World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(REVIEW ARTICLE)

Check for updates

Lightweight cryptography for IoT: A comprehensive survey of algorithms, implementations, and standardization

Faisal Abdullah Althobaiti *

Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.

World Journal of Advanced Engineering Technology and Sciences, 2025, 15(03), 516-525

Publication history: Received on 20 April 2025; revised on 01 June 2025; accepted on 04 June 2025

Article DOI: https://doi.org/10.30574/wjaets.2025.15.3.0967

Abstract

The Internet of Things (IoT) paradigm has enabled seamless connectivity among billions of constrained devices—sensors, actuators, and smart appliances—yet securing these networks remains a critical challenge. Traditional cryptographic primitives often exceed typical IoT endpoints' power, memory, and latency budgets. This paper thoroughly surveys lightweight cryptography specifically designed for Internet of Things (IoT) environments. We first introduce a taxonomy of algorithmic families—including block and stream ciphers, public-key schemes, and hash functions—and review recent proposals that achieve high-security margins with minimal resource footprints. Next, we analyze efficiency optimizations at both the algorithmic and architectural levels, covering hardware accelerators, side-channel countermeasures, and software-only implementations. We then examine ongoing efforts toward standardization and interoperability, focusing on emerging post-quantum candidates suitable for constrained platforms. We illustrate practical deployment trade-offs in throughput, energy consumption, and resilience under realistic attack models through detailed case studies- spanning smart metering, industrial control, and wearable devices. Finally, we identify open challenges and research directions, including further reductions in computational overhead, integration of quantum-resistant primitives, and the development of unified frameworks for large-scale, energy-efficient key management. Our survey highlights the pressing need for quantum-aware, scalable, and lightweight cryptographic solutions to safeguard the next generation of IoT applications.

Keywords: Lightweight Cryptography; Internet of Things (IoT) Security; Resource-Constrained Devices; Post-Quantum Cryptography; Energy-Efficient Key Management

1. Introduction

The Internet of Things (IoT) paradigm has experienced unprecedented expansion in recent years, integrating billions of heterogeneous, resource-constrained devices—such as sensors, actuators, and smart appliances—into virtually every facet of modern life [4], [9]. These endpoints often operate under severe power, memory, and processing budgets, with typical microcontrollers clocked between 8 MHz and 32 MHz, on-chip RAM in the 4 KB–64 KB range, and energy reserves that must support multi-year deployments without maintenance [4], [7]. Concurrently, as IoT deployments scale from smart homes to industrial control systems and large-scale smart-city infrastructures, data integrity, confidentiality, and availability in transit are essential. These networks have become paramount [1], [2], [9]. Traditional cryptographic primitives—while robust in conventional computing environments—incur prohibitive computational and energy overheads when ported to constrained platforms, with algorithms like RSA-2048 consuming tens of millijoules per operation and large symmetric ciphers such as AES-256 demanding substantial code footprints beyond the capabilities of many class-0 and class-1 devices [8], [5]. Beyond traditional confidentiality threats, IoT endpoints face device spoofing, supply-chain tampering, and remote firmware exploits, as evidenced by large-scale botnet-driven

* Corresponding author: Faisal Abdullah Althobaiti

denial-of-service campaigns leveraging compromised sensors [2], [9]. This disparity between security requirements and physical capabilities has resulted in lightweight cryptography, a discipline focused on creating primitives and protocols that minimize computational cycles, memory usage, and communication overhead while upholding clearly defined security standards [6], [7].

This survey comprehensively examines lightweight cryptographic solutions tailored to IoT ecosystems in response to this pressing need. We first introduce a structured taxonomy of algorithmic families—including block ciphers (e.g., PRESENT, Simon, Speck), stream ciphers (e.g., Grain, Trivium), authenticated-encryption schemes (e.g., Ascon, GIFT-COFB), public-key constructs (e.g., elliptic-curve cryptography and emerging post-quantum candidates), and hash functions (e.g., PHOTON, Keccak-based SHA-3)—highlighting their design rationales and security properties [3], [6]. Next, we analyze efficiency optimizations at both the algorithmic level (e.g., reduced-round variants, bit-sliced implementations) and the architectural level (e.g., hardware accelerators, side-channel countermeasures), quantifying trade-offs in throughput, energy consumption, and resilience against realistic attack models [1], [5]. We then review standardization efforts—including NIST's Lightweight Cryptography project, the ISO/IEC 29192 series, and the IETF's SCHC framework—that drive interoperability and deployment guidance for constrained platforms [6]. Detailed case studies spanning smart metering, industrial automation, and wearable health monitoring illustrate real-world performance benchmarks and deployment considerations. By synthesizing design principles, performance optimizations, and standardization frameworks, this survey offers system architects and developers a unified reference for selecting and deploying lightweight cryptographic primitives in practical IoT applications. Our analysis aims to bridge the gap between theoretical research and practical deployment, fostering resilient and scalable security for the next generation of connected devices.

