

REVIEW

Open Access



# A review of lightweight intrusion detection and security mechanisms for resource constrained IoT and WSNs

Osama A. Khashan<sup>1\*</sup>

\*Correspondence:

Osama A. Khashan  
okhashan@ra.ac.ae

<sup>1</sup>Homeland Security Department,  
Rabdan Academy, P.O. Box 114646,  
Abu Dhabi, United Arab Emirates

## Abstract

Resource-constrained Internet of Things (IoT) devices and wireless sensor networks (WSNs) remain difficult to secure using conventional security mechanisms because their sensing, forwarding, and processing nodes operate with limited memory, computation, energy, bandwidth, and software-update capability. These limits expose constrained deployments to attacks on end nodes, routing paths, wireless links, and sensed data. Recent research has therefore proposed lightweight intrusion detection (IDS) and protection methods that reduce input dimensionality, model size, computation, communication overhead, authentication cost, or energy use. The literature, however, remains uneven in how it defines lightweightness, relates lightweight design to security risk, and demonstrates deployability under constrained operation. This article reviews recent lightweight security risk analysis and IDS research for constrained IoT and WSN systems, with emphasis on studies published from 2023 to 2026. It examines feature selection, compact machine learning, compressed deep learning, TinyML, edge-assisted detection, federated learning, lightweight authentication, trust-based scoring, cryptography, and energy-aware routing. The review compares technique families, attack coverage, resource-efficiency evidence, datasets, metrics, and reporting practices. It also separates IoT and WSN assumptions because IoT IDS papers often rely on flow datasets and edge deployment, whereas WSN studies more often address routing attacks, energy depletion, topology instability, and node compromise. The main contribution is a resource-aware taxonomy and comparative evaluation framework that separate lightweight security claims from deployment evidence, clarify IoT and WSN evaluation assumptions, and identify the minimum reporting requirements needed to judge operational feasibility. The reviewed studies indicate that resource metrics should be treated as primary evaluation outcomes and that lightweight security must be assessed through the joint assessment of detection quality, risk reduction, and operational feasibility.

**Keywords** Lightweight security, Intrusion detection systems (IDS), Internet of Things (IoT), Wireless sensor networks (WSNs), Resource-constrained devices, Lightweight authentication, Lightweight encryption



## 1 Introduction

Internet of Things (IoT) devices and wireless sensor network (WSN) deployments place sensing, communication, and decision support close to the physical environment. This proximity is useful in monitoring and control applications, but it also changes the security problem. A constrained node is not merely a small computer; it is often a battery-powered device with a narrow radio channel, limited memory, and low tolerance for protocol overhead. Intrusion detection and security risk analysis in such environments therefore cannot be judged only by detection accuracy. A detector may classify attacks correctly, yet still reduce network lifetime, increase packet loss, or become impractical in duty-cycled deployments if it consumes excessive memory or forces frequent transmissions [1]. Prior IoT work shows that security and privacy need to be considered with application requirements, system architecture, and deployment constraints [2]. Standards-oriented reviews also show that interoperability and security are closely linked to device heterogeneity [3].

Existing intrusion detection systems (IDS) surveys for IoT provide useful discussions of learning methods, datasets, validation settings, deployment strategies, and attack taxonomies [4, 5]. However, most of these surveys address IDS development from a broad IoT security perspective. They do not fully examine the narrower question of what makes an IDS or protection method lightweight enough for resource-constrained operation. Since 2023, lightweight IDS research has expanded across feature grouping, ensemble feature selection, quantized autoencoders, compact deep learning (DL) models, and edge-oriented learning [6]. This growth has created a useful but uneven evidence base. Some studies define lightwightness by the number of selected features, others by model size or inference time, and fewer by memory use, energy consumption, communication cost, or hardware execution. As a result, lightweight claims are not always comparable and may overstate deployability.

This review treats lightweight IDS for constrained IoT and WSN systems as a distinct security problem. Traditional heavy IDS architectures, generic DL models, and cloud-only detection designs are not treated as core contributions unless they are explicitly adapted through compression, feature reduction, edge partitioning, federated optimization, or low-cost protocol design. This boundary keeps the review narrower than a general IoT IDS survey and allows a more precise comparison of how different lightweight families reduce overhead and where their security evidence remains incomplete.

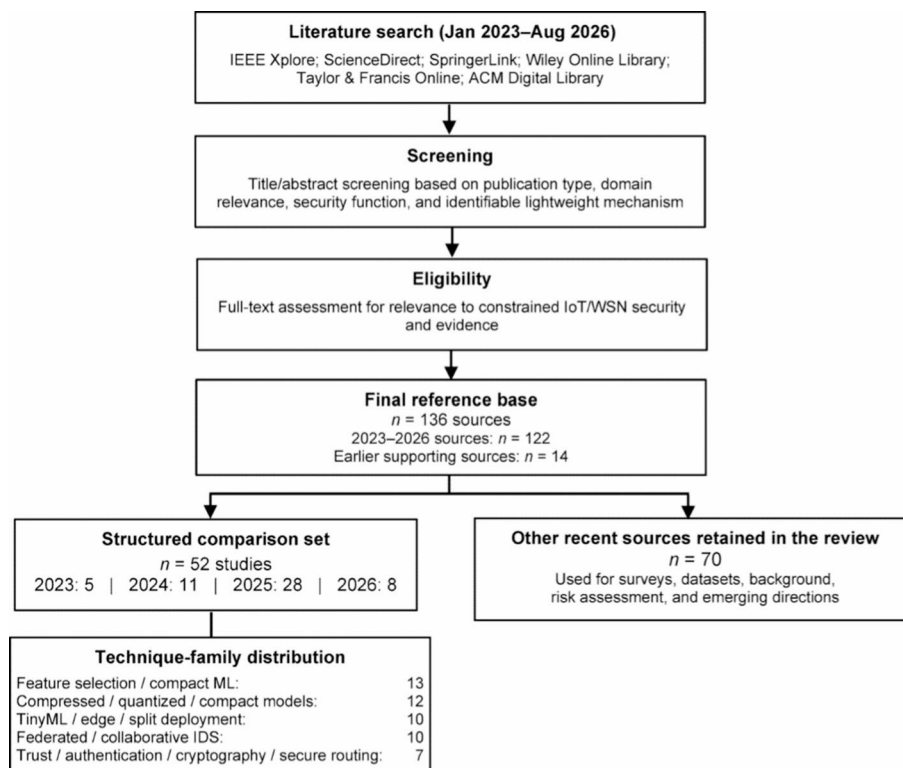
The review has four objectives. First, it identifies recent journal studies that address lightweight security risk analysis, IDS, or protection in IoT, Internet of Medical Things (IoMT), Industrial IoT (IIoT), WSN, edge IoT, IoV, UAV-IoT, and related constrained settings. Second, it develops a taxonomy that separates input reduction, compact machine learning (ML), compressed DL, TinyML, federated IDS, trust/risk scoring, lightweight authentication, cryptography, and secure routing. Third, it compares these families using attack coverage, resource indicators, datasets, and deployment evidence. Fourth, it synthesizes recurring weaknesses, including accuracy-driven evaluation, limited hardware measurement, weak treatment of energy and communication cost, dataset mismatch, and limited linkage between detection outputs and actionable risk reduction. The review therefore provides a constraint-aware comparison rather than an algorithm catalogue.

## 2 Review methodology and selection logic

The review follows a structured critical-review method for study identification, selection, evidence extraction, and comparative analysis. Established literature-review guidance [7] informed the review process. PRISMA [8] was not applied as a formal review protocol because this article was designed as a structured critical review rather than a systematic review or meta-analysis. Figure 1 summarizes the documented study-selection stages and retained-source counts.

The main literature search covered publications from January 2023 to August 2026, with particular attention to recent 2025–2026 studies in international peer-reviewed journals. A final search update was conducted on 28 August 2026. Searches were conducted in IEEE Xplore, ScienceDirect, SpringerLink, Wiley Online Library, Taylor & Francis Online, and the ACM Digital Library. The search terms were organized into three groups: application domain (IoT, IIoT, IoMT, WSN, edge IoT, IoV, and UAV-IoT), security function (intrusion detection, IDS, authentication, trust, cryptography, and secure routing), and lightweight design (lightweight, resource-constrained, feature selection, compression, quantization, TinyML, federated learning, and energy-efficient). Terms within each group were joined by OR, and the three groups were joined by AND. The syntax was adjusted to the search functions of each database. Earlier studies were retained only for foundational methods, benchmark datasets, standards, established security assumptions, or relevant background. The main corpus was limited to peer-reviewed journal research and review articles. Conference papers and preprints were considered only when needed to trace a method or identify a later journal version.

Search results were consolidated and duplicate journal records were identified using title, authorship, and DOI information before screening. Study selection was conducted



**Fig. 1** Literature search, screening, eligibility assessment, and selection of the structured comparison set

in two stages. Title and abstract screening first assessed publication type, domain relevance, security function, and the presence of an identifiable lightweight mechanism. Retained studies were then examined in full text for relevance to constrained IoT/WSN security and for evidence supporting the claimed lightweight design. Studies that did not satisfy the inclusion criteria in Table 1 were excluded at the corresponding screening stage.

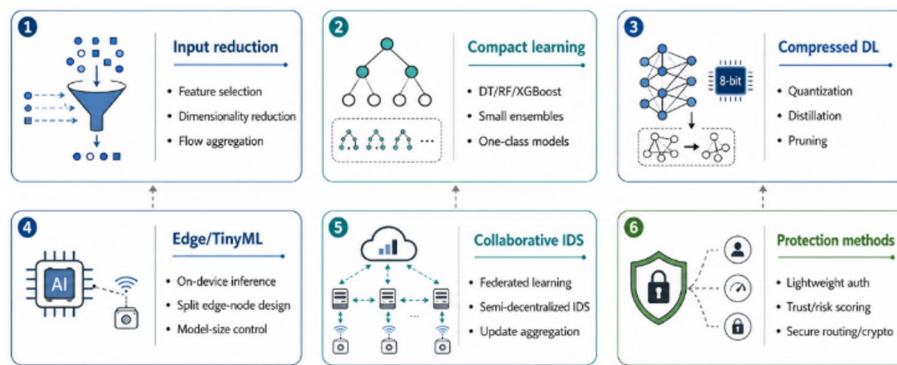
Eligible environments included IoT, IIoT, IoMT, WSNs, edge IoT, IoV, UAV-assisted IoT, and related constrained cyber-physical settings. Eligible mechanisms included feature reduction, model compression, quantization, compact learning, TinyML, edge-assisted processing, federated learning (FL), trust/risk scoring, lightweight cryptography, low-overhead authentication, and energy-aware routing. For each study, the review extracted the application domain, security function, lightweight mechanism, dataset or simulation environment, deployment layer, detection or protection metrics, resource indicators, implementation evidence, and reported limitations. Figure 2 summarizes the resulting technique families.

The final reference base contains 136 sources: 122 publications from 2023 to 2026 and 14 earlier sources retained for foundational methods, benchmark datasets, standards, or background. The 52-study comparison set in Sect. 5 was drawn from the 122 recent sources. Selection for this set required representation of a principal technique family, an identifiable lightweight or resource-reduction mechanism, and sufficient security, resource, or deployment evidence for study-level comparison. The other 70 recent sources support quantitative evidence, datasets, surveys, domain context, risk assessment, background, and emerging directions but are not part of the Table 5 comparison.

No standardized study-quality or risk-of-bias instrument was applied. Studies included in the 52-study comparison had to satisfy the minimum evidence-quality criterion in Table 1, while deployment-evidence maturity was assessed separately through four tiers. E0 denotes design-level evidence: the lightweight claim is supported by

**Table 1** Inclusion and exclusion criteria used for study selection

| Criterion                | Included                                                                                                                                                                                                                                               | Excluded                                                                                                                                                       |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Publication type         | Peer-reviewed journal research articles and relevant review articles.                                                                                                                                                                                  | Conference-only papers, preprints without a journal version, and non-peer-reviewed publications.                                                               |
| Publication period       | January 2023–August 2026 for the principal recent-literature analysis; earlier studies retained only for foundational methods, benchmark datasets, standards, or established security assumptions.                                                     | Older generic IDS studies without direct relevance to constrained IoT or WSN security.                                                                         |
| Domain                   | IoT, IIoT, IoMT, WSNs, edge IoT, IoV, UAV-IoT, and other constrained device networks                                                                                                                                                                   | Enterprise or cloud network IDS studies without constrained-device deployment context.                                                                         |
| Technique                | Feature selection, compact ML, compressed DL, TinyML, federated IDS, trust-based scoring, lightweight authentication or cryptography, and secure or energy-aware routing.                                                                              | Cloud-only IDS, generic DL models, or security methods without resource adaptation.                                                                            |
| Evidence                 | An identifiable lightweight mechanism supported by at least one explicit resource, design, implementation, or deployment indicator, including feature/model reduction, memory, latency, energy, communication cost, or constrained-hardware execution. | Studies reporting only conventional detection metrics without an identifiable lightweight mechanism, except where retained explicitly as comparative contrast. |
| Minimum evidence quality | Clear security objective and method, identifiable evaluation setting, appropriate evaluation metrics, and sufficient methodological detail to interpret the reported evidence. Resource and deployment indicators were recorded where applicable.      | Insufficient methodological or evaluation information to determine what was tested or to interpret the reported security or lightweight claim.                 |



**Fig. 2** Taxonomy of lightweight security and IDS technique families for resource-constrained IoT and WSNs. Auth, authentication; crypto, cryptography; DL, deep learning; DT, decision tree; IDS, intrusion detection system; ML, machine learning; RF, random forest; TinyML, tiny machine learning; XGBoost, Extreme Gradient Boosting

feature reduction, architectural simplification, model reduction, complexity analysis, or detection results, but no direct resource measurement is reported. E1 denotes quantitative software or simulation evidence: at least one resource or operational measure is reported, such as model footprint, execution time, simulated energy, communication overhead, memory estimate, or network lifetime. E2 denotes platform-profiled evidence: resource use is measured directly on a representative embedded, gateway, edge, or other target-class platform. E3 denotes constrained-hardware validation: the method is implemented on the intended or closely representative constrained hardware, and resource use is measured under an operational or networked workload.

Each study receives the highest tier supported by its reported evidence. Tier assignment was based only on evidence reported in the article; unreported implementation or resource characteristics were not inferred. The tier reflects deployment-evidence maturity rather than the overall scientific quality of the study. The resulting tier distribution is examined in Sect. 6.1 as part of the critical evaluation of the reviewed evidence.

Throughout the review, lightweight denotes an explicit reduction in resource or operational cost, while resource-efficient denotes a quantitatively demonstrated reduction relative to a stated baseline or reference condition. Claims of constrained-device suitability require at least E2 evidence, while deployment-ready is reserved for E3 evidence. Fixed numerical thresholds are not imposed because resource budgets differ across device classes and deployment layers.

### 3 Positioning of the review and evaluation lens

#### 3.1 Existing surveys and review gap

Existing surveys provide useful context for this review. Broad IoT security surveys usually discuss architectures, attacks, datasets, privacy issues, and general security requirements, but they are not designed to examine recent lightweight methods at the level of feature count, model footprint, inference cost, and deployment suitability [2–5]. More focused reviews on lightweight IDS and feature selection have narrowed the discussion, yet many remain organized around a single method family and do not sufficiently connect IDS performance with risk reduction and protection cost in constrained systems [6]. A recent PRISMA-ScR review also examined deployability evidence in lightweight IoT-edge IDSs, with particular attention to hardware and operational evidence [136]. Its scope remains centered on IDS and IoT-edge deployment, whereas the present review

additionally considers WSN-specific security, authentication, cryptography, trust, secure routing, and risk management.

Here, lightweight security is treated as a set of complementary mechanisms rather than as IDS alone. IDS remains central to the analysis, but security in constrained IoT and WSN environments also depends on low-cost authentication, secure routing, trust scoring, and energy-aware protection. A feature-selected IDS can detect Distributed Denial-of-Service (DDoS) traffic flows, a trust-based routing rule can keep a suspicious node out of a route, and a lightweight authentication protocol can limit replay or impersonation before traffic reaches a classifier. These mechanisms need to be discussed together, but only when their resource assumptions and security roles are kept distinct.

