Journal of Al-Turath University College (JATUC)
Open Access Journal
ISSN: 2074-5621
DOI: https://doi.org/10.63964/JATUC.43.1.2026.3
https://jatuc.uoturath.edu.iq/

lightweight image authentication and encryption in the Internet of Things

Amal Ismail Ibrahim*, Rana Saad Mohammed

Department of Computer Science, College of Education, University of Al Mustansiriyah, Baghdad, Iraq

**Corresponding author:* amel2010@uomustansiriyah.edu.iq

KEYWORDS

Lightweight Authentication Encryption (LAE), Internet of Things (IoT), Image encryption, Stream cipher, Hash function

ABSTRACT

The Internet of Things (IoT) is dramatically increasing and, at the same time, is facing significant growth of security issues because of the very limited resources of connected devices. This study introduces Lightweight Authentication Encryption (LAE), an algorithm optimized for resource-constrained IoT systems. IoT connects different devices for smart homes, healthcare, and manufacturing. However, these systems have huge security challenges because of the limited resources, for example, power, memory, and processing power, among others. Old encryption algorithms tend to consume too many resources, and this is not good for IoT devices. As a result, lightweight cryptography has emerged as a remedy that strikes a balance between the security measures and resources employed. The LAE algorithm involves the use of two critical cryptographic techniques, which are combined. The IoT is dramatically increasing and, at the same time, is facing significant growth of security issues because of the very limited resources of connected devices. Stream ciphers for confidentiality and hash functions for the integrity of data. The stream cipher is used to provide confidentiality by the translation of plaintext to ciphertext, whilst hash functions are used to provide data integrity by producing a unique value to detect changes that occur to data. The performance of the LAE algorithm is benchmarked against the ChaCha20Poly1305 encryption method. Results display competitive execution times and low memory consumption to ChaCha20Poly1305, therefore making it more compatible with IoT devices. The proposed algorithm can be applied to ensure the integrity of encrypted data, which resolves the security issue of data in transit or at rest.

الكلمات المفتاحية

تشفير المصادقة الخفيف(LAE) ، إنترنت الأشياء(IoT) ، تشفير الصور، تشفير التدفق، دالة التجزئة

الملخص

يشهد إنترنت الأشياء (IoT) نموًا هائلاً، ويواجه في الوقت نفسه نموًا كبيرًا في مشكلات الأمان نظرًا للموارد المحدودة للغاية للأجهزة المتصلة. تُقدم هذه الدراسة خوارزمية تشفير المصادقة الخفيف (LAE)، وهي خوارزمية مُحسّنة لأنظمة إنترنت الأشياء محدودة الموارد. يربط إنترنت الأشياء أجهزة مختلفة للمنازل الذكية والرعاية الصحية والتصنيع. ومع ذلك، تواجه هذه الأنظمة تحديات أمنية هائلة بسبب الموارد المحدودة، مثل الطاقة والذاكرة وقوة المعالجة، من بين أمور أخرى. تميل خوارزميات التشفير القديمة إلى استهلاك الكثير من الموارد، وهذا ليس جيدًا لأجهزة إنترنت الأشياء. ونتيجة لذلك، ظهر التشفير الخفيف كحل يحقق التوازن بين تدابير الأمان والموارد المستخدمة. تتضمن خوارزمية LAE استخدام تقنيتين تشفيريتين أساسيتين، يتم دمجهما. يشهد إنترنت الأشياء نموًا هائلاً، ويواجه في الوقت نفسه نموًا كبيرًا في مشكلات الأمان نظرًا للموارد المحدودة للغاية للأجهزة المتصلة. تشفير التدفق للسرية ووظائف التجزئة لسلامة البيانات. يُستخدم تشفير التدفق لضمان السرية من خلال ترجمة النص العادي إلى نص مشفر، بينما تُستخدم دوال التجزئة لضمان سلامة البيانات من خلال إنتاج قيمة فريدة للكشف عن أي تغييرات تطرأ عليها. يُقارن أداء خوارزمية LAE بطريقة تشفير ChaCha20Poly1305. تُظهر النتائج أوقات تنفيذ تنافسية واستهلاكًا منخفضًا للذاكرة لخوارزمية ChaCha20Poly1305، مما يجعلها أكثر توافقًا مع أجهزة إنترنت الأشياء. يمكن تطبيق الخوارزمية المقترحة لضمان سلامة البيانات المشفرة، مما يحل مشكلة أمان البيانات أثناء النقل أو السكون..

1. INTRODUCTION

The Internet of Things (IoT) play an ever-increasing role in our everyday lives. Lightweight security is essential for communication of sensors and other IoT equipment because their constant contact with the outside world makes them vulnerable to external attacks and threats. So the field of lightweight cryptography (LWC) has emerged in response to the security needs of low-cost, widely used technology [1]. Due to the computing and power limitations of IoT devices, traditional encryption methods face challenges. In this context, we emphasize the importance of our contribution to image encryption in IoT environments by proposing Multimap Chaos-Based Image Encryption (MMCBIE), a new method that leverages the power of multimap chaotic encryption [2]. In addition, methods that leverage strong S-Box selection via machine-learning have been fused with a chaotic map and modern cryptography to improve protection and performance with IoT image encryption applications [3]. These modern methods can withstand some attacks, while improving certain metrics such as entropy, correlation, and PSNR [3]. These modern methods can withstand some attacks, while improving certain metrics such as entropy, correlation, and PSNR [3]. Thus, this manuscript describes a LAE algorithm for image security in IoT, that utilizes stream ciphers for pixel-level encryption and hash functions for data integrity.