2. Key Challenges in IoT Security

IoT security faces unique challenges due to the extreme heterogeneity, massive scale, and stringent resource limitations of connected devices. Security mechanisms need to be re-evaluated in environments where every millijoule of energy and every bite of data are crucial. In what follows, we examine three primary challenge domains—resource constraints, communication overhead, and scalability—and highlight why each motivates the adoption of lightweight cryptographic designs tailored to IoT ecosystems.

2.1. Resource Constraints

IoT endpoints—from battery-powered environmental sensors to wearable health trackers—typically rely on low-power microcontrollers operating at clock frequencies between 8 MHz and 32 MHz, with RAM footprints often limited to 4 KB–64 KB and non-volatile storage in the order of tens to hundreds of kilobytes [4], [9]. Their energy reservoirs, whether coin-cell batteries (≈100 mAh) or small Li-ion packs (≈1000 mAh), must support months or years of field operation without recharge or replacement. In such contexts, conventional public-key schemes like RSA-2048 or even symmetric algorithms configured for high security (e.g., AES-256) impose prohibitive demands: an RSA-2048 encryption on a 16-bit MCU can consume tens of millijoules and require seconds of processing time, rapidly depleting limited energy stores [8]. Similarly, a full-scale AES-256 implementation may occupy 10 KB or more of code space—beyond the capabilities of many class-0 or class-1 constrained devices [7]. Lightweight cryptography addresses these limitations by minimizing computational complexity, memory footprint, and energy per operation. For instance, block ciphers such as PRESENT and LED can be implemented in hardware using fewer than 2,000 gate equivalents (GE), while software-oriented designs like SPECK and Simon maintain code sizes under 1 KB and execute in just a few hundred cycles per 64-bit block [7]. On the energy front, targeted optimizations—such as reducing round counts or employing precomputed lookup tables—can slash per-packet cryptographic overhead by up to 60%, extending battery lifetime substantially without undermining security margins [1], [5].

2.2. Communication Overhead

Beyond pure computation, cryptographic measures introduce additional bits on the wire—authentication tags, initialization vectors, and protocol metadata—that inflate message sizes. In narrowband LPWAN technologies (e.g., LoRaWAN, NB-IoT, and 6LoWPAN), where payloads may be limited to dozens of bytes per transmission, even a 10%–20% ciphertext expansion can translate into missed deadlines, increased collisions, and higher energy drain due to retransmissions [2], [6]. For example, AES-GCM provides authenticated encryption at the cost of roughly 16 bytes of tag data per message, resulting in 30%–50% higher processing latency and an equivalent rise in transmission energy compared to lighter alternatives such as ChaCha20-Poly1305 or POLY1305-based stream ciphers [6].

Lightweight schemes mitigate these burdens by streamlining protocol handshakes, minimizing padding requirements, and employing compact message formats. Techniques such as header compression (e.g., SCHC for 6LoWPAN) and

session resumption can reduce per-transaction overhead by 40% or more, preserving precious bandwidth in high-frequency reporting scenarios like smart grids or industrial telemetry [4]. Nonetheless, designers must navigate inherent trade-offs: smaller tags lower overhead but reduce attack resilience, while aggressive compression may complicate interoperability with legacy stacks [4].

2.3. Scalability

The most daunting challenge in IoT security is scale. Deployments range from a few dozen devices in a smart home to millions of sensors in an Industrial IoT (IIoT) installation or smart-city infrastructure [9]. Managing cryptographic keys at this scale demands robust schemes for secure provisioning, updates, and revocation that impose minimal burden on constrained hardware and network links [4]. While computationally lightweight, symmetric key approaches require efficient group key management or frequent rekeying to maintain confidentiality across dynamic topologies, protocols like Diffie-Hellman remain costly on low-end MCUs [1]. Compounding this is the protocol heterogeneity endemic to IoT: MQTT, CoAP, Zigbee, Bluetooth Low Energy, and emerging standards like OPC UA each embed distinct security models and lifecycle workflows [6]. Ensuring consistent end-to-end protection across gateways, edge nodes, and cloud platforms calls for interoperable frameworks and unified policy enforcement. Mechanisms such as lightweight PKI, identity-based cryptography, or key-update delegation can help distribute trust without overloading devices but require careful architectural planning to avoid single points of failure and to guarantee forward secrecy [9], [4]. IoT security's defining constraints—scarce compute and memory resources, narrowband communications, and massive device populations—demand cryptographic innovations that diverge sharply from traditional designs. Lightweight cryptography bridges security imperatives and the physical realities of ubiquitous, constrained devices, enabling robust protection without sacrificing performance or longevity. Table 1 concisely overviews the primary IoT security challenge domains, their key issues, and representative lightweight cryptographic strategies.

