest overhead of 355.55 units. Similarly, KSA-PRESENT has a significant overhead of 329.57 units, just below SMQTT.

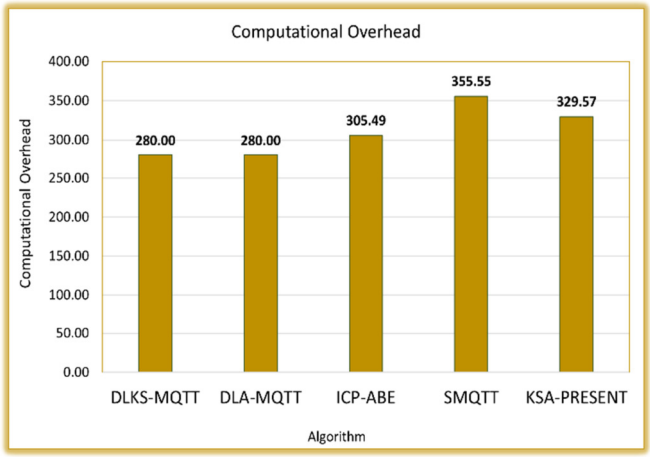

Fig. 4. Computational overhead comparison.

Figure 5 compares the execution times of the algorithms, providing insight into their computational efficiency. DLKS-MQTT and DLA-MQTT achieve the shortest execution times

of 0.40 s, reflecting their lightweight operations and optimized design. In contrast, ICP-ABE takes significantly longer, with an execution time of 1.6 s, whereas SMQTT exhibits the highest execution time of 2.8 s, highlighting its computational intensity. KSA-PRESENT also has a significant execution time of 1.8 s.

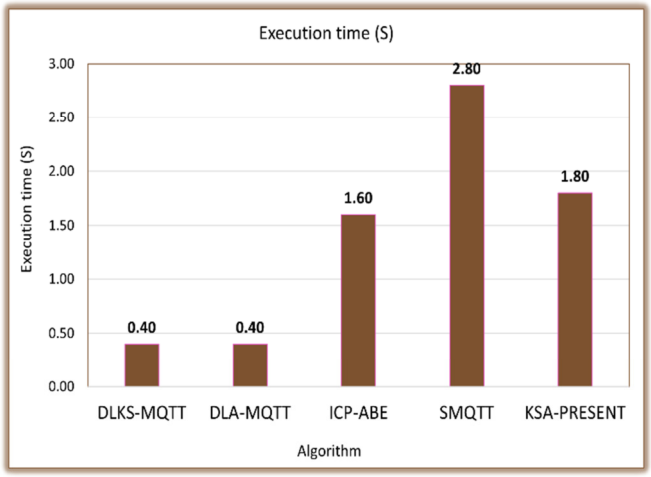

Fig. 5. Execution time (s) comparison.

V. CONCLUSION

In this work, we present a novel and robust solution for enhancing security in an Internet of Things (IoT) environment based on Message Queuing Telemetry Transport (MQTT) via a Dynamic Lightweight Key Sharing (DLKS)-MQTT mechanism. The DLKS-MQTT integrates ephemeral key generation, a streamlined handshake protocol, and lightweight cryptographic operations to address the challenges of resource efficiency and robust security. DLKS-MQTT is compared to existing solutions in terms of energy consumption, computational overhead, communication efficiency, and execution time, and its superiority is demonstrated, making it suitable for resource-constrained IoT devices. The use of a 128-bit key derived from a Linear Congruential Generator (LCG) provides very strong resistance to brute force, cryptanalytic attacks, and provides a sufficiently large key space in line with modern cryptographic standards. Nonces and lightweight hashing are included to prevent replay attacks and provide data integrity to protect the communication process from tampering and eavesdropping. The scalability and adaptability of the proposed mechanism makes it suitable for use in a variety of IoT applications. Its ability to withstand emerging threats could be further enhanced by integrating post-quantum cryptographic algorithms and adaptive machine learning techniques in future work. Overall, the DLKS-MQTT design represents a significant step forward in securing MQTT communications.

REFERENCES

[1] M. Attaran, "The impact of 5G on the evolution of intelligent automation and industry digitization," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 5, pp. 5977–5993, May 2023, https://doi.org/10.1007/s12652-020-02521-x.

[2] C. Patel and N. Doshi, "A Novel MQTT Security framework In Generic IoT Model," *Procedia Computer Science*, vol. 171, pp. 1399–1408, Jun. 2020, https://doi.org/10.1016/j.procs.2020.04.150.

[3] W. Robert *et al.*, "A Comprehensive Review on Cryptographic Techniques for Securing Internet of Medical Things: A State-of-the-Art, Applications, Security Attacks, Mitigation Measures, and Future Research Direction," *Mesopotamian Journal of Artificial Intelligence in Healthcare*, vol. 2024, pp. 135–169, Nov. 2024, https://doi.org/10.58496/MJAIH/2024/016.

[4] A. Banks and G. Rahul, "MQTT Version 3.1.1, " OASIS Standard. https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html.

[5] P. P. Ray, "A survey on Internet of Things architectures," *Journal of King Saud University - Computer and Information Sciences*, vol. 30, no. 3, pp. 291–319, Jul. 2018, https://doi.org/10.1016/j.jksuci.2016.10.003.

