Wiley
International Journal of Distributed Sensor Networks
Volume 2025, Article ID 9639728, 21 pages
https://doi.org/10.1155/dsn/9639728

Review Article

Comprehensive Analysis of Lightweight Cryptographic Algorithms for Battery-Limited Internet of Things Devices

Nahom Gebeyehu Zinabu,[1] **Yihenew Wondie Marye**,[2] **Kula Kekeba Tune**,[1] **and Samuel Asferaw Demilew**[3]

[1]*Department of Software Engineering, College of Engineering, Addis Ababa Science and Technology University, Addis Ababa, Ethiopia*
[2]*Associate Professor in School of Electrical and Computer Engineering (SECE), Addis Ababa Institute of Technology (AAiT), Addis Ababa University (AAU), Addis Ababa, Ethiopia*
[3]*Department of Information Technology, College of Computing, Debre Berhan University, Debre Berhan, Ethiopia*

Correspondence should be addressed to Nahom Gebeyehu Zinabu; nahom.gebyehu@aastustudent.edu.et

Received 6 November 2024; Accepted 21 April 2025

Academic Editor: Pierre Leone

The growth of billions of devices functioning in resource-constrained situations in the Internet of Things (IoT) era poses serious security issues. As a result of their limited processing power efficiency, memory, and battery life, traditional cryptographic algorithms like elliptic curve cryptography (ECC), Rivest–Shamir–Adleman (RSA), and Advanced Encryption Standard (AES) are sometimes too resource-intensive for these devices. Lightweight cryptography has become a crucial field as a remedy, created especially to maintain security while maximizing energy efficiency and reducing resource usage. With an emphasis on contrasting well-known lightweight block ciphers and pointing out areas for future research, this paper examines the most recent developments in lightweight cryptographic methods. In order to synthesize insights into present issues and future goals, it explores cutting-edge methods that improve performance, security, and energy efficiency. Among the main trends that were covered were the trade-offs between resource limitations and security strength, hardware–software co-optimization, block and stream cipher optimization, and hybrid encryption techniques. The study's conclusion emphasizes how urgently more research is needed to create cryptographic solutions that can sufficiently safeguard IoT devices.

Keywords: battery-limited devices; block cipher algorithm; embedded system security; energy-efficient cryptography; IoT security; lightweight cryptography; low-power cryptographic algorithms and performance analysis

1. Introduction

The exponential growth of Internet of Things (IoT) devices, ranging from smart meters to medical implants, has heightened the need for robust security mechanisms [1]. However, these devices often lack the hardware resources of traditional computing systems, making the use of standard cryptographic algorithms impractical. Lightweight cryptography addresses this challenge by providing adequate security while minimizing resource consumption, such as efficiency, central processing unit (CPU) cycles, memory usage, and power [2, 3]. This review offers an overview of lightweight cryptographic algorithms, with a specific focus on those suited for block cipher operations in resource-constrained devices. As IoT devices increase, securing them with traditional methods like Advanced Encryption Standard (AES) or Rivest–Shamir–Adleman (RSA) is not feasible due to their computational and power demands. Instead, lightweight cryptographic algorithms are designed to offer better security while conserving critical resources like battery power, memory, and processing time. The study presents a review of improved methods for lightweight cryptographic encryption algorithms, emphasizing their relevance in constrained environments. The motivation for this work is the growing adoption of IoT devices and the increasing security challenges associated with their widespread use. It

emphasizes the need for cryptographic methods that balance security, performance, and resource consumption and presents the key research questions (RQs) guiding the review: What are the most common methods to improve lightweight cryptographic algorithms? How do these improvements affect performance in constrained environments? What are the evaluation criteria used to measure efficiency in lightweight algorithms? The objectives of this review are to identify the most recent advancements in lightweight cryptographic algorithms, analyze improvements in block and stream cipher designs, explore optimization strategies for resource efficiency without sacrificing security, and summarize challenges and propose future research directions.

The remaining sections of this paper are organized as follows: Section 2 presents a thorough analysis of related works, highlighting key contributions to the field, identifying current research gaps, and outlining the approaches this study is aimed at addressing. Section 3 describes the research methodology, including the design approach and the evaluation metrics used in the analysis. Section 4 provides the results and a detailed discussion of the findings in light of the study's objectives. This section focuses on analyzing and interpreting the performance of various lightweight cryptographic algorithms in terms of execution time, memory usage, and energy consumption. Section 5 concludes the study by summarizing the key findings, and Section 6 offers recommendations for future research in the area of lightweight cryptography for battery-constrained IoT devices.

2. Related Work

In this section, we looked at similar works and tried to cover all the lightweight cryptographic algorithms available until 2024. This work examines various existing cryptographic cipher algorithms in terms of security, efficiency, energy and power, hardware and software efficiency, throughput, and other evaluation criteria. It also covers lightweight block ciphers, lightweight stream ciphers, and lightweight hash functions.

2.1. Overview Cryptography. Cryptography has emerged as a key technique of data security with the growth of digital connection. Algorithms that protect sensitive data have been developed as a result of researchers' thorough study of encryption approaches. Among these, lightweight cryptography is notable for being a specific method made for low-resource devices, which sometimes have limitations like memory, processing power, and battery life. Even though they are strong, standard encryption techniques may be too taxing for these kinds of devices. We performed a thorough literature analysis, looking through books, journals, and conference proceedings to investigate this area. We paid special attention to studies that were published up to 2024. Our literature review led us to identify five essential parameters of a successful lightweight cryptographic algorithm [4, 5]:

1. Efficiency: measured by the times needed to encrypt and decrypt data.
2. Security: avalanche and diffusion properties are used to assess security.
3. Key generation: making sure that key expansion is robust and secure.
4. S-box design: enhanced for resistance to cryptanalysis, differential uniformity, and nonlinearity.
5. Resource utilization: examined in terms of memory usage, CPU usage, and throughput.
6. Randomness: making sure the cryptographic output is mathematically sound.

Battery-scarce IoT devices are smart devices with many sensors that send data to a base station for use in a variety of applications such as home automation, healthcare, surveillance, and environmental monitoring. Their growing popularity is fueled by their energy efficiency and low power consumption. However, these devices are powered mostly by batteries, which eventually reduce and must be replaced, causing considerable maintenance issues. This is especially important for sensors installed in difficult-to-reach locations, such as medical implants, remote monitoring systems, and structural health monitoring of buildings and bridges. In order to get over this restriction, scientists have looked at alternate energy sources that increase the lifespan of devices and lower maintenance needs. However, in order to provide safe data transmission and storage without depleting these devices' limited resources, lightweight encryption is essential.

Lightweight cryptographic algorithms are perfect for battery limited applications, for example, RFID tags, smart cards, and sensor nodes, because they are particularly developed to strike a compromise between high security and low computational and memory requirements. These devices' effectiveness enables them to carry out cryptographic operations without requiring major performance sacrifices. Lightweight encryption is crucial in the context of embedded systems and the IoT to protect data confidentiality and privacy while extending device life. There are three basic categories into which lightweight cryptographic ciphers can be generally divided: [5–9].

a. Hashing functions: these are used to ensure data integrity and authenticity.
b. Stream cyphers: these provide fast, real-time data encryption.
c. Block ciphers: encrypt fixed-size data blocks, ensuring structured security mechanisms.

Our primary focus will be on lightweight block ciphers, which are the foundation for many cryptographic algorithms.

2.2. Hash Function–Based Lightweight Algorithms. Cryptographic hash functions are cryptographic primitives that maintain data integrity and authentication. For secure communication to occur in a variety of networks and applications, data integrity and authenticity must be guaranteed. Requiring a lot of resources, current message authentication algorithms are not appropriate for systems with limited resources, such as real-time and IoT systems [10]. As a result, the following lightweight cryptographic hash function algorithms have been proposed by researchers.

Noura et al. [11] propose a lightweight keyed hash function scheme that reduces the total number of rounds to one, which is advantageous for resource-constrained systems [11]. The proposed solution attempts to achieve desired cryptographic properties, such as resistance against collision with low computation and complexity and message and key avalanche effects. Their security and performance analysis indicates that the recommended keyed hash function has low resource overhead and is immune to known security vulnerabilities. The following are some possible issues with the paper and recommendations for future development [11].

Security analysis: The paper should provide a more comprehensive security analysis of the proposed keyed hash function scheme. This could include a detailed discussion of potential vulnerabilities, threat models, and a comparison with existing secure hash functions to demonstrate the effectiveness of the proposed scheme.

Performance evaluation: While the paper briefly mentions performance analysis, a more in-depth evaluation of the computational performance, resource utilization, and comparison with existing hash function schemes would enhance the paper's credibility.

Real-world implementation: It would be beneficial to include a section discussing the practical implementation aspects of the proposed scheme. This could involve considerations for integration into existing systems, interoperability, and potential challenges in real-world deployment.

Cryptanalysis and attacks: The paper could benefit from a discussion of potential cryptanalysis techniques and attacks that could be relevant to the proposed scheme. Addressing these potential weaknesses and providing countermeasures would strengthen the paper's contribution.

Experimental validation: If applicable, the inclusion of experimental validation, such as implementation on real devices or simulation in a controlled environment, would provide empirical evidence of the scheme's effectiveness and practicality.

Comparison with state of the art: A more detailed comparison with state-of-the-art lightweight cryptographic algorithms and hash function schemes would help situate the proposed scheme within the current research landscape.

Addressing these potential shortcomings in future work would enhance the credibility, applicability, and impact of the proposed keyed hash function scheme based on the RC4 stream cipher.

In Madushan et al. [12], the major contribution of the review lies in its comprehensive analysis of the NIST lightweight cryptography finalists and their fault analyses. The authors provide a detailed overview of the specifications, structures, design primitives, security parameters, advantages, and disadvantages of the finalists. Additionally, the specific focus on fault analyses, including the application of different fault attacks and the vulnerabilities of lightweight ciphers to such attacks, adds significant value to the cryptographic community's understanding of the security implications for resource-constrained IoT devices. This review serves as a valuable resource for researchers, practitioners, and stakeholders involved in the development and deployment of lightweight cryptographic algorithms in IoT and other resource-constrained environments. However, the review identifies several open research problems that warrant further investigations. Specifically, the authors note the limited number of investigations of side-channel attacks on the NIST lightweight cryptography finalists and emphasize the need for future research to focus on analyzing these algorithms from the perspective of side-channel leakage and resistance against side-channel attacks. Additionally, the authors highlight the need to explore further fault models for specific ciphers, such as Elephant and Grain-128AEAD, as well as the importance of investigating quantum-safe lightweight cryptography for future standardization. These identified gaps in research present valuable opportunities for future studies to enhance the understanding of the security implications and resilience of lightweight cryptographic algorithms in resource-constrained environments.