Table 1 Summary of IoT Security Challenges and Lightweight Cryptographic Solutions

Challenge Domain	Key Issues	Lightweight Cryptography Strategies	References
Resource Constraints	• MCU clocks 8–32 MHz; RAM 4–64 KB; storage tens–hundreds KB • Batteries ≈100–1000 mAh, months–years lifetime	• Ultra-low-gate-count ciphers (PRESENT, LED) • Code-compact SPECK/SIMON (< 1 KB) • Reduced rounds, lookup tables	[4], [9], [8], [7], [1], [5]
Communication Overhead	• LPWAN payloads tens of bytes • 10–20 % ciphertext/tag expansion → collisions, retransmits	• Compact tags (e.g., POLY1305-based) • Header compression (SCHC) • Session resumption, streamlined handshakes	[2], [6], [4]
Scalability	• Millions of devices, dynamic topologies • Heterogeneous protocols (MQTT, CoAP, BLE, Zigbee, OPC UA)	• Lightweight PKI/identity-based schemes • Efficient group key management • Delegated key-update frameworks	[9], [4], [6], [1]

3. Lightweight Cryptography Solutions

3.1. Algorithmic Approaches

Lightweight cryptographic primitives encompass a spectrum of symmetric-key ciphers, stream ciphers, asymmetric schemes, and hash functions tailored to minimize computational cycles, code footprint, and energy consumption on highly constrained hardware. Among symmetric-key block ciphers, PRESENT processes 64-bit blocks using only 1570 gate equivalents—ideal for RFID tags and sensor nodes—while Simon and Speck support configurable block and key lengths from 32 to 128 bits, with Simon-64/128 implementations on 65 nm ASICs achieving throughput beyond 200 Mbps and remarkably low per-bit energy costs [6], [7]. Stream ciphers such as Grain and Trivium further reduce latency and power draw: Grain's hardware core requires just 1294 gate equivalents and consumes approximately 3.3 μW,

enabling continuous operation in battery-powered smart meters, whereas Trivium's minimal state machine supports efficient FPGA and software deployments with sub-microsecond keystream generation [6], [7]. Constrained devices demanding authenticated encryption can leverage duplex-based constructions like Ascon, which combines AEAD and hashing in under 4 KB of code and data memory with fewer than 2500 gate equivalents, striking a balance between integrity, confidentiality, and footprint [9]. In the asymmetric domain, elliptic-curve cryptography (ECC) significantly reduces key sizes relative to RSA—256-bit ECC keys deliver security comparable to RSA-3072 yet permit key-pair generation in tens of milliseconds on 16-bit microcontrollers—and libraries such as TinyECC employ fixed-point optimizations and precomputation to achieve practical performance on 8-bit platforms [2], [8]. Lightweight hash functions such as PHOTON can be instantiated in hardware with roughly 1120 gate equivalents for medical implants, and Keccak-based SHA-3 offers configurable security levels within a 2–4 KB software footprint, providing robust collision resistance and side-channel resilience [3], [6]. Beyond these core algorithms, performance is enhanced through hardware accelerators—dedicated crypto cores or reconfigurable logic—that reduce energy per operation by 50–80 % compared to software-only implementations [5], while software optimizations such as bit-sliced S-box designs, loop unrolling, and table-based lookups exploit instruction-level parallelism on 32-bit MCUs to accelerate cipher rounds without significant code-size inflation [4]. Countermeasures against side-channel attacks, including masking, shuffling, and threshold implementations, introduce randomized intermediate states to obfuscate power and electromagnetic leakages with typically under 15 % overhead [5]. Collectively, this algorithmic innovation enables secure, efficient operation in diverse IoT scenarios—from wearable health monitors and smart meters to industrial sensor networks and remote telemetry—by carefully aligning primitive selection and parameterization with stringent resource envelopes, thereby delivering provable security guarantees without compromising device performance or longevity.

3.2. Efficiency and Performance

Efficiency and performance represent critical considerations in lightweight cryptography, especially for resource-constrained IoT endpoints, which must operate within extremely stringent power, memory, and processing budgets while maintaining robust defenses against side-channel leaks. By simplifying round functions and leveraging hardware parallelism, ciphers such as PRESENT on a 65 nm ASIC achieve encryption power consumptions as low as 2.5 μW per 64-bit block—over 70 % less than AES-128's approximately 10 μW—dramatically extending coin-cell-powered wearable sensor lifespans by more than 70 % [7]. Software-oriented designs like SPECK deliver similarly impressive savings, requiring only 1.8 μJ per block on a 32 MHz microcontroller, facilitating multi-year operation in solar-powered environmental monitors [6]. Memory footprints are also minimized: SIMON's reference implementation occupies under 1 KB of static code and employs register-based round keys to fit within 4 KB of RAM, while PHOTON's hardware core uses merely 1120 gate equivalents, and its software variant demands just 1.5 KB of flash—considerably smaller than SHA-2's typical > 5 KB profile—notably freeing precious storage for application logic in implants and smart thermostats [3]. Low-clock devices similarly benefit from high-speed execution, with SPECK-128/128 completing encryption in around 10 μs on 16-bit controllers to sustain throughput above 5 Mbps for automotive telemetry and PRESENT pipelines finishing in 32 clock cycles to ensure sub-microsecond latency for industrial control loops [6], [7]. Finally, integrated countermeasures such as masking, shuffling, and bit-sliced implementations introduce minimal overhead—masked PRESENT variants incur under 15 % additional gates—yet effectively mitigate differential power and timing analysis attacks, ultimately underpinning real-time security in unshielded IoT environments [5], [7].

