Lightweight Cryptographic and Scalable IoT Systems for Encryption across GSM-MQTT Architectures in Resource-Constrained Aquaculture Environment

Rupali P. Shete
Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune Campus, Lavale, Pune, Maharashtra, India
rupali.shete.phd2020@sitpune.edu.in

Anupkumar M. Bongale
Department of Artificial Intelligence and Machine Learning, Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune Campus, Lavale, Pune, Maharashtra, India
anupkumar.bongale@sitpune.edu.in (corresponding author)

Deepak Dharrao
Department of Computer Science and Engineering, Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune Campus, Lavale, Pune, Maharashtra, India
deepak.dharrao@sitpune.edu.in

Received: 7 May 2025 | Revised: 2 June 2025 | Accepted: 6 June 2025

 DOI: https://doi.org/10.48084/etasr.10367

ABSTRACT

Internet of Things (IoT) technologies in environmental monitoring provide real-time remote data acquisition capabilities and decision-making for different domains, including aquaculture. This study investigates a secure water quality monitoring system designed specifically for fish farms through IoT technology with an emphasis on lightweight encryption methods combined with real-time GSM communications. The primary objective is to protect and maintain the confidentiality, integrity, and operational efficiency of vital pH, Dissolved Oxygen (DO), and Temperature (T) measurements collected by low-power microcontroller devices. A wireless data transmission system was based on Arduino with calibrated analog sensors and the SIM800L GSM module to implement the MQTT protocol for data transfer. The data security assessment involved the implementation and testing of the AES-128-CBC and SPECK-128-CBC symmetric encryption algorithms under equivalent real-time circumstances. SPECK-128-CBC delivered faster encryption (209.64 µs) than AES-128-CBC (235.08 µs) along with improved memory efficiency and more compact payloads, indicating its best fit for constrained environments. AES-128-CBC demonstrated a slightly higher level of ciphertext entropy, reaching 5.18 bits/byte compared to 5.09 of SPECK-128-CBC. Using a weighted scoring method that weighted entropy at 40%, processing speed at 30%, and memory and payload efficiency at 30%, SPECK achieved 0.9931 while AES reached 0.9484. These results suggest that SPECK-128-CBC offers an energy-efficient encryption solution to provide optimal security for GSM-based IoT systems in aquaculture environments.

Keywords-IOT security; lightweight encryption; resource constrained devices; aquaculture; AES-128; SPECK

I. INTRODUCTION

IoT systems in critical domains, including environmental monitoring and aquaculture, require secure data transmission. Fish farms with IoT devices struggle to use secure communication channels because they use GSM networks combined with lightweight protocols such as MQTT. Cyber security threats in these networks arise from attacks that range from eavesdropping on transmitted data to spoofing device identities to inject false data through replay attacks, in which previous data transmissions are repeated, and data tampering that maliciously modifies transmitted messages [1]. These security threats represent crucial risks in fish farm systems that rely on continuous sensor data for alert detection and management decisions. Data inaccuracies or tampering can

generate inappropriate system responses, resulting in adverse effects on water conditions that could even cause aquatic species to die [2, 3].

Aquaculture systems must protect data confidentiality and ensure integrity while validating its authenticity because precise and timely decision-making is vitally important. Encryption methods are used to protect data, making them unreadable from unauthorized parties. This context requires lightweight encryption because the system operates with power-limited microcontrollers that handle data in potentially challenging environments [4]. The need for lightweight encryption within embedded IoT devices emerges from published studies, which have shown that secure traditional algorithms require excessive processing power. Lightweight ciphers, such as SPECK, demonstrate practical performance-security trade-offs with limited resources [5]. The implementation of suitable encryption approaches offers dual benefits in protecting against external threats while preserving the reliability of alerting and decision-making that sustains aquaculture practices.

Research interest in lightweight cryptography has grown significantly to solve deficiencies within embedded IoT systems. In [6], lightweight block ciphers were analyzed to emphasize the need for efficient cryptographic solutions for resource-constrained IoT environments. AES-128, SPECK, and ASCON were measured on Arduino and Micro boards to determine execution time, memory consumption, system reliability, latency rates, and data transfer speed, with SPECK being superior for IoT devices with limited resources. In [7], lightweight cryptographic algorithms used for power-limited microcontrollers in IoT systems were examined by resource-requirement analysis and evaluation of power usage, memory utilization, latency, and throughput. In [8], DNA-LWCS was presented as a lightweight DNA-based cryptography system to improve IoT security. This method generates encryption keys through DNA sequences along with Elliptic Curve Cryptography (ECC) to protect data transmission. The evaluation results showed that DNA-LWCS provides better security and operational efficiency for lightweight IoT systems compared to existing methods.