2.3. Stream Cipher–Based Lightweight Algorithms. Noura et al. [13] suggested a stream cipher-based lightweight algorithm called LESCA stream cipher to provide a lightweight cryptographic solution with low computational complexity and minimal resource consumption. It is aimed at achieving a balance between security and performance, especially for power-constrained devices and real-time applications. The cipher utilizes dynamic key-dependent cryptographic primitives and a selective update process to enhance security without compromising efficiency. Additionally, it is aimed at achieving high throughput and error tolerance, making it suitable for noisy wireless channels. The main contribution of the LESCA stream cipher lies in its ability to provide a lightweight and efficient cryptographic solution for emerging systems, particularly in the context of IoT networks. By introducing dynamic cryptographic primitives, selective update processes, and flexible block lengths, LESCA attempts to balance security and performance. Its key contributions include enhancing security, reducing computational complexity, and improving error tolerance, all while maintaining high efficiency and flexibility. However, in the future, it would be important to conduct a thorough analysis of potential drawbacks or limitations of the LESCA stream cipher. Some areas of focus for this analysis could include resistance to advanced attacks, resource constraints, and standardization considerations. By conducting a comprehensive analysis of these potential drawbacks, researchers and practitioners can gain a deeper understanding of the practical implications and limitations of LESCA, leading to further refinement and improvement of the cipher for future applications.

According to [14], this survey has several popular stream ciphers that are recommended for securing data in IoT devices due to their efficiency, lightweight nature, and suitability for constrained environments. Here are some of the commonly recommended stream ciphers for securing data on IoT devices: The Fruit stream cipher has been highlighted in the survey for its good resistance to known attacks. It offers a balance between security and efficiency, making it a suitable choice for securing data in IoT devices [14]. Enocoro128 is another stream cipher known for its large throughput, which can be beneficial for IoT applications where data transmission speed is crucial. Its efficiency and performance make it a popular choice for securing data in constrained environments [14]. The F-FCSR stream cipher is recognized for its large throughput and efficiency, making it well-suited for resource-constrained devices like IoT sensors and smart devices. Its lightweight design and cryptographic capabilities make it a recommended choice for securing data in IoT ecosystems [15].

The WG stream cipher is known for its speed and efficiency, making it a suitable option for securing data in IoT devices with limited resources. Its fast encryption capabilities and lightweight nature make it a popular choice for applications where performance is critical. The Grain stream cipher is designed to be fast and efficient, making it a preferred choice for securing data in IoT devices. Its cryptographic strength and suitability for constrained environments make it a reliable option for ensuring data security in IoT ecosystems [15]. The MICKEY-128 stream cipher is recognized for its efficiency and cryptographic capabilities, making it a recommended choice for securing data in IoT devices. Its performance and suitability for resource-constrained environments make it a valuable cryptographic solution for IoT applications [15].

These stream ciphers offer a combination of security, efficiency, and performance, making them well-suited for securing data in IoT devices operating in constrained environments. By leveraging these popular stream ciphers, IoT devices can ensure the confidentiality and integrity of data exchanged within IoT networks while optimizing resource utilization and energy efficiency. However, according to [16], the survey "A Survey on Stream Ciphers for Constrained Environments" explores the use of stream ciphers to enhance the security of data in IoT devices operating in resource-constrained environments. It discusses the characteristics of lightweight stream ciphers, their advantages over block ciphers, and their suitability for IoT applications. The survey highlights popular stream ciphers such as Fruit, Enocoro128, F-FCSR, WG, Grain, and MICKEY-128, emphasizing their resistance to attacks, throughput, and efficiency for securing data in IoT devices. The paper suggests future research directions, including the development of new lightweight stream ciphers, security analysis, performance optimization, integration with IoT security protocols, real-world evaluations, and standardization efforts to advance the use of stream ciphers in securing IoT devices effectively.

According to [17], cloud services have revolutionized the storage and sharing of data, but they also pose security challenges. The analysis of popular cloud services like Wuala, DropBox, and Google Drive revealed their security methods and encryption algorithms. While AES provides high security but has vulnerabilities, RC4 offers speed but is prone to key-related issues. SHA1 is fast but lacks robust protection. An improved stream cipher based on RC4-128 was developed to enhance security without compromising performance. Continuous advancements in encryption algorithms are crucial for safeguarding data in cloud services. But, the study on the *Improved Secure Stream Cipher for Cloud Computing* includes a focus on a specific set of cloud services, potential oversight of all vulnerabilities in encryption algorithms, and a possibly restricted evaluation of the improved stream cipher. As future works for the study on the *Improved Secure Stream Cipher for Cloud Computing* involve exploring broader analyses of encryption algorithms, integrating advanced encryption techniques, investigating the impact of the improved stream cipher in various cloud computing environments, and staying updated on evolving cybersecurity challenges and encryption technologies [17].

2.4. Block Cipher Lightweight Algorithms. Sherine Jenny et al. [18] proposed the design of a compact S_box for resource-constrained applications. S-box, or substitution box, is an important component in many cryptographic algorithms. It is a mathematical function that maps a certain number of input bits to a certain number of output bits. S-boxes are used to add confusion and diffusion to the plaintext, making it more difficult for attackers to decipher the encrypted message. Robustness, balancing, differential uniformity, nonlinearity, linear approximation, algebraic complexity, fixed and opposite fixed points, and the bit avalanche criterion are among the attributes that make an S-box strong. The design of a compact S-box is specifically tailored for resource-constrained applications, such as those found in IoT devices. As such, the compact S-box is designed to be smaller in size and consume less power than a regular S-box. The compact S-box proposed in this article is [18] designed using a combination of Boolean functions and Galois field arithmetic, which reduces the number of gates required for implementation and improves its efficiency. While the compact S-box proposed in this article is specifically designed for use in lightweight block ciphers for resource-constrained applications, it is possible that it could be used in other applications as well. However, further research would be needed to determine its suitability for other applications [19].

Thakor et al. [8] proposed lightweight cryptographic algorithms for resource-constrained IoT devices. In this study, none of the LWC algorithms fulfill all the criteria of hardware and software performance metrics, but they perform at their best in the specified environment. However, new attacks are reported with the growth of new LWC algorithms, which shows an inevitable and never-ending research process. The war between cybersecurity experts and attackers always opens a door of opportunities for new research in the field of cybersecurity, especially lightweight cryptography.

Hossain et al. [20] proposed the design and development of a novel symmetric algorithm for enhancing data security

in cloud computing to secure and enhance the protection of data stored in the cloud. A hash value is also generated for authentication. The algorithm developed by the authors ensures data confidentiality, integrity, and authenticity for data stored in the cloud. But it works on text format data only, and key and hash code exchange is less secure and efficient.

Thabit et al. [21] proposed a new lightweight cryptographic algorithm (NLCA) for enhancing data security in cloud computing. A NLCA is proposed to improve cloud computing protection with low processing overhead and high performance. It is intended to improve data security in cloud computing environments. The primary goal of the NLCA is to encrypt data using a 16-byte (128-bit) block cipher and a 16-byte (128-bit) key. A symmetric-key algorithm is needed for the encryption process, and each encryption round always depends on mathematical functions to produce confusion and diffusion. Because of its quick data collection and processing times, it is even more useful in the field of cloud computing. However, the suggested work is comparable to AES's current output. It is intended to be used with a 16-byte (128-bit) block cipher.

Usman et al. [22] reviewed on lightweight encryption for the low-powered IoT devices. Since the large key sizes in the asymmetric algorithms are unsuitable for the IoT, the lightweight encryption algorithms typically use symmetric key algorithms. The encryption and decryption processes are complementary to one another when employing Feistel networks in symmetric key algorithms. This lowers latency, preserves memory and circuitry in the limited device, and reduces code size. The development of lightweight encryption algorithms is essential because the IoT's resource-constrained devices cannot use the most advanced encryption algorithms. On the other hand, the algorithm that was developed performed insignificantly.

Sawant et al. [23] discussed the "PRESENT block cipher algorithm" and its suggested design and execution. Lightweight and simple to implement in both hardware and software, the "PRESENT block cipher algorithm" uses substitution and permutation blocks of only 4 bits. Sixty-four bits are used for blocks, and either 80 or 128 bits are used for keys. For the present block cipher, there are a total of 31 rounds. When this algorithm is implemented, an encryption module specifically for FPGA data transmission is created, improving security. This guarantees that unauthorized parties cannot access the data and that it is protected during transmission. Furthermore, the algorithm can be implemented on some of the smallest FPGAs available on the market due to its lightweight nature. Because of its ability to balance security, effectiveness, and adaptability, the "PRESENT block cipher algorithm" is a good choice for a wide range of everyday encryption and decryption applications, from secure communication in contemporary digital ecosystems to the protection of personal data. On the other hand, comparing the algorithm's performance to other lightweight encryption algorithms may be a research gap that needs to be filled in the future. Future studies could therefore concentrate on assessing the algorithm's efficiency, security, and speed, as well as contrasting it with other lightweight encryption algorithms to identify its advantages and disadvantages. Furthermore, additional research could examine the algorithm's possible weaknesses and look into ways to strengthen its defenses against possible intrusions.

In June 2013, the National Security Agency (NSA) released the Speck family of portable block ciphers to the general public. Speck has been optimized for performance in software implementations, while Simon, Speck's sister algorithm, has been optimized for hardware implementations. Speck is an add-rotate-xor (ARX) cipher [13]. Unlike other lightweight block ciphers currently in use, Simon and Speck support a wide range of block and key sizes, which allows the cryptography to be precisely tailored to a given application. Thanks to the approach we took to their design, we are confident that Simon and Speck will continue to deliver exceptional performance on tomorrow's IoT devices [13].

In Seddiq et al. [24], its lightweight design ensures quicker encryption and decryption times with less computational overhead. The algorithm improves security robustness against possible attacks by combining the Feistel network and substitution–permutation network (SPN) structures, in addition to multiple rounds and key addition layers. The ability to select 10, 16, or 20 rounds freely allows for adaptability to various security needs. The algorithm's efficacy in delivering secure encryption is confirmed by evaluation metrics like implementation time tests, avalanche effect tests, and randomness tests. Because of its adaptability, it can be used in a variety of sectors where resource constraints necessitate secure data encryption. Overall, the algorithm is an effective encryption solution for protecting sensitive data while preserving peak performance because of its careful consideration of both speed and security. The article's suggested hybrid lightweight cipher algorithm is made to offer safe data encryption in settings with limited resources. It benefits a range of industries and applications, such as the following [25]:

IoT: This technology can be used to secure IoT devices that have limited resources by providing safe data transfer and communication. Mobile and wireless communication: This provides quick and effective encryption to improve data security in mobile devices and wireless networks. Healthcare: By safeguarding private patient data during transmission and storage, this sector enhances data security. Financial services: These provide confidentiality and integrity for financial transactions and customer data in financial institutions. Smart grids: These improve communication security in energy management by offering a secure encryption solution for smart grid systems. Industrial control systems: These strengthen cybersecurity in sectors that depend on control systems to safeguard vital data and infrastructure. Government and defense: By offering security features to safeguard classified information, this helps government organizations and the defense industry. Cloud computing: This provides safe data transmission and storage with confidentiality and integrity, improving data security in cloud environments. All things considered, the hybrid lightweight cipher algorithm finds extensive use in many sectors where effective encryption and data security are critical, particularly in

settings with constrained computational capacity. However, the limited scope of testing and real-world implementation on the hybrid lightweight cipher algorithm may need to be addressed in future research. The algorithm's performance and security robustness in real-world, diverse environments need to be further validated through extensive testing and deployment, even though simulations and theoretical analysis show promise. Future research should concentrate on carrying out extensive real-world testing, benchmarking against current ciphers, thorough security analysis, scalability assessment, and alignment with industry standards and cryptography guidelines in order to improve the algorithm's credibility and applicability. By taking care of these issues, the algorithm's practical implementation across a range of industries and applications can be validated and optimized, guaranteeing its security and dependability in real-world situations [25].

