Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 28 (2025), No. 5-A, pp. 1803–1812
DOI : 10.47974/JDMSC-2180

Evaluating lightweight encryption schemes for resource-constrained devices in wireless sensor networks

Prema Sahane [†]
Department of Computer Engineering
JSPM's Rajarshi Shahu College of Engineering
Tathawade
Pune
India

Monali Gulhane *
Department of Computer Science & Engineering
Symbiosis Institute of Technology
Nagpur Campus
Symbiosis International (Deemed University)
Pune 440008
Maharashtra
India

Nitin Rakesh [§]
Department of Computer Science & Engineering
Symbiosis Institute of Technology
Nagpur Campus
Symbiosis International (Deemed University)
Pune
India

S. Mohan Mahalakshmi Naidu [‡]
Department of Electronics & Telecommunication
International Institute of Information Technology (I²IT)
Pune 411057
Maharashtra
India

[†] *E-mail:* premasahane@gmail.com
* *E-mail:* monali.gulhane4@gmail.com (Corresponding Author)
[§] *E-mail:* nitin.rakesh@gmail.com
[‡] *E-mail:* mohans@isquareit.edu.in

Madhur Grover [@]
Chitkara Centre for Research and Development
Chitkara University
Baddi 174103
Himachal Pradesh
India

Vibhor Mahajan [#]
Centre of Research Impact and Outcome
Chitkara University
Rajpura 140417
Punjab
India

Abstract

Given the broad selection of remote sensor systems (WSNs) over diverse divisions, it is basic to prioritize information security, particularly in situations with restricted assets. This article gives a thorough appraisal of lightweight encryption plans outlined particularly for gadgets with constrained assets that are utilized in remote sensor systems. This investigate endeavors to assess the security, execution, and fittingness of three encryption schemes—Trivium, Jump, and TINYAES—in the setting of a recreated Web of Things (IoT) situation in farming. The think about utilizes Contiki-NG as the reenactment computer program to mimic a endless agrarian setting loaded with sensor hubs situated all through the areas. Security quality, asset utilization, throughput, inactivity, usage complexity, vitality effectiveness, and strength to imperatives are all assessed by means of thorough experimentation. Besides, novel parameters are presented to encourage a comprehensive comparison, such as vitality utilization per bundle, key setup time, and parcel misfortune rate. The comes about of this consider give important bits of knowledge that can educate the choice and execution of lightweight encryption strategies to defend information transmission in remote sensor systems with constrained assets, with a particular center on rural IoT applications.

Subject Classification: *68M01.*

Keywords: *Lightweight encryption, Resource-constrained devices, Wireless sensor networks, Security evaluation, Agricultural IoT.*

1. Introduction

Within the quickly advancing scene of remote sensor systems (WSN) and Web of Things (IoT) applications, guaranteeing the security of information transmission has risen as a basic concern. WSN contain interconnected sensor hubs able of detecting, preparing, and transmitting information over tremendous

[@] *E-mail:* **madhur.grover.orp@chitkara.edu.in**
[#] *E-mail:* **vibhor.mahajan.orp@chitkara.edu.in**

topographical ranges. These systems play a significant part in different spaces, counting natural observing, healthcare, and mechanical automation [1], [2]. The fundamental significance of information security in WSN cannot be exaggerated. As these systems collect and transmit touchy data such as natural information, quiet wellbeing records, or mechanical parameters, they are helpless to different security dangers, counting listening in, information altering, and denial-of-service attacks [3]. Besides, with the expansion of resource-constrained gadgets in IoT applications, such as sensors and actuators with constrained computational capabilities and vitality assets, conventional encryption plans may demonstrate unreasonable due to their tall computational overhead and vitality utilization. The rise of resource-constrained gadgets in IoT applications advance worsens the security challenges in WSN. These gadgets frequently work in resource-constrained situations, where components such as restricted preparing control, memory, and vitality accessibility posture critical limitations on the execution of strong encryption mechanisms [4], [5]. Thus, there's a squeezing require for lightweight encryption schemes that can give satisfactory security whereas minimizing computational overhead and vitality utilization. In any case, in spite of the developing request for lightweight encryption arrangements, there exists an inquiry about hole in terms of comprehensive assessment ponders that evaluate the execution and security of encryption plans in practical WSN scenarios. Existing thinks about frequently center on hypothetical investigations or restricted exploratory assessments, falling flat to capture the complexities of real-world sending scenarios [6]. This paper points to address this inquire about crevice by showing a comprehensive assessment of lightweight encryption plans custom-made for resource-constrained gadgets in WSNs. The essential goals of this think about are twofold to begin with, to assess the execution and security of chosen encryption plans in WSN, considering variables such as throughput, inactivity, vitality effectiveness, and strength to constraints; and moment, to supply bits of knowledge for selecting appropriate encryption plans in down to earth WSN organizations. The commitment of this paper lies within the advancement of a comprehensive assessment system that considers the one of a kind characteristics and limitations of resource-constrained gadgets in WSN.