In [9], a lightweight cryptographic solution was presented for medical IoT devices, combining Combined Transformation and Expansion (CTE) with a dynamic chaos system. This system provides efficient encryption-decryption operations at low memory consumption. In [10], contemporary lightweight cryptographic protocols for IoT networks were reviewed, examining and comparing performance metrics and cryptographic capabilities of existing block ciphers. Lightweight cryptographic algorithms are indispensable for IoT security but require additional research to achieve optimal operational efficiency. In [11], lightweight cryptographic algorithms developed to protect IoT devices were evaluated. This study examined 54 lightweight cryptographic primitives, consisting of block ciphers, stream ciphers, hash functions, and ECC variants. The research findings emphasized the importance of proper lightweight cryptographic algorithm choices to secure IoT systems with limited resources.

Secure IoT systems have incorporated recent advances to achieve a balance between encryption strength and computational efficiency for embedded applications. This study combined essential studies focusing on AES-128 and SPECK-128 encryption along with the integration of mobile applications and secure data transmission frameworks. Numerous studies have analyzed lightweight encryption ciphers that operate on microcontroller-based IoT systems. AES functions as the main data security standard but imposes substantial memory usage requirements along with processing latency on low-power devices. SPECK has emerged as an effective replacement for AES because it runs faster with reduced memory requirements while generating appropriate randomness [12, 13]. In [14], a classification and recommendations were presented for IoT security threats at different architectural levels, proposing device-specific end-to-end encryption. Modular resilient frameworks represent a critical necessity, which particularly applies to environmental and smart grid installations. Other studies explored hardware-specific enhancements, such as memory-efficient AES hardware accelerators and reprogrammable architectures, although they are often unsuitable for field deployment due to cost and complexity. In [13], the function of AES-128 was evaluated in wireless LoRaWAN networks to measure timing and energy compared to GSM-based systems. Multiple experimental studies have shown that the entropy value and key rotation systems are crucial for long-term encryption maintenance. Shannon entropy analysis is an essential quantity to assess both message encryption quality and resilience against cryptanalysis attacks. Static pool rotation and other dynamic key management methods successfully reduce prediction vulnerabilities without requiring intense computation. MQTT is consistently employed in various IoT research projects that require fast and lightweight communication. Strong proof of MQTT reliability through GSM connections exists in water quality monitoring systems and smart agriculture deployments, proving its suitability for resource-limited environments.

Recent advances in lightweight cryptography have focused on balancing strong encryption with the constraints of embedded IoT environments. Various models have been proposed to optimize execution time, memory usage, entropy integrity, and full-stack system scalability. In [15], a data security model merged Physically Unclonable Functions (PUFs) with lightweight encryption to secure IoT sensor-level information within energy-restricted IoT environments. In [16], an AI-assisted encryption mechanism employed 3D Arnold transforms with quantum logic diffusion to protect autonomous IoT system data. In [17], an encryption system was proposed to protect medical data in IoT networks using ECC along with a Caesar cipher and biometric authentication methods. Here, the "Lab-in-a-Box" framework was established, which integrates AES-128 and XTEA encryption into a modular IoT system. In [18], AES was modified to optimize power consumption along with security enhancements to maintain high encryption strength. The experimental setup verified Shannon entropy and proved the effectiveness of modified lightweight encryption models for IoT deployment. In [19], integrity mechanisms were combined with encryption and hash functions to increase confidentiality in smart home networks. These findings

demonstrate agreement about entropy as an encryption measurement standard, the value of key rotation during encryption, and the advantages of IoT framework designs that merge GSM with MQTT and AES/SPECK with mobile application development.

Recent studies on secure and scalable IoT communication architectures have increasingly focused on balancing robust data protection with the processing and energy constraints of low-power distributed devices. Software-based systems have been proposed to provide secure edge and mobile IoT communication with lightweight cryptographic methods and distributed authentication mechanisms [20]. In [21], a five-layer IoT architecture integrated blockchain elements and a new data link layer, demonstrating better scalability and improved secure speeds in decentralized IoT distribution networks. In [22], LS3 was proposed, which applied ECC and symmetric keys for developing lightweight secure transmission protocol. In [23], a trust-based architecture combined blockchain technology with directed acyclic graph structures. In [24], a symmetric block-cipher encryption model was proposed for resource-constrained devices.

