SSR Journal of Artificial Intelligence (SSRJAI)

OPEN ACCESS

Homepage: https://ssrpublisher.com/ssrjai/

Volume 3, Issue 7, 2026 ISSN: 3049-0413

Detection Depth and Distributed Trust in AI-Based IoT Intrusion Detection: A Systematic Analysis

Gilbert Imuetinyan Osaze Aimufua, Hashim Abdul Isah

Centre for Cyberspace Studies, Nasarawa State University, Keffi, Nigeria

Received: *05.06.2026* | ***Accepted:*** *23.07.2026* | ***Published:*** *26.07.2026*

***Corresponding author:** Hashim Abdul Isah

DOI: *10.5281/zenodo.21590526*

Abstract **Original Research**

Blockchain-integrated deep learning intrusion detection systems for the Internet of Things have attracted growing research attention, yet the relationship between detection depth and blockchain trust scope in these architectures has not been examined systematically. This analysis codes 50 published studies against seven dimensions: protocol-awareness level (PA-LEVEL), blockchain role, blockchain integration depth, consensus mechanism, federated learning use, deployment domain, and study quality. A three-tier PA-LEVEL taxonomy distinguishes flow-level statistical detection (L1), protocol field awareness (L2), and protocol state and semantic awareness (L3). Forty-eight of the 50 included studies operate at L1 irrespective of blockchain integration depth or deployment domain, and 18 studies describe blockchain integration without evaluation. End-to-end detection-to-logging latency, the operationally critical trust metric, is reported in only four studies.

Three gap clusters are identified and mapped to an established gap taxonomy. The first is a methodological gap: the dominant evaluation datasets do not meet IoT representativeness criteria for device traffic, protocol coverage, or attack specificity. The second is an empirical gap: blockchain trust properties are asserted without benchmarking. The third is a knowledge gap with a practical-knowledge dimension: no study co-designs detection depth and trust scope as jointly constrained variables derived from a shared threat model. The three gaps form a dependency chain that constrains the order in which they can be resolved, and a research agenda addressing each cluster in sequence is proposed.

Keywords: Intrusion detection systems, Internet of Things, Deep learning, Blockchain, Protocol-awareness, Trust management, Systematic analysis.

1. Introduction

IoT deployments now extend across industrial control systems, healthcare infrastructure, and smart-city services. The devices in these deployments communicate over protocols designed for low-overhead operation rather than security: MQTT, CoAP, Modbus, LoRaWAN, and Zigbee (Meneghello, Calore, Zucchetto, Polese and Zanella, 2019). The attack surface created by this proliferation is substantial: device authentication is weak or absent in many commercial products, protocol fields carry no integrity protection by default, and the traffic generated by constrained nodes is heterogeneous enough to defeat detection methods trained on generic network data (Meneghello et al., 2019).

Deep learning has been adopted as the primary detection mechanism in response to these conditions (Liao, Murah, Hasan, Aman, Fang, Hu and Khan, 2024), and high classification

accuracy is reported across a growing body of experimental evaluations. The detection problem, however, has not been solved: accuracy figures obtained in laboratory evaluation conditions do not translate directly into operational trust, and the datasets on which most detection models are trained do not capture the protocol-level characteristics of the environments they claim to protect (Ferrag, Maglaras, Moschoyiannis and Janicke, 2020).

Blockchain has been proposed as a trust layer for AI-based intrusion detection in IoT environments, on the premise that immutable, decentralised logging of detection decisions provides verifiable audit evidence that a centralised architecture cannot (Liu, Wang, Yan, Wan and Jäntti, 2023). The combination is theoretically motivated: detection generates output records that require credibility for downstream action, and blockchain provides tamper-evident storage with consensus-backed integrity. Federated learning has been incorporated into a further subset of these systems to address the data-sharing constraints that arise when detection nodes are distributed across organisational boundaries (Friha, Ferrag, Shu, Maglaras, Choo and Nafaa, 2022). The integration of blockchain and federated learning in distributed IoT security architectures has been examined in vehicular and agricultural domains (Javed, Hassan, Shahzad, Ahmed, Singh, Baker and Gadekallu, 2022; Friha et al., 2022), and the broader case for blockchain-based trust management in IoT has been surveyed systematically (Liu et al., 2023).