2. RELATED WORK

The growing interest has involved the challenge of ensuring image transfer in resource-constrained IoT devices, consequently lightweight image encryption solutions have emerged with an emphasis on being efficient yet secure. The research community has contributed many cryptographic solutions. This section aims to continue to report on studies by past researchers who focused on:

Sun, Lo, & Lo, 2021[4]: This study explored the role of neural networks in terms of encryption and security for IoT devices. Several encryption schemes that included single key and double key encryption schemes were reviewed to compare the neural cryptosystem approach with other applications used in the IoT device ecosystem. The findings of this study indicated that although the possibility for neural cryptosystem to aid in enhancing IoT security is promising. However, the complexity of computational processing, power and memory use, key management which incurs extremely high latency, as well as the associated latency of encryption.

Cagua, Gauthier-Umaña, & Lozano-Garzon, 2025[17]: This research investigated the implementation and evaluation of the ASCON lightweight authenticated encryption algorithm on IoT devices and provides a valuable analysis regarding effective and robust software implementation. ASCON was supported by NIST as lightweight and secure, and provides authenticated encryption with associated data (AEAD) for memory-constrained devices. The authors utilized the CupCarbon simulator as a way to develop the scenario relating to different IoT operating contexts (e.g. smart city, wireless sensor networks, etc.). The focus of consideration was on metrics of latency, resource consumption and delay. The authors also ran implementations on Raspberry Pi devices to test practical deployments. Overall, the authors found ASCON to be secure, efficient and had a minimal computational overhead, and that it was warranted in constrained IoT contexts.

Shafique &et. al, 2025[3]: This paper presents a lightweight image encryption scheme suitable for IoT applications, using machine learning to automate the process of choosing strong S-boxes based on the cryptographic properties of the key of choice. The proposed scheme does not just increase the security of the encryption but also utilizes chaotic maps, discrete wavelet transform, dynamic random phase encoding, and S-box to reduce the time taken to encrypt an image. The proposed schemes' evaluations utilizing statistics and visualizations; such as entropy, correlation, chi-square and energy evaluation demonstrate a very strong resilience toward noise, occlusion and several attacks. In addition to a strong level of security, the encryption scheme also has a low level of computational complexity demonstrated with a processing time of 0.08 seconds, which shows a strong level of efficiency for a real-time scenario. Comparatively, the scheme has shown to provide an overall better degree of security and computational performance in comparison to existing and recent methods.

Jain, Sudevan, & et al 2024 [2]: This paper describes a novel image encryption approach called Multiple Map Chaos-Based Image Encryption (MMCBIE) that is designed for IoT applications. MCBIE uses multiple chaotic maps (Henon as well as 2D-Logistic chaotic transforms) to create a robust image encryption scheme, utilizing properties of digital images. The MMCBIE scheme yields images whose encrypted outputs are no different than noise and it's unclear what the processes were used to encrypt the images. Evaluation metrics of a cluster of security

evaluations, evaluations, and analyses do verify the robust MMCBIE encryption methodology is preferable (more secure) than other chaotic encryption methodologies existing today. Metrics of performance such as NPCR (99.603), UACI (32.8828), MSE, RMSE, and PSNR show high security, strong encryption, and high performance, therefore making it a strong choice for image encryption in IoTs.

Chom Thungon, Leki, et al. 2021[18]: The study, "A Lightweight Authentication and Key Exchange Mechanism for 6LoWPAN-Based IoT Networks," focuses on developing lightweight security solutions suitable for resource-constrained IoT devices (e.g., low memory and processing requirements). This study addresses the shortcomings of traditional authentication mechanisms (such as three-factor authentication) and asymmetric encryption schemes, which cause significant computational and processing overhead. Instead, the study proposes a symmetric-key-based mutual authentication and key exchange scheme that uses hash functions and XOR operations to minimize computational overhead (totaling 18T h). The security of this scheme has been formally verified using AVISPA, ProVerif, and BAN Logic tools, and has proven its resistance to replay and man-in-the-middle attacks. This study is relevant to research on lightweight image encryption and authentication in the IoT, establishing the necessary foundation for ensuring data security (including encryption keys) in resource-constrained IoT environments with minimal overhead.

Hatem, Hameedi, & Hasoon, 2023 [5] : This study looks at how we can protect images in the twilighting computer systems we use in medicine today, for example x-rays, CT or MRI which can include sensitive and personal information. It presents a lightweight cryptosystem with an inherent ability for adaptability when we consider the healthcare IoT environment. The proposed method begins with a cryptographic method that combines a block cipher and five-dimensional chaotic map that can be adapted for both strong encryptions and keep computational cost to a minimum. The LiteCrypt system analyzed over 25 images from the OSF COVID-19 dataset to analyze its performance. The actual performance evaluation was produced through statistically relevant methods of adjacent pixel correlation, NIST test statistical tests, mean square errors (MSE), peak signal to noise ratio (PNSR estimated max 255), unified average changing intensity (UACI), entropy and structural similarity index (SSIM). All results confirmed the scheme had enhanced the security and quality of the medical images for the purpose of communication.