II. METHODOLOGY

This section discusses the system integration of two symmetric encryption algorithms, AES-128-CBC and SPECK-128-CBC, embedded within the microcontroller firmware. These algorithms are suitable for constrained IoT environments due to their strengths in cryptographic resistance and computational speed. The key mechanism is added with rotational capabilities to increase encryption entropy without requiring sophisticated key exchange protocols. Figure 1 presents the architecture of the proposed IoT system. The module enables secure observation of water quality in fish farming systems. System design unites hardware elements with software features to maintain uninterrupted sensing operations and reliable data transfer. The system architecture consists of an Arduino microcontroller integrated with three environmental sensors: pH, Dissolved Oxygen (DO), and Temperature (T). These sensors provide critical data for Water Quality Index (WQI) calculation, which is treated as sensitive information that requires end-to-end protection.

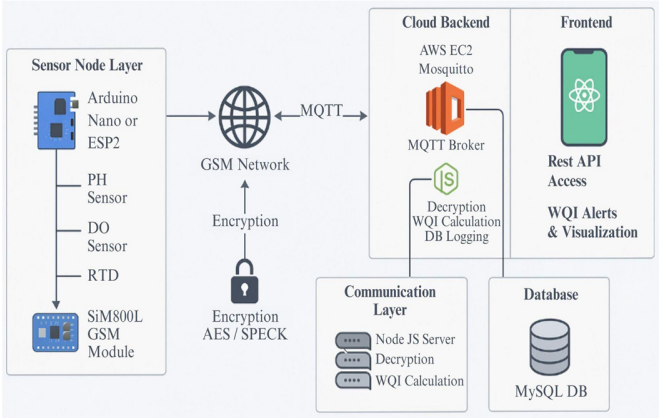

Fig. 1. The architecture of the proposed IoT-based secure environmental monitoring.

This study analyzes encryption and decryption times, memory usage values, payload size, and ciphertext entropy measurements. The microcontroller firmware was designed to encrypt all sensor data before transmission using one of the two embedded encryption modes, AES or SPECK, selected at runtime. The method includes AES-128-CBC and SPECK-128-CBC cryptography in the microcontroller firmware, making it possible to test the encryption speed with real sensor data and transmission circumstances. The SIM800L GSM module establishes GPRS connectivity to the cloud, and the system performs accuracy tests during calibration before connecting signals to the microcontroller through analog and UART channels. Sensor readings were collected every 15 minutes between 9:00 am and 6:00 pm during a seven-day deployment in ponds located in Pune, India (Latitude: 18° 25' 59.646" N, Longitude: 74° 27' 51.5124" E). The system design was modified to provide lightweight secure asynchronous communication capabilities. The Arduino firmware on the controller periodically collects sensor data, which are encrypted through AES-128-CBC or SPECK-128-CBC, and formats them into JSON. On the backend, a Node.js server, subscribed to MQTT, decrypts the incoming payloads based on the key_id embedded in each message and computes the WQI based on preset environmental thresholds. The processed data are logged in structured CSV files for analysis. A mobile application, developed in React Native, used RESTful APIs from the backend to display real-time water quality metrics securely to farm operators.

III. SECURITY IMPLEMENTATION

This study evaluates the transmission security of real-time water quality data in GSM-based IoT systems using two symmetric block ciphers: AES-128 and SPECK-128. AES-128 is standardized by the National Institute of Standards and Technology (NIST) and uses 128-bit blocks and keys through a ten-round substitution-permutation design. The system uses TinyAESLib on Arduino and the Node.js built-in crypto module. SPECK-128 is a lightweight cipher developed by the National Security Agency (NSA), which operates through 27 ARX-based rounds of Add, Rotate, and XOR operations using a 128-bit key and 64-bit block size. It operates manually in both Arduino and Node.js frameworks, providing exact control capabilities and performance-tuning mechanisms. Both cryptographic algorithms receive a key rotation update to increase entropy and prevent repeated patterns in the ciphertext. The system uses a predefined key system instead of real-time negotiation due to the inability of GSM to support negotiations effectively. Evaluation of AES and SPECK uses metrics such as encryption and decryption times, payload size measurements, memory consumption levels, and ciphertext entropy values. The results provide recommendations for choosing encryption methods to match the needs of secure low-power IoT applications during remote GSM-based operations.