The main gap in current review coverage is not the absence of lightweight IDS studies. The gap is the lack of a consistent basis for comparing resource cost, security role, and deployment relevance. One study may report a compact model with high accuracy [9], another may reduce the input space through ensemble feature selection [10], a third may apply quantization [11], and another may use FL to limit raw-data sharing [12]. All may be described as lightweight, but they do not impose the same computation, memory, energy, communication, or deployment burden. Recent FL and collaborative IDS studies also show that distributed detection must be evaluated through client workload, update traffic, and non-IID behavior [13–18]. Related work on trust, authentication, cryptography, routing, datasets, and deployment-oriented evaluation further shows why protection support, benchmark coverage, and resource evidence should be separated [19–35]. Table 2 positions this review relative to related survey directions and explains why a resource-aware comparison remains necessary.

The taxonomy developed in this review is an analytical synthesis of recurring lightweight-design strategies identified across the reviewed literature rather than a new IDS architecture or security algorithm. It organizes heterogeneous studies according to the mechanism used to reduce resource cost, the security function performed, the intended deployment layer, and the evidence supporting constrained operation. The categories are not mutually exclusive because an individual approach may combine multiple mechanisms, such as feature selection with quantization or federated learning with lightweight cryptography.

### 3.2 Threat, resource, and evidence evaluation framework

A lightweight protection method should be read through two linked views: the threat model and the resource model. In IoT flow datasets, the reported attack classes usually

**Table 2** Positioning of this review relative to relevant survey directions

| Survey direction               | Typical coverage                                              | Limitation for this topic                                | Role in this review                        |
|--------------------------------|---------------------------------------------------------------|----------------------------------------------------------|--------------------------------------------|
| Broad intelligent IoT surveys  | Applications, architectures, security/privacy challenges      | Too broad for technique-level lightweight IDS comparison | Provides organization and context [2]      |
| IoT standards/security surveys | Interoperability, protocols, standards, security requirements | Limited study-level comparison of lightweight methods    | Clarifies system constraints [3]           |
| General IoT IDS surveys        | Attacks; learning methods, datasets, deployment strategies    | Often mixes heavy and lightweight methods                | Defines IDS background [4, 5]              |
| Feature-selection IDS surveys  | Feature reduction and classifier selection                    | Does not cover cryptography, trust, routing, or FL       | Supports lightweight IDS taxonomy [6]      |
| WSN security reviews           | Routing attacks, WSN constraints, energy risks                | Often separate from modern IoT datasets and compact ML   | Grounds the WSN-specific analysis [33, 34] |

include DDoS, botnet traffic, scanning, reconnaissance, brute-force access, and malware communication [31, 32]. In WSN and RPL-like environments, the threat surface also includes sinkhole, wormhole, Sybil, selective forwarding, replay, rank manipulation, and energy-drain attacks [33, 34]. The first group is normally studied through traffic records. The second requires topology-aware and energy-aware interpretation because the effect of an attack may appear through route changes, packet loss, or battery depletion rather than through flow features alone.

The resource model is no less important. A node-level IDS may be constrained by CPU cycles, RAM, flash storage, battery capacity, and duty-cycle behavior. An edge-level IDS can use more computation, but it may still face limited bandwidth, intermittent connectivity, and privacy restrictions. A federated IDS reduces raw-data exchange but adds model updates and communication rounds. A blockchain-assisted method may improve auditability, but the benefit may come with storage, communication, and consensus overhead. These trade-offs make accuracy an incomplete measure for judging lightweight security methods [11, 35].

For cross-study comparison, the review considers four dimensions: security function, lightweight mechanism, resource effect, and deployment evidence. Security function identifies whether a method detects, prevents, limits, or responds to a specified threat. The lightweight mechanism identifies the source of resource reduction, such as feature reduction, model compression, reduced numerical precision, edge partitioning, lower protocol cost, or reduced communication. Resource evidence covers computation, memory, energy, latency, communication cost, and other measures relevant to the intended execution layer. Deployment evidence is assessed through the E0–E3 classification defined in Sect. 2.

The reported resource values are not combined into a universal numerical score because the reviewed studies differ in hardware platform, workload, dataset, unit of measurement, and execution layer. Each lightweight claim is instead interpreted according to the resource that is reduced or measured and the maturity of the supporting deployment evidence. Table 3 relates the main threat types to their relevant resource constraints, while Table 4 summarizes the execution layers and resource-evidence requirements across the reviewed deployment settings.

### 3.3 Intrusion detection, risk assessment, and management Across IoT ecosystems

IoT ecosystems differ in device capability, communication structure, service criticality, mobility, and tolerance for security overhead. These differences affect the threat landscape, the role of IDS, and the evidence required to establish practical lightweight operation. General IoT environments commonly involve heterogeneous constrained devices and network-level threats such as DDoS, botnet activity, scanning, reconnaissance, malware communication, and unauthorized access. IIoT environments place greater emphasis on service availability, process continuity, reliability, timing constraints, privacy, and industrial edge or gateway operation. Recent federated and edge-oriented IIoT studies illustrate the importance of communication cost, client heterogeneity, latency, and deployment hardware in addition to detection performance [68, 119, 129].

IoMT introduces additional requirements because medical sensors, wearable devices, and healthcare gateways handle sensitive health information and may support continuous or time-critical monitoring. Security mechanisms in this environment must

**Table 3** Threat types and the role of resource constraints in lightweight evaluation

| Threat               | Typical evidence sources                               | Why resource constraints matter                                        | Relevant method families                            |
|----------------------|--------------------------------------------------------|------------------------------------------------------------------------|-----------------------------------------------------|
| DDoS/botnet          | Flow datasets such as CICIoT2023, Bot-IoT, and TON_IoT | High traffic volume can exhaust energy and radio capacity              | Feature selection, compact ML, and edge IDS         |
| Sinkhole/wormhole    | WSN/RPL simulations or routing traces                  | Topology manipulation affects both detection accuracy and routing cost | Secure routing, trust scoring, and WSN IDS          |
| Sybil/replay         | Protocol traces and authentication analysis            | Lightweight identity checks are required before traffic analysis       | Lightweight authentication and trust models         |
| False data injection | Sensor readings and application-level behavior         | Detection must separate malicious data from noisy or faulty sensing    | Anomaly detection and trust/risk scoring            |
| Node capture         | Physical compromise models or behavioral assumptions   | Cryptographic keys and routing trust may be compromised                | Authentication, key management, and local trust     |
| Energy drainage      | Traffic, MAC/routing activity, and duty-cycle behavior | Attack impact is measured through network lifetime, not only packets   | Energy-aware IDS, secure routing, and rate limiting |

**Table 4** Deployment assumptions and resource-evidence requirements across the reviewed settings

| Setting       | Typical execution layer                   | Dominant constraints                                             | Principal lightweight evidence                                                                    |
|---------------|-------------------------------------------|------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| IoT           | End device, gateway, or edge              | Memory, computation, bandwidth, and latency                      | Feature/model size, memory, inference latency, communication cost, and target platform            |
| IIoT          | Field device, industrial gateway, or edge | Availability, reliability, timing, and processing load           | Latency, throughput, memory, hardware load, and deployment layer                                  |
| WSN           | Sensor node, cluster head, or sink        | Battery energy, radio communication, memory, and topology        | Energy, communication/routing overhead, packet delivery, memory, and network lifetime             |
| TinyML        | Microcontroller or embedded end device    | RAM, flash, energy, and inference time                           | RAM, flash footprint, latency, power/energy, and target board                                     |
| Edge IDS      | Gateway or edge server                    | Throughput, latency, buffering, and offloading cost              | Latency under load, throughput, memory, transferred traffic, and node-edge cost separation        |
| Federated IDS | Distributed clients and aggregator        | Local training, update traffic, client heterogeneity, and energy | Update size, communication rounds, local workload, energy, aggregation cost, and non-IID behavior |

therefore account for patient safety, privacy, reliability, response latency, and constrained-device resources [91]. Recent IoMT studies address different parts of this problem. ElRabaa et al. [125] retain the computational simplicity of Naïve Bayes but adapt its statistical model to skewed IoMT traffic. Jodayree et al. [126] focus on protocol-aware zero-day detection and cross-dataset generalization, while Hosain and Çakmak [127] combine IDS with explainability. Palaniappan and Sengan [128] reduce the input space through hybrid feature selection before deep classification. Together, these studies show that IoMT security may emphasize computational simplicity, generalization, interpretability, or feature reduction, but these properties represent different forms of deployment evidence and should not be treated as equivalent lightweight claims.

Mobility-oriented ecosystems impose a different set of constraints. IoV security must operate under rapidly changing connectivity, low-latency communication,

heterogeneous in-vehicle and external networks, and strict real-time requirements. Recent lightweight IoV research applies compact sequence modeling, feature reduction, knowledge distillation, and differentiated in-vehicle and external-network detection to address these constraints [49]. UAV-IoT systems face similar mobility and connectivity issues together with platform energy and computational limits. Knowledge-distillation approaches in [48] and the distilled and pruned UAV detector proposed by Medhi et al. [130] reduce model complexity and inference cost while preserving detection capability. These examples reinforce the need to interpret IDS performance according to the execution environment rather than assuming that one lightweight design is equally suitable across IoT ecosystem variants.

This distinction becomes particularly important when generic and lightweight IDS approaches are evaluated using the same benchmark datasets. Similar accuracy, precision, recall, or F1-score values establish comparable detection performance under the selected experimental conditions, but they do not establish equivalent deployment suitability. Generic IDS may rely on larger models, extensive preprocessing, or comparatively powerful edge or cloud resources, whereas a lightweight solution should demonstrate a measurable reduction in one or more relevant costs. Such evidence may include feature-processing burden, model footprint, memory demand, inference latency, energy consumption, communication overhead, or hardware execution [125, 128–130]. For example, recent IIoT work combines federated learning with lightweight model components for resource-constrained 5G-IIoT settings [129], while UAV-oriented work reports substantial reductions in model size and inference time [130]. The benchmark dataset should therefore establish the detection context, while resource and deployment measurements should establish whether the reported lightweight claim is supported for the intended ecosystem.

IDS is also one component of a broader security-risk process. IDS identifies evidence of malicious or abnormal activity, whereas risk assessment considers threats, vulnerabilities, likelihood, consequences, and the importance of affected assets when mitigation priorities are determined. Waqdan et al. [131] provide a recent taxonomy of IoT security-risk assessment and management approaches. Czekster et al. [132] propose lightweight dynamic risk assessment for resource-constrained medical IoT through scenario-based simulation and threat modeling. Zhai et al. [133] apply STRIDE and DREAD to identify and prioritize threats across healthcare IoT devices. More recently, Prasanna et al. [135] integrate IDS, vulnerability scanning, and CVSS-based severity information within an IoT risk-scoring framework. These studies demonstrate that successful attack classification does not by itself determine the significance or priority of a security event.

Risk management is particularly important in resource-constrained systems because mitigation also consumes resources. Node isolation, additional authentication, key renewal, trust reduction, rerouting, increased monitoring, or communication restriction can introduce computation, energy, delay, and message overhead. The appropriate response therefore depends on detection confidence, attack severity, asset importance, residual risk, available resources, and the operational consequence of the response.

This perspective links IDS with operational risk management and complements the threat- and resource-aware evaluation framework in Sect. 3.2 and the cross-family risk analysis in Sect. 5.4.

## 4 Lightweight technique families

### 4.1 Feature selection and compact ML

Feature selection is among the most developed lightweight IDS approaches in the reviewed corpus. Its role is straightforward: the detector processes fewer traffic, protocol, or behavioral variables before classification begins. This reduction can lower pre-processing time, model complexity, memory demand, and edge-execution cost. He et al. [36] used feature grouping for IoT IDS so that related variables are organized before classification. Fatima et al. [10] used feature selection as the core of ELIDS, targeting DDoS detection in resource-constrained IoT through an ensemble selection process. Zhang et al. [9] used two-stage feature selection with Bayesian optimization, linking feature reduction with model tuning rather than treating the two steps separately.

Several recent studies take this idea further by making the reduction part of the detection pipeline itself. Li and Yao [37] used a two-stage design in which the early stage reduces the detection burden before the final decision stage. Kushwaha et al. [38] adopted an mFCBF-based filter to obtain a smaller feature subset for IoT IDS. Mohammed et al. [39] used PCA-based dimensionality reduction with ensemble learning for real-time IIoT IDS. Almalawi [40] proposed a clustering-based lightweight IDS for IoT using Monte Carlo cross-entropy, which adds an optimization-oriented direction to input reduction. Benaddi et al. [41] used chi-square feature selection in a hybrid IoT IDS, where the reduced feature space supports lighter classification. The common point across these studies is not the particular selector used, but the attempt to reduce the amount of information handled by the classifier without abandoning conventional IDS models.

This family is attractive because it can improve deployability without requiring a complete change in the security architecture. In many IoT settings, gateways or edge devices already collect flow statistics, and a smaller input space can be introduced with limited disruption. The more difficult issue is feature stability. A subset selected from one dataset may not hold when device behavior, packet timing, routing state, or attack composition changes. This issue is more evident in WSN settings, where residual energy, routing behavior, and topology changes may be more useful than ordinary traffic features.

Compact classical ML overlaps with feature selection, but the two are not identical. Decision trees, random forests, shallow ensembles, one-class models, and support vector machine (SVM)-based methods are often easier to interpret than large neural networks. Their cost still depends on model structure and deployment point. A random forest may require considerable memory, k-nearest neighbors (KNN) can be expensive at inference time, and an ensemble that is reasonable at an edge gateway may be too heavy for a sensor node. Pandey et al. [42] combined tabu search with random forest for WSN IDS, while Talukder et al. [43] used data balancing and dimensionality reduction in a hybrid WSN IDS model. Srivastava and Sinha [44] presented a lightweight hybrid NIDS for IoT that combines signature and anomaly logic to improve robustness without depending on one detector. Zakariyya et al. [45] focused on resource-aware ML for IoT security monitoring, where model cost is treated as part of the security argument rather than as a secondary detail. These works are useful because they bring the discussion closer to WSN and IoT requirements, although their lightweight evidence still depends on how clearly energy, memory, and runtime cost are reported.

Selected feature count, preprocessing cost, model size, inference latency, memory use, and deployment layer provide the evidence needed to assess a lightweight claim. Without these details, a study may demonstrate useful simplification, but the evidence for constrained deployment remains limited.

#### 4.2 Compressed DL, quantization, and distillation

A second group of lightweight IDS studies retains the DL model while reducing its cost. This differs from feature selection, where the main reduction occurs before classification. In compressed DL, the reduction is applied to model representation, numerical precision, parameter structure, or the teacher-student relationship. The approach is attractive when attack behavior is nonlinear or sequential, but a smaller neural model is not automatically suitable for constrained deployment. The compression procedure should be visible in the method, and the evaluation should show whether the reduction is useful beyond detection accuracy.

Sharmila and Nagapadma [11] provide a clear example of anomaly detection on resource-constrained IoT devices using a quantized autoencoder on the RT-IoT2022 dataset. The relevance of this study comes from the use of quantization as part of the detector design. In [46], dynamic quantization is combined with feature engineering for an IoT IDS, which places the study between input preparation and model-cost reduction. These works are not ordinary DL-based IDS studies with a lightweight label added later. Their value, however, still depends on whether the reported reduction includes the surrounding pipeline, especially feature construction, normalization, and other preprocessing steps. Wang et al. [47] follow a different compression route by using an improved BERT-of-Theseus strategy for IoT IDS. The method reduces a larger representation model through teacher-student replacement, so the lightweight claim is tied to model substitution rather than to feature filtering. A similar logic is followed in [48], where knowledge distillation is used for an IoT/UAV IDS so that a smaller student model can retain relevant detection behavior from a larger teacher model. A more recent example is LiteKD [117], which transfers knowledge from a CNN-BiGRU teacher to a compact depth-wise separable CNN-GRU student through adaptive-temperature distillation. The student retains approximately 99% of the teacher performance while reducing model size, parameter count, and computational overhead by more than 98.8%. Its suitability for resource-constrained edge or mobile settings, however, also depends on practical deployment evidence.

Recent work also includes compact sequence and quantized non-neural models. The Mamba-based IDS in [49] uses compact sequence modeling for intrusion detection in constrained vehicular networks, where the expected benefit is the ability to handle temporal attack behavior with lower sequence-model cost. The quantized XGBoost study in [50] uses quantization in a tree-based detector for botnet detection, which is a different form of cost reduction from distillation or neural compression. These studies should therefore be interpreted separately: Mamba-like models address sequence behavior; distillation reduces the cost of learned representations, and quantized tree models target storage or arithmetic efficiency in a structured classifier. More explicit model-footprint evidence is reported by Ullah et al. [116], who proposed KronNet, a compact feed-forward IDS that combines Kronecker-based representation with post-training quantization. The reported post-quantization footprints were 4.96 KB for CICIoT2023 and 4.59

KB for Bot-IoT, with inference times of approximately 0.21 ms. These results provide a useful example of lightwightness supported by direct model-size and runtime measurements rather than detection performance alone.