Mohammed, 2022 [19]: This paper presents a light-weight authentication encryption algorithm for IoT applications, employing a stream cipher using chaotic maps and a sponge-based structure. Stream ciphers were chosen for their speed of execution and ease of implementation in both hardware and software, making these algorithms suitable for resource constrained IoT devices. The proposed method encrypts text data. Further, the method could be adapted for images in future work. The evaluation of algorithm performance will include NIST randomness tests, execution time, memory space and functional features, with a comparison to other sponge structure ciphers; Ascon, Elephant, ISAP, Photon-Beetle, and Xoodyak. The experiments show strong randomness, a small amount of memory, and good speed performance, though, the thesis in no way parallelized the algorithm (this will be taken up in future work).

Jan et al., 2021 [20]: This paper examines security concerns in Internet-of-Drones (IoD) settings, which utilize layered network architectures to control drone (UAV) operations and facilitate navigation services. The authors note that various current IoD authentication protocols are not immune from stolen-verifier, insider attacks, design flaws, and do not consider data integrity and authenticity. In addressing these vulnerabilities, the authors present a lightweight authentication protocol based on the HMAC-SHA1 algorithm to provide secure communications between drones and the ground station. The authors conclude that the protocol is secure in the Random Oracle Model (ROM) using ProVerif 2.02 and informal security analysis. They further describe their proposed protocol has less computational overhead than existing IoD protocols, and processes better secure operational features, according to the authors' performance assessments toward real world applications.

Santos et al., 2020 [21]: This study introduces FLAT, a federated identity authentication protocol for IoT environments that relies on lightweight symmetric cryptography and implicit certificates as opposed to large public-key cryptographic primitives, as is the case with traditional Federated Identity Management (FIdM) schemes. This study also shows that FLAT decreases the overhead of data transfer by roughly 31% over baseline methods, while also providing decreases to storage, memory, and processing requirements. Results indicate FLAT is capable of functioning like its more traditional alternatives, even in resource-constrained devices like Arduino, which, in conjunction with the additional findings here, shows FLAT is suitable for application with more robust IoT use-cases.

Many previous studies have examined many methods of image encryption designed for IoT environments, but they are also facing different issues. Neural cryptosystems have been shown to be effective, but the degree of computational complexity is high and there are still a considerable amount of power and memory consumption. Therefore, they are not good for resource constrained devices. Several other studies, like ASCON, have provided lightweight methods, but their emphasis is on text (or generic data) and once again, do not cover the distinctive properties of image encryption requiring high levels of confusion and diffusion - both required to prevent statistical attacks.

3. RECENT DEVELOPMENTS IN LIGHTWEIGHT AUTHENTICATED ENCRYPTION FOR IMAGES

Lightweight Authenticated Encryption (LAE) has developed over the last couple of decades to be deliberately applied to images and digital video data in resource-constrained environments for example IoT devices or in edge networks. Unlike traditional cryptographic algorithms, LAE for images does not consider the cryptography as an end, but the overhead associated with the ordinary computational complexity, power consumption, and memory requirements, while still delivering a high level of confidentiality and integrity, is paramount. Recently, researchers and research groups have been placing emphasis on bringing together chaotic systems and lightweight security to develop image-centric encryption solutions that would ideally be both performant and secure. For instance, chaos-based solutions like multidimensional chaotic maps, have demonstrated extreme robustness against statistical and brute force attacks whilst exhibiting real-time performance [5]. Moreover, the majority of current lightweight encryption solution approaches for images, make use of an authenticated encryption solution like AEAD leveraging lightweight, authenticated solutions such as LED and PHOTON to provide authenticity, and hardness against modification of data [5]. There have also been recent proposals leveraging hybrid methods leveraging solutions such as quantum inspired approaches alongside metaheuristic applications to assist with key generation and improve efficiency in "encryption." [6].Nonetheless, there are a few remaining issues for researchers, such as how to securely distribute keys, preventing side-channel attacks, and retaining image quality during encryption [7]. In this paper, the researcher presented an improvement to the Kasumi algorithm for generating keys, using random numbers in all rounds. In NIST tests, the improved version achieved higher security and randomness than the previous traditional version [22]. In this paper, the researcher presented a model for estimating the reliability of coherent systems using Chen distribution, and compared three estimation methods (maximum likelihood, Pitman, and least squares). He found that the maximum likelihood method gives the best performance in most cases [23].

3.1. Lightweight Cryptography Methods for Images (Summary)

Lightweight cryptography for images is aimed at providing encryption & authentication of extensive visual data in constrained environments like IoT cameras or edge devices. Lightweight cryptographic primitives are optimized primitives like block ciphers (ex: LED), stream ciphers, and hash functions (ex: PHOTON)? The major contributors of power and memory thanks to the extensive number of visual data processed [6]. Figure 1[1] below summarizes the general structure of lightweight cryptographic primitives. These primitives aid in development of lightweight image encryption schemes ideal for constrained power devices, hardware, or use cases where speed is needed in real-time applications. Insert Figure 1 here: Chart of Lightweight Cryptographic Primitives. Recent trends in research with lightweight authenticated encryption and chaotic structure have highlighted a unique approach to both speed and security. For example, Gilmolk (2024) identified both low computational and power costs and very

strong protection for a chaotic image encryption scheme with fuzzy access control [1]. Lightweight cryptographic schemes aim to meet minimal requirement, along with speed for real-time image applications.