A. Algorithm Selection

AES-128-CBC and SPECK-128-CBC were selected due to their contrasting strengths and relevance to real-world deployment scenarios. AES-128 is known for its robust security and widespread adoption, making it a reliable benchmark to evaluate encryption performance. In contrast,

SPECK-128, is a lightweight cipher optimized for resource-constrained environments, offering efficient performance through simple ARX-based operations. Algorithm 1 represents the secure data encryption pipeline using AES-128-CBC and SPECK-128-CBC. Sensor readings (T, DO, and pH) are collected using Arduino, encoded, and padded into 128-bit blocks. A random 128-bit Initialization Vector (IV) is generated for the CBC mode. In the AES-128-CBC scheme, each plaintext block is XORed with the previous ciphertext (or IV for the first block) and then processed through 10 rounds of SubBytes, ShiftRows, MixColumns, and AddRoundKey transformations using a 128-bit symmetric key. In contrast, SPECK-128-CBC employs a Feistel structure optimized for lightweight devices. Each 128-bit block is split into two 64-bit words; each round performs circular shifts, modular additions, and XOR operations using round keys derived from a simple key schedule. These minimalistic operations make SPECK significantly faster on constrained platforms. The resulting ciphertext blocks and final IV are transmitted over MQTT.

```
Algorithm 1: Secure IoT Data Encryption
for Aquaculture using AES-128-CBC and
SPECK-128-CBC
Input:  Sensor readings T, DO, pH
Output: Encrypted payload transmitted
        securely via MQTT
while system is active do
  (T, DO, H) ← Collect Sensor Data();
  M ← Encode And Pad(T // DO // pH);
  IV ← Random 128 Bit IV();
  if cipher == "AES" then
    for each block Mi in M do
     X ← Mi ⊕ IV // XOR plaintext with IV
                 // or previous
    ciphertext Ci ← AES128(X)
    // Perform 10 rounds of SubBytes,
    // ShiftRows, MixColumns, AddRoundKey
    IV ← Ci;
  else if cipher == "SPECK" then
    for each block Mi in M do
      X ← Mi ⊕ IV
      for i = 0 to Tr - 1 do
        XL, XR ← Split(X);
        XL ← ((XL ≫ α) + XR) ⊕ ki;
        XR ← (XR ≪ β) ⊕ XL;
        X ← XL // XR;
        Ci ← X;
        IV ← Ci;
Payload ← {C1, C2, ..., Cn, IV};
MQTT Publish (Payload);
```

Evaluating both algorithms under identical hardware and network conditions enables the investigation of their trade-offs in terms of encryption strength, execution time, memory usage, and ciphertext entropy. This allows a balanced investigation of standard-compliant versus lightweight encryption methods, making it suitable for constrained IoT systems.

B. Data Collection Method

Synchronized time-based data were used to evaluate the real-world performance between AES-128-CBC, SPECK-128-CBC, and an unencrypted baseline configuration. A continuous system operation over seven days through the setup in fish ponds in Pune, India served as data collection where the sensors recorded measurements every 15 minutes from 9.00 to 18.00 IST. The testing method produced data points that measured short-term performance as well as long-term trends through a process that generated about 350 measurements during the 7-day test period. Each transmission cycle of the Arduino-based IoT device retrieved live sensor measurements from its pH, DO, and T sensors. Table I shows a sample of the collected data.

TABLE I. SAMPLE DATA COLLECTED AT THE SITE

Water temperature T (°C)	pH of water	Dissolved oxygen DO (mg/L) in water
27.67	8.50	8.40
27.56	8.40	8.50
27.45	8.40	8.50
27.41	8.40	8.40
27.29	8.80	8.40

The experimental data were transmitted as plaintext with the baseline mode or encrypted by AES-128 or SPECK-128 with Base64 encoding and then distributed through the GSM link to an MQTT broker with the SIM800L module. The encryption process time on the microcontroller was measured through the micros() function, which tracked the precise microseconds required to encrypt each message. The published message included this value for analysis. The decryption time measurement was performed on the Node.js backend through time measurements before and after running the decryption function. The measurement of processing overhead indicates how much time is needed to complete the work and thus determines the overall system latency. Memory consumption of Flash and RAM was measured using compile-time summary reports accessible through the Arduino IDE. A single recording of memory usage values per algorithm implementation occurred during each mode's execution to compare the resource utilization of AES and SPECK. The evaluation of ciphertext entropy occurred using plaintext log files to store encrypted payloads collected from each cycle:

$$H = -\sum_{i=1}^{n} p(x_i) log_2 p(x_i) \quad (1)$$

where $p(x_i)$ is the probability of character x_i appearing in the ciphertext, and H is the entropy value (bits per byte). All collected parameters were compiled into structured CSV files for each test condition. This uniform data logging framework facilitated consistent analysis, allowing comparative evaluation of encryption schemes in terms of both security and performance within a constrained real-time GSM-based IoT deployment.