3.3. Standards and Guidelines

Standardization ensures seamless interoperability across the heterogeneous spectrum of Internet of Things (IoT) devices—from ultra-constrained sensor nodes to high-capacity edge gateways—by delivering rigorously vetted algorithm portfolios, clear implementation profiles, and comprehensive deployment guidance that collectively balance security requirements with stringent resource constraints. In this context, the National Institute of Standards and Technology (NIST) launched its Lightweight Cryptography (LWC) project to solicit, evaluate, and endorse cryptographic primitives optimized for constrained environments, culminating in its 2022 selection of authenticated-encryption with associated data (AEAD) and hash candidates such as Ascon and ISAP; Ascon's 320-bit permutation and duplex construction can be implemented in under 10 KB of combined code and data memory, making it ideal for battery-powered sensors and RFID tags, while ISAP's sponge-based design augmented with lightweight masking delivers robust side-channel resistance with hardware footprints below 2000 gate equivalents and software implementations under 2 KB of flash memory [6]. To facilitate adoption and interoperability, NIST also publishes official test vector packages, reference implementations, and conformance testing frameworks that enable vendor certification and cross-platform assurance [6]. Complementing these efforts, ISO/IEC 29192 provides a structured framework for lightweight cryptography: Part 2 defines block cipher families such as PRESENT, SIMON, and SPECK with hardware profiles targeting sub-2000 GE gate counts and software code sizes below 1 KB; Part 3 specifies stream cipher constructions; and Part 5 details lightweight hash functions like PHOTON and Keccak (SHA-3), complete with normative test vectors, domain parameters, integrity requirements, and algorithm agility guidelines to ensure seamless updates in response to

emerging threats [3], [7]. Meanwhile, the Internet Engineering Task Force (IETF) has standardized the Static Context Header Compression (SCHC) framework for header compression and fragmentation over LPWAN protocols—including LoRaWAN and NB-IoT—thereby reducing per-packet cryptographic overhead and defining security contexts and key-identifier management for embedded AEAD schemes [2]. In parallel, the Common Criteria (ISO/IEC 15408) offers evaluation assurance levels tailored to lightweight cryptographic modules, enabling system integrators to obtain recognized security certifications under global regulatory regimes [7]. Collectively, these international standards and guidelines furnish system architects, firmware developers, and security evaluators with a coherent foundation for selecting, implementing, and deploying lightweight cryptographic primitives that deliver provable security assurances without exceeding the severe power, memory, and processing constraints of modern IoT applications. Table 2 illustrates the classification of lightweight cryptography approaches across algorithm design, performance/implementation optimizations, and standardization, highlighting example primitives, their primary advantages, and foundational references.

Table 2 The key dimensions of lightweight cryptographic solutions

Dimension	Example Primitives / Techniques	Key Benefits	References
Symmetric Key – Block Ciphers	PRESENT, SIMON, SPECK	Ultra-small hardware footprint (e.g. 1570 GE for PRESENT); adjustable block/key sizes for throughput tuning	[6], [7]
Symmetric Key – Stream Ciphers	Grain, Trivium	Bit-level encryption with low latency; minimal hardware cost (1294 GE for Grain)	[6], [7]
Asymmetric Key	ECC (256-bit), TinyECC	RSA-3072-equivalent security with much faster key ops (10 ms vs. 500 ms); precomputation for 8-bit MCUs	[2], [8]
Hash Functions	PHOTON, Keccak-based SHA-3	Integrity assurance; compact cores (1120 GE PHOTON) and 2–4 KB software footprint	[3], [6]
Hardware Accelerators	Dedicated crypto cores, FPGA implementations	50–80 % reduction in energy per operation compared to pure software	[5]
Software Optimizations	Bit-sliced S-boxes, loop unrolling, lookup tables	Exploits instruction-level parallelism on 32-bit MCUs for faster execution	[4]
Side-Channel Countermeasures	Masking, hiding, constant-time routines	Mitigates power/timing leaks with minimal overhead (<15 % gate increase)	[5], [7]
Standardization & Interoperability	Ascon, GIFT-COFB (NIST LWC); ISO/IEC 29192; IETF SCHC	Vetted algorithm portfolios and profiles; harmonized deployment for LPWANs	[2], [6], [9]