The recurring weakness in this family is not compression itself, but the limited connection between compression results and deployment evidence. A compressed model may be smaller than the original model, and a quantized detector may use fewer bits, but deployment suitability still depends on where the detector runs, how much memory it needs, how long inference takes, and whether preprocessing is also affordable. If feature extraction, normalization, update frequency, or edge coordination remains expensive, the lightweight claim becomes weaker even when the final detector is compact. Stronger compressed IDS studies therefore evaluate compression ratio, model size, latency, memory use, and deployment layer together rather than reporting detection performance alone.

#### 4.3 TinyML, edge-aware IDS, and split deployment

TinyML and edge-aware IDS should not be classified only as model-compression approaches. In these studies, IDS functions are often distributed across the sensing device, gateway, edge server, or cloud boundary. This placement matters because the cost of detection is often shifted from one layer to another rather than removed completely. Nassef et al. [51] present an edge-oriented IIoT IDS with federated learning and preliminary Raspberry Pi profiling. The study provides edge-level resource evidence, while further TinyML adaptation is required for more constrained endpoints. More direct on-device evidence is provided by Ortiz-Garcés et al. [118] through a NAS-optimized TinyML IDS designed for ultra-low-power IoT edge devices. Hardware-level evaluation reports 2.48 ms inference latency, less than 20 KB RAM use, and 0.39 mJ per inference, together with cross-dataset evaluation across heterogeneous IoT traffic sources. Tiwari et al. [52] provide a more deployment-oriented example by combining feature selection, feature reduction, model quantization, and Azure IoT Edge deployment for edge-based IIoT security. Alani [53] extends the discussion to industrial flow-based IDS, where efficiency must be considered alongside explainability if the detector is expected to support operational decisions.

This layering is also visible in studies that place heavier analysis outside the constrained device. Gao et al. [54] propose a lightweight multiclass IDS at the edge. This is not the same task as binary screening, because multiclass detection must separate multiple attack types and usually requires a richer decision model. Khashan [55] presents a WSN-specific split-deployment design in which suspicious behavior is first handled at the cluster head level and deeper malicious node analysis is then moved to a server-side model. The value of such designs is that they do not force the weakest node to perform all IDS functions.

The lightweight claim in this family can come from different sources: a small local detector, an efficient edge- or server-side classifier, or a staged pipeline that avoids unnecessary offloading. These forms are related but not interchangeable. A lightweight edge method may still be unsuitable for a constrained node if the node must collect features, buffer traffic, transmit intermediate data, or communicate repeatedly with an edge server. The evaluation should therefore reflect the intended deployment layer. A WSN sensor, a microcontroller board, a Raspberry Pi-class gateway, an industrial edge server,

and a vehicle gateway have different limits in memory, energy, latency, and communication. Node-level claims require evidence on RAM and flash usage, local delay, and energy consumption. Edge-level claims require evidence on throughput, latency under load, update cost, and communication savings. Flow-window construction, feature extraction, neighborhood statistics, and routing evidence should also be counted when they are required by the detector. Without these details, the results mainly show efficiency on the selected platform, but they do not fully establish suitability for constrained IoT devices or WSN nodes.

#### 4.4 Federated and collaborative lightweight IDS

FL and collaborative IDS belong to the lightweight-security discussion because they change how IDS evidence is shared and learned. Instead of collecting traffic records or device logs at a central point, each participant keeps its data locally and sends model updates. This is useful in IoT environments where traffic traces, industrial records, or site-specific behavior cannot be freely transferred across gateways, devices, or organizations [56]. This does not make FL lightweight by default. FL reduces raw-data movement but introduces local training, update transmission, aggregation, synchronization, and protection against poisoned updates [13]. In constrained IoT or WSN settings, repeated training rounds can become more expensive than running a small local detector. FL-based IDS should therefore be judged by communication rounds, update size, client workload, aggregation cost, and participating device type, not by accuracy alone.

Recent studies approach this issue from different directions. Danquah et al. [14] propose a computationally efficient FL-based IDS for IoT botnet attack detection, where distributed client training reduces the need for centralized raw-data pooling while feature selection and dimensionality reduction lower the computational burden. Alsaleh et al. [15] address client heterogeneity through a semi-decentralized FL design with BiLSTM, which is relevant when participating devices differ in bandwidth, data distribution, and computation capacity. Tursynbek et al. [16] move the discussion closer to constrained hardware by considering secure FL for IoT microcontrollers. Bukhari et al. [17] add a WSN-specific example by using FL with an SCNN-Bi-LSTM model, allowing sensor nodes to participate in collaborative IDS training without exposing their local data. Peng et al. [18] provide a distinct efficiency-oriented FL-based IDS direction through FD-IDS, which combines mutual-information feature selection with round-wise knowledge distillation to reduce feature redundancy, limit computational overhead, and improve learning under non-IID IoT data. Zhou et al. [119] propose a lightweight federated IIoT IDS based on two-stage knowledge distillation. The framework combines personalized and self-knowledge distillation with heterogeneity-aware aggregation and achieves over 97% accuracy on ToN-IoT and Edge-IIoTset while addressing model efficiency, non-IID data, and privacy.

In FL-based IDS, the deployment layer cannot be treated as a secondary design detail. A scheme trained across edge gateways cannot be evaluated in the same way as one involving microcontrollers, sensor nodes, or other low-power IoT devices. Each setting places a different burden on the client, especially during local training, update transmission, and aggregation. For this reason, the evaluation should make the training setting explicit. This includes the training location, the size of exchanged updates, the number of communication rounds, and the workload assigned to each participant. It should also

account for non-IID data, poisoning attempts, and hardware constraints, since these factors directly affect whether FL remains practical in constrained deployments. Accuracy is still part of the evidence, but it is not sufficient to justify a lightweight FL-based IDS unless the added training and communication costs remain manageable under IoT or WSN constraints.

#### 4.5 Lightweight trust, authentication, cryptography, and secure routing

Lightweight protection in constrained IoT and WSNs is not limited to traffic-based IDS. In many deployments, protection starts earlier in the communication path. It may verify a device before access, assign trust before route selection, protect sensed data before transmission, or exclude an unreliable forwarding node before classification is needed. These mechanisms serve a different role from IDS because they reduce the opportunity for attack rather than only detecting traces after traffic becomes observable.

Recent studies present various lightweight protection approaches. Pathak et al. [19] propose NATURE as a trust evaluation model for resource-constrained industrial WSNs, where node reliability supports secure cooperation and routing decisions. Alghamdi [20] presents a lightweight IIoT authentication protocol that focuses on low communication and computation cost while maintaining resistance to common protocol attacks. Sai et al. [21] address fault-tolerant WSN authentication using elliptic-curve mechanisms, with attention to execution time and memory usage. Jan et al. [22] introduce FEELAP, a fuzzy-extractor-based lightweight authentication protocol for edge-IoT e-healthcare that reduces the burden on constrained healthcare devices by offloading computation to the edge server. Khashan et al. [23] propose an energy-efficient proxy re-encryption scheme for secure data exchange in WSNs, with emphasis on computation cost, energy consumption, secure re-encryption, and network efficiency. Selvi et al. [24] propose ABE-TBSRA, where trust-aware routing and attribute-based encryption are combined to protect WSN communication paths. Kavitha et al. [120] propose a knowledge-based adaptive routing method that combines attack detection with energy-aware route adaptation in WSNs. The approach jointly considers security, energy efficiency, quality of service, and adaptability.

These studies should be compared according to the function they protect. Trust models need supporting results on trust accuracy, false isolation, message overhead, residual energy, and resistance to collusion or bad-mouthing. Authentication protocols need reported measurements on message count, computation cost, storage demand, delay, and resistance to replay, impersonation, stolen-verifier, and man-in-the-middle attacks. Lightweight cryptography and selective encryption require results on encryption time, energy use, key management, storage, and communication overhead. Secure routing needs evaluation results on packet delivery ratio, routing overhead, route stability, and network lifetime. These requirements differ from ordinary traffic-classification metrics, but they are essential for lightweight protection in IoT and WSNs.

### 5 Cross-family comparative analysis and evaluation evidence

This section moves from technique-level discussion to cross-family comparison. It examines how recent studies substantiate lightweight claims rather than merely identifying the methods they use. A feature-selected IDS, a compressed DL detector, an edge-assisted model, an FL-based IDS, and a lightweight authentication protocol may all be

described as lightweight, but each one reduces a different cost and protects a different part of the system. The comparison is therefore organized around method function, reduced overhead, deployment context, and the evidence required before a study can be considered suitable for IoT or WSN operation.

The representative comparison includes 52 recent studies: 5 (9.6%) published in 2023, 11 (21.2%) in 2024, 28 (53.8%) in 2025, and 8 (15.4%) in 2026, with coverage for 2026 extending through August. By technique family, the comparison includes 13 feature-selection or compact-ML studies, 12 compressed, quantized, or compact-model studies, 10 TinyML, edge, or split-deployment studies, 10 federated or collaborative IDS studies, and 7 studies on trust, authentication, cryptography, or secure routing. The studies further span six deployment settings, seven principal datasets, and fourteen detection and resource metrics, as detailed in Tables 4, 6, and 7.

### 5.1 Study-level comparison of recent lightweight studies

The study-level comparison focuses on the mechanism used to reduce resource cost, the intended deployment setting, the evidence that supports the lightweight claim, and the main limitation of that evidence. Table 5 compares the 52 representative studies across these dimensions and reports the corresponding deployment-evidence tier. Technique families are retained as group headings rather than repeated in a separate column.

### 5.2 Dataset and metric evidence

Evaluation evidence is the point at which a lightweight claim becomes credible or remains mainly descriptive. A result based only on accuracy, precision, recall, or F1-score shows detection behavior, but it does not show whether the method fits a sensor node, WSN cluster head, microcontroller, or edge gateway. Dataset selection and metrics should therefore be interpreted together with the deployment assumptions behind each study. A node-level IDS needs memory, flash, latency, energy, and local communication evidence. A gateway or edge IDS needs throughput, buffering, latency under load, and update cost. A federated IDS needs communication rounds, update size, and local training cost. A protocol or routing method needs message count, computation delay, packet delivery, residual energy, and route overhead.

The original source for each dataset was cited whenever available. Edge-IIoTset [31] and TON\_IoT [25] support IoT/IIoT evaluation through network and telemetry views. CICIoT2023 [26] provides broad large-scale IoT attack coverage. Bot-IoT [27] and N-BaIoT [28] are useful for botnet-oriented evaluation. WSN-DS [29] is closer to routing and behavioral patterns in WSNs, while RT-IoT2022 [11] has been used in resource-constrained anomaly detection. These datasets are useful, but they do not represent the same security setting. A flow-based IoT dataset should not be used as proof for WSN routing attacks, and a WSN dataset should not be used as support for smart-home botnets or IIoT telemetry.

Metric interpretation requires the same caution. Accuracy can hide class imbalance, weighted F1 can hide weak minority-class detection, and ROC-AUC does not show the cost of response. False positives consume energy and communication resources, while delayed detection can allow forwarding, routing, or energy-drain attacks to continue. A lightweight evaluation is strongest when detection metrics and resource metrics are reported together [11, 16, 46, 51, 52]. Table 6 summarizes the evaluation role and

**Table 5** Study-level comparison and deployment-evidence maturity of the 52 representative lightweight IDS and protection studies

| Refs.                                     | Tier | Lightweight mechanism                             | Deployment context                         | Evidence contribution                                                                                | Main evidence limitation                                            |
|-------------------------------------------|------|---------------------------------------------------|--------------------------------------------|------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| Feature selection and compact ML          |      |                                                   |                                            |                                                                                                      |                                                                     |
| [36]                                      | E0   | Input reduction before classification             | IoT flow IDS                               | Feature grouping and reduced classifier input                                                        | Cross-dataset validation remains limited                            |
| [10]                                      | E0   | Reduced feature subset                            | DDoS in constrained IoT                    | Reduced feature set for DDoS detection                                                               | Attack coverage is mainly limited to DDoS                           |
| [9]                                       | E1   | Feature selection with Bayesian optimization      | IoT IDS                                    | Feature reduction, model tuning, and reported model footprint                                        | Limited relevance to WSN routing behavior                           |
| [37]                                      | E0   | Stage-wise reduction before final detection       | IoT IDS                                    | Two-stage reduction of the detection workload                                                        | End-to-end cost of both stages is not reported                      |
| [38]                                      | E0   | Filter-based feature reduction                    | IoT IDS                                    | Reduced feature set with classifier evaluation                                                       | No direct hardware or energy measurement                            |
| [40]                                      | E0   | Clustering and Monte Carlo cross-entropy          | IoT IDS                                    | Optimization-based lightweight IDS design                                                            | Direct deployment and energy measurements are absent                |
| [41]                                      | E0   | Chi-square feature selection with hybrid IDS      | IoT IDS                                    | Reduced input space for hybrid classification                                                        | Feature stability beyond the evaluated setting is not established   |
| [39]                                      | E0   | Dimensionality reduction with ensemble learning   | IIoT IDS                                   | Dimensionality reduction with real-time IIoT evaluation                                              | Execution-layer evidence remains limited                            |
| [57]                                      | E0   | Feature selection and hybrid data balancing       | Industrial IoT                             | Feature reduction with imbalance treatment                                                           | End-to-end resource cost and deployment assumptions remain unclear  |
| [42]                                      | E0   | Optimization-assisted random forest               | WSN IDS                                    | WSN-specific optimization and detection results                                                      | Energy evidence remains limited                                     |
| [43]                                      | E0   | Data balancing with dimensionality reduction      | WSN IDS                                    | Input reduction with imbalance treatment                                                             | Runtime and memory demands are not fully characterized              |
| [44]                                      | E0   | Combined signature and anomaly detection          | IoT network security                       | Hybrid signature and anomaly detection                                                               | Runtime and execution-layer evidence remain limited                 |
| [45]                                      | E0   | Resource-aware ML design                          | IoT security monitoring                    | Resource-aware ML evaluation                                                                         | Comparable energy and hardware measurements remain limited          |
| Compressed, quantized, and compact models |      |                                                   |                                            |                                                                                                      |                                                                     |
| [11]                                      | E2   | Model quantization                                | Resource-constrained IoT anomaly detection | 92.23% memory-size compression, 70.01% lower memory use, and shorter prediction time on Raspberry Pi | Preprocessing cost is not included in the reported resource profile |
| [46]                                      | E0   | Dynamic quantization with feature engineering     | IoT IDS                                    | Feature engineering with dynamic quantization                                                        | Attack validation remains limited beyond the evaluated setting      |
| [58]                                      | E0   | Data purification and separable convolution       | IoT IDS                                    | Separable-convolution lightweight CNN design                                                         | Preprocessing cost is not quantified in the deployment evidence     |
| [59]                                      | E0   | TCP/IP packet-header image construction           | IoT networks                               | Compact packet-header representation for IDS                                                         | Header-conversion and preprocessing costs are not quantified        |
| [47]                                      | E0   | Teacher-student model replacement                 | IoT IDS                                    | Compression route and model substitution                                                             | Final footprint and preprocessing cost are not reported             |
| [60]                                      | E0   | Deep convolutional model with lightweight framing | IoT IDS                                    | Lightweight deep-convolutional IDS design                                                            | Memory, energy, and preprocessing costs are not directly reported   |