Prior surveys have catalogued deep learning architectures for IoT IDS (Ferrag et al., 2020; Liao et al., 2024) and blockchain-based trust management in IoT (Liu et al., 2023), and the combination of blockchain with federated learning in distributed IoT security has been reviewed in domain-specific settings (Javed et al., 2022; Friha et al., 2022). None of these surveys introduces a protocol-awareness taxonomy for the detection layer, none codes integration depth at the blockchain evaluation level, and none examines whether the two components are architecturally aligned. What the literature has not examined is whether the detection component and the trust component in these architectures are designed to be mutually consistent: whether the depth at which a detection node operates is matched to the scope of the trust assertion the blockchain layer is expected to support. This gap is established empirically from the 50-study coding reported in Section 6.4.

This paper reports a systematic analytical study: a structured coding and synthesis of published work rather than a meta-analysis or reproducible systematic review, and distinct from prior surveys in that it introduces an original protocol-awareness taxonomy and applies it to examine the co-design relationship between detection depth and blockchain trust scope. The analysis addresses the gap identified above through a systematic examination of 50 published studies on blockchain-integrated deep learning IDS in IoT, IIoT, and healthcare IoT deployments. Each study was coded against seven dimensions: protocol-awareness level (PA-LEVEL), blockchain role, blockchain integration depth (BC-DEPTH), consensus mechanism (BC-CONSENSUS), federated learning use, deployment domain, and study quality. The PA-LEVEL dimension applies a three-tier taxonomy, developed for this analysis, to capture the degree to which a detection node can observe the protocol-level behaviour through which IoT attacks are conducted: L1 designates flow-level statistical feature extraction, L2 designates protocol field awareness, and L3 designates protocol state and semantic awareness. The analysis is guided by four research questions:

1. At what protocol-awareness level do blockchain-integrated deep learning IDS operate, and how does that level distribute across deployment domains and blockchain integration depths?
2. How do the consensus mechanisms and integration depths adopted in the reviewed literature relate to the operational trust requirements of the target deployment domains?
3. What evaluation substrates do the included studies use (datasets, representativeness criteria, and federated configurations), and do those substrates support the detection and trust claims made?

4. What gap types, under the taxonomy of Miles (2017), characterise the distance between the current state of the literature and a co-designed detection-trust architecture?

The remainder of this paper is organised as follows. Section 2 describes the search strategy, coding instrument, and quality assessment rubric. Section 3 examines detection architecture, PA-LEVEL distribution, and detection performance. Section 4 analyses blockchain integration depth, consensus mechanism selection, and benchmarking gaps. Section 5 addresses evaluation substrate and the role of federated learning. Section 6 synthesises the three gap clusters and proposes a dependency-ordered research agenda.

2. Methodology

2.1. Corpus Assembly and Eligibility Criteria

The study set was assembled through a unified keyword search across four databases: IEEE Xplore, ScienceDirect, Springer Link, and ACM Digital Library. The search string combined three thematic blocks (IoT intrusion detection, deep learning architectures, and blockchain mechanisms) joined by Boolean AND operators. The core IEEE Xplore string was:

("Internet of Things" OR "IoT" OR "IIoT" OR "Industrial IoT" OR "healthcare IoT") AND ("intrusion detection" OR "IDS" OR "anomaly detection") AND ("deep learning" OR "convolutional neural network" OR "LSTM" OR "recurrent neural network" OR "federated learning") AND ("blockchain" OR "distributed ledger" OR "smart contract")

Per-database syntax adaptations substituted field-code prefixes and wildcard conventions as required by each database interface. A date filter of 2018–2026 was applied at database level where supported and at screening stage otherwise. Figure 1 summarises the full search and screening workflow.