C. Data Analysis Procedure

The proposed GSM-based IoT fish farming system required measurements of encryption time, decryption time, payload size, ciphertext entropy, and memory usage to evaluate the performance of AES-128-CBC and SPECK-128-CBC.

D. Normalized Scores and Final Comparison

The scoring results shown in Table II reveal that SPECK-128-CBC reached a higher total score of 0.9931 while AES-128-CBC achieved 0.9484. AES had slightly better entropy but SPECK outperformed it in all other categories, including processing speed, memory footprint, and payload efficiency. A similar approach was undertaken in [25], emphasizing weighted scoring based on normalized performance parameters to evaluate trust and efficiency in IoT environments. These results show that SPECK-128-CBC provides an optimal solution for IoT applications that require lightweight performance along with moderate cryptographic security capabilities. AES-128-CBC functions as a reliable encryption option but offers better security when hardware capabilities are adequate.

TABLE II. WEIGHTED SCORES OF AES-128-CBC AND SPECK-128-CBC FOR EACH PARAMETER

Metric	Weight (%)	AES-128-CBC	SPECK-128-CBC
Ciphertext Entropy	40	1	0.9826
Processing time (encryption+decryption avg)	30	0.9226	1
Payload size	10	0.8928	1
Memory efficiency (Flash+RAM)	20	0.9118	1
Final weighted score	100	0.9484	0.9931

IV. RESULTS AND DISCUSSION

A 7-day comparison between AES-128-CBC and SPECK-128-CBC was performed. A security system without an encryption function provided a point of reference. The encryption-decryption process along with payload size, memory usage, encryption time, decryption time, and ciphertext entropy measurements were taken on the IoT device and back-end server during each operational period. Similar comparative analyses between algorithms for such parameters have been conducted in related studies [13], supporting the relevance of multimetric evaluation in constrained environments. According to the experimental results, SPECK-128-CBC showed a faster speed compared to AES-128-CBC. Figure 2 shows that SPECK encrypted the data within an average time of 209.64 µs, while the decryption process took 143.08 µs, compared to the AES times of 235.08 and 147.23 µs, respectively. Unencrypted transmission with the baseline method required short decryption times (~41 µs) due to its lack of cryptographic operations. The encryption processes of SPECK-128-CBC outperform those of AES-128-CBC, while decryption runs slightly faster. The performance advantage of SPECK results from its compact design, along with its minimal computational requirements, which enables usage in limited environments and real-time systems. On the contrary, AES-128-CBC, while highly secure, incurs greater processing overhead due to its more complex key schedule and block operations. The scenario without encryption shows the shortest times because it skips all cryptographic processing, thus avoiding the additional time needed for encryption and decryption.

The adoption of SPECK as a lightweight cryptographic algorithm remains essential for IoT deployments because it provides security while maintaining performance but brings upon data security challenges. These differences are essential in real-time systems because they determine the effectiveness of the alert mechanism through responsiveness. The SPECK method outperforms AES because its ARX-based structure requires less execution power due to its minimal computations compared to AES substitution-permutation operations and S-box transformations. Resource-constrained microcontrollers would need more resources to process AES due to its heavier structural requirements despite its well-known robustness.

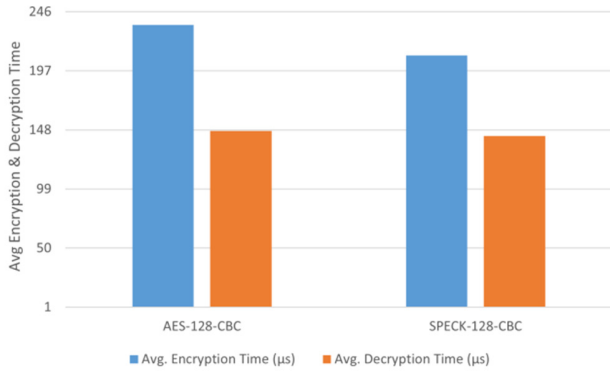

Fig. 2. Average encryption and decryption times (µs) for AES-128-CBC, SPECK-128-CBC, and the baseline configuration.