**Table 5** (continued)

| Refs.                              | Tier | Lightweight mechanism                                 | Deployment context       | Evidence contribution                                                              | Main evidence limitation                                                       |
|------------------------------------|------|-------------------------------------------------------|--------------------------|------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| [48]                               | E0   | Knowledge distillation                                | IoT/UAV IDS              | Student-model reduction                                                            | UAV setting may not generalize to fixed IoT or WSNs                            |
| [117]                              | E1   | Adaptive-temperature knowledge distillation           | IoT IDS                  | More than 98.8% reduction in model size, parameters, and computational overhead    | No constrained-hardware validation                                             |
| [49]                               | E0   | Compact sequence modeling                             | IoT/constrained networks | Temporal modeling with lighter sequence design                                     | Direct hardware and resource measurements remain limited                       |
| [50]                               | E0   | Quantized tree-based model                            | IoT botnet detection     | Quantized tree-based botnet detector                                               | Evidence is botnet-specific and lacks direct constrained-platform validation   |
| [116]                              | E1   | Kronecker-based compact architecture and quantization | IoT IDS                  | Model footprint and inference-time reduction                                       | Broader device-level hardware validation is not reported                       |
| [61]                               | E0   | Quantum SVM                                           | IoT networks             | Explores compact alternative classification for IoT IDS                            | Practical constrained-device evidence remains limited                          |
| TinyML, edge, and split deployment |      |                                                       |                          |                                                                                    |                                                                                |
| [51]                               | E2   | Edge-oriented federated detection                     | IIoT                     | Edge-level resource profiling and distributed detection                            | The footprint remains large for highly constrained endpoints                   |
| [118]                              | E3   | NAS-optimized compact architecture                    | IoT edge                 | Less than 20 kB RAM, 2.48 ms inference, 0.39 mJ/inference, and hardware validation | Long-term update and drift behavior remain untested                            |
| [52]                               | E2   | Feature reduction, quantization, and edge deployment  | Edge-based IIoT          | Latency and deployment setup                                                       | Node-level energy cost remains limited in the reported evidence                |
| [62]                               | E0   | Lightweight edge DL                                   | Smart-agriculture IoT    | Supports application-specific edge IDS deployment                                  | Application-specific setting may not generalize to other IoT domains           |
| [63]                               | E0   | DL-inspired edge IDS                                  | Edge IoT                 | Supports edge-level IDS placement                                                  | Node and edge costs are not separated                                          |
| [53]                               | E0   | Efficient flow-based design with explainability       | Industrial IoT           | Efficient flow-based IDS with explanation support                                  | The resource cost of explanation is not quantified                             |
| [54]                               | E0   | Lightweight edge-side classification                  | Edge IoT                 | Multiclass edge detection                                                          | Multiclass cost may exceed binary screening                                    |
| [64]                               | E0   | AI-driven deep IDS                                    | Cloud-based IIoT         | Useful contrast for high-performance IIoT detection                                | Cloud setting does not prove constrained-node deployability                    |
| [65]                               | E0   | DL-based IoT IDS                                      | IoT networks             | Scalable DL-based detection                                                        | Detection accuracy is not paired with direct resource or hardware measurements |
| [55]                               | E1   | Cluster-head (CH) screening with server-side analysis | WSN IDS                  | Split CH/server IDS with execution-time, power, and network-lifetime results       | CH and server costs are not fully separated                                    |
| Federated and collaborative IDS    |      |                                                       |                          |                                                                                    |                                                                                |
| [56]                               | E0   | Distributed training without raw-data pooling         | IoT networks             | Data scaling and FL performance                                                    | Update size and communication-round cost are not quantified                    |
| [14]                               | E0   | Feature selection with FL                             | IoT botnet detection     | Client training and reduced feature burden                                         | Communication cost is not reported                                             |

**Table 5** (continued)

| Refs.                                                   | Tier | Lightweight mechanism                                        | Deployment context       | Evidence contribution                                          | Main evidence limitation                                                                    |
|---------------------------------------------------------|------|--------------------------------------------------------------|--------------------------|----------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| [15]                                                    | E0   | Heterogeneity-aware FL with BiLSTM                           | Heterogeneous IoT IDS    | Non-IID and client heterogeneity treatment                     | Client workload and communication cost are not quantified                                   |
| [66]                                                    | E0   | Federated learning over edge clients                         | Heterogeneous IoT        | Edge-FL across heterogeneous IoT clients                       | Client workload and update traffic are not quantified                                       |
| [67]                                                    | E0   | Federated learning using edge devices                        | Transportation IoT       | Extends FL-based IDS to transportation IoT                     | Device capability and communication budget remain insufficiently characterized              |
| [16]                                                    | E3   | Lightweight cryptography with FL                             | IoT microcontrollers     | ESP32-S3 hardware validation of secure FL update exchange      | Cryptographic operations remain part of the total client and communication cost             |
| [17]                                                    | E0   | FL with SCNN-Bi-LSTM                                         | WSN IDS                  | Privacy-preserving collaborative training                      | Sensor-node energy cost is not directly quantified                                          |
| [18]                                                    | E0   | Feature selection and round-wise knowledge distillation      | Non-IID IoT IDS          | Feature reduction, distillation, and non-IID learning          | FL update traffic remains part of the cost                                                  |
| [119]                                                   | E0   | Two-stage knowledge distillation with heterogeneity-aware FL | IIoT                     | Lightweight personalized FL under non-IID data                 | Client-side energy and communication evidence remain limited                                |
| [68]                                                    | E0   | Lightweight FL design                                        | Industrial IoT           | IIoT-oriented federated detection                              | Non-IID behavior, communication rounds, and update size remain insufficiently characterized |
| Trust, authentication, cryptography, and secure routing |      |                                                              |                          |                                                                |                                                                                             |
| [19]                                                    | E1   | Lightweight trust scoring                                    | Industrial WSNs          | Trust accuracy, residual energy, and overhead evidence         | Collusion and bad-mouthing validation remains limited                                       |
| [20]                                                    | E1   | Low-cost IIoT authentication                                 | IIoT                     | Message count, computation cost, and attack resistance         | Post-authentication compromise remains possible                                             |
| [21]                                                    | E1   | Elliptic-curve-based fault-tolerant authentication           | WSNs                     | Execution time and memory evidence                             | Key-management assumptions constrain interpretation                                         |
| [22]                                                    | E1   | Fuzzy-extractor-based lightweight authentication             | Edge-IoT e-healthcare    | Computation, energy, and communication-cost evidence           | Authentication does not replace runtime IDS                                                 |
| [23]                                                    | E1   | Energy-efficient proxy re-encryption                         | WSN secure data exchange | Computation and energy evidence for proxy re-encryption        | The mechanism does not address later node misbehavior                                       |
| [24]                                                    | E1   | Trust-aware routing with attribute-based encryption          | WSNs                     | Packet delivery, residual energy, and route-stability evidence | Route-level protection does not cover application-layer abuse                               |
| [120]                                                   | E1   | Two-level RL route mutation                                  | Ad hoc WSN               | Attack detection with route adaptation and energy awareness    | Evidence depends on the evaluated topology and traffic conditions                           |

limitations of common IoT and WSN datasets. Table 7 clarifies what common detection and resource metrics can and cannot establish in constrained deployments.

**Table 6** Dataset relevance and limitations for resource-aware IoT/WSN security evaluation

| Dataset           | Evaluation context                         | Main relevance                                                     | Resource-aware limitation                                                     |
|-------------------|--------------------------------------------|--------------------------------------------------------------------|-------------------------------------------------------------------------------|
| Edge-IIoTset [31] | IoT/IIoT centralized and federated IDS     | Modern IoT/IIoT coverage with centralized and FL use               | Does not by itself prove node-level deployability                             |
| TON_IoT [25]      | IoT/IIoT telemetry and network IDS         | Multi-source IoT/IIoT environment with telemetry and network views | Feature extraction and deployment cost may vary by data source                |
| CICIoT2023 [26]   | Large-scale IoT attack benchmarking        | Broad attack coverage and recent IoT attack scale                  | Class imbalance, traffic realism, and device-level cost should be checked     |
| Bot-IoT [27]      | Botnet and DoS/DDoS analysis               | Useful for botnet-heavy IDS and high-volume traffic attacks        | May bias evaluation toward traffic-volume patterns                            |
| N-BaloT [28]      | IoT botnet detection                       | Device-level botnet behavior from compromised IoT devices          | Limited attack diversity beyond botnet behavior                               |
| WSN-DS [29]       | WSN IDS and routing-behavior analysis      | Relevant to WSN-specific attacks and sensor-network behavior       | Older and narrower than recent IoT datasets                                   |
| RT-IoT2022 [11]   | Resource-constrained IoT anomaly detection | Used in quantized IDS evaluation                                   | Needs broader external validation across attack types and deployment settings |

**Table 7** Detection and resource metrics for interpreting lightweight security evidence

| Metric                           | Evidence provided                          | Evidence not captured in constrained deployment              |
|----------------------------------|--------------------------------------------|--------------------------------------------------------------|
| Accuracy                         | Overall correctness                        | Class imbalance, false alarms, and rare-attack weakness      |
| Precision/Recall/F1              | Detection balance                          | Energy, memory, and communication cost                       |
| Macro-F1/per-class recall        | Minority-class behavior                    | Resource cost and operational effect of each decision        |
| ROC-AUC                          | Threshold behavior                         | Practical cost of threshold selection and response           |
| False-positive rate              | Alarm burden                               | Energy and communication spent on unnecessary responses      |
| Model size                       | Storage footprint                          | Feature extraction, preprocessing, and communication cost    |
| Inference latency                | Runtime delay                              | Energy use and full pipeline delay                           |
| Memory/RAM                       | Execution feasibility                      | Radio cost, update overhead, and flash constraints           |
| Flash footprint                  | Storage feasibility on constrained devices | Runtime memory, feature buffer, and update cost              |
| Energy consumption               | Battery effect                             | Often omitted, simulated only, or measured under one setting |
| Communication overhead           | Network burden                             | FL rounds, routing updates, and radio duty-cycle effect      |
| FL update size/rounds            | Cost of collaborative learning             | Local training energy and aggregation vulnerability          |
| Packet delivery/network lifetime | WSN operational effect                     | Application-level detection quality and false isolation      |
| Hardware test                    | Deployment realism                         | Often limited to one board, one gateway, or one scenario     |

### 5.3 Resource-efficiency interpretation across technique families

Resource efficiency differs across lightweight security mechanisms because each targets a different source of overhead. Feature-selection methods reduce classifier input and processing burden, while compression and quantization reduce model footprint and inference cost. TinyML, edge-assisted, and split-deployment approaches also determine where computation is performed. Federated learning redistributes training and communication costs, whereas authentication, cryptography, trust management, and secure routing primarily affect computation, message exchange, energy consumption, and network operation [1, 11, 13–24, 36–56].

Feature-selection and compact-learning studies commonly demonstrate efficiency through reduced dimensionality and lower classifier complexity [36–39, 73–75, 83]. These reductions are useful when they also decrease processing cost, storage demand,

or inference time. However, the benefit may be limited when feature generation remains computationally expensive. Model-oriented approaches provide more direct evidence when compression, pruning, quantization, or compact architectures are supported by measurements of model footprint, memory demand, or execution latency [11, 46–50, 76–82, 86].

Deployment-oriented approaches require a different interpretation. TinyML studies provide stronger evidence when RAM, flash, latency, and power are measured on embedded hardware [77, 85]. Edge-assisted approaches shift part of the processing burden to gateways or edge servers [51, 52, 74, 87, 101, 103]. Split-processing designs distribute security functions across constrained nodes, cluster heads, and server-side resources [55]. Resource claims in these cases should distinguish the cost incurred at each deployment layer rather than treat edge-level efficiency as equivalent to sensor-level efficiency.

Federated IDS replaces centralized raw-data transfer with local training and repeated model-update exchange [13–18, 56, 69–72, 88, 89]. Its lightweight assessment therefore depends on update size, communication rounds, client workload, aggregation cost, and hardware capability. Outside IDS classification, lightweight authentication and cryptography are better evaluated through computation time, message size, memory demand, throughput, and energy [1, 20–23, 91–98]. Trust and secure-routing mechanisms are more appropriately assessed through routing overhead, packet delivery, residual energy, and network lifetime [19, 24, 55, 90, 100, 105].

Table 8 presents representative quantitative resource and deployment evidence across the reviewed mechanisms. It covers model or input reduction, memory footprint, latency, energy or power, communication cost, and hardware execution where these measures are reported. The values are interpreted within their original experimental settings rather than ranked directly across heterogeneous platforms. The Tier column indicates the maturity of the supporting deployment evidence according to the classification defined in Sect. 2.

#### 5.4 Security-risk interpretation across technique families

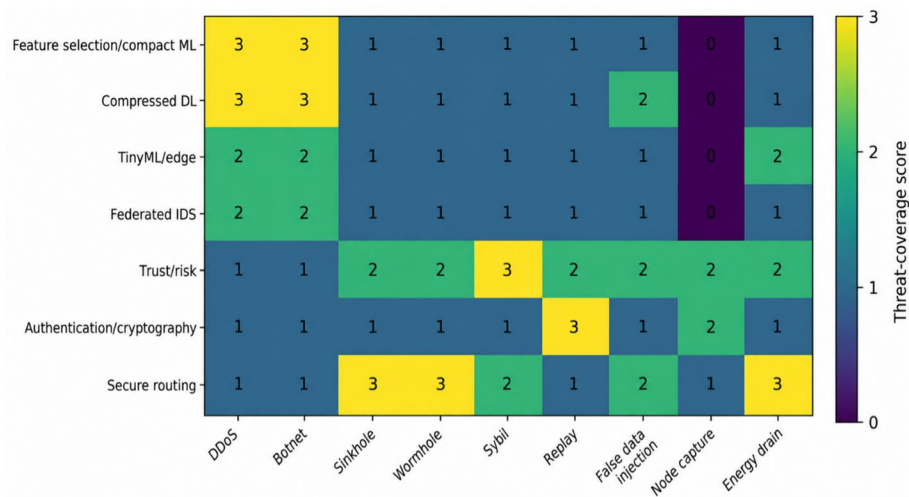
After the resource-efficiency discussion, the remaining question is what type of security risk each family can credibly reduce. A lightweight method should be interpreted according to its point of intervention in the security pipeline. Detection-oriented studies act after traffic, node behavior, or routing state becomes observable. Protection-oriented studies act earlier, either before or during communication, by restricting unauthorized access, limiting key misuse, or preventing unsafe routing decisions. These roles are complementary, but they do not support the same security claim.

The scope of evidence in each result is therefore important. A DDoS detector should not be treated as evidence against sinkhole routing. A secure routing protocol does not by itself establish botnet detection. A lightweight authentication scheme does not provide post-access anomaly detection. These examples do not reduce the value of the individual studies. They show that risk coverage must be interpreted according to the attack surface addressed by the method and the additional cost introduced by the defense.

Figure 3 summarizes the relative relevance of each technique family to the main threat categories. The ordinal score represents threat coverage rather than comparative

**Table 8** Representative quantitative resource and deployment evidence across lightweight IoT and WSN security studies

| Refs. | Tier | Setting and lightweight approach                        | Quantitative resource evidence                                                                                   | Deployment evidence                                            |
|-------|------|---------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| [11]  | E2   | Resource-constrained IoT; quantized autoencoder         | 92.23% memory-size compression; 70.01% lower average memory utilization; prediction time reduced from 35 to 15 s | Raspberry Pi, 2 GB RAM, 1.2 GHz ARM Cortex-A53                 |
| [16]  | E3   | IoT federated IDS with lightweight cryptography         | SPECK: 504 B RAM, 31.328 kB ROM; 0.1348/0.1362 $\mu$ s/B encryption/decryption; 138.3/134.1 mJ                   | ESP32-S3 clients with hardware-validated FL implementation     |
| [22]  | E1   | Edge-IoT healthcare; lightweight authentication         | 10.07 ms computation cost; 109.56 mJ energy; 3392-bit communication cost                                         | Edge-assisted IoT authentication                               |
| [52]  | E2   | Edge-based IIoT IDS; feature reduction and quantization | 4-bit/8-bit quantization; approximately 25% latency reduction                                                    | Azure IoT Edge deployment                                      |
| [55]  | E1   | WSN split IDS; CH screening and server-side analysis    | 124.63 ms execution time; up to 70% lower CH power consumption; 2.7x longer network lifetime                     | WSN CH/server split deployment                                 |
| [77]  | E2   | Lightweight TCN for edge IoT                            | Model size reduced from 248.5 to 89.1 kB; 0.19–0.27 ms inference; 4.22–4.64 W average power                      | Raspberry Pi 4 with physical power measurement                 |
| [51]  | E2   | IIoT edge/FL IDS                                        | 8–12 MB model weights; approximately 300–500 MB RAM; 120–180 ms/sample; 20% lower FL communication overhead      | Preliminary Raspberry Pi 4 profiling and 10-node FL simulation |

**Fig. 3** Qualitative threat-coverage map of the reviewed lightweight security technique families. Ordinal scale: 0=no direct coverage; 1=limited or indirect coverage; 2=partial coverage; 3=primary or direct coverage

detection performance. Table 9 translates the same risk view into evidence requirements at the technique-family level.

Overall, the cross-family comparison shows that lightweightness is supported most convincingly when security performance and resource reduction are evaluated at the intended deployment layer. Feature reduction, model compression, TinyML, edge offloading, federated learning, authentication, and secure routing reduce different forms of overhead and can shift cost across system layers. The reviewed evidence should therefore be interpreted through the combined consideration of security function, measured resource effect, deployment context, and operational constraints rather than detection accuracy alone.