2.5. Lightweight S-Box Design Algorithms. In Aboytes-González et al. [26], the enhanced security of S-boxes resulting from the proposed method in the study can have several potential applications in real-world scenarios: improved cryptosystems, secure communication, network security, IoT security, blockchain technology, and data privacy. Generally, the application of enhanced S-box security can have a broad impact on various sectors where data security and privacy are critical, ultimately contributing to a more secure digital environment. But the paper does not extensively discuss the potential trade-offs or drawbacks of implementing the proposed method in practical cryptographic systems. It is essential to consider factors such as computational overhead, implementation complexity, and compatibility with existing cryptographic protocols when introducing new methods for enhancing S-box security.

According to [27], implementing traditional cryptography algorithms on resource-constrained IoT devices faces challenges such as limited memory, low processing power, small physical area, energy consumption, complexity, security risks, compatibility issues, and scalability concerns. Adapting these algorithms for IoT devices requires optimization, simplification, and customization to ensure efficient and secure operation within the device constraints. The novel 5-bit S-box design proposed in the article addresses the limitations of lightweight cryptography by offering resource efficiency, flexibility for various block sizes, enhanced security through a chaotic mapping technique, a comparative advantage over existing S-box designs, and an overall balanced solution to meet the requirements of resource-constrained IoT devices [27].

The research on the novel 5-bit S-box design for lightweight cryptography algorithms can enhance security for IoT devices in various industries such as IoT security, smart homes, healthcare, industrial IoT, transportation, logistics, retail, supply chain, and financial services. The applications include securing communication, protecting sensitive data, preventing cyber threats, ensuring privacy, and safeguarding critical infrastructure across different sectors. However, one limitation of the paper on the novel 5-bit S-box design for lightweight cryptography algorithms is the absence of real-world implementation and testing on a diverse range of IoT devices. Future research should focus on conducting real-world testing, scalability assessment, energy efficiency analysis, interoperability testing, robustness evaluation, performance benchmarking, and user feedback validation to enhance the credibility and applicability of the proposed 5-bit S-box design for IoT devices [27].

According to [28], the paper "Design of a Lightweight Cryptographic Scheme for Resource-Constrained Internet of Things Devices" introduces the Small lightweight cryptographic algorithm (SLA) specifically designed for IoT devices. SLA is structured as a SPN with 16 rounds and supports key lengths of 80/128 bits. It incorporates nonlinear S-box and linear permutation layers to ensure security and efficiency. The S-box in SLA enhances the complexity of the cipher design by introducing nonlinearity, which helps in thwarting attacks and improving robustness. However, "Design of a Lightweight Cryptographic Scheme for Resource-Constrained Internet of Things Devices" lacks detailed analysis or discussion on the potential vulnerabilities or weaknesses of the SLA. While the paper highlights the security features and resistance to certain attacks, a more in-depth exploration of possible weaknesses, such as side-channel attacks or implementation vulnerabilities, could provide a more comprehensive understanding of the algorithm's limitations. Additionally, the paper could benefit from discussing real-world implementation challenges or performance issues that may arise when deploying the SLA in practical IoT scenarios. Addressing these limitations would enhance the overall robustness and applicability of the proposed cryptographic scheme [28].

According to [29], the proposed S-box algorithm plays a crucial role in enhancing data encryption for network security in several ways: resistance to cryptanalysis, well-designed S-boxes are resistant to various cryptographic attacks such as linear and differential cryptanalysis. They are selected based on specific requirements to ensure that products can withstand rigorous scrutiny. Strengthening the key, S-boxes are integral in symmetric key ciphers for key mixing and expansion. They ensure that the secret key is applied in a complex and uncertain manner, enhancing the overall encryption process. Enhanced security layers, in ciphers like the AES, S-boxes contribute to multiple layers of security. The unique transformations of each S-box in AES make it robust against various attacks, adding to the overall security of the encryption. Confidentiality, S-boxes introduce nonlinearity into encryption processes, making it challenging for hackers to deduce the relationship between cipher text and plaintext without knowledge of the secret key. This feature helps maintain the confidentiality of sensitive data. Diffusion and avalanche effect, S-boxes play a role in diffusion and avalanche effects in cryptographic products. A small change in the key or plaintext results in a significant change in the cipher text output, making decryption more challenging for adversaries [29]. Generally, the S-box algorithm is a critical component in data encryption for network security, providing essential cryptographic strength and resilience needed to protect data during storage and transmission in the face of evolving cyber threats. As future

research directions, the authors suggest potential future research directions, such as optimizing computational performance, exploring resilience to quantum attacks, and integrating the novel S-box algorithm into existing cryptographic systems. These recommendations pave the way for further advancements in network security and encryption technologies [29].

2.6. Lightweight Key Generation Algorithms. Sittampalam et al. [30] proposed enhanced symmetric cryptography for IoT using a novel random secret key (RSK) approach. This paper proposes a novel RSK technique to enhance the security of symmetric LWC algorithms for IoT applications. The experimental results show that the RSK is more resistant to key-based attacks, uses fewer resources, and provides more effective protection for symmetric LWC algorithms in all IoT systems. Nevertheless, this work only addresses key generation, not the more power-intensive step of symmetric cryptography algorithms.

Sohel Rana et al. [31] proposed a novel neural network (NN)–based key scheduling method for lightweight block ciphers. A multilayer feed-forward NN with a single hidden layer and a nonlinear activation function that satisfies the Shannon confusion properties is the foundation of the suggested NN technique. This is an illustration of the ideal NN for the key scheduling process, consisting of four input, four hidden, and four output neurons. This architecture generates five unique keys from 64-bit input data. Sufficient diffusion is achieved by using nonlinear bit shuffling. With less power and memory usage, the 4-4-4 NN technique ensures that secure keys are generated with an avalanche effect of greater than 50%. Nevertheless, this algorithm's speed and performance fall short of what is needed for battery-scarred IoT devices.

2.7. Comparison of the Existing Algorithm. In this section, we provide a comparative analysis of selected lightweight cryptographic algorithms, emphasizing their structural and cryptographic characteristics.

The comparison is based on recent research contributions, with a focus on S-box size, algebraic degree, linear proximity, author information, and publishing source, along with the major insights from the literature summarized in Table 1.

This analysis aids in determining the strengths and limitations of existing techniques, as well as potential areas for development in lightweight cipher design for IoT devices with limited battery life.

According to Table 2, prior studies have focused on various aspects of lightweight cryptographic algorithms, including their implementation on constrained devices and specific improvements in security or efficiency. Several studies have focused on the applicability of lightweight cryptography in the IoT space. For example, Li et al. provided an overview of lightweight encryption algorithms specifically designed for low-power IoT systems [27, 36]. Their work, however, did not address recent innovations in hybrid encryption or hardware-software co-optimizations. Other researchers, such as Gupta et al., conducted a comprehensive review of lightweight block ciphers, focusing on optimization techniques such as reduced S-box complexity and optimized round functions [38]. While this work offered valuable insights into block ciphers, it lacked an evaluation of lightweight stream ciphers and energy-efficient public key algorithms, and Table 3 provides a summary of the literature review on lightweight cryptographic techniques.

2.8. Popular Lightweight Block Cipher Algorithms. Explain the fundamentals of lightweight cryptographic encryption algorithms. For example, the article discusses why traditional encryption methods (such as AES) are unsuitable for constrained devices and highlights key lightweight encryption standards such as PRESENT, SIMON, SPECK, KATAN, KTANTAN, AES, DESL, RECTANGLE, CLEFIA, HIGHT, CAMELIA, TEA, and TWINE.

The PRESENT lightweight cipher is an SPN network block cipher with a 64-bit block size and 80- or 128-bit keys. The PRESENT lightweight cipher is well-known for its simplicity and compatibility with hardware implementations [5].

SIMON and SPECK: The NSA developed SIMON optimized for hardware and SPECK optimized for software, both of which are lightweight block ciphers with block and key sizes ranging from 32 to 128 bits. They are highly adaptable and efficient across multiple platforms [35, 49, 50].

KATAN is designed for minimal hardware implementation, with a 32-bit block size and 80-bit key sizes. KTANTAN is a simplified version with a hardcoded fixed key [51, 52].

AES variants: Although the standard AES is computationally expensive, lightweight variants of AES are often used in constrained environments. Optimizations include reducing the number of rounds or simplifying key schedules [51, 52]. Table 4 shows the performance of popular lightweight block cipher algorithms.

3. Research Methodology

To evaluate the performance and usability of lightweight cryptographic algorithms for battery-constrained IoT devices, the following important assessment metrics are considered:

1. Memory usage: Memory consumption describes how much RAM (volatile) and ROM/Flash (nonvolatile) memory a cryptographic algorithm uses while it is running. Reducing memory utilization is essential because the majority of IoT devices have very little memory. This metric consists of the following:

   ✓ Code size: The size of the binary that is produced by the algorithm.

   ✓ Runtime memory includes state registers, key schedules, and temporary buffers that are utilized for encryption and decryption.

   ✓ Optimization objective: In order to facilitate deployment on microcontrollers such as ARM

TABLE 1: Comparison of existing lightweight cryptographic algorithms.

Algorithms	Recent work (year)	S-box size	Algebraic degree	Linear proximity	Authors	Publisher	Critics
DES (1976) [32]	N/A	6-bit input, 4-bit output	1	2	National Bureau of Standards (NBS)	US government	Simple, fixed S-box but vulnerable to differential cryptanalysis
AES (2000) [32, 33	2021: Enhancements in hardware implementations	8 bits	2	4	Rijmen, Joan, Vincent Rijmen, and Anton Shamir	Springer	Complex, nonlinear design, good resistance to attacks
PRESENT (2007) [34]	2023: Integration into lightweight IoT protocols	4 bits	1	3	Bogdanov et al.	Springer	Lightweight, designed for resource-constrained devices
KLEIN (2010) [34]	2022: Analysis of differential fault attacks	4 bits	1	4	Bogdanov et al.	Springer	Lightweight, similar to PRESENT, but different design
SPECK (2013) [35]	2020: Optimized implementations for various platforms	4 bits	1	4	Beaulieu et al.	Springer	Ultralightweight, designed for resource-constrained devices
TWIN (2012) [36]	2019: Hardware implementation with improved performance	4 bits	1	4	Keliher et al.	Springer	Lightweight, similar to PRESENT and KLEIN, but different design

TABLE 2: Summary of some research articles, proposed techniques, major contributions, and gaps Observed of lightweight algorithms.