4. Security Considerations

4.1. Adapted Security Levels

Adapted security levels require mapping application risk profiles to cryptographic strength quantified as security bits, the base-2 logarithm of the work factor necessary to compromise a primitive [8]. NIST SP 800-57 recommends 128-bit security for highly sensitive or regulated data, 112-bit for moderate-risk scenarios, and 80-bit for low-impact use cases [5]. This guidance informs adaptive strategies balancing security with energy, memory, and latency in resource-constrained IoT endpoints. Risk-based selection exemplifies this balance, as a networked door lock may adopt an 80-bit cipher to deter opportunistic intruders, whereas a cardiac implant exchanging personal health data demands complete 128-bit security to resist targeted cryptanalysis and ensure confidentiality [7]. Algorithm agility allows devices to embed multiple lightweight primitives or parameterized variants such as reduced-round modes, enabling dynamic adjustment of security levels in response to emerging threat intelligence with negligible firmware overhead [1]. Key management and rotation employ compact key-derivation functions for renewal and protocol-level replay protection to thwart message forgery without burdensome state tracking [5]. These measures must be balanced against energy consumption, as AES-256's roughly 20 μJ per 128-bit block contrasts sharply with SPECK's approximate 1.8 μJ, driving many constrained designs to default to 96- or 80-bit security profiles unless protection is essential [6].

4.2. Post-Quantum Security

Quantum adversaries threaten to dismantle RSA and ECC through Shor's algorithm and to halve the adequate security of symmetric primitives via Grover's search [4]. In response, lightweight post-quantum cryptography (PQC) research has emphasized the design of key encapsulation mechanisms (KEMs) and digital signature schemes tailored for constrained environments. Among the NIST PQC finalists, CRYSTALS-Kyber and Saber offer IND-CCA security with public keys below 1 KB and ciphertexts of 800 to 1500 bytes—significantly more compact than many lattice-based alternatives—yet their implementations typically require 8–12 KB of RAM and on the order of millions of clock cycles per encapsulation or decapsulation operation [4]. To facilitate gradual migration, hybrid key-exchange protocols combine classical elliptic-curve handshakes with streamlined PQC runs, preserving backward compatibility but approximately doubling computational and communication overhead during initial exchanges. Concurrently, research into lightweight PQC variants has explored parameter reductions, polynomial-compression techniques, and dedicated hardware accelerators, with prototype implementations of reduced-parameter NewHope and NTRU achieving sub-10 ms encapsulation on 32 MHz microcontrollers, albeit at modest reductions in security margins [6]. Ultimately, ensuring the long-term confidentiality of IoT deployments requires an agile, quantum-ready cryptographic ecosystem that integrates adaptable security levels, emerging PQC frameworks, efficient key-management and firmware-update mechanisms, and tamper-resistant hardware, thereby future-proofing systems without breaching the severe power, memory, and processing envelopes characteristic of modern IoT devices. Table 3 shows the core security strategies for lightweight cryptography in IoT, summarizing how devices can adapt classical security levels to resource constraints and prepare for quantum-era threats by integrating emerging PQC frameworks.

Table 3 The core security strategies for lightweight cryptography in IoT

Category	Key Strategies	References
Adapted Security Levels	• Map risk profiles to "security bits" (80/112/128-bit) to match application sensitivity • Risk-based selection (e.g., 80-bit for low-risk door locks vs. 128-bit for medical sensors) • Algorithm agility via multiple primitives or reduced-round variants • Periodic key management and rotation with compact KDFs and replay protection • Balance energy/latency (AES-256 ≈ 20 μJ vs. SPECK ≈ 1.8 μJ per block)	[5], [8], [7], [1], [6]
Post-Quantum Security	• NIST PQC candidates (CRYSTALS-Kyber, Saber) with ≪ 1 KB keys and 800–1500 B ciphertexts, requiring ~8–12 KB RAM and millions of cycles • Hybrid key exchange (ECC + PQC) for backward compatibility • Lightweight PQC variants via parameter tuning and compression, enabling sub-10 ms on 32 MHz MCUs at slightly reduced margins	[4], [6]

5. Implementation Aspects

5.1. Hardware vs. Software Implementation

Implementation platform profoundly influences throughput, energy consumption, and update agility in lightweight cryptographic deployments: hardware implementations such as ASICs deliver unparalleled per-operation efficiency and performance, exemplified by a PRESENT core on a 65 nm ASIC sustaining 100 Mbps encryption while drawing under five μW, and enabling on-chip integration of physical unclonable functions for secure key storage and resistance to invasive probing—capabilities critical for always-on environmental sensors and tamper-sensitive applications [3]. In contrast, FPGAs offer post-fabrication reconfigurability, allowing a single gateway device to switch between ciphers such as SIMON and SPECK as security policies evolve; a 40 nm FPGA implementation of SIMON-64/128 demonstrates throughput above 150 Mbps at approximately 20 mW but incurs three to five times higher dynamic power and elevated bill-of-materials costs compared to ASICs, factors that must be weighed in large-volume or cost-sensitive production [6], [3]. Software implementations on general-purpose microcontrollers running lightweight cryptographic libraries such as mbedTLS or TinyCrypt maximize flexibility and firmware-update agility—supporting over-the-air patching of vulnerabilities or introduction of new primitives without hardware redesign—yet place greater demand on CPU resources, potentially increasing latency in real-time control loops and lacking the tamper-resistance of isolated hardware zones or PUF-backed key storage [8]. On a 16-bit MCU clocked at 16 MHz, for example, SPECK-128/128 encrypts a 128-bit block in roughly 50 μs at an energy cost of about 10 μJ per operation—approximately five times the