**Table 9** Risk-reduction roles and evidence requirements across lightweight technique families

| Technique family                            | Security role                                                                  | Evidence needed                                                                                                    | Main limitation                                                                                |
|---------------------------------------------|--------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| Feature selection + compact ML              | Low-cost traffic or behavior detection through reduced input and model burden  | Selected features, preprocessing cost, latency, model size, feature stability, and per-class detection results     | Generalization across datasets and deployments; weak coverage of routing and energy behavior   |
| Compressed DL                               | Detection of complex or sequential attack patterns with lower model cost       | Compression ratio, model size, RAM use, latency, accuracy loss after compression, and preprocessing cost           | Rare-class sensitivity, off-device preprocessing, and device-level feasibility                 |
| TinyML/edge IDS                             | Local or near-local response through node, gateway, or edge placement          | Hardware target, RAM, flash, energy use, latency, update cost, and deployment layer                                | Device variability and unclear separation between node-level and edge-level evidence           |
| Federated IDS                               | Collaborative detection with reduced raw-data pooling                          | Communication rounds, update size, client workload, non-IID behavior, poisoning resistance, and aggregation cost   | Update overhead, client trust, aggregation vulnerability, and feasibility on low-power clients |
| Trust/risk scoring                          | Behavior-based node assessment and risk-based isolation                        | Trust accuracy, false isolation rate, routing evidence, residual energy, recovery behavior, and message overhead   | Collusion, bad-mouthing, on-off behavior, sparse evidence, and trust bootstrapping             |
| Lightweight authentication and cryptography | Prevention of replay, impersonation, session abuse, and key-related exposure   | Message count and size, computation cost, storage demand, delay, key-management assumptions, and security analysis | Post-authentication compromise, captured keys, and runtime misbehavior                         |
| Secure routing                              | Reduction of topology manipulation, forwarding abuse, and energy-drain effects | Packet delivery ratio, routing overhead, route stability, residual energy, recovery time, and network lifetime     | Interaction with IDS, false route isolation, energy-drain attacks, and application-layer abuse |

## 6 Critical evaluation of the reviewed literature

The critical evaluation focuses on the remaining limitations in deployment evidence, generalization, robustness, integration, and reproducibility. It considers whether the reported security benefit, resource effect, and deployment layer are sufficiently connected to support operational use in constrained IoT and WSN environments.

### 6.1 Evidence maturity behind lightweight claims

Evidence maturity is interpreted through the E0–E3 classification defined in Sect. 2. Among the 52 studies, 36 were classified as E0, 11 as E1, 3 as E2, and 2 as E3. Most lightweight claims therefore rely on design-level or software/simulation evidence, while direct platform profiling and constrained-hardware validation remain uncommon.

Across the reviewed literature, feature-selection studies commonly report input reduction, compressed models more often report footprint or inference effects, and hardware-oriented studies provide stronger evidence when RAM, flash, latency, energy, or communication costs are measured directly [11, 16, 36, 38, 46, 51, 52]. A relative reduction against a heavier baseline remains useful, but it does not establish constrained-device feasibility unless the target platform and the relevant end-to-end costs are clear.

The principal limitation is therefore uneven evidence rather than the absence of lightweight design. Feature reduction may be well documented while feature-extraction cost remains unmeasured; model compression may be quantified while preprocessing is omitted; FL may report distributed detection performance without equivalent detail on update traffic or local energy; and lightweight protocols may report computation or

security properties without integration with runtime detection or response [13–24]. These gaps define the evidence-maturity problem examined in the following subsections.

Technology maturity remains uneven across the reviewed families. Feature-selection and compact-ML approaches have a broad experimental base, while recent compact and distillation-based methods provide clearer evidence on model size and runtime [116, 117]. TinyML studies become more convincing when memory, latency, energy, and target hardware are measured directly [118]. Javed et al. [134] further demonstrate this direction through a MobileNetv3-based IoT malware detector that combines multi-scale feature extraction with channel-wise pruning to improve detection performance and computational efficiency. Federated IIoT approaches have progressed in model efficiency and the treatment of heterogeneous data [119], but client-side energy and update costs remain less consistently reported. In WSNs, recent adaptive routing methods increasingly consider security and energy efficiency together [120], although their results remain sensitive to topology and traffic assumptions. Deployment maturity therefore depends on resource evidence measured at the intended execution layer.

A more mature lightweight claim is supported when three elements are reported together: an explicit reduction in resource demand, measurements on the intended or a representative hardware platform, and the cost of the complete processing or communication path. Evidence is less conclusive when only relative model reduction or inference time is reported without preprocessing, update, communication, or energy costs. This distinction is important because a large percentage reduction does not necessarily make a method suitable for a constrained device, and cost can be shifted from the endpoint to an edge server or from raw-data transmission to repeated federated updates [11, 51, 77, 102, 106]. Absolute resource demand, deployment layer, and shifted system costs should therefore be considered together when deployment maturity is assessed.

## 6.2 Deployment boundary and end-to-end cost

The boundary between node, gateway, edge, fog, and cloud is a recurring source of ambiguity. Several papers use the language of constrained deployment while evaluating the method at a gateway, edge, or server level [84, 101]. This is acceptable when the method is intended for edge-assisted detection. It becomes problematic when the conclusion implies suitability for constrained devices without showing what the device itself must compute, store, transmit, or update [102].

The full cost of IDS is rarely limited to inference. Before classification, traffic must be captured, features extracted, windows formed, and values normalized or encoded [103, 104]. In WSNs, the cost may also include neighbor observation, route-state collection, CH reporting, and retransmission [105]. In FL, it includes local training, update transmission, aggregation, synchronization, and security checks [106]. In authentication and cryptography, it includes message exchanges, key storage, verification operations, and recovery after compromise. Measuring only the final classifier or protocol operation therefore gives a partial view of deployment cost.

End-to-end cost matters most when the attack itself targets energy or communication. A defense against energy-drain or selective-forwarding attacks should not introduce message overhead that shortens network lifetime. A trust system should not require so much neighborhood reporting that it consumes the energy it is meant to protect [105]. A federated detector should not replace raw-data transfer with model updates that

are equally expensive for the client [106]. A compressed model should not shift cost to feature construction or repeated edge communication [104]. Lightweightness must therefore be evaluated along the path from sensing and communication to detection, protection, and response.

A clear boundary statement is therefore essential. Node-level studies should report the hardware, feature source, local memory, flash footprint, inference time, energy effect, and communication overhead [102]. Gateway-level studies should report throughput, buffering cost, latency under load, update behavior, and the amount of data sent from nodes to the gateway. Edge or fog studies should explain what is offloaded and what remains on the constrained device [101]. FL studies should report local training time, update size, rounds to convergence, client selection behavior, and non-IID effects [106]. Without these details, strong detection results still leave the deployment claim incomplete.

Practical deployment also depends on maintenance and recovery costs that are rarely captured in benchmark evaluation. Model or rule updates must fit available memory, energy, and communication budgets, while intermittent connectivity and hardware variation can affect latency and availability. False alarms may also trigger costly isolation, key refresh, rerouting, or additional verification. Deployment studies should therefore report update mechanisms, fallback behavior, connectivity assumptions, hardware variation, and response cost alongside inference or protocol measurements.

### 6.3 WSN realism and cross-domain generalization

The comparison also shows a persistent difference between IoT and WSN evaluation cultures. IoT studies usually rely on flow records, benchmark datasets, malware traffic, DDoS, botnet activity, scanning, or device-level telemetry. WSN studies more often involve packet delivery, routing behavior, residual energy, topology instability, sinkhole behavior, selective forwarding, and network lifetime [25–29, 31, 34]. These settings overlap, but they should not be treated as interchangeable. A method validated on an IoT flow dataset may classify traffic effectively but still lack evidence for WSN routing abuse. A WSN method may protect forwarding behavior and route selection, but it may not represent smart-home botnets or IIoT process telemetry.

Cross-domain generalization is therefore a central weakness. Many studies are evaluated on one dataset or one simulation setting. Some report high accuracy, but the result may depend on fixed feature distributions, repeated attack patterns, or stable traffic windows. Feature selection can intensify this problem when a small subset captures dataset-specific signatures rather than general attack behavior. Compression can have a similar effect if dominant classes are preserved while rare or low-rate behaviors are weakened. FL reduces centralized data dependence, but non-IID clients can still create local and global drift [9, 10, 15, 18, 30, 69, 70, 83, 106].

For WSNs, generalization also depends on topology and routing assumptions. A result obtained under a fixed topology may not hold under mobility, link fluctuation, duty cycling, or node death. A routing-aware method should be tested under changes in neighbor density, attacker position, traffic rate, and energy level. A trust method should be tested under collusion, bad-mouthing, on-off behavior, and sparse observations. A secure routing method should show not only packet delivery under attack, but also route-recovery cost and energy effect [19, 24, 34, 55, 105]. These requirements are more

demanding than ordinary classification benchmarking, but they match the risk profile of WSN deployments.

A stronger evaluation should include at least one form of external validation. This may be cross-dataset testing for flow-based IoT IDS, topology variation for WSN IDS, device variation for TinyML, non-IID stress testing for FL, or multiple attack intensities for secure routing and trust scoring [25–29, 85, 102, 106]. Such validation does not need to cover every environment, but it should show whether the method remains useful when evaluation conditions move away from the training setting. This is the difference between a promising lightweight model and a method with credible deployment evidence.

#### 6.4 Robustness, drift, and adaptive attacks

Robustness remains underdeveloped. Most studies evaluate the detector or protocol under the attacks represented in the selected dataset or simulation. Fewer examine what happens when the attacker adapts to the lightweight mechanism itself. This matters because lightweight designs often reduce redundancy. Feature selection removes variables, compression reduces representation capacity, TinyML models use small architectures, FL shares updates rather than raw data, and trust methods compress behavior into scores [9, 11, 46, 83, 85]. Each choice can improve efficiency, but it can also reduce the evidence available to resist adaptive behavior.

For feature-reduction IDS, attackers may shift behavior away from the retained features. For compressed DL, rare classes may weaken after pruning, quantization, or distillation. For TinyML, limited model capacity can make the detector brittle under drift. For FL, poisoning, free-riding, and malicious updates can change the global model even when raw data remain local. For trust scoring, collusion and bad-mouthing can manipulate local reputations. For authentication and cryptography, node capture can invalidate assumptions about key secrecy and device identity [19, 20, 92, 105, 106]. These weaknesses do not make the methods unsuitable, but they show that robustness should be a primary evidence category.

Recent studies have begun to address parts of this issue. Some combine feature selection with resource-sensitive model design or evaluate lightweight models under stronger deployment assumptions [74–78, 81, 82, 86, 87]. FL studies increasingly discuss non-IID behavior, poisoning, privacy leakage, update verification, or explainability [13, 16, 18, 30, 69–72, 88, 89]. Newer IoT IDS studies also use incremental learning, contrastive learning, attention, and explainable components to handle changing environments [82, 88, 107–109]. The gap is consistency: robustness is still less visible than accuracy in many evaluations.

A lightweight security method should also be evaluated against recovery cost. Detection alone does not show whether the system can respond safely. False isolation may remove a useful sensor, rerouting may increase energy use, key refresh may add communication overhead, and FL defenses against poisoned updates may require extra filtering or verification [24, 55]. Robustness should therefore include both model stability and response stability. The question is not only whether the method detects or prevents an attack, but whether the response remains safe and affordable under constrained operation.

### 6.5 Integration of prevention, detection, and response

The reviewed literature still treats prevention, detection, and response as separate tracks. IDS papers classify traffic or behavior, authentication papers prove access control and message protection, cryptography papers report computation and communication cost, secure routing papers report packet delivery and energy, and trust papers report node scores or route decisions [19–24, 81, 82, 91–98, 105]. Each line is valuable, but constrained security is rarely achieved by one layer. A realistic architecture usually needs several low-cost components that cover different points in the security pipeline. This makes integration evidence as important as the performance of each isolated mechanism.

The lack of integration creates two problems. First, papers may overstate scope. A lightweight authentication scheme can reduce unauthorized access, but it cannot prove that a captured device will be detected later. A compact IDS can detect DDoS traffic, but it does not protect against key misuse. A trust-aware routing method can reduce forwarding abuse, but it may not identify application-layer manipulation. Second, the combined cost of multiple lightweight components is rarely measured. A system using authentication, IDS, trust updates, and secure routing may be secure in principle, but the total overhead may still be too high for the intended device or network [84, 92, 102, 106].

Future work should move from isolated lightweight modules toward layered evaluation. A useful design would show how authentication reduces entry-level exposure, how IDS monitors residual behavior, how trust scoring handles repeated misbehavior, and how routing or response policies limit damage. Such evaluation does not need every possible security layer, but it should make the relationship between selected layers explicit. An authentication method can be evaluated with lightweight IDS to show whether post-access attacks remain visible. A secure routing method can be evaluated with traffic-based IDS to show whether routing and application threats can be separated. An FL-based IDS can be tested with poisoning protection and update-size reporting to show both security and communication cost [30, 69–72, 88, 89].

Response policy is another missing part of integration. Many IDS papers stop at classification. Yet constrained systems need careful responses because isolation, rerouting, key refresh, or rate limiting can consume energy or interrupt legitimate sensing. A lightweight detector that triggers frequent expensive responses may be impractical even if inference is cheap. A trust model that quickly isolates nodes may protect the network but may also remove legitimate nodes under noisy links. Future studies should report not only detection metrics, but also the cost and safety of the proposed response [24, 55, 91, 105].

### 6.6 Explainability, reproducibility, and reporting consistency

Explainability is increasingly important in lightweight security, but it remains difficult to balance with resource constraints. In IoT and WSN deployments, a security decision may trigger isolation, routing change, key refresh, or alarm escalation. Operators need to understand why a node, flow, or route was treated as suspicious. At the same time, explanation methods can add computation and memory overhead. A black-box model may be compact but difficult to trust. An explainable model may be easier to justify but more expensive to run. Recent explainable, attention-based, and adaptive IDS studies

move the field in a useful direction, but the cost of explanation is rarely treated as part of the lightweight budget [53, 82, 88, 89, 108–113].

Reproducibility is another limiting factor. Lightweight claims depend on implementation choices, including feature extraction, class balancing, train-test splitting, normalization, model conversion, hardware platform, and measurement method. If these details are missing, results are difficult to verify. Reproducibility is not only a code-availability issue. A study can publish code but still omit selected feature lists, exact hardware, window construction, energy model, or FL client settings. For systematic comparison, these details are essential [7, 8, 83, 85].

Reporting consistency is one of the most practical ways to strengthen the field. A lightweight IDS paper should report the target layer, selected features, model size, RAM, flash, latency, energy, communication cost, dataset, attack classes, and class-imbalance handling. A WSN paper should add packet delivery, routing overhead, residual energy, topology conditions, and lifetime. An FL paper should add rounds, update size, local training cost, aggregation method, client heterogeneity, non-IID split, and poisoning assumptions. A protocol paper should add message count, storage, computation, delay, security assumptions, and compromise model. These items are not secondary details; they determine whether the evidence supports constrained deployment [84, 102, 105].

Taken together, these gaps show that the literature has moved beyond simple accuracy reporting, but not yet toward a consistent deployment-evidence standard. The strongest studies now connect security benefit with a measurable resource effect; the weaker ones still leave the resource cost, deployment layer, or response consequence outside the evaluation. Table 10 summarizes the main evidence gaps that remain after comparing the reviewed studies. The table is not a second technique taxonomy. It identifies where an otherwise promising lightweight claim becomes incomplete.

## 7 Deployment evidence and future directions

The critical evaluation indicates that future research should not only propose additional lightweight models. The field needs clearer evidence rules. A new detector, protocol, or routing scheme should state which cost is reduced, which security role is supported, and which device or deployment layer is assumed. It should also provide enough detail for another researcher to repeat the evaluation. This section converts the gaps identified in Sect. 6 into a reporting framework and research agenda.

The framework below is intentionally practical. It does not require every paper to report every possible metric. It requires each paper to report the metrics that match its own lightweight claim. A feature-selection study should report input-level evidence. A compressed model should report model-level cost. A TinyML study should report hardware results. An FL study should report communication and client-side costs. A protocol or routing study should report message count, computation, and network-lifetime results. The purpose is not to make evaluation heavier; it is to make lightweight evidence clearer and easier to compare.