2. Literature review

As it connects a vast network of devices to collect and share data, the Internet of Things (IoT) is growing. This interdependence necessitates strong security measures to protect sensitive data and system integrity. IoT security requires device authentication. IoT devices with limited resources often cannot use traditional cryptographic methods. Many studies have examined lightweight IoT authentication. Sadhukhan et al. [7] used ECC to authenticate remote IoT users. They tried to balance computational efficiency and security. Saqib et al. [8] proposed a lightweight three-factor authentication mechanism for critical IoT applications. User authentication was based on low-cost cryptographic primitives and pre-established shared secrets. Darbandeh et al. [9] suggested a

key agreement and lightweight user authentication scheme for WSN, which named IoT network. This was applicable for their demand for resource limited sensor nodes, because of a low computation and communication overhead. Alsaeed et al. [10] study lightweight cryptography schemes for an IoT-compatible use. In their comprehensive work on lightweight cryptography for IoT, investigate fog computing, block-chain, and lightweight group authentication for the enormous IoMT deployments, with emphasis on scalability and resource efficiency. Rao et al. [11] stress the relevance of memory budget, processing capability, and power consumption requirements. Itoo et al. [12] have proposed a lightweight secure privacy-preserving authentication and key exchange scheme for smart agriculture monitoring systems. They are designed with secure communications, low latency, and low-energy consumption in mind. Elliptic Curve Cryptography is a known lightweight cryptographic technique which has smaller key size compared with the public key cryptography. Elhoseny et al. [13] used ECC in a secure WSN data routing schema to show its efficacy in resource-constrained environments.

3. Methodology

A. *Selection of Encryption Schemes*

A.1. Trivium

Trivium is a stream cipher designed for lightweight cryptographic applications, especially suited for hardware implementations. This process results in the generation of a keystream consisting of bits that can be XORed with plaintext to make ciphertext or with plaintext to produce plaintext. Trivium operates by initializing three shift registers and then iteratively updating their contents based on nonlinear feedback functions. Trivium is described as follows:

- Initialization: Trivium initializes three move registers, indicated as A, B and C with beginning values based on the mystery key and initialization vector (IV). These move registers are regularly spoken to as clusters of bits.

$$A = (a_1, a_2, \dots a_{93})$$
$$B = (b_1, b_2, \dots . b_{84})$$
$$C = (c_1, c_2, \dots c_{111})$$

- Update Functions: Trivium characterizes three overhaul capacities g,f, and h h to overhaul the substance of the move registers in each emphasis. These capacities are nonlinear and depend on the current state of the move registers.

$$g(x1, x2, x3) = (x1 + x2)mod2 + x3$$
$$f(x1, x2, x3, x4) = (x1 + x2 + x3)mod2 + x4$$
$$h(1, x2, x3, x4, x5) = (1, x2, x3, x4)mod2 + x5$$

- Update Operation: In each emphasis, Trivium upgrades the substance of the move registers based on the overhaul capacities and input from past emphases.

$$a_{i+1} = g(a_i, a_{i-66}, a_{i-93})$$
$$b_{i+1} = f(b_i, b_{i-69}, b_{i-84}, b_{i-68})$$
$$c_{i+1} = h(c_i, c_{i-78}, c_{i-111}, c_{i-65}, c_{i-110})$$

- Key stream Generation: Trivium produces a key stream by combining the yield bits of the move registers at each cycle.

$$z_i = (a_i + b_i + c_i) mod 2$$

- Output: The key stream bits z are XORed with the plaintext or ciphertext to create the scrambled or unscrambled information.

$$\frac{ciphertext}{pliantext} = Data \oplus keystream$$

A.2. Lightweight Encryption Algorithm for Pervasive computing (LEAP)

Jump is outlined to supply lightweight encryption for resource-constrained gadgets in unavoidable computing situations. It works on fixed-size squares of information, regularly 64 or 128 bits, and employments a settled key estimate. Jump utilizes a streamlined round-based encryption structure for effective usage on resource-constrained gadgets.