Figure 3 shows that AES-128-CBC encryption produces the largest average payload size among the methods tested. When implementing AES-128-CBC encryption, the resulting data size reaches its maximum value. The transmission of extra bytes occurs because AES-128-CBC has a 16-byte IV and must pad data to meet block boundary requirements. SPECK-128-CBC maintains a smaller IV and basic padding rules which enables it to generate a medium-sized encrypted message. The baseline scenario provides the smallest transmission because it sends data without encryption additions. IoT networks must balance encryption security against network transmission speed through constrained network conditions. The SPECK protocol achieves the right balance between security level and data transmission efficiency, making it suitable for bandwidth-constrained and energy-sensitive GSM-based systems.

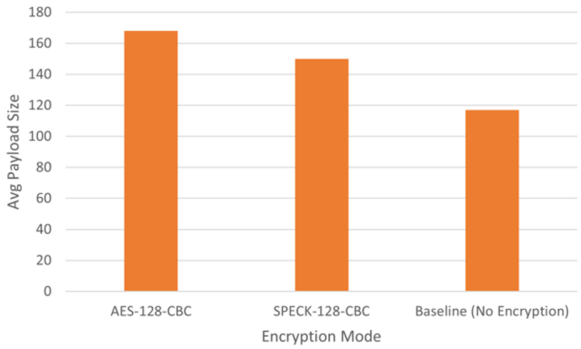

Fig. 3. Average payload size (bytes) per encryption mode, including baseline.

Figure 4 shows the average byte entropy values of the payloads encrypted with different modes. AES-128-CBC demonstrates a slightly superior entropy value at 5.18 bits/byte than SPECK-128-CBC, which measures 5.09 bits/byte. The entropy measure determines the unpredictable nature of ciphertexts, as higher entropy values indicate better diffusion properties that lead to superior resistance to statistical analysis. AES demonstrates slightly better entropy than other encryption methods, which indicates that it scatters input data uniformly throughout the ciphertext and improves cryptographic security. The encryption schemes demonstrate high levels of randomness through both values, which proves their effectiveness in hiding original message content. The results show that both ciphers work well for IoT security, but AES provides marginally superior entropy-based protection.

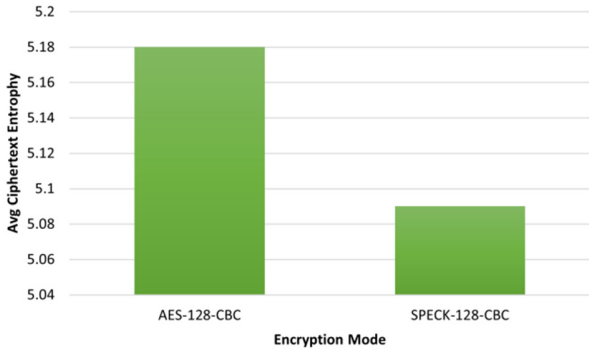

Fig. 4. Average ciphertext entropy (bits/byte) calculated using Shannon's formula.

The evaluation results were consolidated using a weighted scoring approach (Table II). Performance metrics were normalized based on their importance within the requirements of the IoT system and assigned weights. SPECK-128-CBC obtained a total score of 0.9931 exceeding the score of 0.9484 obtained by AES-128-CBC. These results demonstrate that SPECK-128-CBC delivers better performance in GSM-based IoT applications. The cryptographic strength of AES-128-CBC remains high, but SPECK-128-CBC offers better energy efficiency, making it an ideal solution for time-sensitive environmental monitoring systems with resource limitations.

V. CONCLUSION

The study developed a secure GSM-based IoT system that enables real-time monitoring of water quality in aquaculture installations. The system addresses essential requirements for real-time environmental monitoring and alerting in fish farming facilities by tracking critical parameters, including pH, DO, and T. The system implements encryption features on its IoT edge through lightweight symmetric cryptographic algorithms to maintain reliable and secure data transmission on public mobile networks. The encryption methods AES-128-CBC and SPECK-128-CBC were evaluated and compared. Data were collected through an experimental deployment that spanned seven days using a 15-minute logging interval. Weighted scoring was used to balance efficiency and security, weighting entropy at 40%, processing time at 30%, and memory efficiency and payload size at 30%. The SPECK encryption process took 209.64 µs on average, while decryption required 143.08 µs and resource consumption reached 8584 B of Flash memory and 306 B of RAM, which were less than the demands of AES. The encryption output of AES showed slightly higher ciphertext entropy at 5.18 bits/byte compared to 5.09 bits/byte obtained by SPECK. The weighted evaluation determined that SPECK as the preferred choice (0.9931) over AES (0.9484) for use in restricted GSM-based IoT systems. These evaluation results confirm that SPECK-128-CBC offers a superior performance profile over AES-128-CBC by achieving lower execution time and reduced memory usage, making it more suitable for deployment in resource-constrained GSM-based IoT environments.