### 7.1 Reporting requirements for lightweight security studies

The reviewed literature shows that lightweightness has several meanings: fewer features, fewer parameters, lower precision, shorter inference time, lower memory use, fewer protocol messages, smaller keys, lower communication, or lower energy cost. These

**Table 10** Critical evidence gaps across lightweight IoT/WSN security studies

| Evidence gap                           | Most affected studies or families                            | Interpretive effect                                                                          | Recommended evidence                                                                         | Representative refs.                |
|----------------------------------------|--------------------------------------------------------------|----------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|-------------------------------------|
| Accuracy-first evaluation              | ML/DL IDS and many flow-based studies                        | High detection scores do not prove constrained deployment                                    | Report accuracy with model size, latency, memory, energy, and communication cost             | [11, 46, 53, 64, 65, 81, 82]        |
| Feature stability not tested           | Feature selection, compact ML, and PCA-based IDS             | A selected subset may capture dataset-specific signatures rather than stable attack evidence | Report selected features, feature stability, and at least one external validation setting    | [9, 10, 36–43, 57, 73–75, 83]       |
| Preprocessing cost omitted             | Flow-based and compressed IDS                                | Feature construction may be more expensive than final inference                              | Report capture, windowing, feature extraction, normalization, and conversion cost            | [41, 46, 47, 58, 59, 76, 83, 104]   |
| Node-edge boundary unclear             | TinyML, edge, split, and FL studies                          | Cost may be shifted away from constrained nodes rather than reduced                          | Separate node, CH, gateway, edge, and server costs                                           | [51, 52, 62, 63, 84–87, 101–103]    |
| Energy and radio cost underreported    | WSN IDS, trust, routing, FL, and protocol studies            | A defense may consume the same resource targeted by the attack                               | Report residual energy, radio cost, update size, packet delivery, and network lifetime       | [17, 19, 23, 24, 55, 100, 105]      |
| Non-IID and poisoning evidence limited | FL-based IDS                                                 | Collaborative detection may fail under heterogeneous or malicious clients                    | Report non-IID splits, poisoning assumptions, aggregation defense, and client workload       | [13–18, 30, 56, 66–72, 88, 89, 106] |
| Security layer evaluated in isolation  | Authentication, cryptography, IDS, trust, and routing        | One layer cannot justify broad protection claims                                             | Evaluate at least one interaction among prevention, detection, trust, routing, and response  | [1, 19–24, 55, 90–100]              |
| Explainability cost rarely measured    | Compact DL, attention models, edge IDS, and XAI-assisted IDS | Explanations may improve trust but add computation or delay                                  | Report explanation latency, memory cost, stability, and operational usefulness               | [53, 81, 82, 88, 89, 107–113]       |
| Reproducibility details incomplete     | Cross-family comparison                                      | Results cannot be verified or fairly compared                                                | Publish code, features, splits, hardware, hyperparameters, energy model, and FL/WSN settings | [7, 8, 77, 83, 88, 111, 114, 115]   |

reductions are not equivalent. A method may reduce model size while increasing preprocessing. It may reduce raw-data sharing while increasing update traffic. It may reduce authentication time without providing runtime visibility. It may reduce route attacks while requiring additional trust messages. A reporting framework is needed because a single lightweight label hides these differences.

The first reporting item is deployment level. A study should state whether the method targets a sensor node, microcontroller, cluster head, mobile device, gateway, edge server, fog node, cloud platform, or hybrid architecture. This information should appear in the method and evaluation, not only in the abstract. The second item is the cost category reduced by the method. A feature-selection study should report the original and final feature sets, feature-extraction cost, and stability. A compressed DL study should report parameter count, compression ratio, model size, inference delay, memory, and accuracy loss. A TinyML study should report flash, RAM, current draw or energy, device model, and runtime. An FL study should report local training time, update size, communication rounds, aggregation cost, and client heterogeneity. A protocol study should report message count, message size, storage, computation, and security assumptions.

The third item is security scope. A study should state whether it supports traffic detection, anomaly detection, device authentication, session protection, route defense, trust

management, energy protection, or response. The scope should remain narrow when the reported results are narrow. A botnet detector does not cover sinkhole behavior unless routing results are used. A secure routing protocol does not cover malware communication unless application or traffic analysis is included. A lightweight authentication method does not show runtime anomaly detection unless behavior after access is monitored. This discipline makes the contribution more credible, not weaker.

The fourth item is robustness. Future studies should include cross-dataset evaluation when possible, or at least variation in attack intensity, client distribution, topology, device behavior, or traffic period. FL studies should include non-IID testing and poisoning assumptions. WSN studies should include route changes, link noise, attacker position, and residual energy. TinyML and edge studies should include device-specific measurements. Protocol and routing studies should include compromised-node assumptions and recovery behavior. These tests do not need to be exhaustive, but they should show that the method is not tuned only to one stable setting.

The fifth item is reproducibility. A useful lightweight study should provide enough detail for the method to be repeated. This includes feature lists, preprocessing steps, train-test split, class balancing, hyperparameters, hardware, energy model, software versions, and code when possible. For WSN simulations, topology, node count, radio model, mobility assumptions, and traffic profile should be stated. For FL, the number of clients, client sampling, local epochs, optimizer, aggregation rule, update size, and non-IID split should be stated. These details are often treated as implementation matters, but they directly affect the claimed resource cost. Table 11 summarizes these five items as practical reporting requirements.

## 7.2 Future research directions

A first opportunity is end-to-end lightweight evaluation. Many studies optimize one part of the pipeline, but constrained security depends on the whole path. Future work should measure cost from sensing or packet capture to feature construction, inference, response, and update. This is particularly important for flow-based IDS and WSN methods because the cost of building the evidence can be as important as the cost of the classifier. A compact model may look efficient when evaluated on prepared features, but the sensor or gateway may still pay the cost of generating those features.

A second opportunity is cross-family integration. The reviewed literature contains strong individual methods, but fewer studies evaluate layered lightweight protection. A realistic constrained security architecture may combine lightweight authentication, compact IDS, trust scoring, secure routing, and risk-aware response. Future studies should test how these layers interact. Authentication can be paired with post-access anomaly detection, trust-aware routing with traffic-based IDS, and FL detection with update filtering and communication budgeting. The objective is not to add layers without limit, but to show how selected lightweight mechanisms cover different attack surfaces without exceeding the resource budget.

A third opportunity is risk-aware response. IDS outputs should not be treated as final decisions. In constrained systems, response can be costly. Blocking a node, refreshing keys, rerouting packets, reducing trust, or requesting additional verification consumes energy and bandwidth. Future research should design response policies that consider confidence, attack severity, node importance, and resource cost. A low-confidence

**Table 11** Minimum reporting requirements for future lightweight IDS and protection studies

| Study type                  | Minimum evidence                                                                                                          | Security scope                                                        | Deployment details                                                               |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|----------------------------------------------------------------------------------|
| Feature-selection IDS       | Original features, selected features, selector method, pre-processing cost, feature stability, and classification metrics | Traffic or behavior detection only unless additional evidence is used | Dataset, feature source, window construction, model size, latency, and memory    |
| Compact ML IDS              | Model type, number of learners/support vectors, inference time, memory, feature cost, and per-class metrics               | Low-cost classification or anomaly detection                          | Node, gateway, or edge placement; training and inference location                |
| Compressed DL IDS           | Compression method, parameter count, model size, latency, memory, accuracy loss, and preprocessing cost                   | Detection of complex or temporal attack behavior                      | Hardware target, conversion method, and device or edge execution evidence        |
| TinyML/on-device IDS        | RAM, flash, inference latency, energy/current draw, quantization, and device board                                        | Local screening or near-device detection                              | Board type, processor, input source, update method, and duty-cycle effect        |
| Edge/split IDS              | Local task, offloaded task, latency, throughput, communication savings, and update cost                                   | Staged detection or edge-assisted classification                      | Sensor, gateway, edge, and server responsibilities separated                     |
| FL-based IDS                | Number of clients, rounds, update size, local training cost, aggregation rule, non-IID setting, and poisoning defense     | Collaborative detection without raw-data pooling                      | Client hardware, local epochs, communication budget, and aggregation location    |
| WSN IDS                     | Packet delivery, residual energy, routing overhead, topology, lifetime, and attacker position                             | Routing, forwarding, or node-behavior detection                       | Node count, radio model, traffic rate, topology change, and energy model         |
| Authentication/cryptography | Message count, computation cost, storage, key size, delay, and security analysis                                          | Access control, replay prevention, integrity, and confidentiality     | Device class, key storage, communication round trips, and compromise assumptions |
| Trust/secure routing        | Trust update rule, false isolation, packet delivery, route stability, residual energy, and recovery behavior              | Node-behavior control and route protection                            | Neighbor observation cost, trust message cost, topology, and collusion model     |

anomaly may justify monitoring rather than isolation, while a high-confidence routing attack may justify rerouting even if communication increases. This approach links detection to operational risk, which is still weak in many studies.

A fourth opportunity is lightweight robustness. Future work should test drift, rare attacks, poisoning, collusion, and adaptive behavior under resource limits. Robustness should not be measured only by overall accuracy. It should include per-class recall, false alarms, time to detection, recovery cost, and performance under reduced energy or communication budgets. For FL, robustness should include non-IID clients, malicious updates, client dropout, and update compression. For trust and routing, it should include collusion, bad-mouthing, on-off attacks, and noisy links. For compressed models, it should include minority or low-rate attack sensitivity after compression.

A fifth opportunity is low-cost explainability. Explainability is often discussed as a benefit, but its cost is rarely measured. Lightweight IDS needs explanations that support operator decisions without making the detector too heavy. Future work could explore precomputed explanations, feature-level rationales, lightweight rule extraction, confidence calibration, or gateway-side explanation for node-side detections. Explanation

should be judged by usefulness, cost, and stability. An explanation that consumes too much memory or delays response may not be suitable for constrained deployment.

A sixth opportunity is standard reporting and shared evaluation artifacts. The field would benefit from common reporting templates for feature sets, resource metrics, energy models, hardware targets, FL client settings, and WSN topologies. Shared artifacts should not impose a single benchmark. They should allow reviewers to understand how the method was evaluated and whether the reported lightweight benefit is reproducible. Dataset diversity should also be encouraged. Depending on the method, a strong evaluation may combine one flow dataset, one IoT telemetry dataset, and one WSN or routing scenario. This would reduce the risk of fitting the contribution to one dataset type. Another useful direction is scenario-based benchmarking, where studies are evaluated under predefined deployment profiles such as sensor-node IDS, gateway IDS, edge-assisted IDS, FL-based IDS, and WSN routing protection. Such profiles would make results easier to compare without forcing all studies to use the same dataset or hardware.

A seventh opportunity concerns emerging adaptive and foundation-model-based security. LLM-supported IDS can assist with data augmentation, contextual interpretation, and attack analysis, while Agentic AI extends this role toward coordinated reasoning and response [121, 122]. These capabilities are more suitable for edge- or fog-assisted operation than direct execution on highly constrained nodes because memory demand, latency, communication cost, hallucination risk, and action safety remain important concerns. Recent foundation-model research for IoT also emphasizes efficiency, context awareness, safety, and security/privacy across prompt-, agent-, and training-based approaches [124]. This creates an opportunity for Tiny or compressed foundation models, but their suitability for lightweight cybersecurity should be demonstrated through memory, energy, latency, and hardware measurements rather than model capability alone.

Digital Twins and continual learning provide a complementary path to adaptive security. Recent IIoT research combines hybrid Digital Twins with continual learning to adapt IDS to changing traffic and emerging threats under hardware-in-the-loop evaluation [123]. Future work should examine synchronization cost, adaptation memory, update frequency, catastrophic forgetting, and recovery after incorrect adaptation. These emerging approaches can improve adaptability and contextual security analysis, but their maturity in resource-constrained environments still depends on clear deployment and resource evidence.

These opportunities are deliberately framed as evaluation priorities rather than as a separate taxonomy. End-to-end cost, layered protection, risk-aware response, robustness, low-cost explainability, shared reporting artifacts, and emerging adaptive and foundation-model-based security each address a different weakness identified in the review.

## 8 Conclusion

In this review, we have examined lightweight IDS and protection methods for resource-constrained IoT and WSN environments. The analysis shows that lightweight security cannot be judged by accuracy alone or by the word lightweight in a method description. A method may reduce the number of features, simplify the model, compress a neural architecture, move processing to the edge, or reduce protocol messages. These

reductions are useful only when the saved resource is clear and the security function is not claimed beyond the reported results. The reviewed studies show clear progress, but the supporting evidence is still uneven. Feature selection and compact ML can reduce the detection burden, but the cost of feature construction is not always included. Compressed DL and TinyML methods become more convincing when memory, latency, energy, and hardware limits are reported with detection results. FL reduces raw-data sharing, but it adds update traffic, client workload, non-IID behavior, poisoning risk, and aggregation cost. Authentication, cryptography, trust, and secure routing help protect access and communication, but they cannot replace runtime monitoring after a node is admitted. The review also shows that datasets and metrics strongly affect how results should be interpreted. Flow-based IoT datasets do not prove resistance to WSN routing attacks, and WSN routing results do not represent all botnet or IIoT behavior. Similarly, high accuracy or F1-score does not show whether a method can run within the limits of a sensor node, gateway, or edge platform. Overall, the field has made useful progress, but stronger resource-aware reporting is still needed. Future work should state where the method runs, what resource it saves, what attack surface it covers, and what response cost it introduces. A stronger lightweight security study should therefore connect detection or protection quality with realistic resource measurements and a clearly bounded security claim.

**Author contributions**

Osama A. Khashan performed the conceptualization, literature search, study screening, evidence extraction, comparative analysis, methodology development, original-draft preparation, manuscript review, editing, and revision. The author read and approved the final manuscript.

**Funding**

The author received no funding for this study.

**Data availability**

No datasets were generated or analysed during the current study.

**Code availability**

Not applicable.

**Declarations****Ethics approval and consent to participate**

Not applicable.

**Consent for publication**

Not applicable.

**Competing interests**

The authors declare no competing interests.

Received: 6 July 2026 / Accepted: 21 September 2026

Published online: 26 September 2026

**References**

1. Khashan OA, Ahmad R, Khafajah NM An automated lightweight encryption scheme for secure and energy-efficient communication in wireless sensor networks. *Ad Hoc Netw.* 2021. <https://doi.org/10.1016/j.adhoc.2021.102448>.
2. Aouedi O, Vu T-H, Sacco A, Nguyen DC, Piamrat K, Marchetto G, Pham Q-V. A survey on intelligent internet of things: applications, security, privacy, and future directions. *IEEE Commun Surv Tuts.* 2025;27(2):1238–92. <https://doi.org/10.1109/COMST.2024.3430368>.
3. Lee E, Seo Y-D, Oh S-R, Kim Y-G. A Survey on Standards for Interoperability and Security in the Internet of Things. *IEEE Commun Surv Tuts.* 2021;23(2):1020–47. <https://doi.org/10.1109/COMST.2021.3067354>.
4. Khraisat A, Alazab A A Critical Review of Intrusion Detection Systems in the Internet of Things: techniques, Deployment Strategy, Validation Strategy, Attacks, Public Datasets and Challenges. *Cybersecurity.* 2021. <https://doi.org/10.1186/s42400-021-00077-7>.
5. Chaabouni N, Mosbah M, Zemmari A, Sauvignac C, Faruki P. Network Intrusion Detection for IoT Security Based on Learning Techniques. *IEEE Commun Surv Tuts.* 2019;21(3):2671–701. <https://doi.org/10.1109/COMST.2019.2896380>.