Authors	Research article	Proposed techniques	Major contribution	Gap observed
Thakor et al. [27]	Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities	The proposed methodology and techniques involve a holistic approach that includes review, classification, performance evaluation, security analysis, and the identification of future research opportunities	The paper provides a comprehensive review and comparison of different LWC algorithms for resource-constrained IoT devices, highlighting their strengths and limitations	Actually, no limitation. But, it shows the strengths and weaknesses of the work done and the future directions
Anwar et al. [20]	Design and Development of a Novel Symmetric Algorithm for Enhancing Data Security in Cloud Computing	Hybrid cryptography model: The authors proposed a hybrid cryptography model for cloud data security, combining symmetric key (AES) and asymmetric key (hyperelliptic curve cryptography (HECC)) techniques	The paper provides a hopeful solution to the security concerns associated with cloud computing, offering a novel symmetric algorithm and hybrid cryptography and hashing techniques to enhance the security of data stored in the cloud environment	As cloud computing continues to expand, accommodating massive volumes of data and diverse workloads, it becomes crucial to ensure that the proposed security solutions can effectively scale and maintain optimal performance under such demanding conditions
Thabit et al. [21]	Enhanced Symmetric Cryptography for IoT Using Novel Random Secret Key Approach	Feistel and SP architectural approaches: The NLCA algorithm is based on a complex structure and a mixture of Feistel and SP architectural approaches, contributing to its security and reliability	The paper's major contribution is the development of a new lightweight cryptographic algorithm and its analysis in the context of cloud computing, providing a comprehensive approach to addressing data security in cloud environments	But, not focus on ensuring the scalability and adaptability of security measures to meet the evolving needs of cloud computing
Sohel Rana et al. [31]	A New Key Generation Technique Based on Neural Networks for Lightweight Block Ciphers	Design neural network-based key scheduling technique	Demonstrating the effectiveness of the proposed technique through experiments that show its computational efficiency, lower power consumption, and reduced memory usage compared to existing algorithms	One potential gap in this paper is the lack of a comprehensive comparison with other existing lightweight block ciphers Another potential gap is the limited evaluation of the proposed technique's security
Noura et al. [37]	LESCA: LightwEight Stream Cipher Algorithm for Emerging Systems	Development of a novel dynamic key-dependent lightweight stream cipher	The proposed cipher achieves high throughput and demonstrates significant performance enhancements when compared to AES and recent one-round cipher schemes	The paper could not compare it with other lightweight encryption algorithms to determine its relative strengths and weaknesses
Noura et al. [13]	Efficient and Secure Keyed Hash Function Scheme Based on RC4 Stream Cipher	Design a lightweight message authentication algorithm (MAA) based on a one-round compression function using the RC4 stream cipher	This reduction in the number of rounds compared to existing standardized algorithms such as CMAC, GMAC, and HMAC contributes to lower execution time, reduced latency, computing resources, and energy consumption	Security issue: Even though the paper addresses the security concerns associated with the use of the RC4 stream cipher, future research could focus on further strengthening the security aspects of the proposed scheme. This may involve conducting additional security analyses, exploring potential vulnerabilities, and considering alternative lightweight cryptographic primitives to enhance the overall security of the scheme

TABLE 3: Summary of literature review on lightweight cryptographic techniques.

Authors and publication year	Methods	Strengths	Limitations
Abd Al-Rahman et al. [24]	✓The proposed method has a hybrid structure that combines substitution–permutation network (SPN) and Feistel network designs, which are divided into three layers: Nonlinear, linear, and key addition. The algorithm is rigorously tested against NIST standards to ensure its robustness	✓The paper's key contributions include the introduction of a novel hybrid lightweight cipher that balances security and performance, a comprehensive evaluation of its effectiveness, and a framework for future lightweight encryption research	✓However, the study has limitations, such as a narrow scope of testing that may not cover all potential attack vectors, a lack of performance evaluation under extreme conditions, and insufficient comparative analysis with other lightweight algorithms
M.K. Saini and R.K. Saini [39]	✓Conducts a literature review to provide a broad overview of existing research, analyzes specific applications in sectors such as smart homes and industrial environments, and identifies various security threats *Internet of Things* (IoT) devices face, emphasizing the importance of strong security measures	✓The paper's strengths include a thorough overview of *Internet of Things* (IoT) security challenges, a focus on real-world applications, and the identification of critical security requirements such as confidentiality, integrity, and authentication	✓However, it has limitations, such as a lack of empirical data, the rapidly evolving nature of *Internet of Things* (IoT) technology, which may render some information obsolete, and a narrow focus on specific solutions to security challenges
Al-ahdal et al. [40]	✓It employs a comprehensive review methodology to analyze various lightweight cryptographic algorithms suitable for *Internet of Things* (IoT) devices	✓Its strengths lie in its thorough examination of the security challenges posed by *Internet of Things* (IoT), the detailed discussion of lightweight algorithms, and the emphasis on practical implementation metrics such as memory utilization, code size, and energy consumption	✓However, the limitations include a potential lack of empirical data to support the theoretical discussions and the challenge of comparing different algorithms due to variations in hardware implementations, which may affect the generalizability of the findings across diverse *Internet of Things* (IoT) environments
Sallam and Beheshti [41]	✓Survey and comparative analysis of lightweight cryptographic algorithms	✓Comprehensive overview of lightweight cryptographic algorithms ✓Addresses security challenges in *Internet of Things* (IoT) devices ✓Discusses performance metrics and design constraints	✓It focuses primarily on lightweight algorithms, potentially overlooking other cryptographic methods ✓May not cover the latest developments post-2018
Irkhede and Kumar [42]	✓Systematic literature review analyzing security challenges in *Internet of Things* (IoT) devices	✓Comprehensive review of current *Internet of Things* (IoT) security state ✓Identification of four security levels: Authentication, trust, access control, and data protection ✓Focus on real-world applications in industrial and consumer contexts	✓Lack of standardized security protocols for *Internet of Things* (IoT) devices ✓Resource constraints in *Internet of Things* (IoT) devices (computational processing, power, memory) ✓Rapidly evolving threat landscape may render findings outdated quickly
Suryateja et al. [43]	✓The paper comprehensively surveys various lightweight cryptographic algorithms specifically designed for *Internet of Things* (IoT) devices ✓It includes a detailed analysis of existing algorithms, their performance metrics, and categorization based on different taxonomies to aid new researchers in the field	✓The study provides a comprehensive overview of lightweight cryptographic solutions designed for resource-constrained *Internet of Things* (IoT) devices, addressing the critical need for security ✓Lightweight cryptography is crucial for maintaining device performance and ensuring data security	✓The paper may not cover all emerging lightweight cryptographic algorithms, as the field is rapidly evolving ✓Additionally, the performance analysis may be limited to specific metrics, which might not encompass all practical scenarios in real-world *Internet of Things* (IoT) applications

TABLE 3: Continued.

Authors and publication year	Methods	Strengths	Limitations
Azish et al. [44]	✓The authors present a bit-banding method for permutations in the PRESENT and GIFT SPN (substitution–permutation network) ciphers. This method is intended to optimize energy, power consumption, and execution time for lightweight cryptographic implementations, particularly in *Internet of Things* (IoT) devices	✓Efficiency: The proposed method is reported to be highly efficient in terms of energy consumption, power usage, and execution time, making it suitable for resource-constrained environments like the Internet of Things ✓Adaptability: The approach can be applied to various SbPN-based primitives, demonstrating its versatility in improving the performance of different cryptographic algorithms ✓Focus on lightweight design: The research addresses the specific needs of lightweight cryptography, which is critical for the heterogeneity of *Internet of Things* (IoT) devices	✓Optimization trade-offs: Higher optimization levels can make debugging difficult and complicate development ✓The method is effective for PRESENT and GIFT ciphers but may need additional validation for other ciphers ✓Complexity in design: The challenge of creating a "one design fits all" lightweight block cipher due to varying application requirements and constraints is recognized
Singh et al. [45]	✓The authors evaluate the software and hardware applicability of lightweight block ciphers and categorize them according to round function design, emphasizing important performance parameters for contexts with limited resources	✓The research classifies lightweight block ciphers by structure, offering insights into their architectures. It provides a comparative analysis of performance trade-offs and emphasizes implementation suitability, aiding developers in selecting ciphers for resource-constrained devices	✓The paper only offers a limited security analysis of the studied ciphers, lacks experimental validation, and might not cover the most recent developments in cryptography
Singh et al. [46]	✓The authors use modeling and optimization to improve efficiency and performance in KLEIN cipher architectures with scalar and pipelined designs for FPGA and ASIC platforms	✓Optimized KLEIN cipher designs with customizable implementations for both FPGA and ASIC platforms are presented in the study for high-speed performance on resource-constrained *Internet of Things* (IoT) devices	✓The study only looks at the KLEIN cipher, which limits its relevance to other *Internet of Things* (IoT) ciphers. It also prioritizes performance enhancement above a thorough security investigation of the proposed architectural modifications
Singh et al. [47]	✓The authors propose two CLEFIA encryption architectures that provide 52.95% and 117.22% improvements on Artix-7 FPGA, respectively: Round-based for increased throughput and pipelined for greater frequency. Both are implemented on ASIC and FPGA, and metrics such as NPCR, UACI, and PSNR are used to measure security and evaluate performance. Image encryption is also used to evaluate security	✓The proposed architectures optimize performance with notable throughput and frequency gains, providing strong security for *Internet of Things* (IoT) picture encryption while being appropriate for high-speed applications. The reliability of the encrypted photos is ensured by the study's thorough assessment of security parameters	✓Although the research is suited for resource-constrained contexts, it lacks specific hardware requirements, which could impact scalability, and it only examines the CLEFIA cipher, restricting its applicability to other lightweight ciphers
Pulkit Singh et al. [48]	✓The authors emphasize the KLEIN cipher while concentrating on lightweight cryptographic methods for the Internet of Things. On the Spartan and Virtex FPGA platforms, they suggest three hardware configurations for KLEIN, assessing performance in terms of throughput, frequency, and efficiency. Metrics such as correlation, NPCR, UACI, MSE, PSNR, and entropy are used to evaluate security	✓The proposed approach offers comprehensive insights into protecting resource-constrained *Internet of Things* (IoT) devices by concentrating on lightweight cryptographic algorithms like KLEIN. With practical implementation on popular FPGA platforms for real-world applicability, it offers a thorough assessment of security and performance	✓Applicability to other lightweight algorithms is limited by the KLEIN cipher's focus, and the lack of specific hardware requirements may have an impact on the scalability and adaptation of different *Internet of Things* (IoT) devices

TABLE 4: The popular lightweight block cipher algorithm performance.

Ciphers	Structure	Block size (bits)	Key size (bits)	Number of rounds	Application	Attack threatening cipher
AES [33, 53]	Substitution–permutation network	128	128, 192, or 256	10, 12, 14	Internet of Things (IoT), secure communications, smart cards	Side-channel attacks, considered very secure
DESL [54]	Feistel	64	56	16	Internet of Things (IoT), low-resource devices	Brute force
PRESENT [5, 55]	Substitution–permutation network	64 bits	80 or 128	31	RFID, Internet of Things (IoT) devices	Differential and linear cryptanalysis
SIMON [50]	Feistel	32, 48, 64, 96, 128	64, 72, 96, 128, 144, 192, 256	32–72	Low-power, constrained environments	Linear and differential cryptanalysis
SPECK [35]	ARX (addition, rotation, XOR)	32, 48, 64, 96, 128	64, 72, 96, 128, 144, 192, 256	22–34	Optimized for software, constrained devices	Similar vulnerabilities as SIMON
KATAN [16]	Nonlinear shift register	32, 48, 64 bits	80	254	RFID tags, constrained devices	Differential and linear attacks
KTANTAN [16]	Nonlinear shift register	32, 48, 64 bits	80 bits (fixed key)	254	Constrained devices (fixed key)	Linear and differential attacks
RECTANGLE [56]	Substitution–permutation network	64 bits	80 or 128	25	Internet of Things (IoT), RFID, low-power devices	Differential cryptanalysis
CLEFIA [57]	Feistel	128 bits	128, 192, or 256	18, 22, 26	Multimedia, mobile devices, communications	Linear cryptanalysis
HIGHT [58]	Feistel	64 bits	128	32	RFID, sensor networks, low-power devices	Differential cryptanalysis
TEA [59]	Feistel	64 bits	128	64	Embedded systems, software encryption	Related-key attacks
TWINE [36]	Feistel	64	80 or 128	36	Internet of Things (IoT), constrained environments	Differential cryptanalysis

Cortex-M or RISC-V processors, algorithms should be small and able to operate with a small memory footprint.