- Key Expansion: Jump grows the beginning key into a set of circular keys. The beginning key as K and circular keys as K_i for i=0,1,2,…N_r ("N_r= add up to no. of round"). The key development prepare creates the circular keys based on the starting key.

$$K_0, K_1, K_2 \ldots . K_{Nr} = KeyExpansion(K)$$

- Round Function: Jump applies a circular work to the input plaintext square utilizing the circular keys. The circular work ordinarily comprises of a few basic operations such as bitwise XOR, substitution, and stage. Let's signify the input plaintext piece as P and the yield ciphertext square as C. The circular work can be spoken to as

C = RoundFunction(P,K)

- Final Round: Jump may incorporate a last circular to guarantee the security of the encryption prepare. Within the last circular, Jump applies extra operations such as stage or XOR with a final circular key. The ultimate circular can be spoken to as

$$Ciphertext = FinalRound(C, K_{Nr})$$

A.3. TINYAES

TINYAES may be a lightweight usage of the Progressed Encryption Standard (AES), optimized for resource-constrained gadgets. It points to supply proficient encryption with diminished memory utilization, making it appropriate for applications where memory imperatives are basic. TINYAES works on fixed-size pieces of information, regularly 128 bits, and underpins key sizes of 128, 192, or 256 bits.

- Key Expansion: Through the utilization of a key scheduling method, TINYAES is able to grow the initial key into a collection of round keys.

Let K represent the initial key, K_i represent the round keys for each round i. The key expansion process generates the round keys based on the initial key

$$K_0, K_1, K_2 \ldots . K_{Nr} = KeyExpansion(K)$$

- Round Function: TINYAES applies a simplified round function to the input plaintext block using the round keys. The round function typically involves basic operations such as bitwise XOR, substitution, and permutation.

$$C = RoundFunction(P, K_i)$$

A. Comparison of Trivium, LEAP, TINYAES

Following Table 1 presents the technical evaluation of lightweight encryption schemes, simulation environment setup and evaluation parameters

Table 1
Technical Evaluation of Lightweight Encryption Schemes

Feature	Trivium	LEAP	TINY AES
Category	Stream Cipher	Stream Cipher	Lightweight Block Cipher
Key Size	80-bit	128-bit	64-bit, 128-bit
Block Size	1-bit	1-bit	128-bit
Security Strength	Good for resource-constrained applications	Considered secure for lightweight applications	Good for low-security environments
Performance	Very efficient	More efficient than AES	More efficient than AES
Memory Usage	Low	Low	Lower than AES
Hardware Support	Limited	Limited	Limited
Suitability for IoT	Well-suited	Well-suited	Better suited than AES for low-power devices

B. *Simulation Environment Setup*

Parameter	Value
Simulation Tool	Contiki-NG
Network Topology	Agricultural farm environment with sensor nodes distributed across fields
Number of Sensor Nodes	200
Communication Protocol	IEEE 802.15.4
Transmission Range	100 meters
Environmental Interference	Simulated using interference models
Energy Model	Power-saving modes and battery depletion modeled

Contd..

Operating System	Contiki OS
Programming Language	C
Simulation Duration	24 hours
Data Collection Interval	5 minutes
Hardware Platform	Low-power microcontroller

c. *Evaluation Parameters*

Parameter	Description
Security Strength	The ability of the encryption scheme to resist cryptographic attacks and maintain data confidentiality.
Resource Utilization	The level of hardware resources (e.g., memory, CPU) consumed by the encryption scheme.
Throughput	The rate at which data can be encrypted or decrypted, typically measured in packets per second.
Latency	The time delay between inputting data and receiving the corresponding encrypted or decrypted output.
Implementation Complexity	The level of complexity involved in implementing and integrating the encryption scheme into devices.
Energy Efficiency	The ability of the encryption scheme to operate with minimal energy consumption in resource-constrained environments.
Robustness to Constraints	The resilience of the encryption scheme to environmental constraints such as network congestion, interference, or limited energy availability.
Memory Footprint	The amount of memory (RAM and ROM) required to store the encryption scheme and its associated data structures.
Code Size	The size of the executable code required to implement the encryption scheme on the target devices.
Key Setup Time	The time required to initialize encryption keys before data transmission begins.
Energy Consumption per Packet	The amount of energy consumed by the device for encrypting or decrypting each data packet.
Packet Loss Rate	The proportion of data packets that are lost during transmission as a result of overhead associated with encryption or other potential reasons.