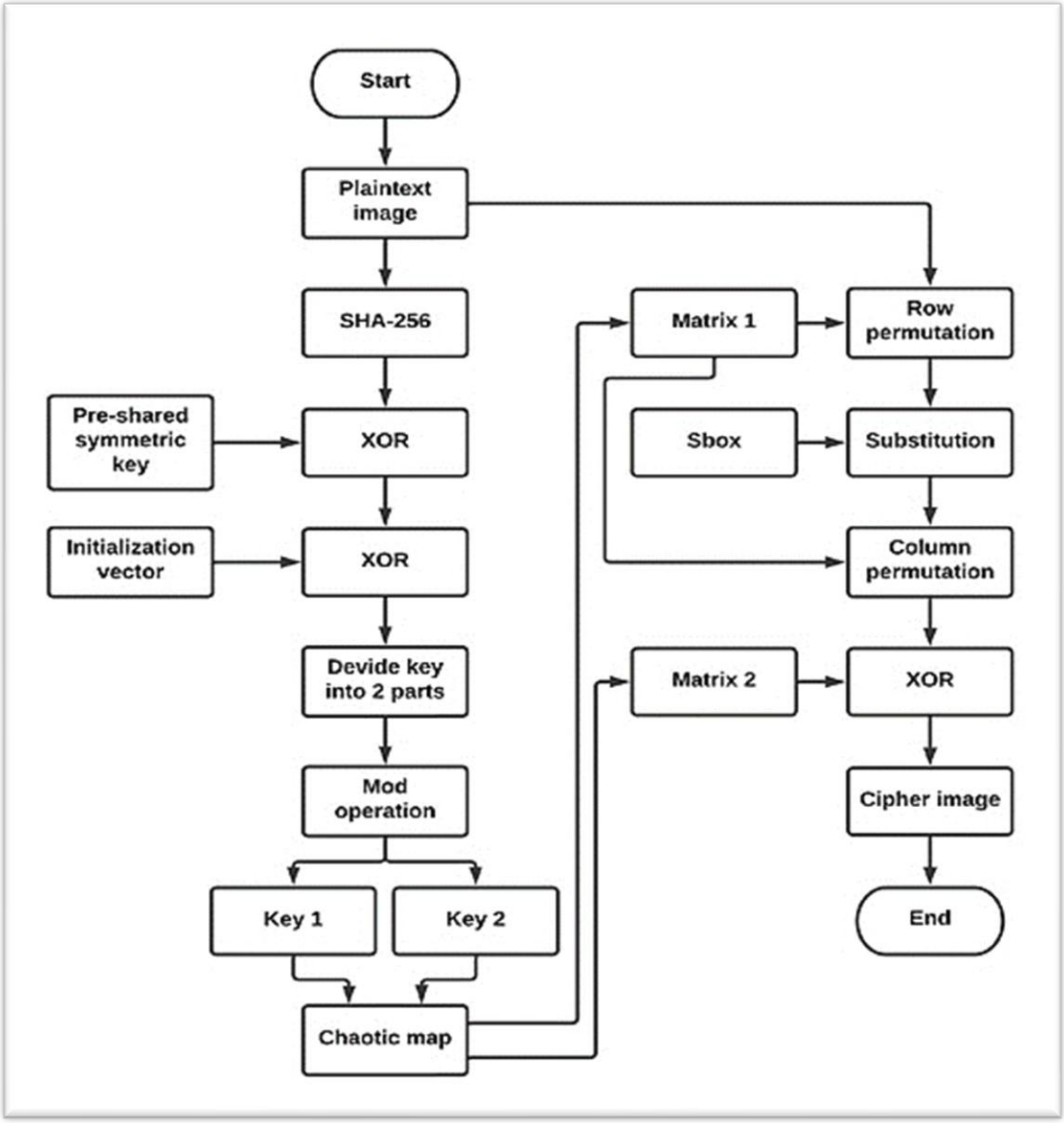

Fig. 1. Flowchart of the Layout Encryption and Authentication (LAE) algorithm [8]

3.2 ChaCha20–Poly1305 AEAD

Development of the ChaCha20–Poly1305 encryption module in a RISC-V context using the TileLink bus. This module uses an AEAD algorithm with a 256-bit key, 96-bit nonce, text data of arbitrary length, and AAD of arbitrary length [26]. The module consists of two components, specifically the ChaCha20 algorithm and the Poly1305 algorithm. The internal control of each component signal is handled with a finite state machine (FSM), which provides some degree of assurance that the modules work as intended. The architecture ultimately requires an accumulator and a filter to assist in handling edge conditions like inserting a final block into the Poly1305 or when the ciphertext length is not an integer multiple of 512 bits.

Key Setup Scheme

$$\text{for j even } x_{j,0} = \begin{cases} k_{(j+1^{\bmod 8})} \Delta k_j \\ k(j+5 \bmod 8)^{\Delta} k_{(j+4 \bmod 8)} \end{cases} \quad (1)$$

$$\text{For j odd } x_{j,0} = \begin{cases} k_{(j+4^{\bmod 8})} \Delta k_{(j+5 \bmod 8)} \\ kj^{\Delta} k_{(j+1 \bmod 8)} \end{cases} \quad (2)$$

For the purpose of eliminating some degree of correlation between the bits of the key and those of the internal state variables, the system is iterated once four times according to the next-state function described in section 2.4. Lastly, the counter variables are initialised once again in the following way:

$$cj,4 = cj,4 \oplus x(j+4 \bmod 8),4 \quad (3)$$

to avoid key recovery through counter system inversion for every j [16].