2. Energy efficiency: Power efficiency assesses the algorithm's energy usage while performing cryptographic operations. It is an essential measure for IoT technologies that run on batteries or use energy. Power efficiency can be expressed as follows:
   - ✓ Energy per operation (e.g., nanojoule/encryption or microjoule/decryption), which is typically assessed using tools like ARM Energy Probe or perf counters.
   - ✓ Typical voltage and frequency circumstances, including average current draw and execution time.
   - ✓ The algorithm's optimization objective is to maximize cryptographic performance while minimizing energy consumption.
3. Security level: The algorithm's resistance to different cryptographic attacks, such as differential and linear cryptanalysis and algebraic attacks, is measured by its security level.
   - ✓ Side-channel assaults (SCAs), like timing or power analysis
   - ✓ Key recovery and brute-force attacks: The S-box characteristics (such as nonlinearity and differential uniformity), key length, number of rounds, and the cipher's general structure all have an impact. Stronger protection is implied by a greater security level, although resource requirements may rise.
4. Utilization of resources: A more comprehensive measure that encompasses hardware complexity (such as the number of gates in hardware implementations), processing time, and instruction count is resource consumption. It shows how well an algorithm works when hardware and processing power are scarce. Important features include the following:
   - ✓ Execution time: In milliseconds or clock cycles, each encryption or decryption is completed.
   - ✓ CPU load: The proportion of processor usage when the system is operating.
   - ✓ Hardware: Gate equivalents (GEs) are used to measure this for ASIC or FPGA implementations.
   - ✓ Optimization objective: Use resources as little as possible without compromising security or energy efficiency.

3.1. Research Approach and Techniques Used. Describe the review procedure in depth. Data extraction techniques, inclusion/exclusion criteria, search strategy, and analysis tools are all included in this. The evaluation procedure was carried out methodically to guarantee the reliability and thoroughness of the results. The procedure included a number of crucial steps.

3.2. Search Strategy. This section explains how and where the relevant literature was searched. This typically includes academic databases such as IEEE Xplore, Springer Link, MDPI Digital Library, and Google Scholar. The search terms should focus on security, cryptography, lightweight cryptography, algorithms, energy efficiency, performance evaluation, etc.

3.3. Inclusion and Exclusion Criteria. Studies were chosen using precise inclusion and exclusion criteria to guarantee quality and relevance.

3.3.1. Inclusion Criteria. The following criteria must be met for inclusion:

- o. Research into lightweight cryptographic methods, particularly for IoT devices
- o. Articles that present benchmarks or empirical statistics (such as execution time, memory utilization, and energy consumption)
- o. English-language articles published in reputable conference proceedings or peer-reviewed journals
- o. Studies that assess defenses against side-channel or cryptanalytic attacks

Exclusion: Studies not relevant to lightweight cryptography, older works without modern relevance, and nonpeer-reviewed sources published before 2006.

3.3.2. Exclusion Requirements

- o. Research that only examines traditional, nonlightweight cryptography
- o. Articles that lack adequate benchmarking results or methodological details
- o. Editorials, articles with merely abstracts, publications in languages other than English, or repeated research

3.4. Quality Assessment. This outlines how the quality of selected papers is assessed. Factors include relevance to RQs, contribution to the field, study design and methodology, and impact factor of the journal or conference.

3.5. Key Challenges. This identifies the challenges encountered by lightweight cryptography [52]:

Energy efficiency: reducing power consumption while encrypting and decrypting

Processing power: achieving low latency while minimizing computational overhead

Memory consumption: reducing the size of encryption algorithms to fit into limited storage space

3.6. RQs. The main RQs guiding this review are as follows:

RQ1: What are the improved methods of lightweight cryptographic encryption algorithms presented in the literature?

RQ2: How do these improvements affect performance in battery-limited IoT devices?

RQ3: What evaluation metrics are used to assess the performance of battery-limited IoT devices?

4. Discussion and Results' Analysis

This section presents the review's findings, which are organized by key areas for improvement in lightweight cryptographic algorithms.

4.1. Block Cipher Enhancements. This reviews the literature on recent block ciphers designed for lightweight applications and discusses improvements such as the following:

Reduced S-box complexity: simplifying the S-box to reduce computation

Efficient key scheduling: optimizing key schedule algorithms for speed and low memory usage

Round function optimization: enhancing round functions to balance security and performance

Several papers proposed improvements to existing block ciphers like PRESENT, CLEFIA, and SIMON. These improvements concentrated on optimizing the S-box, round function, and key scheduling to reduce computational complexity and power consumption. For example, Xie et al. introduced an optimized version of the PRESENT cipher that achieved a 30% reduction in power consumption while maintaining security [44, 55].

4.2. Stream Ciphers for Lightweight Encryption. This investigates improvements in stream ciphers, with a focus on their suitability in resource-constrained environments. Examples include the Grain and Trivium families. In the field of stream ciphers, lightweight algorithms like Grain and Trivium have been improved by reducing state size and round count [60]. Martinez et al. proposed a hybrid approach that combines stream ciphers with block cipher-like properties, resulting in a balance of speed and security [60].

4.3. Lightweight Public Key Cryptography. Recent research has focused on lightweight public key algorithms, particularly elliptic curve cryptography (ECC), to address the resource constraints of IoT devices. Efficient curve selections and optimized implementations have made ECC viable for IoT applications, with several studies reporting significant improvements in execution time and memory footprint. Additionally, research has analyzed enhancements in lightweight ECC and RSA specifically designed for IoT. This includes novel techniques for faster key generation, encryption, and decryption in constrained environments [30, 61, 62].

4.4. Energy-Efficient Cryptography. This is aimed at reducing the energy consumption of cryptographic operations, such as hardware accelerations, utilizing cryptographic hardware modules to speed up operations and optimize software implementations, fine-tuning software algorithms for minimal power usage [52, 63].

Significance of energy-efficient cryptography is as follows.

IoT devices: IoT devices, such as smart thermostats, security cameras, and wearable health monitors, require energy-efficient cryptographic solutions to secure data without depleting their limited battery resources [63–66].

Wireless sensor networks (WSNs): WSNs consist of specially distributed sensors that monitor physical or environmental conditions. These networks require secure, energy-efficient cryptographic protocols to ensure data confidentiality, integrity, and authenticity over long periods, often in remote locations [63–66].

Mobile devices and wearables: Devices like smartphones and fitness trackers need energy-efficient cryptography to secure sensitive user data without quickly draining the battery [63–69].

RFID tags and smart cards: These devices typically operate with minimal power and need lightweight, energy-efficient cryptographic algorithms to provide secure authentication and data protection in environments like contactless payment systems [63–66].

Automotive and industrial IoT: Energy-efficient cryptography is essential for secure communications between connected cars or in industrial IoT systems, where embedded devices monitor and control machinery with limited power availability.

Challenges and future directions are as follows [63–66, 70].

Balancing security and efficiency: As cryptographic algorithms become lighter and more energy-efficient, ensuring they remain secure against emerging threats (including quantum attacks) is a challenge. There is a constant trade-off between minimizing energy consumption and maintaining strong cryptographic guarantees.

Postquantum cryptography: The development of energy-efficient cryptographic algorithms that are also resistant to quantum computing attacks is an area of ongoing research. Quantum-safe cryptosystems, like lattice-based cryptography, must be optimized for low-power devices to maintain energy efficiency.

Adaptable cryptography: Dynamic cryptographic systems that can adapt based on available energy resources are being explored. These systems would adjust the level of security or type of algorithm used based on the current energy constraints of the device.

4.5. Hybrid Encryption Methods. This section discusses the combination of multiple lightweight techniques (e.g., combining stream and block ciphers) to achieve both flexibility and security. A new trend identified in the review is hybrid cryptographic methods combining lightweight block ciphers and stream ciphers. This approach has been shown to offer enhanced flexibility and security without sacrificing performance. Hybrid schemes, such as the one proposed by Zhao et al., can dynamically adjust encryption strategies based on the available resources [71].

TABLE 5: Comparison of the lightweight block cipher algorithms based on different parameters.

Algorithms	Block size (bits)	Key size (bits)	Memory consumption	Power efficiency	Security level	Resource consumption
PRESENT [5, 55]	64	80, 128	Low	High	Medium	Low
SIMON [13, 50]	32–128	64–256	Very low	Very high	Medium	Very low
SPECK [13, 35]	32–128	64–256	Very low	Very high	Medium	Very low
KATAN [16]	32, 48, 64	80	Low	High	Medium	Low
KTANTAN [16]	32, 48, 64	80	Low	High	Medium	Very low
AES [72]	128	128, 192, 256	Medium	Medium	High	Medium
DESL [54]	64	56	Low	Medium	Low	Low
RECTANGLE [22]	64	80, 128	Low	High	Medium	Low
CLEFIA [23]	128	128, 192, 256	High	Medium	High	High
HIGHT [13]	64	128	Very low	High	Medium	Very low
TEA [24]	64	128	Low	High	Medium	Low
TWINE [21]	64	80, 128	Low	High	Medium	Low

TABLE 6: Comparison of different lightweight block cipher algorithm performance.

Algorithm	Memory	Power	Security	Resource
PRESENT	6.5	5.0	6.2	2.5
SIMON	4.0	5.5	7.3	1.2
SPECK	5.1	3.2	8.0	1.7
KATAN	6.5	5.4	9.2	3.3
KATYAN	8.6	6.8	9.2	1.1
AES	9.0	7.9	10.1	3.8
DESL	9.1	8.8	9.2	3.2
RECTANGLE	6.7	4.6	8.3	1.5
CLEFIA	5.7	5.6	7.2	1.5
HEIGHT	4.9	4.1	6.9	1.1
TEA	4.3	3.7	6.2	3.3
TWINE	5.1	4.4	7.1	2.9

4.6. Challenges and Research Opportunities. A significant challenge in lightweight cryptography is achieving maximum energy efficiency while maintaining security. Emerging research focuses on designing energy-efficient hardware and software implementations. The research opportunity is to investigate power consumption models and create hybrid techniques that incorporate energy-saving mechanisms while ensuring a high level of security.

4.7. Comparison of Lightweight Block Cipher Algorithms. The choice of lightweight block cipher depends on the specific use case and constraints of the IoT device. In Table 5, important characteristics of common lightweight block ciphers are compared.