energy of a dedicated hardware core—while avoiding the nonrecurring engineering expenses associated with ASIC development [6]. Ultimately, the choice between hardware and software implementations hinges on application requirements for throughput, power envelope, update agility, cost, and physical security within the severe resource constraints of modern IoT ecosystems.

5.2. Optimization Strategies

Bridging security requirements and stringent resource constraints in IoT endpoints demands a coherent suite of optimizations that begins with careful algorithm selection and parameter tuning—preferring hardware-friendly primitives such as PRESENT or LED and software-oriented ciphers like SPECK or Simon and even reducing Simon's rounds from 31 to 24 to achieve nearly 20 % energy savings at an acceptable security trade-off [7]—while memory-oriented refinements, including storing S-box tables in a flash rather than SRAM and precomputing round constants, further minimize runtime state and offload critical loops [4]. Hardware acceleration via integrated cryptographic co-processors or ARM TrustZone's CryptoCell modules can halve elliptic-curve key-exchange latencies—shrinking a 20 ms software handshake to under 10 ms—vital for latency-sensitive applications like V2X and industrial automation [8]. Protocol-level strategies leverage lightweight stacks such as CoAP over DTLS with pre-shared keys or OSCORE and apply SCHC header compression to reduce per-packet overhead by roughly 40 % on narrowband LPWANs [6]. At the code level, aggressive compiler flags (–O2 with link-time optimization), inline assembly for S-box routines, loop unrolling, and table-based bit slicing exploit MCU instruction-level parallelism to speed up cipher rounds without significant code-size inflation [4]. Dynamic voltage and frequency scaling tailors CPU performance to cryptographic workloads for additional energy savings, while runtime profiling informs branch-prediction hints and prefetch strategies to minimize stalls. On FPGAs, partial reconfiguration loads only required crypto modules, lowering static power and freeing on-chip memory. Finally, lightweight side-channel hardening techniques—such as threshold implementations and randomized instruction scheduling—inject randomness into intermediate states to mitigate power and electromagnetic leakage with under 15 % overhead, preserving device longevity and security integrity [5]. Collectively, these layered optimizations enable robust, efficient cryptography that seamlessly coexists with the severe power, memory, and processing budgets inherent in modern IoT systems. Table 4 illustrates the primary implementation choices for lightweight cryptography in IoT—contrasting hardware versus software realizations and summarizing key optimization strategies to boost throughput, minimize energy, enhance security, and maintain interoperability.

Table 4 The primary implementation choices for lightweight cryptography in IoT

Category	Key Strategies	References
Hardware Implementation	• ASICs: ultra–low energy (e.g., PRESENT core at 100 Mbps for <5 μW @ 65 nm) with on-chip PUF storage for tamper resistance • FPGAs: post-fabrication reconfigurability (e.g., SIMON-64/128 at >150 Mbps for ~20 mW @ 40 nm) but 3–5× higher power and BOM cost	[3], [6]
Software Implementation	• General-purpose MCUs: lightweight libraries (mbedTLS, TinyCrypt) in C/assembly (SPECK-128/128 ≈ 50 μs/block at ~10 μJ on a 16 MHz, 16-bit MCU) • OTA updates for algorithm agility, though with higher CPU load, latency, and limited tamper protection	[6], [8]
Optimization Strategies	• Algorithm selection & parameter tuning (e.g., reduce Simon rounds for ~20% energy savings at slight security loss) • Hardware acceleration (CryptoCell co-processors halve ECC times) • Protocol tuning (CoAP/DTLS + SCHC header compression) • Code-level refinements (–O2, inline asm, bit-slicing) [4] • Side-channel hardening (masking/shuffle at ~10–15% gate overhead)	[7], [8], [6], [4], [5]

6. Case Studies

We examine three representative applications to demonstrate how lightweight cryptography enables secure, efficient IoT deployments—each highlighting specific algorithmic choices, performance metrics, and deployment considerations.

6.1. Smart Healthcare

Wearable health monitors demand both strong confidentiality and minimal power draw to safeguard sensitive patient data over extended operating periods. In one implementation, a continuous-glucose monitor employs the PRESENT block cipher in hardware, achieving 64-bit block encryption with 80-bit keys while consuming only 2.5 μW per operation. This means the battery can last for more than a year in a single coin-cell battery. [7]. For the key establishment, the device leverages TinyECC—an optimized elliptic-curve library—which completes a 256-bit key-exchange handshake in under 15 ms on an 8-bit microcontroller, consuming less than 50 μJ. Combining a sub-2 μs encryption routine with a compact public-key setup ensures end-to-end security without compromising wearability or form-factor constraints [8].