5. Results and Output

Table 2
Evaluation of encryption schemes

Parameter	Trivium	LEAP	TINY AES
Security Strength	High	Medium	High
Resource Utilization	Low	Low	Low
Throughput (packets/sec)	1200	900	950
Latency (ms)	1.1	1.3	1.4
Implementation Complexity	Low	Low	Low

Contd..

Energy Efficiency	High	High	High
Robustness to Constraints	Excellent	Excellent	Excellent
Memory Footprint (KB)	3	2	2.5
Code Size (KB)	8	7	5.5
Key Setup Time (ms)	4	8	6
Energy Consumption per Packet (mJ)	0.2	0.4	0.3
Packet Loss Rate (%)	0.3	0.7	0.4

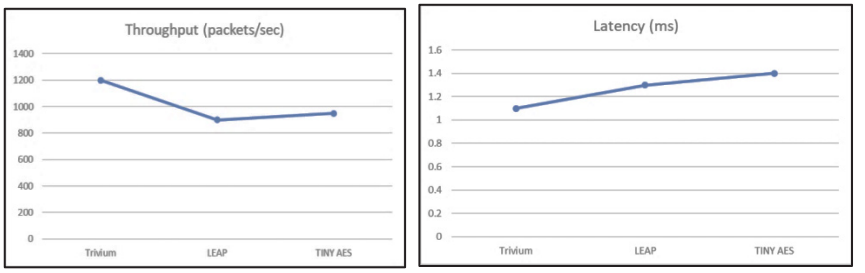

Figure 1
Throughput and Latency comparison

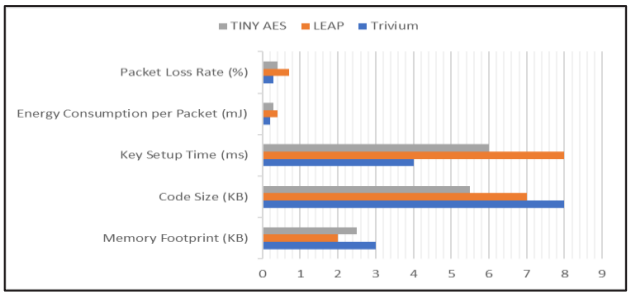

Figure 2
Various parameters comparison

Multiple metrics showed significant results for TRIVIUM, LEAP, and TINY AES. Trivium had excellent security shown in Table 2 and Figure 1-2, low resource use, and easy implementation. The device could handle 1200 packets per second with 1.1 millisecond latency. With a 3 KB memory footprint and 8 KB code size, Trivium was energy efficient and resilient. Trivium had a critical setup time of 4 milliseconds, 0.2 mJ per packet energy consumption, and 0.3% packet loss. LEAP combined moderate security with low resource consumption and implementation complexity. The system had 900/sec packet throughput and 1.3 ms latency. In spite of its 2 KB memory impression and 7 KB code estimate, Jump was vitality proficient and flexible. Jump had an 8-millisecond key setup time, 0.4-millijoule vitality utilization per bundle, and 0.7% parcel misfortune. Like Tritium, Modest AES had solid security, moo

asset utilization, and simple execution. Idleness was 1.4 milliseconds and throughput was 950 bundles per moment. Due to its 5.5 KB code estimate and 2.5 KB memory impression, Little AES was vitality productive and versatile. Minor AES key setup took 6 milliseconds, devoured 0.3 millijoules per bundle, and misplaced 0.4% of parcels.

6. Conclusion and Future Scope

The assessment of lightweight encryption plans for remote sensor systems with low-resource gadgets has uncovered their viability, security, and common sense. Trivium, Jump, and Minor AES appeared distinctive qualities and shortcomings in numerous parameters, giving information transmission alternatives in resource-constrained situations. Trivium had way better vitality proficiency and throughput, but Jump had lower idleness and code measure. Modest AES adjusted security and memory diminishment. The comes about illustrate the significance of considering different components when choosing a remote sensor organize encryption convention. The impacts of joining blockchain and machine learning into lightweight encryption for low-resource gadgets may uncover novel ways to secure remote sensor systems in an assortment of applications. Considering the combination of hardware-accelerated cryptographic calculations and inventive encryption strategies may reduce security dangers and moderate framework assets. Within the age of inescapable computing, this investigate is fundamental for remote sensor organize security and unwavering quality.