Eligibility was determined by six inclusion criteria:

- Studies addressing AI- or ML-based intrusion detection in an IoT, IIoT, or healthcare IoT environment (**IC1**).
- Incorporated blockchain in any operational role, whether tamper-proof logging, federated aggregation security, trust management, smart contract access control, or evidence storage (**IC2**).
- Reported at least one quantitative detection performance metric (**IC3**).
- Appeared in a peer-reviewed journal or conference proceedings (**IC4**).
- Was available in English full text (**IC5**).
- Was published from 2018 onward (**IC6**).

Three exclusion criteria removed studies that:

- Presented a purely conceptual architecture with no implemented or evaluated detection system (**EC1**).
- Described blockchain as future work only with no implemented component (**EC2**).
- Duplicated findings from another included publication by the same authors on the same system (**EC3**).

Three borderline decision rules addressed cases that the primary criteria did not resolve unambiguously:

- **BR1** specified that studies implementing but not benchmarking a blockchain component were included and coded accordingly; IC3 applies to the detection layer only.
- **BR2** defined inclusion conditions for studies whose primary deployment target was enterprise infrastructure but whose evaluation used an IoT-specific dataset.
- **BR3** addressed federated learning architectures using Merkle tree-based or zero-knowledge proof audit mechanisms in lieu of a distributed ledger: inclusion required the authors to characterise the mechanism as blockchain-inspired or blockchain-equivalent.

Seven candidate studies were excluded after full-text review: four from the blockchain-IDS pool (S16, S25, S57, S59) for failing IC3, IC1, IC2, and IC2 respectively, and three framework-bridging candidates (Wang2019, Sammour2026, PeralesGomez2023) under EC1, as each

presented a conceptual architecture proposing blockchain IDS integration without an implemented or evaluated detection system within an IoT environment. The final set comprises 50 studies (B01–B50), ordered alphabetically by first author.