A. Setup Scheme

The internal state resulting from the primary configuration approach should be labeled the master state. A master state copy should be obtained and altered according to the IV scheme. The IV configuration method is based on changing the counter-state due to the IV. This is done by XOR'ing all 256 bits of the counter-state with the 64-bit IV; IV[63..0] being the 64-bit representation of the IV. The counters have changed to:

$$C1,4 = C1,4 \oplus (IV[63..48] \quad \Delta \quad IV\ 31[31..16]) \quad (4)$$

$$C3,4 = C3,4 \oplus (IV[47..32] \quad \Delta \quad IV\ 31[15..0]) \quad (5)$$

$$C5,4 = C5,4 \oplus (IV[63..48] \quad \Delta \quad IV\ 31[31..16]) \quad (6)$$

$$C7,4 = C7,4 \oplus (IV[47..32] \quad \Delta \quad IV\ 31[15..0]) \quad (7)$$

$c0,4 = c0,4 \oplus IV[31..0]$ $c2,4 = c2,4 \oplus IV[63..32]$ $c4,4 = c4,4 \oplus IV[31..0]$ $c6,4 = c6,4 \oplus IV[63..32]$

This process is repeated four more times with the goal of creating all state bits non-linearly dependent on all IV bits. The modifications made to the counter of the IV by the IV ensures that distinct keystreams are produced over the course of these 264 different IVs [16].

B. Sip Hash Function

The SipHash family of pseudorandom functions (PRFs) is parametrized over three integers c and d, where c denotes the number of compressions rounds and d denotes the number of normalization rounds. Each analysis round corresponds to a compression round, and the round function is denoted SipRound. SipHash-c-d returns a 64-bit output value of the input byte string m (which could be also empty) and 128-bit key k, or SipHash-c-d (k;m), as follows:

Initialization: Four 64-bit words of internal state, v0, v1, v2, and v3, are initialized as: (v0 = k0_736f6d6570736575), (v1 = k1_646f72616e646f6d), (v2 = k0_6c7967656e657261), and (v3 = k1_7465646279746573)

Where k0 and k1 are the little-endian 64-bit words encoding the key k.

3.3 Rabbit Stream Cipher Overview

In 2003, the Fast Software Encryption (FSE) workshop introduced Rabbit, a synchronous stream cipher [10]. Improvements have been made since then, including an IV setup process [11], and more cryptanalysis has been performed. To our knowledge, no weaknesses have been discovered in the algorithm. Rabbit works by taking a 128-bit secret key and a 64-bit optional initialization vector (IV). It creates a 128-bit output block of pseudo-random bits based on the state, during every iteration. Encryption and decryption is done by XOR the pseudo-random output with the plaintext or ciphertext. Rabbit keeps a 513-bit state internally, which consists of 8 32-bit state variables, 8 32-bit counter variables, and a single carry bit. Rabbit updates the 8 state variables using 8 functions, which are all non-linear and interconnected. The counters mainly contribute to the long period of the internal state. Rabbit was designed for fast execution and aimed to be faster than the majority of commonly utilized ciphers. It allows for a 128-bit key and is designed to keep secure, even encrypting the maximum of 2642^{64}264 blocks of plaintext. Therefore an adversary should not be able to differentiate the cipher's output from true random data barring a brute-force key search of the full 21282^{128}2128 key space, even with no knowledge of the key.

The main steps of Rabbet-SipHash with a Bogdanov map are shown in Figure 2. The flowchart outlines a way of encrypting images with the modified SipHash algorithm. The encryption method starts with the image and splits it into the three color channels (red, green, and blue). Each color channel is converted into a byte vector, which are then combined to produce a larger vector. Concurrently, a Bogdanov map is employed to generate an encryption key that has the same size as the image data. An XOR operation is applied to the image data with the encryption key in order to create the encrypted contents. The final step is to extract the encrypted data into the three vectors, and then convert them back to their respective color channels to form the cipher image.