[6] T. Eisenbarth, S. Kumar, C. Paar, A. Poschmann, and L. Uhsadel, "A Survey of Lightweight-Cryptography Implementations," *IEEE Design & Test of Computers*, vol. 24, no. 6, pp. 522–533, Nov. 2007, https://doi.org/10.1109/MDT.2007.178.

[7] J.-Y. Lee, W.-C. Lin, and Y.-H. Huang, "A lightweight authentication protocol for Internet of Things," in *2014 International Symposium on Next-Generation Electronics*, Kwei-Shan, Tao-Yuan, Taiwan, 2014, pp. 1–2, https://doi.org/10.1109/ISNE.2014.6839375.

[8] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: A survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, Jun. 2017, https://doi.org/10.1016/j.jnca.2017.04.002.

[9] S. Al Salami, J. Baek, K. Salah, and E. Damiani, "Lightweight Encryption for Smart Home," in *2016 11th International Conference on Availability, Reliability and Security*, Salzburg, Austria, 2016, pp. 382–388, https://doi.org/10.1109/ARES.2016.40.

[10] B. B. Ehui, Y. Han, H. Guo, and J. Liu, "A Lightweight Mutual Authentication Protocol for IoT," *Journal of Communications and Information Networks*, vol. 7, no. 2, pp. 181–191, Jun. 2022, https://doi.org/10.23919/JCIN.2022.9815201.

[11] P. Singh, B. Acharya, and R. K. Chaurasiya, "Lightweight cryptographic algorithms for resource-constrained IoT devices and sensor networks," in *Security and Privacy Issues in IoT Devices and Sensor Networks*, S. K. Sharma, B. Bhushan, and N. C. Debnath, Eds. Cambridge, MA, USA: Academic Press, 2021, ch. 8, pp. 153–185, https://doi.org/10.1016/B978-0-12-821255-4.00008-0.

[12] R. Pothumarti, K. Jain, and P. Krishnan, "A lightweight authentication scheme for 5G mobile communications: a dynamic key approach," *Journal of Ambient Intelligence and Humanized Computing*, Jan. 2021, https://doi.org/10.1007/s12652-020-02857-4.

[13] M. Abdelrazig Abubakar, Z. Jaroucheh, A. Al-Dubai, and X. Liu, "Blockchain-based identity and authentication scheme for MQTT protocol," in *Proceedings of the 2021 3rd International Conference on Blockchain Technology*, Shanghai, China, 2021, pp. 73–81, https://doi.org/10.1145/3460537.3460549.

[14] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, May 2018, https://doi.org/10.1016/j.future.2017.11.022.

[15] S. Balbal and S. Bouamama, "Minimizing IoT Security Deployment Costs using the Dominating Set Approach," *Engineering, Technology & Applied Science Research*, vol. 14, no. 6, pp. 18324–18329, Dec. 2024, https://doi.org/10.48084/etasr.8725.

[16] B. Mopuru and Y. Pachipala, "Advancing IoT Security: Integrative Machine Learning Models for Enhanced Intrusion Detection in Wireless Sensor Networks," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 14840–14847, Aug. 2024, https://doi.org/10.48084/etasr.7641.

[17] M.-L. Messai, "AdaPtive and rObust Key pre-distribution for multi-phase IoT networks," *International Journal of Communication Systems*, vol. 37, no. 13, Sep. 2024, Art. no. e5824, https://doi.org/10.1002/dac.5824.

[18] T. Noguchi, M. Nakagawa, M. Yoshida, and A. G. Ramonet, "A Secure Secret Key-Sharing System for Resource-Constrained IoT Devices using MQTT," in *2022 24th International Conference on Advanced*

Communication Technology, PyeongChang, Kwangwoon Do, Republic of Korea, 2022, pp. 147–153, https://doi.org/10.23919/ICACT53585.2022.9728781.

[19] J. Furtak, "The Cryptographic Key Distribution System for IoT Systems in the MQTT Environment," *Sensors*, vol. 23, no. 11, Jun. 2023, Art. no. 5102, https://doi.org/10.3390/s23115102.

[20] M. Iqbal, A. M. Ari Laksmono, A. T. Prihatno, D. Pratama, B. Jeong, and H. Kim, "Enhancing IoT Security: Integrating MQTT with ARIA Cipher 256 Algorithm Cryptography and mbedTLS," in *2023 International Conference on Platform Technology and Service*, Busan, Republic of Korea, 2023, pp. 91–96, https://doi.org/10.1109/PlatCon60102.2023.10255171.

[21] I. R. Alzahrani, "Semantic IoT Transformation: Elevating Wireless Networking Performance through Innovative Communication Paradigms," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 15717–15723, Aug. 2024, https://doi.org/10.48084/etasr.7784.

[22] A. H. A. Saq, A. Zainal, B. A. S. Al-Rimy, A. Alyami, and H. A. Abosaq, "Intrusion Detection in IoT using Gaussian Fuzzy Mutual Information-based Feature Selection," *Engineering, Technology & Applied Science Research*, vol. 14, no. 6, pp. 17564–17571, Dec. 2024, https://doi.org/10.48084/etasr.8268.