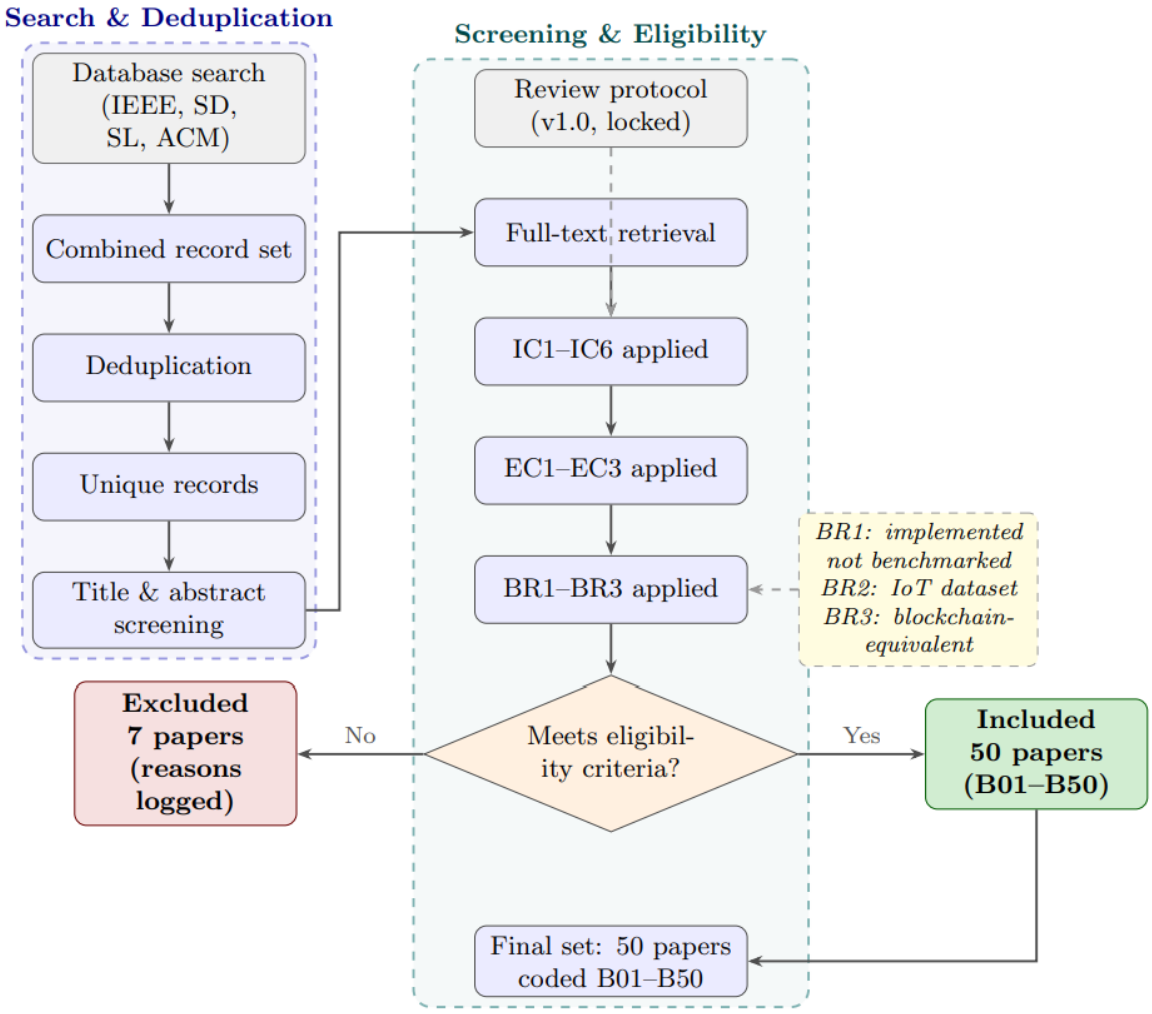

Figure 1: Search, deduplication, and eligibility screening work- follow. IC = inclusion criterion; EC = exclusion criterion; BR = borderline decision rule.

2.2. Coding Instrument

Each included study was coded on seven dimensions drawn from two analytical layers. The detection layer captured the protocol-awareness level of the deep learning component, the blockchain role and integration depth, the consensus mechanism, the use of federated learning, and the deployment domain. The contextual layer recorded the publication year and an overall quality rating. Table 1 presents the full coding schema.

Table 1: Coding schema applied to all 50 included studies.

Code Dimension	Layer	Value set
PA-LEVEL	Detection	L1 (flow statistics) / L2 (protocol-typed fields) / L2.5 (payload/byte) / L3 (finite state machine [FSM] state)
BC-ROLE	Blockchain	Tamper-proof logging / Trust management / federated learning (FL) aggregation security / Smart contract access control / Evidence storage / Multiple / None
BC-DEPTH	Blockchain	Described only / Implemented (not benchmarked) / Implemented and benchmarked
BC-CONSENSUS	Blockchain	PoA / PoW / PoS / PBFT / Hyperledger Fabric / Other / Not specified
FL	Detection	Yes / No
DOMAIN	Contextual	Consumer IoT / IIoT / Healthcare IoT / Smart city / 5G-6G / Multi-domain
QUALITY	Contextual	Strong / Adequate / Weak

The PA-LEVEL dimension classifies each study's detection node on four levels. L1 denotes detection operating on flow-level statistics with no reference to protocol-typed fields; L2 denotes detection on protocol header fields typed by specification; L2.5 denotes inspection of payload bytes or raw packet content; and L3 denotes detection that models protocol finite state machine transitions. PA-LEVEL coding was assigned by reading the feature set and input representation reported in each study's methodology section. All seven coding dimensions were assigned by a single coder. No second-coder validation was conducted, and no interrater reliability figure is available. For dimensions with binary or categorical choices and unambiguous criteria (BC-DEPTH, BC-CONSENSUS, FL, DOMAIN), coding decisions are directly verifiable from the study texts. For PA-LEVEL, where feature descriptions vary in specificity across studies, a conservative default rule was applied: where field names were not listed explicitly, L1 was assigned. The PA-LEVEL distribution reported should therefore be treated as a lower-bound estimate of protocol-awareness depth rather than a precisely calibrated measurement. This is acknowledged as a study limitation.