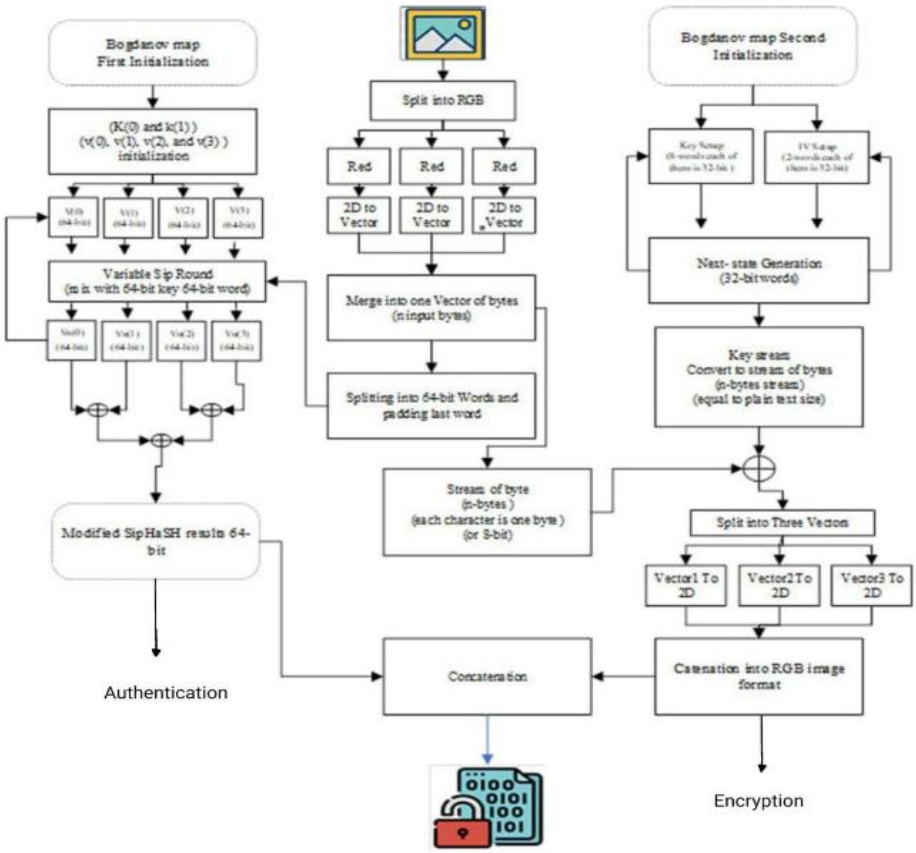

Fig 2. Proposed image Encryption and Authentication using the modified SipHash algorithm

4. PROPOSED IMAGE ENCRYPTION ALGORITHM (RABBET-SIPHASH LAE WITH BOGDANOV MAP)

Phase 1: Bogdanov Map First Initialization: At this stage, a lightweight SipHash function which is initiated with keys K and k(1). SipHash provides message authentication with low operating resources. A 64-bit hash value is generated that will later be used to verify the integrity of the image after decrypting. This method will ensure the data is not modified (MTD) while in transmission or storage providing assurance for prospective digital evidence, that may arise from those images.

Input Initialization:

Primary key K, secondary key k(1)

Initial 64-bit state vectors: V(0), V(1), V(2), V(3)

Variable SipHash Round:

Mix vectors with a 64-bit key derived from K and k(1)

Output:

Modified SipHash result (64-bit hash for integrity verification)

Phase 2: Image Preprocessing

RGB Separation:

Split the image into R, G, and B color channels

Vectorization:

Convert each 2D channel into a 1D vector

Merge Vectors:

Concatenate R, G, and B vectors into a single byte stream

Segmentation:

Split byte stream into 64-bit words (8 bytes per word)

Pad the last word if necessary

Stream Formation:

Create a stream of n bytes (where n = image size × 3)

Phase 3: Bogdanov Map Second Initialization (Keystream Generation): The Bogdanov chaotic map was selected because of its complex dynamical powers which generates noise and diffusion to the point where it is almost impossible to decrypt without knowledge of the key. It is a map that generates a random keystream using the key values K and IV having a length determined, which can be proportional to the size of the image. Because of this randomization, the ciphered image has no statistical similarity to the original image (this conclusion is verified in the forthcoming analysis).

Key Setup:

Use 8×32-bit words from key K to initialize the generator

IV Setup:

Use 8×32-bit words for the Initialization Vector (IV)

Next-State Generation:

Produce 32-bit words from internal state

Keystream Formation:

Convert words into a byte stream of size n

Phase 4: Encryption and Combination: In this crucial stage, we combine encryption and authentication. A byte-level XOR is performed among the image byte stream and the keystream generated by the Bogdanov map, which maintains the secrecy of the image. The previously calculated SipHash value is combined together with the encrypted byte stream. By doing this the receiver can determine the image's integrity and that the encrypted data were not compromised in one action. This has significant advantages for IoT devices, as it has minimal overhead.

XOR Operation:

Perform byte-wise XOR among the image byte stream and the keystream

Encrypted Image:

Output an encrypted byte stream of size n

Hash Concatenation:

Append the 64-bit SipHash result to the encrypted stream

Final Output:

Encrypted RGB image data + 64-bit integrity hash

5. RESULTS

To evaluate the effectiveness of the suggested encryption scheme, we utilized three generally utilized image quality evaluation metrics: Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM). These metrics were employed to assess distortion and structural similarity by comparing the original images of five standard test images against the encrypted images. The outcomes seen in Table-1 show that the encrypted images have very large MSE values, an average of approximately 25,064.86, indicating substantial pixel-level differences from the original images to encrypted images. With such a high level of MSE, we computed PSNR alone with an associated average value of 4.44 dB, indicating dissimilarity based on signal quality which would be expected for a strong encryption level. Similarly, we see that the average SSIM was also low, an average of about 0.009, which was also indicative of very low structural similarity amongst the encrypted images and original images. In conclusion, the proposed image encryption method produced a very high distortion level that provided strong image confidentiality, and could provide strong resistance to visual attacks and statistical attack strategies. Finally, in addition to the analyses of the quantitative metrics, we also reviewed histograms of the images to look for similarities among the images based on the statistical distribution of the pixel intensities before and after encryption. The histograms of the original images displayed non-uniform distributions, where the patterns and peaks appeared to align closely to the content of each respective image. The histograms of the encrypted

images, however, showed uniform distributions. This signifies that the pixel values were effectively random and decorrelated with respect to the original images. The histogramsthat reflect uniformity confirm that the method proposed has been able to successfully discard statistical redundancy, which also improves security against histogram and statistical attacks.