6.2. Smart Grids

In advanced metering infrastructures, millions of smart meters periodically transmit consumption data over narrowband links. A field trial deployed the SIMON-64/128 cipher in firmware, balancing throughput and energy. On a 32 MHz ARM Cortex-M0+, SIMON achieved sustained 200 kbps encryption rates with an average power draw of 3 μW per packet, supporting sub-second readout intervals while preserving meter battery life for up to five years [6]. To streamline multicast billing updates, the utility implements group key rotation via a lightweight key-derivation function, limiting the need for frequent full rekeying and reducing network overhead by 30% compared to unicast approaches [2].

6.3. Smart Cities

Traffic-sensor networks in urban environments require integrity protection to prevent data spoofing that could disrupt signal timing or emergency routing. Deploying the PHOTON-128 hash function on low-cost microcontroller boards (1120 GE hardware footprint) allows sub-100 μs tag generation per 64-bit sensor reading, with negligible impact on sampling rates [3]. Coupled with a compressed header scheme (SCHC) over LoRaWAN, each packet carries a 32-bit MAC without exceeding the network's 51-byte payload limit. This arrangement delivers real-time data validation—supporting adaptive traffic-light coordination—while adhering to city-wide deployments' strict latency and bandwidth constraints [9]. Table 5 concisely compares three representative IoT deployments—highlighting the chosen lightweight cryptographic primitives, their measured performance, and the practical considerations that guided each implementation.

Table 5 The concise comparison of three representative IoT deployments

Application	Cryptographic Primitive & Implementation	Performance Metrics	Deployment Considerations	References
Smart Healthcare	• PRESENT in hardware: 64-bit blocks, 80-bit keys, 2.5 μW per operation • TinyECC for 256-bit key exchange on 8-bit MCU: < 15 ms, < 50 μJ	• > 1 year battery life on coin-cell cell • Sub-2 μs encryption routine	• Wearable form factor • End-to-end confidentiality with minimal energy draw	[7], [8]
Smart Grids	• SIMON-64/128 in firmware on 32 MHz ARM Cortex-M0+	• 200 kbps sustained encryption • 3 μW average per packet	• Sub-second readout intervals • Group key rotation reduces network overhead by 30 %	[6], [2]
Smart Cities	• PHOTON-128 hash on low-cost MCU (1120 GE footprint) • SCHC header compression over LoRaWAN	• < 100 μs MAC tag generation per 64-bit reading • 32-bit MAC within 51-byte payload limit	• Real-time integrity protection • Adheres to LoRaWAN payload/latency constraints	[3], [9]

7. Future Research Directions

As the IoT ecosystem expands into more critical and diverse application domains, the imperative to advance lightweight cryptography along several fronts has become increasingly apparent. Foremost among these is the integration of post-quantum security guarantees into resource-constrained platforms. While lattice-based schemes such as CRYSTALS-Kyber have demonstrated strong resistance to Shor-style attacks, their current implementations typically require upwards of 8–12 KB of RAM and substantial CPU cycles—far beyond the 4 KB memory budgets typical of many IoT sensors [4]. Therefore, Future work must explore algorithmic refinements, parameter tuning, and hardware accelerators that can trim footprint and latency without eroding the core security properties. Parallel to this, developing side-channel countermeasures tailored for lightweight primitives remains essential. Conventional masking and randomization techniques often incur prohibitive overheads in gate count or energy; research into low-cost, provably secure masking schemes and architectural obfuscation techniques will be vital to protect devices from differential power and electromagnetic analysis attacks, particularly in medical and industrial settings [7].

Equally pressing is the need for scalable key management and seamless interoperability across the heterogeneous IoT protocols and standards landscape. Efficient, lightweight key-exchange protocols—potentially leveraging elliptic-curve or identity-based constructions—must be devised to support millions of devices while minimizing handshake complexity and network traffic [9]. Concurrently, closer harmonization of lightweight cryptographic profiles across bodies such as NIST, ISO/IEC, and IETF will facilitate vendor-agnostic deployments and simplify certification pathways [6]. Beyond foundational algorithmic and protocol concerns, emerging cross-disciplinary approaches promise to unlock new efficiencies: ultra-low-power algorithms optimized for energy-harvesting nodes could enable perpetual, battery-less operation in remote monitoring scenarios [9], while machine-learning techniques could dynamically tailor cipher parameters—such as key sizes or round counts—based on real-time assessments of device workload, threat level, and energy availability [10]. Finally, as blockchain and distributed-ledger technologies gain traction in supply-chain and asset-tracking applications, lightweight consensus mechanisms and compact signature schemes will be required to bring tamper-evident trust guarantees to deeply constrained devices. Collectively, these research directions outline a roadmap for sustaining robust, agile security in the next generation of pervasive, resource-limited IoT deployments.

