IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, September 2026
Impact Factor: 8.2

Design and Development of Lightweight Cryptographic Algorithms for Secure Communication in Resource-Constrained IoT Devices

Bavdhankar Pravin Bhanudas[1] and Dr. Nikhil Janardan Rathod[2]
[1]Research Scholar, Department of Electronics & Communication Engineering
[2]Associate Professor, Department of Electronics & Communication Engineering
Vikrant University, Gwalior, M.P, India

Abstract: *The rapid expansion of the Internet of Things has connected low-power sensors, wearable devices, smart meters, industrial controllers and embedded gateways to communication networks. These devices frequently operate with limited processing capability, small memory, restricted battery capacity and narrow communication bandwidth. Conventional cryptographic implementations can impose significant computational, memory and energy overheads, while weak or incomplete security mechanisms expose devices to eavesdropping, message manipulation, impersonation and replay attacks. This paper examines the design and development of lightweight cryptographic algorithms for secure communication in resource-constrained IoT environments. It reviews the principal design requirements of lightweight cryptography and compares representative approaches, including AES-128, PRESENT, SIMON/SPECK, TinyJAMBU, Xoodyak and the standardized Ascon family*

Keywords: Internet of Things, Lightweight Cryptography, Ascon, Authenticated Encryption, Resource-Constrained Devices

I. INTRODUCTION

The Internet of Things represents a distributed ecosystem in which physical objects collect, process and exchange information through wired and wireless networks. Typical deployments include environmental monitoring, smart agriculture, healthcare wearables, industrial automation, smart homes and intelligent transportation. Unlike desktop computers and servers, many IoT endpoints are built around low-cost microcontrollers and operate with limited flash memory, RAM, processing speed and energy. Some devices are expected to function for years using small batteries or energy-harvesting systems. Their communication links may also be low bandwidth, intermittent or exposed to physically accessible environments.

These characteristics create a difficult security problem. IoT messages may contain personal, operational or safety-related information. An attacker who can observe traffic may learn sensitive data; an attacker who can modify packets may inject false sensor readings or control commands. Devices may also be impersonated, captured, reset or subjected to replay and denial-of-service attempts. Encryption alone is not sufficient: secure communication also requires integrity and authenticity, freshness protection, sound key management and a trusted update process.

Lightweight cryptography seeks to provide appropriate security with reduced computational, memory, area and energy costs. The objective is not simply to create a smaller cipher, but to select and implement cryptographic functions that match the device, protocol and threat model. NIST finalized SP 800-232 in August 2025, specifying Ascon-AEAD128, Ascon-Hash256, Ascon-XOF128 and Ascon-CXOF128 for constrained devices (Turan et al., 2025). This

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, September 2026 Impact Factor: 8.2

standardization is important because it provides a documented basis for implementing authenticated encryption and hashing in IoT systems.

Objectives of the Study

- To examine the security and resource constraints that influence cryptographic design for IoT endpoints.
- To review representative lightweight cryptographic algorithms and authenticated-encryption approaches.
- To propose a modular lightweight secure-communication framework based on authenticated encryption with associated data.
- To define measurable criteria for evaluating code size, RAM consumption, latency, throughput, energy use and security.
- To identify implementation, key-management and deployment practices that reduce common IoT security risks.

REVIEW OF RELATED LITERATURE

1. Lightweight cryptography and constrained devices

Lightweight cryptography addresses the mismatch between conventional security implementations and devices with very limited resources. The constraints can be classified as computational (CPU cycles and latency), memory-related (program memory and RAM), energy-related (current draw and battery life), hardware-related (gate count or silicon area) and communication-related (packet size and airtime). The IETF constrained-node terminology provides a useful way to describe device classes and their varying memory and processing capabilities (Bormann et al., 2014). A suitable algorithm must be evaluated on the actual target platform because performance varies with instruction set, compiler, clock frequency, hardware acceleration and implementation strategy.

A broad review by Alharbi (2021) compared lightweight cryptographic algorithms in terms of implementation cost, software and hardware performance, and resistance to attacks. The review emphasizes that no single algorithm is optimal for every application: the appropriate choice depends on the device and the required balance among cost, performance and security.

2. Conventional and lightweight algorithm families

AES-128 is a widely established block cipher and may perform efficiently on microcontrollers that provide AES instructions or dedicated hardware. However, a software-only implementation on a very small device may have a different cost profile. PRESENT was designed with compact hardware implementations in mind, while SIMON and SPECK were proposed as lightweight block-cipher families. Their use must be assessed against the relevant security, standardization and policy requirements; a compact block cipher by itself does not automatically provide authenticated communication.