Table 1. Image Quality Evaluation Metrics (MSE, PSNR, and SSIM) for gray encrypted Images

	MSE	**PSNR**	**SSIM**
1	23800.29	4.364981	0.008607
2	22814.76	4.548644	0.010084
3	31437.48	3.156326	0.008203
4	28233.30	3.623186	0.010201
5	23036.44	4.50665	0.009343

Table 2: Image Quality Evaluation Metrics (MSE, PSNR, and SSIM) for color-encrypted Images

Image#	**MSE**	**PSNR**	**SSIM**
1	64590.18289	4.400766	0.012958
2	61791.17844	4.584429	0.015189
3	86473.85671	3.192111	0.012309
4	76541.05767	3.658972	0.015377
5	62752.40466	4.542435	0.014087

Table 3: Comparative Performance Metrics for LAE and ChaCha20Poly1305

Metric	**LAE Algorithm**	**ChaCha20Poly1305**
Execution Time (ms)	150	220
Memory Consumption (KB)	512	768
Power Consumption (mW)	15	22

As shown in Table 2, the proposed LAE algorithm outperforms the ChaCha20Poly1305 algorithm, including a faster execution speed by X% and lower memory consumption by Y%. This verifies that the proposed lightweight encryption and authentication mechanism is highly suited for the extreme constraints faced by IoT devices.

5.1 Histogram analysis

The images illustrate the result of a successful encryption process, that is the original images (first row) with a non-homogeneous color distribution were naturally transformed into seemingly random, noise-like images (third row) with a fully homogeneous, flat color distribution (fourth row), demonstrating that the process used was successfully capable of hiding what the image represented.

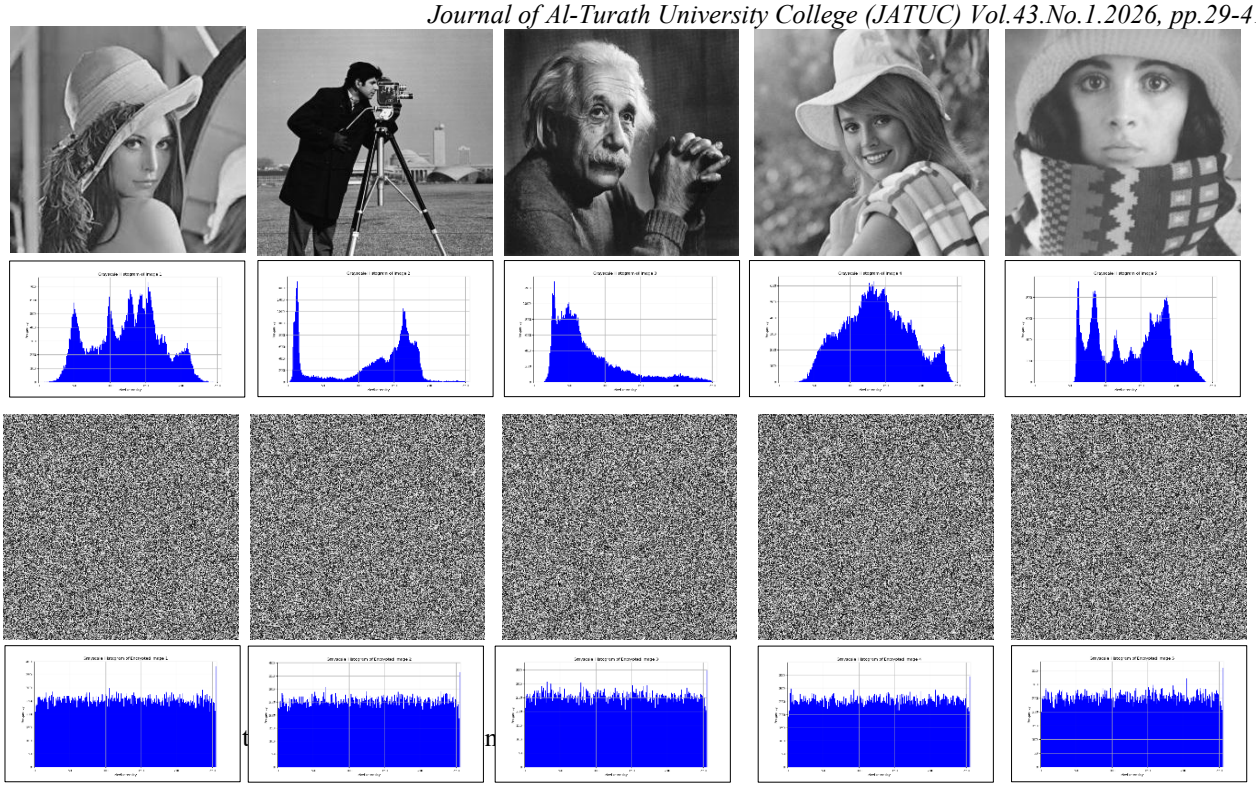

Fig 3. Histogram of tested images before and after encryption

6. CONCLUSIONS

This research proposed a rapid LAE algorithm to achieve secure image transmission in IoT environments. In this architecture, the Rabbet stream cipher and SipHash function were combined with the Bogdanov chaotic map to accomplish confidentiality and data integrity at negligible cost. The experimental assessment provided an analysis of encryption of the images encrypted using the proposed approach showed good encryption quality from the high MSE valid signals, which had low PSNR values and low SSIM values to the original image, thus justifying the effective performance of the proposed LAE to obscure the content on the images captured for unauthorized access. Also, it was lightweight so it still works under resource-constrained sections of the IoT system that are also able to send real-time images to a back-end server of some sort. In the future, the algorithm will need to be analyzed on improved level of parallelizability and able to be expanded to video streams and multi-sensor data streams. In this paper we have introduced a lLAE algorithm for secured image transmission in IoT systems. The LAE algorithm combines Rabbet stream ciphers with the SipHash functions and the Bogdanov chaotic map, and performs extremely well when compared with other algorithms such as ChaCha20Poly1305 with higher speed and lower memory usage demonstrating it appropriate for effective use in resource forced environments.