References

[1] P. Ramadevi, S. Ayyasamy, Y. Suryaprakash, C. Anilkumar, S. Vijayakumar, and R. Sudha, "Security for wireless sensor networks using cryptography," *Meas. Sensors*, vol. 29, no. May, p. 100874 (2023), doi: 10.1016/j.measen.2023.100874.

[2] C. Patel and N. Doshi, "Secure Lightweight Key Exchange Using ECC for User-Gateway Paradigm," *IEEE Trans. Comput.*, vol. 70, no. 11, pp. 1789–1803 (2021), doi: 10.1109/TC.2020.3026027.

[3] F. Alcaraz Velasco, J. M. Palomares, and J. Olivares, "Lightweight method of shuffling overlapped data-blocks for data integrity and security in WSNs," *Comput. Networks*, vol. 199, no. September, p. 108470 (2021), doi: 10.1016/j.comnet.2021.108470.

[4] H. He, J. Zhang, J. Gu, Y. Hu, and F. Xu, "A fine-grained and lightweight data access control scheme for WSN-integrated cloud computing," *Cluster Comput.*, vol. 20, no. 2, pp. 1457–1472 (2017), doi: 10.1007/s10586-017-0863-y.

[5] Y. Li and Y. Tian, "A Lightweight and Secure Three-Factor Authentication Protocol With Adaptive Privacy-Preserving Property for

Wireless Sensor Networks," *IEEE Syst. J.*, vol. 16, no. 4, pp. 6197–6208 (2022), doi: 10.1109/JSYST.2022.3152561.

[6] M. Khalifa, F. Algarni, M. Ayoub Khan, A. Ullah, and K. Aloufi, "A lightweight cryptography (LWC) framework to secure memory heap in Internet of Things," *Alexandria Eng. J.,* vol. 60, no. 1, pp. 1489–1497 (2021), doi: 10.1016/j.aej.2020.11.003.

[7] D. Sadhukhan, S. Ray, G. P. Biswas, M. K. Khan, and M. Dasgupta, "A lightweight remote user authentication scheme for IoT communication using elliptic curve cryptography," *J. Supercomput.*, vol. 77, no. 2, pp. 1114–1151 (2021), doi: 10.1007/s11227-020-03318-7.

[8] M. Saqib, B. Jasra, and A. H. Moon, "A lightweight three factor authentication framework for IoT based critical applications," *J. King Saud Univ. - Comput. Inf. Sci.,* vol. 34, no. 9, pp. 6925–6937 (2022), doi: 10.1016/j.jksuci.2021.07.023.

[9] R. N. Wadibhasme, A. U. Chaudhari, P. Khobragade, H. D. Mehta, R. Agrawal and C. Dhule, "Detection And Prevention of Malicious Activities In Vulnerable Network Security Using Deep Learning," 2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET), Nagpur, India, pp. 1-6 (2024), doi: 10.1109/ICICET59348.2024.10616289.

[10] N. Alsaeed, F. Nadeem, and F. Albalwy, "A scalable and lightweight group authentication framework for Internet of Medical Things using integrated blockchain and fog computing," *Futur. Gener. Comput. Syst.,* vol. 151, no. June 2023, pp. 162–181 (2024), doi: 10.1016/j.future.2023.09.032.

[11] V. Rao and K. V. Prema, "A review on lightweight cryptography for Internet-of-Things based applications," *J. Ambient Intell. Humaniz. Comput.,* vol. 12, no. 9, pp. 8835–8857 (2021), doi: 10.1007/s12652-020-02672-x.

[12] S. Itoo, A. A. Khan, M. Ahmad, and M. J. Idrisi, "A Secure and Privacy-Preserving Lightweight Authentication and Key Exchange Algorithm for Smart Agriculture Monitoring System," *IEEE Access*, vol. 11, no. April, pp. 56875–56890 (2023), doi: 10.1109/ACCESS.2023.3280542.

[13] M. Elhoseny, H. Elminir, A. Riad, and X. Yuan, "A secure data routing schema for WSN using Elliptic Curve Cryptography and homomorphic encryption," *J. King Saud Univ. - Comput. Inf. Sci.,* vol. 28, no. 3, pp. 262–275 (2016), doi: 10.1016/j.jksuci.2015.11.001.

Received November, 2024