This study provides essential insights into aquaculture operations that operate under restricted power supply, bandwidth, and processing capabilities. This study demonstrates a lightweight and secure IoT framework that brings scalability to embedded systems while offering a reproducible testing approach for cryptographic system efficiency. The proposed architecture serves as a base for future enhancements that will include secure actuator control mechanisms and dynamic key management strategies along with predictive models for smart aquaculture operations. Although this study focuses on performance-based evaluation of encryption schemes, future work will include a formal threat analysis to assess resilience against real-world attack scenarios in secure IoT aquaculture systems.

REFERENCES

[1] K. M. Ahmed, R. Shams, F. H. Khan, and M. Á. Luque-Nieto, "Securing Underwater Wireless Sensor Networks: A Review of Attacks and Mitigation Techniques," *IEEE Access*, vol. 12, pp. 161096–161133, 2024, https://doi.org/10.1109/ACCESS.2024.3490498.

[2] S. Jennings *et al.*, "Aquatic food security: insights into challenges and solutions from an analysis of interactions between fisheries, aquaculture, food safety, human health, fish and human welfare, economy and environment," *Fish and Fisheries*, vol. 17, no. 4, pp. 893–938, 2016, https://doi.org/10.1111/faf.12152.

[3] J. Gladju, B. S. Kamalam, and A. Kanagaraj, "Applications of data mining and machine learning framework in aquaculture and fisheries: A review," *Smart Agricultural Technology*, vol. 2, Dec. 2022, Art. no. 100061, https://doi.org/10.1016/j.atech.2022.100061.

[4] V. Rao and K. V. Prema, "A review on lightweight cryptography for Internet-of-Things based applications," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 9, pp. 8835–8857, Sep. 2021, https://doi.org/10.1007/s12652-020-02672-x.

[5] V. A. Thakor, M. A. Razzaque, and M. R. A. Khandaker, "Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities," *IEEE Access*, vol. 9, 2021, https://doi.org/10.1109/ACCESS.2021.3052867.

[6] Amrita, C. P. Ekwueme, I. H. Adam, and A. Dwivedi, "Lightweight Cryptography for Internet of Things: A Review," *EAI Endorsed Transactions on Internet of Things*, vol. 10, Mar. 2024, https://doi.org/10.4108/eetiot.5565.

[7] J. Soto-Cruz, E. Ruiz-Ibarra, J. Vázquez-Castillo, A. Espinoza-Ruiz, A. Castillo-Atoche, and J. Mass-Sanchez, "A Survey of Efficient Lightweight Cryptography for Power-Constrained Microcontrollers," *Technologies*, vol. 13, no. 1, Dec. 2024, Art. no. 3, https://doi.org/10.3390/technologies13010003.

[8] S. Aqeel, A. S. Khan, I. A. Abbasi, F. Algarni, and D. Grzonka, "Enhancing IoT security with a DNA-based lightweight cryptography system," *Scientific Reports*, vol. 15, no. 1, Apr. 2025, Art. no. 13367, https://doi.org/10.1038/s41598-025-96292-0.

[9] A. M. Rasheed and R. M. S. Kumar, "Lightweight Cryptographic Algorithms for Medical IoT Devices using Combined Transformation and Expansion (CTE) and Dynamic Chaotic System," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 4, 2024, https://doi.org/10.14569/IJACSA.2024.0150472.

[10] J. Khalid, "Lightweight Cryptography Algorithms for Internet of Things Enabled Networks. A Comparative Study," *International Journal of Innovations in Science & Technology*, vol. 6, no. 4, pp. 2124–2139, Dec. 2024.

[11] I. El Gaabouri, M. Senhadji, and M. Belkasmi, "A Survey on Lightweight Cryptography Approach For IoT Devices Security," in 2022 5th International Conference on Networking, Information Systems and Security: Envisage Intelligent Systems in 5g//6G-based Interconnected Digital Worlds (NISS), Bandung, Indonesia, Mar. 2022, pp. 1–8, https://doi.org/10.1109/NISS55057.2022.10085144.