6. Fatima M, Rehman O, Rahman IMH, Ajmal A, Park SJ Towards Ensemble Feature Selection for Lightweight Intrusion Detection in Resource-Constrained IoT Devices. *Future Internet*. 2024. <https://doi.org/10.3390/fi16100368>.
7. Kitchenham B, Charters S. Guidelines for Performing Systematic Literature Reviews in Software Engineering, Version 2.3, EBSE Technical Report EBSE-2007-01, Keele Univ. and Univ. Durham, 2007.
8. Page MJ, et al The PRISMA 2020 Statement: an Updated Guideline for Reporting Systematic Reviews. *BMJ*. 2021. <https://doi.org/10.1136/bmj.n71>.
9. Zhang D, Huang D, Chen Y, Lin S, Li C. A Lightweight IoT Intrusion Detection Method Based on Two-Stage Feature Selection and Bayesian Optimization. *AIMS Electron Electr Eng*. 2025;9(3):359–89. <https://doi.org/10.3934/electreng.2025017>.
10. Fatima M, Rehman O, Ali S, Niazi MF. ELIDS: ensemble Feature Selection for Lightweight IDS against DDoS Attacks in Resource-Constrained IoT Environment. *Future Gener Comput Syst*. 2024;159:172–87. <https://doi.org/10.1016/j.future.2024.05.013>.
11. Sharmila BS, Nagapadma R Quantized Autoencoder (QAE) Intrusion Detection System for Anomaly Detection in Resource-Constrained IoT Devices Using RT-IoT2022 Dataset. *Cybersecurity*. 2023. <https://doi.org/10.1186/s42400-023-00178-5>.
12. Javeed D, Saeed MS, Adil M, Kumar P, Jolfaei A A federated learning-based zero trust intrusion detection system for Internet of Things. *Ad Hoc Netw*. 2024. <https://doi.org/10.1016/j.adhoc.2024.103540>.
13. Alsaleh SS, Menai MEB, Al-Ahmadi S. Federated learning-based model to lightweight IDSs for heterogeneous IoT networks: state-of-the-art, challenges, and future directions. *IEEE Access*. 2024;12:134256–72. <https://doi.org/10.1109/ACCESS.2024.3460468>.
14. Danquah LKG, Appiah SY, Mantey VA, Danlard I, Akowuah EK. Computationally Efficient Deep Federated Learning With Optimized Feature Selection for IoT Botnet Attack Detection. *Intell Syst Appl*. 2025;25(200462). <https://doi.org/10.1016/j.iswa.2024.200462>.
15. Alsaleh S, Menai MEB, Al-Ahmadi S, Heterogeneity-Aware A. Semi-Decentralized Model for a Lightweight Intrusion Detection System for IoT Networks Based on Federated Learning and BiLSTM. *Sensors*. 2025;25(4). <https://doi.org/10.3390/s25041039>.
16. Tursynbek Y, Albanbay N, Djenouri D, Latif S, Akhmediyarova A, Alibiyeva Z, Alimkulova J, Oralbekova D. Secure Federated Intrusion Detection for Resource-Constrained IoT Devices Using Lightweight Cryptography: a Hardware-Validated Study. *Future Internet*. 2026;18(6):306. <https://doi.org/10.3390/fi18060306>.
17. Bukhari SMS, Zafar MH, Abou Houran M, Moosavi SKR, Mansoor M, Muaz M, Sanfilippo F. Secure and Privacy-Preserving Intrusion Detection in Wireless Sensor Networks: federated Learning With SCNN-Bi-LSTM for Enhanced Reliability. *Ad Hoc Netw*. 2024;155(103407). <https://doi.org/10.1016/j.adhoc.2024.103407>.
18. Peng H, Wu C, Xiao Y. FD-IDS: federated learning with knowledge distillation for intrusion detection in Non-IID IoT Environments. *Sensors*. 2025;25(14). <https://doi.org/10.3390/s25144309>.
19. Pathak V, Singh K, Khan T, Shariq M, Chaudhry SA, Das AK. A Secure and Lightweight Trust Evaluation Model for Enhancing Decision-Making in Resource-Constrained Industrial WSNs. *Sci Rep*. 2024;14(28162). <https://doi.org/10.1038/s41598-024-75414-0>.
20. Alghamdi AM. Design and Analysis of Lightweight and Robust Authentication Protocol for Securing the Resource Constrained IIoT Environment. *PLoS ONE*. 2025;20(2):e0318064. <https://doi.org/10.1371/journal.pone.0318064>.
21. Sai KS, Bhat R, Hegde M, Andrew J. A Lightweight Authentication Framework for Fault-Tolerant Distributed WSN. *IEEE Access*. 2023;11:83364–76. <https://doi.org/10.1109/ACCESS.2023.3302251>.
22. Jan SU, Tariq MU, Khashan OA, Alzahrani N, Ghani A. FEELAP: fuzzy Extractor-Based Efficient Lightweight Authentication Protocol for Edge-IoT Ecosystem in e-Healthcare. *IEEE Open J Comput Soc*. 2025;6:1727–40. <https://doi.org/10.1109/OJCS.2025.3616014>.
23. Khashan OA, Khafajah NM, Alomoush W, Alshinwan M. Innovative Energy-Efficient Proxy Re-Encryption for Secure Data Exchange in Wireless Sensor Networks. *IEEE Access*. 2024;12:23290–304. <https://doi.org/10.1109/ACCESS.2024.3360488>.
24. Selvi M, Santhosh Kumar SVN, Thangaramya K, Abdul Gaffar H. Energy Efficient Trust Aware Secure Routing Algorithm With Attribute Based Encryption for Wireless Sensor Networks. *Sci Rep*. 2025;15(19724). <https://doi.org/10.1038/s41598-025-03558-8>.
25. Alsaedi A, Moustafa N, Tari Z, Mahmood A, Anwar A. TON\_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems. *IEEE Access*. 8:165130–165150. <https://doi.org/10.1109/ACCESS.2020.3022862>.
26. Neto ECP, Dadkhah S, Ferreira R, Zohourian A, Lu R, Ghorbani AA. CIIoT2023: a Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. *Sensors*. 2023;23(13). <https://doi.org/10.3390/s23135941>.
27. Koroniots N, Moustafa N, Sitnikova E, Turnbull B. Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset. *Future Gener Comput Syst*. 2019;100:779–96. <https://doi.org/10.1016/j.future.2019.05.041>.
28. Meidan Y, Bohadana M, Mathov Y, Mirsky Y, Breitenbacher D, Shabtai A, Elovici Y. N-Balot: network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. *IEEE Pervasive Comput*. 2018;17(3):12–22. <https://doi.org/10.1109/MPRV.2018.03367731>.
29. Almomani I, Al-Kasasbeh B, Al-Akhras M. WSN-DS: a Dataset for Intrusion Detection Systems in Wireless Sensor Networks. *J Sens*. 2016;2016:4731953. <https://doi.org/10.1155/2016/4731953>.
30. Anwar RW, Abrar M, Salam A, Ullah F. Federated learning with LSTM for intrusion detection in IoT-based wireless sensor networks: a multi-dataset analysis. *PeerJ Comput Sci*. 2025;11:e2751. <https://doi.org/10.7717/peerj-cs.2751>.
31. Ferrag MA, Friha O, Hamouda D, Maglaras L, Janicke H. Edge-IIoTset: a New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning. *IEEE Access*. 2022;10:40281–306. <https://doi.org/10.1109/ACCESS.2022.3165809>.
32. Alsamiri J, Alsubhi K. Internet of Things Cyber Attacks Detection Using Machine Learning. *Int J Adv Comput Sci Appl*. 2019;10(12). <https://doi.org/10.14569/IJACSA.2019.0101280>.
33. Raza S, Wallgren L, Voigt T. SVELTE: real-Time Intrusion Detection in the Internet of Things. *Ad Hoc Netw*. 2013;11:2661–74. <https://doi.org/10.1016/j.adhoc.2013.04.014>.
34. Albinali H, Azzedin F. Toward RPL Attacks and Mitigation Taxonomy: systematic Literature Review Approach. *IEEE Trans Netw Serv Manag*. 2024;21(5):5215–38. <https://doi.org/10.1109/TNSM.2024.3386468>.

35. Jan SU, Ahmed S, Shakhov V, Koo I. Toward a Lightweight Intrusion Detection System for the Internet of Things. *IEEE Access*. 2019;7:42450–71. <https://doi.org/10.1109/ACCESS.2019.2907965>.
36. He M, Huang Y, Wang X, Wei P, Wang X. A Lightweight and Efficient IoT Intrusion Detection Method Based on Feature Grouping. *IEEE Internet Things J*. 2024;11(2):2935–49. <https://doi.org/10.1109/JIOT.2023.3294259>.
37. Li Z, Yao W. A Two Stage Lightweight Approach for Intrusion Detection in Internet of Things. *Expert Syst Appl*. 2024;257. <https://doi.org/10.1016/j.eswa.2024.124965>.
38. Kushwaha JP, Bhadauria S, Tapaswi S. mFCBF Based Lightweight Intrusion Detection System for IoT Networks. *Cluster Comput*. 2025;28. <https://doi.org/10.1007/s10586-024-05022-0>.
39. Mohammed TJ, Alnoor A, Abdelfattah F, Chew X, Khaw KW. Lightweight Principal Component Analysis-Driven Ensemble Framework for Real-Time Intrusion Detection in Industrial IoT Networks. *Cybersecurity*. 2026;9:46. <https://doi.org/10.1186/s42400-025-00445-7>.
40. Almalawi A. A lightweight intrusion detection system for Internet of Things: clustering and Monte Carlo cross-entropy approach. *Sensors*. 2025;25(7). <https://doi.org/10.3390/s25072235>.
41. Benaddi H, Jouhari M, Elharrouss O. A lightweight hybrid approach for intrusion detection systems using a chi-square feature selection approach in IoT. *Internet Things*. 2025;32(101624). <https://doi.org/10.1016/j.iot.2025.101624>.
42. Pandey VK, Prakash S, Gupta TK, Sinha P, Yang T, Rathore RS, Wang L, Tahir S, Bakhsh ST. Enhancing Intrusion Detection in Wireless Sensor Networks Using a Tabu Search Based Optimized Random Forest. *Sci Rep*. 2025;15(18634). <https://doi.org/10.1038/s41598-025-03498-3>.
43. Talukder MA, Khalid M, Sultana N. A Hybrid Machine Learning Model for Intrusion Detection in Wireless Sensor Networks Leveraging Data Balancing and Dimensionality Reduction. *Sci Rep*. 2025;15. <https://doi.org/10.1038/s41598-025-87028-1>.
44. Srivastava A, Sinha D. ARLHNIDS-IoT: an accurate and robust lightweight hybrid-NIDS for IoT network security. *Comput Secur*. 2025;156(104515). <https://doi.org/10.1016/j.cose.2025.104515>.
45. Zakariyya I, Kalutarage H, Al-Kadri MO. Towards a robust, effective and resource efficient machine learning technique for IoT security monitoring. *Comput Secur*. 2023;133. <https://doi.org/10.1016/j.cose.2023.103388>.
46. Misrak SF, Melaku HM. Lightweight Intrusion Detection System for IoT With Improved Feature Engineering and Advanced Dynamic Quantization. *Discover Internet Things*. 2025;5. <https://doi.org/10.1007/s43926-025-00203-8>.
47. Wang Z, Li J, Yang S, Luo X, Li D, Mahmoodi S. A Lightweight IoT Intrusion Detection Model Based on Improved BERT-of-Theseus. *Expert Syst Appl*. 2024;238(122045). <https://doi.org/10.1016/j.eswa.2023.122045>.
48. Wisanwanichthan T, Thammawichai M. A Lightweight Intrusion Detection System for IoT and UAV Using Deep Neural Networks with Knowledge Distillation. *Computers*. 2025;14(7):291. <https://doi.org/10.3390/computers14070291>.
49. Feng H, Sun M, Wang Y. MambaCNN: a Lightweight Intrusion Detection System Based on Mamba for the Internet of Vehicles. *Cybersecurity*. 2026;9(64). <https://doi.org/10.1186/s42400-025-00473-3>.
50. Rauf Ali Khan M, Barnawi AY, Munir A, Alsalmán Z, Satan DM, Sanunga. Lightweight Quantized XGBoost for Botnet Detection in Resource-Constrained IoT Networks. *IoT*. 2025;6(4). <https://doi.org/10.3390/iot6040070>.
51. Nassef L, Alghamdi MI, Chaabane SB, Abbas Q, Alawad WM, Albalawi OH, Alqaisi OI, Fakieh B. Lightweight and Energy-Aware Intrusion Detection for Industrial IoT Using TinyML and Edge AI. *Sci Rep*. 2026;16(18524). <https://doi.org/10.1038/s41598-026-50690-0>.
52. Tiwari RS, Lakshmi D, Das TK, Tripathy AK, Li K-C. A Lightweight Optimized Intrusion Detection System Using Machine Learning for Edge-Based IIoT Security. *Telecommun Syst*. 2024;87(3):605–24. <https://doi.org/10.1007/s11235-024-01200-y>.
53. Alani MM. An Explainable Efficient Flow-Based Industrial IoT Intrusion Detection System. *Comput Electr Eng*. 2023;108. <https://doi.org/10.1016/j.compeleceng.2023.108732>.
54. Gao W, Wang M, Pei Y, Li F, Wang C. A Lightweight Multi-Classification Intrusion Detection Model for Edge IoT Networks. *Electronics*. 2026;15(5). <https://doi.org/10.3390/electronics15050938>.
55. Khashan OA. Dual-stage machine learning approach for advanced malicious node detection in WSNs. *Ad Hoc Netw*. 2025;166. <https://doi.org/10.1016/j.adhoc.2024.103672>.
56. Albanbay N, Tursynbek Y, Graffi K, Uskenbayeva R, Kalpeyeva Z, Abilkaiyr Z, Ayapov Y. Federated Learning-Based Intrusion Detection in IoT Networks: performance Evaluation and Data Scaling Study. *J Sens Actuator Netw*. 2025;14(4). <https://doi.org/10.3390/jsan14040078>.
57. Mallidi SKR, Ramisetty RR. A multi-level intrusion detection system for industrial IoT using bowerbird courtship-inspired feature selection and hybrid data balancing. *Discover Comput*. 2025;28. <https://doi.org/10.1007/s10791-025-09632-z>.
58. Yang T, Chen J, Deng H, He B. A lightweight intrusion detection algorithm for IoT based on data purification and a separable convolution improved CNN. *Knowl -Based Syst*. 2024;304(112473). <https://doi.org/10.1016/j.knosys.2024.112473>.
59. El-Sherif M, Khattab A, El-Soudani M. Intrusion detection using TCP/IP single packet header binary image for IoT networks. *Cybersecurity*. 2025;8. <https://doi.org/10.1186/s42400-025-00441-x>.
60. Hnamte V, Najar AA, Laldinsanga C, Hussain J, Hmingliana L. A lightweight intrusion detection system using deep convolutional neural network. *Comput Electr Eng*. 2025;127(Part A):110561. <https://doi.org/10.1016/j.compeleceng.2025.110561>.
61. Kumar R, Swarnkar M. QulIDS: a quantum support vector machine-based intrusion detection system for IoT networks. *J Netw Comput Appl*. 2025;234. <https://doi.org/10.1016/j.jnca.2024.104072>.
62. Villafranca A, Cano M-D. A lightweight edge-DL intrusion detection system for IoT sustainable smart-agriculture. *Internet Things*. 2025;34. <https://doi.org/10.1016/j.iot.2025.101818>.
63. Aldaei A, Ahanger TA, Ullah I. Deep learning-inspired IoT-IDS mechanism for edge computing environments. *Sensors*. 2023;23(24). <https://doi.org/10.3390/s23249869>.
64. Qureshi SS, He J, Qureshi SU, Zhu N, Wajhat A, Nazir A, Ullah F, Wadud A. Advanced AI-driven intrusion detection for securing cloud-based industrial IoT. *Egypt Inf J*. 2025;30(100644). <https://doi.org/10.1016/j.eij.2025.100644>.
65. Hossain MA. Deep learning-based intrusion detection for IoT networks: a scalable and efficient approach. *EURASIP J Inf Secur*. 2025;2025(28). <https://doi.org/10.1186/s13635-025-00202-w>.
66. Mahadik SS, Pawar PM, Muthalagu R. Edge-federated learning-based intelligent intrusion detection system for heterogeneous Internet of Things. *IEEE Access*. 2024;12:81736–57. <https://doi.org/10.1109/ACCESS.2024.3410046>.
67. Bhavsar MH, Bekele YB, Roy K, Kelly JC, Limbrick D. FL-IDS: federated learning-based intrusion detection system using edge devices for transportation IoT. *IEEE Access*. 2024;12:52215–26. <https://doi.org/10.1109/ACCESS.2024.3386631>.
68. Lee S-J, Lee I-G. Lightweight federated learning-based intrusion detection system for industrial internet of things. *ICT Express*. 2025;11(4):690–5. <https://doi.org/10.1016/j.icte.2025.05.002>.