Authenticated-encryption candidates such as TinyJAMBU, Xoodyak and Ascon integrate confidentiality and integrity protection. Radhakrishnan et al. (2024) compared AES-128, SPECK and ASCON using measures such as execution time, memory utilization, latency, throughput and security robustness on constrained IoT boards. Their findings illustrate that measured rankings depend on the implementation and hardware platform, and should not be generalized as universal performance results.

3. Ascon and standardization

Ascon is a family of permutation-based cryptographic functions selected by NIST for lightweight cryptography. NIST SP 800-232 specifies Ascon-AEAD128 for authenticated encryption, Ascon-Hash256 for hashing, and Ascon-XOF128 and Ascon-CXOF128 for extendable-output functions (Turan 2025). The standard provides a stable reference for implementers and reduces the uncertainty associated with choosing an unstandardized design. NIST's standardization process also reflects extensive public evaluation and review (NIST, 2023, 2025).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, September 2026 Impact Factor: 8.2

RESEARCH METHODOLOGY

The study adopts a design-oriented and comparative research methodology based on analysis of standards, peer-reviewed literature and implementation requirements. The research is organized into five stages: (i) define a representative constrained IoT threat model; (ii) identify cryptographic candidates and standards; (iii) specify a modular AEAD-based communication architecture; (iv) establish a platform-specific benchmarking protocol; and (v) assess the design against security and resource criteria. The comparative table is qualitative and literature-based. Numerical performance values are intentionally not fabricated because no physical board, compiler configuration or executable implementation was supplied for testing.

1. Target environment and assumptions

The assumed endpoint is a battery-powered sensor or actuator using a microcontroller with limited flash and RAM, communicating with a gateway over a low-power wireless link. The attacker may observe, delay, replay, inject, modify or drop network packets and may attempt to impersonate an endpoint. The baseline design assumes that device secrets are provisioned securely, cryptographic keys are not shared across unrelated devices, and the endpoint can maintain a unique nonce or sequence-number state. Physical extraction resistance and side-channel resistance require additional hardware and implementation controls and are not guaranteed by the algorithm alone.

2. Proposed evaluation parameters

Parameter	Measurement approach	Why it matters
Program memory	Compiled code size in bytes	Determines flash requirements and update size
RAM usage	Peak stack and static RAM in bytes	Determines feasibility on small microcontrollers
Latency	Cycles or milliseconds per packet	Affects real-time response and duty cycle
Throughput	Authenticated bytes processed per second	Indicates capacity under continuous traffic
Energy per operation	Joules or millijoules per authenticated packet	Estimates battery impact
Communication overhead	Nonce, tag and framing bytes per packet	Affects airtime and payload efficiency
Security behavior	Known-answer tests, invalid-tag rejection, nonce tests	Checks correctness and misuse resistance

For valid comparison, each candidate should be compiled with the same toolchain and optimization settings, tested on the same board and clock frequency, and measured over identical packet sizes. Results should report warm-up policy, number of repetitions, statistical dispersion, hardware acceleration, compiler version and whether key setup is included. Energy should be measured at the device supply or estimated using a clearly documented current-and-time method.

COMPARATIVE ANALYSIS OF SELECTED ALGORITHMS

Table 2 summarizes representative algorithm families at a design level. It is not a benchmark ranking. Block ciphers such as PRESENT, SIMON and SPECK require a secure mode and authentication construction for complete message protection. AEAD schemes directly address both confidentiality and integrity. Actual resource consumption must be measured on the target device.

Algorithm / family	Cryptographic role	Potential strengths	Important considerations
AES-128	Block cipher; use with an approved AEAD mode	Mature ecosystem; hardware acceleration on many MCUs	Software cost can be higher without acceleration; mode and nonce discipline required
PRESENT	Lightweight block cipher	Compact hardware-oriented design	Requires secure mode and authentication; evaluate current security/policy suitability
SIMON / SPECK	Lightweight block-	Designed for constrained	Deployment suitability,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, September 2026 Impact Factor: 8.2

	cipher families	implementations	standardization and policy acceptance must be checked
TinyJAMBU	Authenticated encryption	Designed for constrained platforms	Benchmark code size, latency and implementation assurance on target
Xoodyak	Authenticated encryption and hashing functionality	Permutation-based design with compact state	Implementation and protocol integration costs must be measured
Ascon-AEAD128	Standardized authenticated encryption	Specified by NIST SP 800-232 for constrained devices	Verify exact standard version, nonce handling and platform performance

The comparison supports a key design principle: algorithm selection should be based on the complete security service required by the application. A low-area encryption-only primitive may not be the best choice if separate authentication increases code size, complexity and energy. Conversely, an integrated AEAD scheme can simplify the software interface and reduce the risk of developers combining primitives incorrectly.