[12] I. Makarenko, S. Semushin, S. Suhai, S. M. Ahsan Kazmi, A. Oracevic, and R. Hussain, "A Comparative Analysis of Cryptographic Algorithms in the Internet of Things," in *2020 International Scientific and Technical Conference Modern Computer Network Technologies (MoNeTeC)*, Oct. 2020, pp. 1–8, https://doi.org/10.1109/MoNeTeC49726.2020.9258156.

[13] S. Maitra and K. Yelamarthi, "Rapidly Deployable IoT Architecture with Data Security: Implementation and Experimental Evaluation," *Sensors*, vol. 19, no. 11, Jan. 2019, https://doi.org/10.3390/s19112484.

[14] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019, https://doi.org/10.1109/ACCESS.2019.2924045.

[15] P. Velmurugan, K. Senthil kumar, S. S. Sridhar, and E. Gotham, "An advanced and effective encryption methodology used for modern IoT security," *Materials Today: Proceedings*, vol. 81, pp. 389–394, Jan. 2023, https://doi.org/10.1016/j.matpr.2021.03.424.

[16] B. Li, Y. Feng, Z. Xiong, W. Yang, and G. Liu, "Research on AI security enhanced encryption algorithm of autonomous IoT systems," *Information Sciences*, vol. 575, pp. 379–398, Oct. 2021, https://doi.org/10.1016/j.ins.2021.06.016.

[17] M. A. Khan, M. T. Quasim, N. S. Alghamdi, and M. Y. Khan, "A Secure Framework for Authentication and Encryption Using Improved ECC for IoT-Based Medical Sensor Data," *IEEE Access*, vol. 8, pp. 52018–52027, 2020, https://doi.org/10.1109/ACCESS.2020.2980739.

[18] P. Satyanarayana, N. Sriramdas, B. Madhavi, A. M, N. V. Phani Sai Kumar, and V. Gokula Krishnan, "Enhancement of Security in IoT Using Modified AES Algorithm for IoT Applications," in *2023 International Conference on Sustainable Communication Networks and Application (ICSCNA)*, Theni, India, Nov. 2023, pp. 380–386, https://doi.org/10.1109/ICSCNA58489.2023.10370606.

[19] B. V. Sundaram, M. Ramnath, M. Prasanth, and V. J. Sundaram, "Encryption and hash based security in Internet of Things," in *2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*, Chennai, India, Mar. 2015, pp. 1–6, https://doi.org/10.1109/ICSCN.2015.7219926.

[20] J. Cecílio, A. Oliveira de Sá, and A. Souto, "Software-Based Security Framework for Edge and Mobile IoT," *Ada Lett.*, vol. 44, no. 1, pp. 95–99, Sep. 2024, https://doi.org/10.1145/3706601.3706618.

[21] M. Jahangir, M. J. Lee, B. H. Kwan, T. Kamal, T. Y. Chai, and W. S. Loh, "Improved Layered Architecture: Integration of IoT with Blockchain for Increased Security and Scalability," in *2024 5th International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, Bangkok, Thailand, Sep. 2024, pp. 1–7, https://doi.org/10.1109/AiDAS63860.2024.10730319.

[22] I. Al-Hejri, F. Azzedin, S. Almuhammadi, and M. Eltoweissy, "Lightweight Secure and Scalable Scheme for Data Transmission in the Internet of Things," *Arabian Journal for Science and Engineering*, vol. 49, no. 9, pp. 12919–12934, Sep. 2024, https://doi.org/10.1007/s13369-024-08884-z.

[23] N. Garcia, E. Hammad, and A. Farraj, "Soft-Trust Based Architecture for NextG IIoT/IoET Security, Authentication and Authorization," in *2023 IEEE Texas Power and Energy Conference (TPEC*, Feb. 2023, pp. 1–6, https://doi.org/10.1109/TPEC56611.2023.10078555.

[24] A. H. Al-Omari, "Lightweight Dynamic Crypto Algorithm for Next Internet Generation," *Engineering, Technology & Applied Science Research*, vol. 9, no. 3, pp. 4203–4208, Jun. 2019, https://doi.org/10.48084/etasr.2743.

[25] V. Srivatsan and V. Pathari, "Design and Development of a Simulation Environment for IoT Devices," in *2022 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, Thiruvananthapuram, India, Mar. 2022, pp. 207–211, https://doi.org/10.1109/SPICES52834.2022.9774116.