Table 6 presents a comparison of lightweight block cipher algorithms, highlighting significant variations in memory, power, security, and resource use. Although AES has the strongest security (10.1), it is best suited for contexts where security is a top priority because it uses more memory (9.0), power (7.9), and resources (3.8). Strong security is also offered by DESL, KATAN, and KATYAN (9.2), with KATYAN being particularly resource-efficient (1.1). The memory and power consumption of SIMON, SPECK, and TEA are all minimal; SIMON has the least memory (4.0) and SPECK the lowest power (3.2), making them perfect for extremely limited IoT applications. Because they provide balanced performance, RECTANGLE, CLEFIA, and TWINE are flexible choices. The choice of cipher is determined by the particular application priorities, such as hardware simplicity, energy efficiency, or security.

Figure 1 compares 12 lightweight cryptographic algorithms on a scale of 1 (*low*) to 10 (*high*) based on four crucial factors: memory, power, security, and resource usage. Despite its minimal resource needs, AES performs well in terms of memory and power and receives the highest security grade. PRESENT, KATAN, and DESL are ideal for IoT devices with constrained resources due to their powerful memory and power efficiency. SPECK and SIMON offer good power performance but poor resource and security ratings. RECTANGLE and CLEFIA attain balance, but TWINE and TEA perform mediocrely overall. HEIGHT has the lowest ratings on the majority of metrics, making it unsuitable for applications that need to be secure or efficient.

Each cipher is designed for different priorities, balancing block size, key size, and efficiency, particularly for constrained environments like IoT devices. Lightweight block ciphers are designed to be suitable for resource-constrained devices such as IoT devices, RFID tags, and embedded systems. They prioritize efficiency in terms of memory footprint, power consumption, and computational cost while maintaining a reasonable level of security. This comparative analysis focuses on five prominent lightweight block ciphers: PRESENT, SIMON, SPECK, KATAN, and AES-128. The table provides key metrics: block size, key size, memory footprint, power efficiency, and security level.

Block size: The size of the data block that the cipher can encrypt or decrypt at once. A larger block size generally offers better security but also requires more computational resources. PRESENT: 64 bits, the smallest among the compared ciphers. SIMON, SPECK: variable block sizes from 32 to 128 bits, offering flexibility; KATAN: 32 bits, similar to PRESENT; and AES-128: 128 bits, the largest block size, providing the highest level of security.

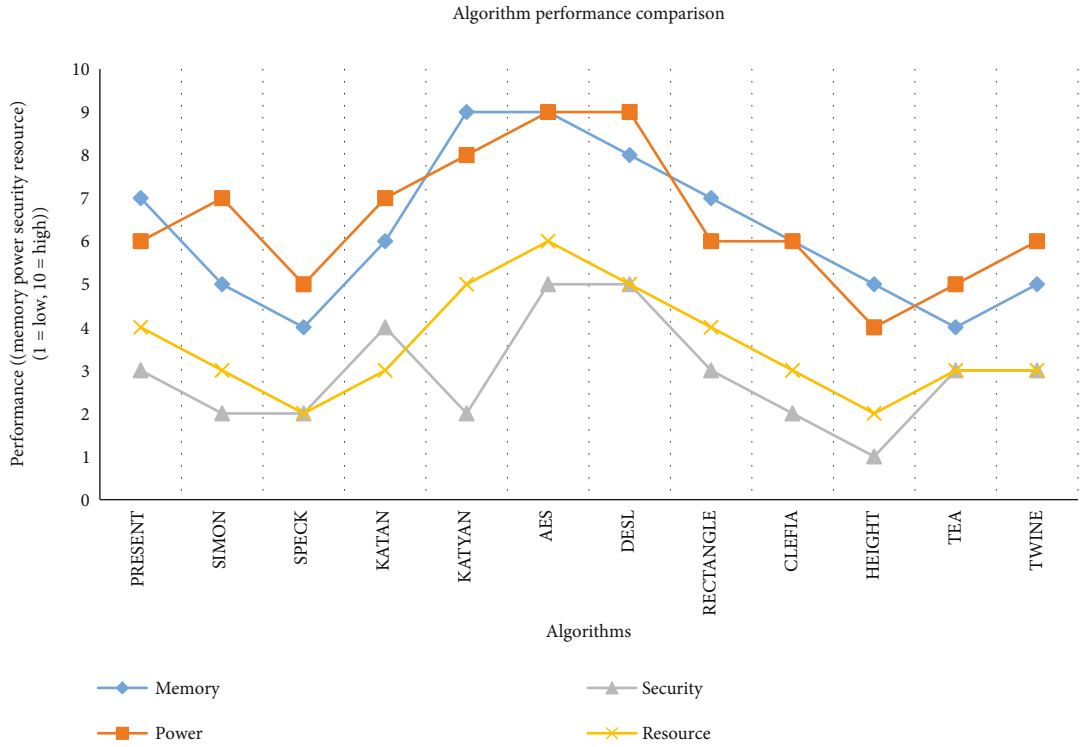

FIGURE 1: Lightweight cryptographic algorithm performance comparison.

Key size: The length of the secret key used for encryption and decryption. A longer key size generally increases security but also demands more computational power. PRESENT: 80 or 128 bits. SIMON, SPECK: 64–256 bits, offering a wide range of key lengths. KATAN: 80 bits and AES-128: 128 bits.

Memory footprint: The amount of memory required to implement the cipher. A smaller footprint is crucial for resource-constrained devices. PRESENT, KATAN: extremely low memory requirements, making them ideal for very small devices. SIMON, SPECK: low memory usage, suitable for most lightweight applications.

AES-128: Moderate memory consumption, compared to the others.

Power efficiency: The amount of power consumed during operation. Lower power consumption is essential for battery-powered devices. KATAN: very high power efficiency, making it suitable for energy-constrained environments; SIMON, SPECK, PRESENT: high power efficiency; AES-128: lower power efficiency compared to the other ciphers.

Security level: The resistance of the cipher to various attacks, such as brute-force attacks and cryptanalytic attacks. A higher security level is generally desired, but it often comes at the cost of increased computational resources.

SIMON, SPECK, AES-128: High-security level, considered to be resistant to known attacks. PRESENT, KATAN: Moderate security level, providing adequate protection for many applications but may be less suitable for high-security scenarios. The selection of a lightweight block cipher depends on the specific requirements of the application. For devices with extremely limited resources, PRESENT or KATAN might be the best choices due to their minimal memory footprint and power consumption. SIMON and SPECK offer a balance between security, efficiency, and flexibility. AES-128 provides the highest level of security but may be less suitable for resource-constrained environments. It is important to consider the trade-offs between security, efficiency, and memory footprint when choosing a lightweight block cipher. A thorough evaluation of the application's needs will help determine the most appropriate option.

TABLE 7: Search strategy and source usage in literature review (number of reviewed paper).

Source	Number of reviewed papers	%
IEEE Xplore	32	44.44%
Springer Link	9	12.50%
MDPI Digital Library	7	9.72%
Google Scholar	14	19.44%
Others	10	13.89%
Total	72	100%

4.8. Analysis of Search Strategy. To calculate the percentage, we used the following formula:

$$\text{Percentage\%} = \frac{\text{number of papers}}{\text{total papers}} * 100. \quad (1)$$

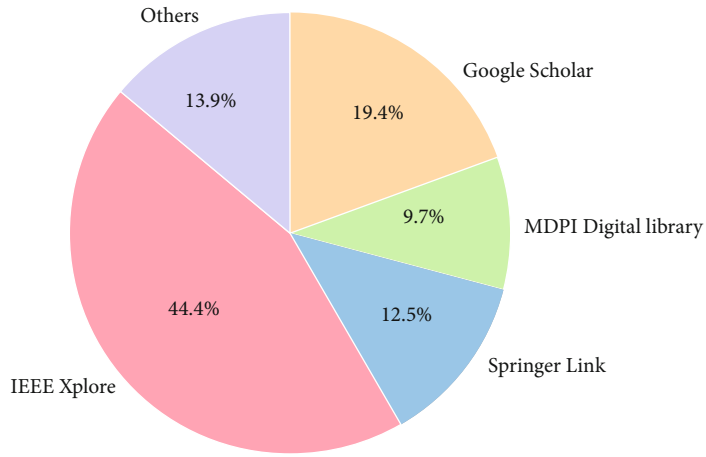

Figure 2: Search strategy and source usage in literature review.

The proportionate contribution of each source to the total number of reviewed papers in a study is calculated using Equation (1). It aids in measuring the contribution of each database or source to the reviewed literature. The percentage values in Table 7 are computed using Equation (1). For instance, 32 of the 72 papers were supplied by IEEE Xplore:

$$\text{Percentage\%} = \frac{32}{72} * 100 = 44.44\%.$$

Similarly, the same formula is used to derive contributions from Springer Link (12.50%), MDPI (9.72%), Google Scholar (19.44%), and other sources (13.89%). Table 7 breakdown makes it evident which databases had the biggest impact on the literature study. With almost half of the reviewed publications coming from IEEE Xplore, it is clear that this source is the most important and relevant for research on lightweight cryptography. By demonstrating a diverse body of literature, this percentage analysis not only assesses the distribution of sources but also improves the study's methodology's legitimacy and transparency.

Figure **2** demonstrates that IEEE Xplore is the most used source of literature, accounting for 44.4% of the review, indicating a strong focus on peer-reviewed, domain-specific research; Google Scholar (19.4%) provided a broad aggregator to supplement database-specific searches; Springer Link (12.5%) and MDPI Digital Library (9.7%) provided access to recent and multidisciplinary studies; and the remaining 13.9% came from other sources, such as Science Direct, Wiley, and arXiv, guaranteeing a thorough and well-rounded review.

4.9. Performance Analysis. This section investigates improvements in lightweight cryptographic methods in terms of performance, security, and resource efficiency, responding to the RQs posed in the Introduction section. It evaluates how much the reviewed cryptographic algorithm enhancements improve performance in terms of speed, power consumption, and memory usage, providing insight into their overall effectiveness.

The reviewed improvements significantly enhanced performance metrics such as energy consumption, memory usage, and computational overhead. For instance, optimizing the S-box structure in block ciphers consistently reduced memory usage by up to 20%, while hardware accelerations in ECC led to a 40% reduction in execution time.

4.10. Security Analysis. This paper discusses how these improvements maintain or enhance the security of lightweight encryption methods. Assess whether the reduction in complexity leads to potential vulnerabilities. Although performance enhancements were evident, certain enhancements raised concerns regarding the security compromises. The reduction of the complexity of ciphers could potentially expose them to cryptanalysis attacks, despite the majority of studies demonstrating that their methodologies maintained adequate security margins.

4.11. Trade-Offs. The improvements in lightweight encryption often involve trade-offs between speed, security, and resource consumption. A common trend observed in the literature is that higher performance often comes at the expense of reduced cryptographic strength, particularly in lightweight block ciphers. This identifies the trade-offs between security, performance, and resource efficiency. For example, increased speed might reduce energy efficiency or weaken cryptographic strength.

5. Conclusions

The increasing prevalence of IoT devices highlights the critical need for effective security measures tailored to resource-constrained environments. Traditional cryptographic algorithms, while robust, are not viable for these devices due to their high computational and power demands. As a response, lightweight cryptography has emerged as a vital alternative, emphasizing energy efficiency, minimal resource consumption, and adequate security. This review has provided an in-depth analysis of the current landscape of lightweight cryptographic algorithms, identifying major advancements, challenges, and potential research avenues.