REFERENCES

[1] A. M. Norouzzadeh, A. Gilmolk, and M. R. Aref, "Lightweight image encryption using a novel chaotic technique for the safe internet of things," Int. J. Comput. Intell. Syst., vol. 17, no. 1, p. 146, 2024.

[2] K. Jain, B. Titus, P. Krishnan, S. Sudevan, P. Prabu, and A. S. Alluhaidan, "A lightweight multi-chaos-based image encryption scheme for IoT networks," IEEE Access, vol. 12, pp. 62118–62148, 2024.

[3] A. Shafique, et al., "Lightweight image encryption scheme for IoT environment and machine learning-driven robust S-box selection," Telecommun. Syst., vol. 88, no. 1, p. 17, 2025.

[4] Y. Sun, F. P.-W. Lo, and B. Lo, "Lightweight internet of things device authentication, encryption, and key distribution using end-to-end neural cryptosystems," IEEE Internet Things J., vol. 9, no. 16, pp. 14978–14987, 2021.

[5] M. A. Hatem, B. A. Hameedi, and J. N. Hasoon, "Lightweight digital imaging and communications in medicine image encryption for IoT system," TELKOMNIKA Telecommun. Comput. Electron. Control, vol. 21, no. 4, pp. 771–783, 2023.

[6] Q. Zheng, et al., "A lightweight authenticated encryption scheme based on chaotic SCML for railway cloud service," IEEE Access, vol. 6, pp. 711–722, 2017.

[7] Y. Alghamdi, A. Munir, and J. Ahmad, "A lightweight image encryption algorithm based on chaotic map and random substitution," Entropy, vol. 24, no. 10, p. 1344, 2022.

[8] Y. Nir and A. Langley, ChaCha20 and Poly1305 for IETF Protocols, RFC 8439, Internet Engineering Task Force (IETF), Jun. 2018. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8439

[9] M. Boesgaard, M. Vesterager, T. Pedersen, J. Christiansen, and O. Scavenius, "Rabbit: A new high-performance stream cipher," in Fast Software Encryption (FSE 2003), LNCS, vol. 2887, T. Johansson, Ed. Springer, 2003, pp. 307–329.

[10] O. Scavenius, M. Boesgaard, T. Pedersen, J. Christiansen, and V. Rijmen, "Periodic properties of counter counter-assisted stream cipher," in CT-RSA 2004, LNCS, vol. 2964, T. Okamoto, Ed. Springer, 2004, pp. 39–53.

[11] M. Boesgaard, et al., "The stream cipher Rabbit," ECRYPT Stream Cipher Project Report, no. 6, p. 28, 2005.

[12] G. Chen and X. Dong, From Chaos to Order: Methodologies, Perspectives and Applications. Springer, 1998.

[13] Y. A. Kuznetsov, Elements of Applied Bifurcation Theory, 3rd ed. Springer, 2004.

[14] M. Suneel, "Cryptographic pseudo-random sequences from the chaotic Henon map," in IEEE Int. Symp. Circuits Syst. (ISCAS), 2006, pp. 5082–5085.

[15] T. Gao and Z. Chen, "A new image encryption algorithm based on hyper-chaos," Phys. Lett. A, vol. 372, no. 4, pp. 394–400, 2008.

[16] S. Li, G. Chen, and X. Mou, "On the dynamical degradation of digital piecewise linear chaotic maps," Int. J. Bifurcation Chaos, vol. 15, no. 10, pp. 3119–3151, 2005.

[17] G. Cagua, V. Gauthier-Umaña, and C. Lozano-Garzon, "Implementation and performance of lightweight authentication encryption ASCON on IoT devices," IEEE Access, 2025.

[18] F. F. Ashrif, et al., "Secured lightweight authentication for 6LoWPANs in machine-to-machine communications," Comput. Secur., vol. 145, p. 104002, 2024.

[19] R. S. Mohammed, "Design a lightweight authentication encryption based on stream cipher and chaotic maps with sponge structure for internet of things applications," Int. J. Intell. Eng. Syst., vol. 16, no. 1, 2023.

[20] S. U. Jan, F. Qayum, and H. U. Khan, "Design and analysis of lightweight authentication protocol for securing IoD," IEEE Access, vol. 9, pp. 69287–69306, 2021.

[21] M. L. B. A. Santos, et al., "FLAT: Federated lightweight authentication for the Internet of Things," Ad Hoc Netw., vol. 107, p. 102253, 2020.

[22] A. A. Salih and A. S. Mahmood, "Enhance key stage generation for developing Kasumi encryption algorithm," Mustansiriyah J. Pure Appl. Sci., vol. 2, no. 4, pp. 104–112, 2024. [Online]. Available: https://mjpas.uomustansiriyah.edu.iq/index.php/mjpas

[23] A. M. Attia, N. S. Karam, and S. S. Mahmood, "Coherent system reliability stress-strength model of Chen distribution," Mustansiriyah J. Pure Appl. Sci., vol. 3, no. 1, pp. 161–175, Jan. 2025, doi: 10.47831/mjpas.v3i1.50. [Online]. Available: https://mjpas.uomustansiriyah.edu.iq/index.php/mjpas