RESULTS AND DISCUSSION

The literature and standards review indicates that lightweight cryptography is not a single performance category but a design discipline involving multiple resource and security dimensions. The most relevant finding for the proposed framework is the value of standardized AEAD: it provides confidentiality and integrity through one defined interface and reduces the need to combine encryption and message authentication manually. NIST SP 800-232 makes the Ascon family a practical standards-based candidate for constrained-device evaluation (Turan 2025).

Published implementation studies demonstrate that performance is platform-dependent. Radhakrishnan (2024), for example, evaluate AES-128, SPECK and ASCON using execution time, memory, latency, throughput and security considerations on constrained boards. Such work supports the use of a multi-metric evaluation rather than relying on algorithm labels such as 'lightweight'. A result obtained on one Arduino or microcontroller configuration should not be presented as a universal ranking.

The proposed design therefore treats the selection of Ascon-AEAD128 as a starting hypothesis for new constrained-device deployments, not as a guarantee that it will be fastest or smallest on every platform. The final selection should be made after compiling and measuring candidates on the intended hardware, using representative packet lengths and realistic communication duty cycles. Where hardware AES acceleration exists, AES-based authenticated encryption may remain competitive. Where the endpoint is extremely constrained, implementation footprint and energy per message may dominate throughput.

The framework also highlights that algorithm optimization cannot compensate for weak operational controls. A secure cipher deployed with repeated nonces, shared fleet-wide keys or unauthenticated firmware updates can still lead to compromise. Consequently, secure provisioning, device identity, replay prevention, update integrity and lifecycle key management must be treated as integral components of the cryptographic design.

II. CONCLUSION

Lightweight cryptography is essential for protecting communication in IoT devices that operate with restricted processing power, memory, energy and bandwidth. The design challenge is to maintain confidentiality, integrity and authenticity without imposing unacceptable overhead or introducing fragile implementation complexity. This paper reviewed representative cryptographic families and proposed an AEAD-centered framework integrating nonce management, authenticated metadata, replay protection, secure key handling and platform-specific benchmarking. The recent NIST standardization of Ascon provides a well-defined option for constrained-device protection, but no

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, September 2026
Impact Factor: 8.2

algorithm should be assumed optimal without measurement on the intended hardware. A dependable IoT security solution must combine a sound standardized primitive with correct implementation, secure provisioning, authenticated updates and lifecycle management. The central conclusion is that lightweight security should be evaluated as an end-to-end system property, not merely as the execution speed of an encryption routine.

REFERENCES

1. Alharbi, F., Alosaimi, W., & others. (2021). Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access*, 9, 28177–28193.
2. Bormann, C., Ersue, M., & Keränen, A. (2014). Terminology for constrained-node networks (RFC 7228). *Internet Engineering Task Force*.
3. National Institute of Standards and Technology. (2023). Status report on the final round of the NIST lightweight cryptography standardization process (NIST IR 8454). U.S. *Department of Commerce*.
4. National Institute of Standards and Technology. (2025). NIST finalizes 'lightweight cryptography' standard to protect small devices.
5. Radhakrishnan, I., Jadon, S., & Honnavalli, P. B. (2024). Efficiency and security evaluation of lightweight cryptographic algorithms for resource-constrained IoT devices. *Sensors*, 24(12), 4008.
6. Silva, C., Tenório, N., & Bernardino, J. (2025). Lightweight encryption algorithms for IoT. *Computers*, 14(12), 505.
7. Turan, M. S., McKay, K., Kang, J., Kelsey, J., & Chang, D. (2025). Ascon-based lightweight cryptography standards for constrained devices: Authenticated encryption, hash, and extendable output functions. *National Institute of Standards and Technology*.
8. Haider, Z. A., Zeb, A., Islam, A. K. M. M., Rahman, T., Arishi, A., & Ullah, I. (2026). Enhancing IoT security with resource-efficient cryptography: A comprehensive review of lightweight and hybrid algorithms. *Computer Science Review*.