69. ugli AAA, Shin JS. SAFE-IDS: a privacy-preserving framework for overcoming non-IID challenges in federated intrusion detection. *Comput Secur*. 2025;155. <https://doi.org/10.1016/j.cose.2025.104492>.
70. Abd Elaziz M, Fares IA, Dahou A, Shrahili M. Federated learning framework for IoT intrusion detection using tab transformer and nature-inspired hyperparameter optimization. *Front Big Data*. 2025;8:1526480. <https://doi.org/10.3389/fdata.2025.1526480>.
71. Özmen G, Yiltas-Kaplan D. Robust and lightweight federated learning for NB-IoT security: a blockchain-verified CNN-RNN approach. *Sensors*. 2026;26(11):3578. <https://doi.org/10.3390/s26113578>.
72. Nishad DK, Singh R, Khalid S. A lightweight blockchain-enabled federated intrusion prevention framework for resource-constrained industrial IoT devices to detect and mitigate emerging cyberattacks. *J Cloud Comput*. 2026;15. <https://doi.org/10.1186/s13677-026-00843-3>.
73. Almotairi A, Atawneh S, Khashan OA, Khafajah NM. Enhancing intrusion detection in IoT networks using machine learning-based feature selection and ensemble models. *Syst Sci Control Eng*. 2024;12(1). <https://doi.org/10.1080/21642583.2024.2321381>.
74. Adjewa F, Esseghir M, Merghem-Boulahia L. From edge transformer to IoT decisions: offloaded embeddings for lightweight intrusion detection. *Sensors*. 2026;26(2):356. <https://doi.org/10.3390/s26020356>.
75. Ismail S, Dandan S, Qushou A. Intrusion detection in IoT and IIoT: comparing lightweight machine learning techniques using TON\_IoT, WUSTL-IIoT-2021, and EdgelloTset datasets. *IEEE Access*. 2025;13:73468–85. <https://doi.org/10.1109/ACCESS.2025.3554083>.
76. Umar HGA, Yasmeen I, Aoun M, Mazhar T, Khan MA, Jaghdam IH, Hamam H. Energy-efficient deep learning-based intrusion detection system for edge computing: a novel DNN-KDQ model. *J Cloud Comput*. 2025;14. <https://doi.org/10.1186/s13677-025-00762-9>.
77. Akhi M, Eising C, Dhirani LL. Securing IoT using lightweight TCN for edge deployment on Raspberry Pi 4. *IEEE Open J Commun Soc*. 2026;7:442–60. <https://doi.org/10.1109/OJCOMS.2025.3649498>.
78. Ding Z, Zheng J, Wu X. NIDS-Mamba: lightweight network intrusion detection for IoT sensor networks via state space models. *Sensors*. 2026;26(9). <https://doi.org/10.3390/s26092766>.
79. Chandroth J, Ali J. Lightweight MLP-based feature extraction with linear classifier for intrusion detection system in Internet of Things. *Electronics*. 2026;15(8):1604. <https://doi.org/10.3390/electronics15081604>.
80. Moucharraf M, Ridouani M, Salahdine F, Kaabouch N. AutoBoost-IoT: a hybrid model for intrusion detection in IoT networks. *Future Internet*. 2026;18(5). <https://doi.org/10.3390/fi18050229>.
81. Khan A, Hussain MA, Anwer F. A hybrid lightweight deep learning-based intrusion detection approach in IoT utilizing feature selection and explainable artificial intelligence. *IEEE Access*. 2025;13:192451–66. <https://doi.org/10.1109/ACCESS.2025.3630753>.
82. Ebrahimi F, Javidan R, Akbari R, Hosseini Y. Intrusion detection in the Internet of Things using convolutional neural networks: an explainable AI approach. *Cybersecurity*. 2025;8. <https://doi.org/10.1186/s42400-025-00369-2>.
83. Otokwala U, Petrovski A, Kalutarage H. Optimized common features selection and deep-autoencoder (OCFSDA) for lightweight intrusion detection in Internet of Things. *Int J Inf Secur*. 2024;23(4):2559–81. <https://doi.org/10.1007/s10207-024-00855-7>.
84. Alani MM, Awad AI. An intelligent two-layer intrusion detection system for the Internet of Things. *IEEE Trans Ind Informat*. 2023;19(1):683–92. <https://doi.org/10.1109/TII.2022.3192035>.
85. Hamdouchi A, Idri A. Evaluating the performance of TinyML singular and ensemble techniques for intrusion detection in IoT networks. *Microprocess Microsyst*. 2025;117(105172). <https://doi.org/10.1016/j.micpro.2025.105172>.
86. Yang Y-M, Chang K-C, Luo J-N. A lightweight training approach for MITM detection in IoT networks: time-window selection and generalization. *Appl Sci*. 2025;15(22). <https://doi.org/10.3390/app152212147>.
87. Sethu Subramanian N, Prabu P, Jain K, Krishnan P. Edge AI Bridge: a micro-layer intrusion detection architecture for smart-city IoT networks. *IoT*. 2026;7(2). <https://doi.org/10.3390/iot7020033>.
88. Bilal MA, Islam IU, Iltaf N, Khan MJ, Khan J. Federated learning with explainable AI for malicious traffic detection in IoT networks. *IEEE Access*. 2025;13:173368–83. <https://doi.org/10.1109/ACCESS.2025.3613459>.
89. Taheri R, Jafari R, Gegov A, Arabikhan F, Ichtev A. Explainable AI for federated learning-based intrusion detection systems in connected vehicles. *Electronics*. 2025;14(22). <https://doi.org/10.3390/electronics14224508>.
90. Sinha P, Sahu D, Prakash S, Rathore RS, Dixit P, Pandey VK, Hunko I. An efficient data driven framework for intrusion detection in wireless sensor networks using deep learning. *Sci Rep*. 2025;15:34046. <https://doi.org/10.1038/s41598-025-12867-x>.
91. Al-Otaibi S, Ayouni S, Sarwar N, Irshad A, Ullah F. AI-driven intrusion detection and lightweight authentication framework for secure and efficient medical sensor networks. *Sci Rep*. 2026;16. <https://doi.org/10.1038/s41598-025-31981-4>.
92. Çetintav I, Sandikkaya MT. A review of lightweight IoT authentication protocols from the perspective of security requirements, computation, communication, and hardware costs. *IEEE Access*. 2025;13:37703–23. <https://doi.org/10.1109/ACCESS.2025.3546147>.
93. Cagua G, Gauthier-Umaña V, Lozano-Garzon C. Implementation and performance of lightweight authenticated encryption ASCON on IoT devices. *IEEE Access*. 2025;13:16671–82. <https://doi.org/10.1109/ACCESS.2025.3529757>.
94. Ahmed S, Ahmad N, Shah NA, Abro GEM, Wijayanto A, Hirs A, Altaf AR. Lightweight AES design for IoT applications: optimizations in FPGA and ASIC with DFA countermeasure strategies. *IEEE Access*. 2025;13:22489–509. <https://doi.org/10.1109/ACCESS.2025.3533611>.
95. Imdad M, Fazil A, Ramli SNB, Ryu J, Mahdin HB, Manzoor Z. DNA-PRESENT: an improved security and low-latency, lightweight cryptographic solution for IoT. *Sensors*. 2024;24(24). <https://doi.org/10.3390/s24247900>.
96. Mohanapriya R, Nithish Kumar V. Modified SPECK (M-SPECK) lightweight cipher architecture for resource-constrained applications. *IEEE Access*. 2025;13:88993–9002. <https://doi.org/10.1109/ACCESS.2025.3570727>.
97. Singamaneni KK. A novel lightweight hybrid cryptographic framework for secure smart card operations. *EURASIP J Inf Secur*. 2025;2025(13). <https://doi.org/10.1186/s13635-025-00204-8>.
98. Aslan B. Energy consumption analysis of ISO/IEC 29192-2 standard lightweight ciphers. *Appl Sci*. 2025;15(7). <https://doi.org/10.3390/app15073928>.
99. Heidari A, Navimipour NJ, Unal M. A secure intrusion detection platform using blockchain and radial basis function neural networks for Internet of Drones. *IEEE Internet Things J*. 2023;10(10):8445–54. <https://doi.org/10.1109/JIOT.2023.3237661>.

100. Mustafa R, Sarkar NI, Mohaghegh M, Pervez S, Vohra O. Cross-layer analysis of machine learning models for secure and energy-efficient IoT networks. *Sensors*. 2025;25(12). <https://doi.org/10.3390/s25123720>.
101. Yang R, He H, Xu Y, Xin B, Wang Y, Qu Y, Zhang W. Efficient intrusion detection toward IoT networks using cloud-edge collaboration. *Comput Netw*. 2023;228(109724). <https://doi.org/10.1016/j.comnet.2023.109724>.
102. Tekin N, Acar A, Aris A, Uluagac AS, Gungor VC. Energy consumption of on-device machine learning models for IoT intrusion detection. *Internet Things*. 2023;21(100670). <https://doi.org/10.1016/j.iot.2022.100670>.
103. Pandey VK, Sahu D, Prakash S, Rathore RS, Dixit P, Hunko I. A lightweight framework to secure IoT devices with limited resources in cloud environments. *Sci Rep*. 2025;15. <https://doi.org/10.1038/s41598-025-09885-0>.
104. Siliveri AK, Rao KRM, Solleti R. Dual-path feature extraction based hybrid intrusion detection in IoT networks. *Comput Electr Eng*. 2025;122. <https://doi.org/10.1016/j.compeleceng.2024.109949>.
105. Wang L, Petrova K, Yang ML. Trust models in wireless sensor networks for defending against denial-of-service attacks: a literature review. *Appl Sci*. 2025;15(6). <https://doi.org/10.3390/app15063075>.
106. Khraisat A, Alazab A, Singh S, Jan T, Gomez A. Survey on federated learning for intrusion detection system: concept, architectures, aggregation strategies, challenges, and future directions. *ACM Comput Surv*. 2024;57(1). <https://doi.org/10.1145/3687124>.
107. Yin Z, Chen H, Ma H, Hu T, Bai L. CAEAIID: an incremental contrast learning-based intrusion detection framework for IoT networks. *Comput Netw*. 2025;262. <https://doi.org/10.1016/j.comnet.2025.111161>.
108. Ishtiaq W, Zannat A, Parvez AHMS, Hossain MA, Kanchan MH, Tarek MM. CST-AFNet: a dual attention-based deep learning framework for intrusion detection in IoT networks. *Array*. 2025;27(100501). <https://doi.org/10.1016/j.array.2025.100501>.
109. Nandanwar H, Katarya R. Securing Industry 5.0: an explainable deep learning model for intrusion detection in cyber-physical systems. *Comput Electr Eng*. 2025;123. <https://doi.org/10.1016/j.compeleceng.2025.110161>.
110. Wang Y, Azad MA, Zafar M, Gul A. Enhancing AI transparency in IoT intrusion detection using explainable AI techniques. *Internet Things*. 2025;33. <https://doi.org/10.1016/j.iot.2025.101714>.
111. Ogunseyi TB, Thiyagarajan G, He H, Bist V, Du Z. Performance analysis of explainable deep learning-based intrusion detection systems for IoT networks: a systematic review. *Sensors*. 2026;26(2). <https://doi.org/10.3390/s26020363>.
112. Alabbadi A, Bajaber F. An intrusion detection system over the IoT data streams using explainable artificial intelligence (XAI). *Sensors*. 2025;25(3). <https://doi.org/10.3390/s25030847>.
113. Jain P, Rathour A, Sharma A, Chhabra GS. Bridging explainability and security: an XAI-enhanced hybrid deep learning framework for IoT device identification and attack detection. *IEEE Access*. 2025;13:127368–90. <https://doi.org/10.1109/ACCESS.2025.3590159>.
114. Rahman MM, Shakil SA, Mustakim MR. A survey on intrusion detection system in IoT networks. *Cyber Secur Appl*. 2025;3(100082). <https://doi.org/10.1016/j.csa.2024.100082>.
115. Mallidi SKR, Ramisetty RR. Advancements in training and deployment strategies for AI-based intrusion detection systems in IoT: a systematic literature review. *Discover Internet Things*. 2025;5(8). <https://doi.org/10.1007/s43926-025-00099-4>.
116. Ullah S, Wu J, Kamal MM, Saudagar AKJ. KronNet: a lightweight Kronecker enhanced feed forward neural network for efficient IoT intrusion detection. *Sci Rep*. 2025;15. <https://doi.org/10.1038/s41598-025-08921-3>.
117. Faiak AM, Rafin SJ, Roy P, Razzaque MA, Hassan MR, Polash M, Hassan MM. LiteKD: a lightweight knowledge-distillation deep learning framework for intrusion detection in IoT networks. *Comput Netw*. 2026;286(112423). <https://doi.org/10.1016/j.comnet.2026.112423>.
118. Ortiz-Garcés I, Román-Cañizares M, Palacios P, Villegas-Ch W. NAS-Optimized TinyML Intrusion Detection for Ultralow-Power IoT Edge Devices. *Internet Things*. 2026;39(102034). <https://doi.org/10.1016/j.iot.2026.102034>.
119. Zhou R, Wang Z, Yang S, He D, Chan S. Federated learning based on two-stage knowledge distillation for intrusion detection in industrial IoT. *Expert Syst Appl*. 2026;299(pt C):130144. <https://doi.org/10.1016/j.eswa.2025.130144>.
120. Kavitha MJ, Geetha MR, Sajjan RI. Knowledge-based adaptive routing for energy efficiency and attack detection in ad hoc wireless sensor networks. *Comput Netw*. 2025;259. <https://doi.org/10.1016/j.comnet.2025.111086>.
121. Ma H, Zhang W, Zhang D, Chen B. An IoT intrusion detection framework based on feature selection and large language models fine-tuning. *Sci Rep*. 2025;15. <https://doi.org/10.1038/s41598-025-08905-3>.
122. Chirag C, Rawat M, Singal G. Design and implementation of IoT intrusion detection centric agentic AI with auxiliary conversational capabilities. *Pervasive Mob Comput*. 2026;123. <https://doi.org/10.1016/j.pmcj.2026.102271>.
123. Melis A, Piroddi A, Lam CT, Pau G, Girau R. Anomaly detection of cyber threats in industrial IoT networks via hybrid digital twins and continual learning. *Internet Things*. 2026;37(101915). <https://doi.org/10.1016/j.iot.2026.101915>.
124. Wei H, Lee DY, Rohal S, Hu Z, Rossi R, Fang S, Pan S. A survey of foundation models for IoT: taxonomy and criteria-based analysis. *CCF Trans Pervasive Comput Interact*. 2026;8:1–29. <https://doi.org/10.1007/s42486-025-00208-w>.
125. ElRabaa R, Rawas S, El-Zaart A. A Gamma-enhanced Naïve Bayes model for robust intrusion detection in IoMT networks. *Discover Internet Things*. 2026;6(10). <https://doi.org/10.1007/s43926-025-00281-8>.
126. Jodayree M, Kavooosi Ghafi A, Amiri S, Shaykholeslami P. Explainable zero-day attack detection in IoMT using transformer-based time-series modeling. *Sci Rep*. 2026;16. <https://doi.org/10.1038/s41598-026-50813-7>.
127. Hosain Y, Çakmak M. XAI-XGBoost: an innovative explainable intrusion detection approach for securing Internet of Medical Things systems. *Sci Rep*. 2025;15(22278). <https://doi.org/10.1038/s41598-025-07790-0>.
128. Palaniappan S, Sengan S. Hybrid feature selection for IoMT based intrusion detection system for integrating mutual information filtering with deep learning based accelerated metaheuristic optimization. *Sci Rep*. 2026;16. <https://doi.org/10.1038/s41598-026-47264-5>.
129. Du J. A novel intrusion detection system for IIoT in 5G networks using attention-augmented federated learning and lightweight transformer architectures. *Sci Rep*. 2026;16(23974). <https://doi.org/10.1038/s41598-026-54748-x>.
130. Medhi J, Liu R, Wang Q, Chen X. A lightweight and efficient intrusion detection system (IDS) for unmanned aerial vehicles. *Neural Comput Appl*. 2025;37:15819–36. <https://doi.org/10.1007/s00521-025-11276-5>.
131. Waqdan M, Louafi H, Mouhoub M. Security risk assessment in IoT environments: a taxonomy and survey. *Comput Secur*. 2025;154(104456). <https://doi.org/10.1016/j.cose.2025.104456>.
132. Czekster RM, Webber T, Bertolin Furstenau L, Marcon C. Dynamic risk assessment approach for analysing cyber security events in medical IoT networks. *Internet Things*. 2025;29. <https://doi.org/10.1016/j.iot.2024.101437>.
133. Zhai B, Akande ON, Agarwal S, Pak W. Security risk assessment of Internet of Things health devices using DREAD and STRIDE models. *Ain Shams Eng J*. 2025;16(11):103721. <https://doi.org/10.1016/j.asej.2025.103721>.

134. Javed S, Wu G, Javed H, Khashan OA, Hassan H, Ghani A. An automated multi-scale and multi-contextual MobileNetv3 for malware detection based on IoT. *Array*. 2026;29. <https://doi.org/10.1016/j.array.2026.100681>.
135. Prasanna K, Prajapati J, Patel U. AI driven cyber threat classification and risk assessment for IoT ecosystem security. *Discover Artif Intell*. 2026. <https://doi.org/10.1007/s44163-026-01748-5>.
136. Islam MM, Salsabil U, Nurmatov M, Hossain S. Lightweight Intrusion Detection Systems for IoT-Edge Environments: a PRISMA-ScR Systematic Review of Deployability Evidence and a Unified Assessment Framework. *Future Internet*. 2026;18(6). <https://doi.org/10.3390/fi18060300>.

### **Publisher's note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.