8. Conclusion

Lightweight cryptography has emerged as a linchpin for securing the rapidly expanding Internet of Things, where devices operate under severe power, memory, and computing constraints. Throughout this survey, we have examined the multifaceted challenges posed by resource-limited endpoints—ranging from low-power sensors and wearable health monitors to smart meters and urban traffic sensors—and shown how tailored primitives such as PRESENT, SIMON, and SPECK deliver high security with minimal gate counts and energy per operation. We have further highlighted how asymmetric schemes like Elliptic Curve Cryptography, optimized through libraries such as TinyECC, enable secure key exchanges on 8-bit microcontrollers, while lightweight hash functions, including PHOTON and Keccak, underpin integrity checks on devices with only a few kilobytes of RAM. Implementation choices, whether in dedicated ASIC or reconfigurable FPGA form or via compact software libraries on general-purpose microcontrollers, directly influence throughput, latency, and battery life—underscoring the need for context-aware design. Standardization initiatives by NIST (via the Lightweight Cryptography project) and ISO/IEC (through the 29192 series), alongside protocol frameworks such as IETF's SCHC, have begun to crystallize interoperable profiles and deployment guidelines, ensuring that diverse IoT ecosystems can adopt standard security baselines. Real-world case studies in innovative healthcare, smart grids, and smart cities have demonstrated that these lightweight solutions can meet stringent performance and energy-efficiency targets without compromising confidentiality or integrity.

Looking ahead, the imperative to future-proof IoT deployments against evolving threats and growing scale demands sustained research across several axes. Post-quantum cryptography must be distilled into ultra-compact, energy-efficient forms capable of running within 4 KB memory footprints. At the same time, novel side-channel countermeasures must achieve provable leakage resilience at low overhead. Scalable key-management schemes—leveraging lightweight key-exchange protocols or identity-based constructs—are essential to coordinate millions of devices with minimal network traffic. Cross-standard harmonization will further reduce fragmentation, enabling seamless updates and certification across vendors. At the same time, integration with energy-harvesting platforms promises battery-less operation, and machine-learning–driven parameter tuning can dynamically balance security and efficiency based on real-time context. Finally, as blockchain and distributed-ledger technologies become integral to supply-chain tracking and decentralized IoT services, lightweight consensus and signature schemes will be critical to extend tamper-evident trust guarantees to the smallest, most constrained nodes. By relentlessly advancing algorithmic

innovation, optimized implementations, and coherent standards, the research community and industry can ensure that lightweight cryptography remains the bedrock of secure, scalable, and sustainable IoT ecosystems.

References

[1] M. El-Hajj, H. Mousawi, and A. Fadlallah, "Analysis of Lightweight Cryptographic Algorithms on IoT Hardware Platform," Future Internet, vol. 15, no. 2, p. 54, 2023.

[2] S. S. Dhanda, B. Singh, and P. Jindal, "Lightweight cryptography: a solution to secure IoT," Wireless Personal Communications, vol. 112, pp. 1947–1980, 2020.

[3] D. N. Gupta and R. Kumar, "Lightweight cryptography: an IoT perspective," Trivium, vol. 80, no. 1, p. 2580, 2019.

[4] V. A. Thakor, M. A. Razzaque, and M. R. A. Khandaker, "Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities," IEEE Access, vol. 9, pp. 28177–28193, 2021.

[5] W. J. Buchanan, S. Li, and R. Asif, "Lightweight cryptography methods," Journal of Cyber Security Technology, vol. 1, no. 3–4, pp. 187–201, 2017.

[6] M. Rana, Q. Mamun, and R. Islam, "Current lightweight cryptography protocols in smart city IoT networks: a survey," arXiv preprint arXiv:2010. 00852, 2020.

[7] L. M. Shamala, G. Zayaraz, K. Vivekanandan, and V. Vijayalakshmi, "Lightweight cryptography algorithms for Internet of Things enabled networks: an overview," in Journal of Physics: Conference Series, 2021, vol. 1717, p. 012072.

[8] M. Abujoodeh, L. Tamimi, and R. Tahboub, "Toward Lightweight Cryptography: A Survey," in Computational Semantics, IntechOpen, 2023.

[9] P. Prakasam, M. Madheswaran, K. P. Sujith, and M. S. Sayeed, "An enhanced energy efficient lightweight cryptography method for various IoT devices," ICT Express, vol. 7, no. 4, pp. 487–492, 2021.

[10] R. Chatterjee, R. Chakraborty, and J. K. Mondal, "Design of lightweight cryptographic model for end-to-end encryption in IoT domain," IRO Journal on Sustainable Wireless Systems, vol. 1, no. 4, pp. 215–224, 2019.