It is evident that while significant progress has been made in optimizing block and stream ciphers, enhancing performance without compromising security remains a complex challenge. Emerging trends such as hybrid encryption methods and hardware–software co-optimizations offer promising directions for future development. The future of lightweight cryptography ultimately rests in its ability to balance the delicate trade-offs between security strength, performance, and resource constraints, leading to safe and effective IoT deployments. In addition, as IoT technology develops further, there is a pressing need for continued research focused on integrating lightweight cryptography with postquantum algorithms and optimizing existing protocols to counteract evolving security threats. This paper highlights the significance of addressing the identified gaps, exploring creative solutions, and fostering collaborative efforts within the research community to ensure that the security of IoT systems meets the demands of future applications.

6. Future Research Directions

The growing complexity and quantity of IoT devices are driving advancements in lightweight cryptography, and several exciting avenues for research are emerging to fill in these gaps and overcome current obstacles.

6.1. Postquantum Lightweight Cryptography. Future research should concentrate on developing lightweight cryptographic algorithms that are resistant to quantum attacks while remaining efficient in terms of power and computational requirements, as the emergence of quantum computing poses a serious threat to established cryptographic protocols. The challenge is to develop energy-efficient postquantum algorithms that specifically address the constraints of resource-constrained IoT devices.

6.2. Automated Testing for Security and Efficiency. For lightweight cryptography methods to be effective in practical applications, automated evaluation is essential. Research might be focused on creating automated benchmarking tools that test, simulate, and assess how well cryptographic algorithms operate in different environments. In IoT situations, using AI and machine learning to dynamically modify cryptographic parameters might improve response to shifting security requirements.

6.3. Biologically Inspired Cryptographic Algorithms. Investigating bio-inspired algorithms could result in novel solutions that naturally maximize resource efficiency and robustness. Future studies could look into using evolutionary algorithms and swarm intelligence concepts to create adaptive cryptography models that can react to threats on their own while using the fewest resources possible.

6.4. Enhancing Algorithm Robustness Against Cryptanalysis. The simpler structures of lightweight cryptographic algorithms make them susceptible to cryptanalytic attacks. In order to increase their resistance without appreciably raising energy demands, research is necessary. This may entail creating sophisticated mathematical techniques and hybrid encryption algorithms that combine symmetric and asymmetric cryptography in order to increase security.

6.5. Energy-Efficient Design Principles. Energy-efficient solutions must be given priority in lightweight cryptography strategies due to the low battery life of IoT devices. In order to ensure sustainable operation in battery-limited conditions, future initiatives should focus on enhancing encryption algorithms to lower energy consumption while maintaining security integrity. Researchers can make substantial progress in lightweight cryptography by tackling these important areas, opening the door for safe and effective IoT applications that can handle the difficulties presented by contemporary technological environments.

Data Availability Statement

This published article contains all of the data collected or analyzed during this investigation.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

The authors are grateful to Wiley Open Access for providing a full waiver of the article publication charge for this paper. There was no specific funding for the research, writing, or publishing of this work.

References

[1] 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* 9 (2021): 28177–28193, https://doi.org/10.1109/ACCESS.2021.3052867.

[2] A. Fotovvat, G. M. E. Rahman, S. S. Vedaei, and K. A. Wahid, "Comparative Performance Analysis of Lightweight Cryptography Algorithms for IoT Sensor Nodes," *IEEE Internet of Things Journal* 8, no. 10 (2021): 8279–8290, https://doi.org/10.1109/JIOT.2020.3044526.

[3] G. Spanos, K. M. Giannoutakis, and K. Votis, "A Lightweight Cyber-Security Defense Framework for Smart Homes," in *Proceedings of the 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)* (IEEE, 2020).

[4] S. Kotel, F. Sbiaa, M. Zeghid, M. Machhout, A. Baganne, and R. Tourki, "Performance Evaluation and Design Considerations of Lightweight Block Cipher for Low-Cost Embedded Devices," in *Proceedings of the 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)* (IEEE, 2016), https://doi.org/10.1109/AICCSA.2016.7945695.

[5] N. G. Zinabu and S. Asferaw, "Enhanced Efficiency of Advanced Encryption Standard (EE-AES) Algorithm," *American Journal of Engineering and Technology Management* 7, no. 3 (2022): https://doi.org/10.11648/J.AJETM.20220703.13.

[6] C. Ramakrishna, G. Kiran Kumar, and A. Mallikarjuna Reddy, "A Survey on Various IoT Attacks and its Countermeasures,"

International Journal of Engineering Research in Computer Science and Engineering 5, no. 4 (2018): 2320–2394.

[7] K. S. Mohamed, "The Era of Internet of Things: Towards a Smart World," in *Energy Systems in Electrical Engineering* (Springer, 2019), 1–19.

[8] D. Sehrawat and N. S. Gill, "Lightweight Block Ciphers for IoT Based Applications: A Review," *International Journal of Applied Engineering Research* 13, no. 5 (2019).

[9] M. A. Alsmirat, M. I. Daoud, and A. M. Elmisery, "Lightweight Cryptographic Algorithms for IoT: A Comparative Study," *IEEE Access* 9 (2021): 150372–150390, https://doi.org/10.1109/ACCESS.2021.3124692.

[10] M. A. Salama, H. M. El-Bakry, and H. M. Soliman, "A Review of Energy-Aware Security Mechanisms for IoT Devices," *IEEE Internet of Things Journal* 9, no. 13 (2022): 10456–10467, https://doi.org/10.1109/JIOT.2021.3128175.

[11] H. Noura, O. Salman, A. Chehab, and R. Couturier, "Efficient and Secure Keyed Hash Function Scheme Based on RC4 Stream Cipher," in *Proceedings of the 2020 IEEE Symposium on Computers and Communications (ISCC)* (IEEE, 2021).

[12] H. Madushan, I. Salam, and J. Alawatugoda, "A Review of the NIST Lightweight Cryptography Finalists and Their Fault Analyses," *MDPI, Electronics* 11, no. 24 (2022): 4199, https://doi.org/10.3390/electronics11244199.

[13] R. Beaulieu, D. Shors, J. Smith, S. T.-C. Bryan, and W. L. Wingers, *Simon and Speck: Block Ciphers for the Internet of Things* (National Security Agency, 2015).

[14] G. R. S. Qaid and N. S. Ebrahim, "A Lightweight Cryptographic Algorithm Based on DNA Computing for IoT Devices," *Security and Communication Networks* 2023, no. 1 (2023): 9967129, https://doi.org/10.1155/2023/9967129.

[15] S. A. Jassim and A. K. Farhan, "A Survey on Stream Ciphers for Constrained Environments," in *Proceedings of the 2021 1st Babylon International Conference on Information Technology and Science (BICITS)*IEEE).

[16] C. De Canniere, O. Dunkelman, and M. Knežević, "KATAN and KTANTAN A Family of Small and Efficient Hardware-Oriented Block Ciphers," in *International Workshop on Cryptographic Hardware and Embedded Systems* (Springer Berlin Heidelberg, 2009), https://doi.org/10.1007/978-3-642-04138-9_20.

[17] S. Gnatyuk, M. Iavich, V. Kinzeryavyy, T. Okhrimenko, Y. Burmak, and I. Goncharenko, *Improved Secure Stream Cipher for Cloud Computing* (State Scientific and Research Institute of Cybersecurity Technologies and Information Protection, 2020).

[18] R. Sherine Jenny, R. Sudhakar, and M. Karthikpriya, "Design of Compact S Box for Resource Constrained Applications," *Journal of Physics, Conference Series* 1767, no. 1 (2021).

[19] B. Pourmohseni, S. Bhasin, and F. Regazzoni, "An Overview of Lightweight Block Ciphers for IoT and Their Power Analysis Resistance," *IEEE Design & Test* 38, no. 2 (2021): 72–81, https://doi.org/10.1109/MDAT.2020.3044416.

[20] M. A. Hossain, A. Ullah, N. I. Khan, and M. F. Alam, "Design and Development of a Novel Symmetric Algorithm for Enhancing Data Security in Cloud Computing," *Journal of Information Security* 10, no. 4 (2019): 199–236, https://doi.org/10.4236/jis.2019.104012.

[21] F. Thabit and S. Alhomdy, "A New Lightweight Cryptographic Algorithm for Enhancing Data Security in Cloud Computing," in *Global transitions* (Elsevier B. V, 2021), 91–99.

[22] M. Usman, "Lightweight Encryption for the Low Powered IoT Devices," in *Computer Engineering* (Chosun University, 2020).

[23] A. G. Sawant, A. Deshpande, M. Daunde, K. Ghadage, and R. Pilane, "Design and Implementation of PRSENT Block Cipher Algorithm," *Journal of Emerging Technologies and Innovative Research* 6, no. 1 (2019).

[24] S. Q. Abd Al-Rahman, O. A. Dawood, and A. M. Sagheer, "A Hybrid Lightweight Cipher Algorithm," *International Journal of Computing and Digital Systems* 11, no. 1 (2022): 463–475, https://doi.org/10.12785/ijcds/110138.

[25] T. Y. Alhamdan and M. A. Alqarni, "Secure and Energy-Efficient Lightweight Cipher Suite for Wireless Sensor Networks," *IEEE Sensors Journal* 22, no. 19 (2022): 18743–18753, https://doi.org/10.1109/JSEN.2022.3187321.

[26] J. A. Aboytes-González, C. Soubervielle-Montalvo, I. Campos-Cantón, O. E. Perez-Cham, and M. T. Ramírez-Torres, "Method to Improve the Cryptographic Properties of s-boxes," *IEEE Access* 11 (2023): 99546–99557, https://doi.org/10.1109/ACCESS.2023.3313180.

[27] V. A. Thakor, M. A. Razzaque, A. D. Darji, and A. R. Patel, "A Novel 5-bit S-box Design for Lightweight Cryptography Algorithms," *Journal of Information Security and Applications* 73 (2023): 103444, https://doi.org/10.1016/j.jisa.2023.103444.

[28] N. Ibrahim and J. Agbinya, "Design of a Lightweight Cryptographic Scheme for Resource-Constrained Internet of Things Devices," *Applied Sciences* 13 (2023): 4398, https://doi.org/10.3390/app13074398.

[29] J. Alqahtani, M. Akram, G. A. Ali, N. Iqbal, A. Alqahtani, and R. Alroobaea, "Elevating Network Security: A Novel S-Box Algorithm for Robust Data Encryption," *IEEE Access* 12 (2024): 2123–2134, https://doi.org/10.1109/ACCESS.2023.3348144.

[30] G. Sittampalam and N. Ratnarajah, "Enhanced Symmetric Cryptography for IoT Using Novel Random Secret Key Approach Conference Paper," in *Proceedings of the 2nd International Conference on Advancements in Computing* (IEEE, 2020), https://doi.org/10.1109/ICAC51239.2020.9357316.

[31] M. Sohel Rana, R. H. Mondal, and A. H. M. Shahariar Parvez, "A New Key Generation Technique Based on Neural Networks for Lightweight Block Ciphers," *International Journal of Advanced Computer Science and Applications* 12, no. 6 (2021): https://doi.org/10.14569/IJACSA.2021.0120623.

[32] W. Stallings, *Cryptography and Network Security Principles and Practice Seventh Edition Global Edition* (Authorized adaptation from the United States edition, entitled Cryptography and Network Security: Principles and Practice, 8th edition, 2018).

[33] N. G. Zinabu and K. Adere, "Enhanced Image Cipher and Decipher Speed of Advanced Encryption Standard Algorithms for EmbeddedDevices" 2022, https://www.researchgate.net/publication/379121266.

[34] S. Banik, A. Bogdanov, and T. Isobe, "Tiaoxin-346: A Lightweight Authenticated Encryption Scheme," *Proceedings of the 30th USENIX Security Symposium (USENIX Security '21)* (ResearchGate, 2021), 2367–2384.

[35] R. A. Lustro, A. M. Sison, and R. Medin, "Performance Analysis of Enhanced SPECK Algorithm," in *Proceedings of the 4th International Conference on Industrial and Business Engineering* (Association for Computing Machinery, 2018), 256–264, https://doi.org/10.1145/3288155.3288196.

[36] W. Li, W. Zhang, G. Dawu, et al., "Security Analysis of the Lightweight Cryptosystem TWINE in the Internet of Things,"

KSII Transactions on Internet and Information Systems 9, no. 2 (2015): 793–810, https://doi.org/10.3837/tiis.2015.02.018.

[37] H. Noura, O. Salman, R. Couturier, and A. Chehab, "LESCA: LightwEight Stream Cipher Algorithm for Emerging Systems," *Ad Hoc Networks* 138 (2023): 102999, https://doi.org/10.1016/j.adhoc.2022.102999.

[38] D. Gupta, A. Pandey, and A. Sharma, "Revisiting Lightweight Block Ciphers: Review, Taxonomy, and Future Directions" 2020, IACR Cryptology ePrint Archive, Paper 2020/1181. Retrieved from https://eprint.iacr.org/2020/1181.

[39] M. K. Saini and R. K. Saini, "Internet of Things (IoT) Applications and Security Challenges: A Review," *NCRIETS – 2019 Conference Proceedings* 7, no. 12 (2019).

[40] A. H. A. Al-ahdal, M. M. H. Ali, A. Alahdal, A. S. M. Qaed, and G. A. Al-Rummana, "Securing the Internet of Things: A Review of Lightweight and Low-Power Cryptography Techniques," *Abhath Journal of Basic and Applied Sciences* 1, no. 2 (2022): 18–26, https://doi.org/10.59846/abhathjournalofbasicandappliedsciences.v1i2.441.

[41] S. Sallam and B. D. Beheshti, "A Survey on Lightweight Cryptographic Algorithms," in *Proceedings of the 2018 IEEE Region 10 Conference (TENCON)* (IEEE, 2018), https://doi.org/10.1109/TENCON.2018.8650352.

[42] T. Irkhede and S. Kumar, "Review on Challenges in IoT Device Security Approaches," *Mathematical Statistician and Engineering Applications* 71, no. 4 (2022): 9056–9067, http://philstat.org.ph.

[43] P. S. Suryateja, R. K. Ramesh, M. H. Anwar, and K. N. Sridhar, "A Survey on Lightweight Cryptographic Algorithms in IoT," *Cybernetics and Information Technologies* 24, no. 1 (2024): 21–34, https://doi.org/10.2478/cait-2024-0002s.

[44] M. T. Azish, I. S. Banday, and S. Banday, "An Efficient Permutation Approach for SbPN-Based Symmetric Block Ciphers," *Cybersecurity* 6, no. 42 (2023): https://doi.org/10.1186/s42400-023-00174-9.

[45] P. Singh, B. Acharya, and R. K. Chaurasiya, "A Comparative Survey on Lightweight Block Ciphers for Resource Constrained Applications," *International Journal of High Performance Systems Architecture* 8, no. 4 (2019): 250–270, https://doi.org/10.1504/IJHPSA.2019.104953.

[46] P. Singh, R. K. Chaurasiya, and B. Acharya, "Modelling and optimisation of High-Speed KLEIN Architectures on FPGA and ASIC Platforms for IoT Applications," *International Journal of Ad Hoc and Ubiquitous Computing* 42, no. 4 (2023): 207–225, https://doi.org/10.1504/IJAHUC.2023.130459.

[47] P. Singh, K. A. K. Patro, R. K. Chaurasiya, and B. Acharya, "Hardware-Software Co-Design Framework of Lightweight CLEFIA Cipher for IoT Image Encryption," *Sādhanā* 47, no. 4 (2022): 213, https://doi.org/10.1007/s12046-022-01994-0.

[48] P. Singh, B. Acharya, and R. K. Chaurasiya, "Lightweight Cryptographic Algorithms for Resource-Constrained IoT Devices and Sensor Networks," in *In Security and Privacy Issues in IoT Devices and Sensor Networks* (Academic Press, 2021), 153–185, https://doi.org/10.1016/B978-0-12-821255-4.00008-0.

[49] D. K. Nalla and A. Chattopadhyay, "An Efficient Lightweight Encryption Scheme for Energy-Constrained IoT Devices," *IEEE Transactions on Circuits and Systems I: Regular Papers* 70, no. 3 (2023): 1001–1014, https://doi.org/10.1109/TCSI.2023.3240123.

[50] K. H. Wan, F. Liu, and M. S. Oscar Dahlsten, "Learning Simon's Quantum Algorithm" 2018, arXiv, v1 quant-ph.

[51] C. C. K. Gaj, "Cryptographic Hardware and Embedded Systems – CHES 2009," in *Proceedings of the 11th International Workshop Lausanne, Switzerland, Lecture Notes in Computer Science series* (Springer, 2009).

[52] K. A. McKay, L. Bassham, M. S. Turan, and N. Mouha, *NISTIR 8114 Report on Lightweight Cryptography* (National Institute of Standards and Technology, 2017).

[53] J. Daemen and V. Rijmen, *AES Proposal: Rijndael* (National Institute of Standards and Technology (NIST), 1999), https://csrc.nist.gov/csrc/media/projects/cryptographic-standards-and-guidelines/documents/aes-development/rijndael-ammended.pdf?utm_source=chatgpt.com.

[54] F.-X. Standaert, G. Piret, G. Rouvroy, J.-J. Quisquater, and J.-D. Legat, "ICEBERG: An Involutional Cipher Efficient for Block Encryption in Reconfigurable Hardware," in *Fast Software Encryption: 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004. Revised Papers 11* (Springer, 2004), 279–298.

[55] X. Xie, Y. Zhang, and J. Wang, *Optimizing PRESENT Cipher for Resource-Constrained Environments: Improvements in Throughput and Energy Efficiency* (Springer Open, 2021), https://cybersecurity.springeropen.com.

[56] W. Zhang, Z. Bao, D. Lin, V. Rijmen, B. Yang, and I. Verbauwhede, "RECTANGLE: A Bit-Slice Lightweight Block Cipher Suitable for Multiple Platforms," *Science China Information Sciences* 58, no. 12 (2015): 1–15, https://doi.org/10.1007/s11432-015-5459-7.

[57] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, *14th International Workshop on Fast Software Encryption (FSE 2007)* (Lecture Notes in Computer Science Series, 2007).

[58] D. Hong, J. Sung, S. Hong, et al., "HIGHT: A New Block Cipher Suitable for Low-Resource Device," *International Association for Cryptologic Research* 4249 (2006): 46–59, https://doi.org/10.1007/11894063_4.

[59] D. J. Wheeler and R. M. Needham, "Implementation of Modified TEA to Enhance Security," in *International Conference on Information and Communication Technology for Intelligent Systems* (Springer International Publishing, 2017).

[60] J. Martinez, A. Smith, and R. Johnson, "A Hybrid Approach Combining Stream Ciphers With Block Cipher-Like Properties for Lightweight Encryption," *Journal of Cryptography and Information Security* 15, no. 4 (2020): 123–145.

[61] N. Koblitz and A. Menezes, "Elliptic Curve Cryptography: The Serpentine Course of a Paradigm Shift," *Journal of Cryptology* 22, no. 3 (2009): 236–264.

[62] A. K. Das, S. Kumari, M. Wazid, X. Li, and Y. Park, "Provably Secure and Lightweight Key Management Protocol for Wireless Body Area Networks," *IEEE Transactions on Dependable and Secure Computing* 17, no. 5 (2020): 870–882, https://doi.org/10.1109/TDSC.2018.2867773.

[63] T. Güneysu and A. Moradi, "Generic Side-Channel Countermeasures for Reconfigurable Devices," in *International Workshop on Cryptographic Hardware and Embedded Systems* (Springer Berlin Heidelberg, 2011).

[64] D. J. Bernstein, T. Lange, and P. Schwabe, "The Security Impact of a New Cryptographic Library," in *Progress in Cryptology–LATINCRYPT 2012: 2nd International Conference on Cryptology and Information Security in Latin America, Santiago, Chile, October 7-10, 2012. Proceedings 2* (Springer Berlin Heidelberg, 2012).

[65] S. Roy and D. Mukhopadhyay, "Lightweight Cryptography for IoT: A Review of Current Challenges and Solutions," *Internet of Things Journal* 5, no. 6 (2018): 4889–4910.

[66] Y. W. Law, S. Dulman, S. Etalle, and P. Havinga, *Assessing Energy-Efficient Security Protocols for Wireless Sensor Networks* (International Journal of Wireless Information Networks, 2006).

[67] I. Radhakrishnan, S. Jadon, and P. Balaji Honnavalli, "Efficiency and Security Evaluation of Lightweight Cryptographic Algorithms for Resource-Constrained IoT Devices," *Sensors* 24, no. 12 (2024): 4008, https://www.mdpi.com/1424-8220/24/12/4008.

[68] M. El-hajj, H. Mousawi, and A. Fadlallah, "Analysis of Lightweight Cryptographic Algorithms on IoT Hardware Platform," *Future Internet* 15, no. 2 (2023): 54, https://www.mdpi.com/1999-5903/15/2/54.

[69] H. Zhang, J. Zhang, and K. Ren, "Energy-Efficient and Secure Data Transmission for IoT Using Lightweight Block Ciphers," *IEEE Internet of Things Journal* 9, no. 6 (2022): 4350–4362, https://doi.org/10.1109/JIOT.2021.3114329.

[70] A. Kumar, V. S. Mohan, N. K. Gupta, R. P. Yadav, and S. Chandra, "Securing the IoT Ecosystem: ASIC-Based Hardware Realization of Ascon Lightweight Cipher," *International Journal of Information Security* 23 (2024): 3653–3664, https://doi.org/10.1007/s10207-024-00904-1.

[71] X. Zhao, Y. Zhang, W. Li, and Z. Wang, "Message forwarding for WSN-Assisted Opportunistic Network in disaster scenarios," *Journal of Network and Computer Applications* 137 (2019): 11–24, https://doi.org/10.1016/j.jnca.2019.04.005.

[72] N. G. Zinabu and S. Asferaw, "Enhanced Security of Advanced Encryption Standard (ES-AES) Algorithm," *American Journal of Computer Science and Technology* 5, no. 2 (2022): https://doi.org/10.11648/J.AJCST.20220502.13.