2.3. Quality Assessment

Studies were rated Strong, Adequate, or Weak against four criteria. A **Strong** rating required all four: a clearly stated research question; evaluation on an IoT-specific dataset (ToN-IoT, BoT-IoT, Edge-IIoTset, MQTTset, or N-BaIoT)

or a physical testbed; an implemented and benchmarked blockchain component; and an ablation or component-level comparison. A study met the **Adequate** threshold if it satisfied criterion one and at least one of criteria two through four. The **Weak** rating applied if only the research question criterion was met. Weak-rated studies are excluded from quantitative claims; Strong and Adequate studies both contribute to the descriptive synthesis.

3. Detection Architecture in Blockchain-Integrated IoT IDS

3.1. Deep Learning Architectures Employed

Architecture type was determined from each study's methodology section by identifying the primary model used for intrusion classification or anomaly scoring. Where a study combined multiple architectures, all were recorded; counts therefore sum to more than 50. Table 2 presents the full distribution.

Table 2: Deep learning architecture distribution across the 50 included studies.

Representative citations indicate studies where the architecture forms the primary detection component. Counts exceed 50 because several studies combine multiple architectures.

Architecture	Role in detection	Count	Representative studies
CNN	Feature extraction from flow vectors	19	Babar et al. (2026); Kolsur and Hosmani (2025); Ullah et al. (2025)
LSTM	Sequential flow modelling	18	Liu et al. (2022); Kolsur and Hosmani (2025); Lilhore et al. (2026)
MLP / DNN	Feature-vector classification	13	Odeh and Abu Taleb (2025); Villegas-Ch et al. (2025); Mahendran et al. (2025)
Ensemble / XGBoost	Multi-classifier or boosting pipeline	12	Rathee et al. (2022); Ahakonye et al. (2025); Alatawi (2025)
BiLSTM	Bidirectional sequential modelling	9	Babar et al. (2026); Ullah et al. (2025); Liu et al. (2022)
Autoencoder	Anomaly detection via reconstruction	8	Mirdula and Roopa (2023); Liu et al. (2022); Villegas-Ch et al. (2025)
GAN	Training data augmentation	6	Villegas-Ch et al. (2025); Ahmed et al. (2022); Alruwaili et al. (2024)
CNN-LSTM (hybrid)	Local feature + temporal dependency	6	Kolsur and Hosmani (2025); Liu et al. (2022); He et al. (2026)

Architecture	Role in detection	Count	Representative studies
Transformer / Attention	Sequence-level attention encoding	5	Parashar et al. (2026); Sengan et al. (2026); Srinivasan and Senthilkumar (2025)
GRU	Lightweight sequential modelling	4	Ahad et al. (2023); Ngo et al. (2024); Aouini and Pekar (2022)
Random Forest	Ensemble classification baseline	3	Mirdula and Roopa (2023); Shahraki et al. (2022); Kumar et al. (2021)

CNN and LSTM architectures appear across all deployment domains and dataset types represented in the study set. No study states an explicit rationale for architecture selection based on the protocol structure or traffic characteristics of the target deployment environment. Liu et al. (2022) deploy a hybrid autoencoder-MLP model on three DDoS-focused datasets (CICDDoS2019, CIC-IDS2017, and BoT-IoT); the architecture is described in terms of layer dimensions, activation functions, and hyperparameters, but no account is given of why an autoencoder-MLP combination is appropriate for the smart transport IoT environment the study targets. Ullah et al. (2025) deploy a Physics-Informed Neural Network at the edge tier of a four-tier IoT architecture and note in their own evaluation that the approach is theoretically motivated but unvalidated beyond the BoT-IoT benchmark. Architecture choice is reported without deployment justification in both cases. The basis for selection can therefore only be inferred from the evaluation dataset used, not confirmed from the studies themselves. This is a limitation of the available reporting rather than a finding about architecture suitability.

3.2. Protocol-Awareness Level

PA-LEVEL was assigned by reading the feature set and input representation reported in each study's methodology section. A study was coded L1 if its detection node operated on flow-level statistics with no reference to protocol-typed fields, regardless of architectural complexity. L3 required explicit modelling of protocol finite state machine (FSM) transitions or equivalent session-state sequencing.

Out of the 50 included studies, 48 are coded L1 and 2 are coded L3. No study is coded L2 or L2.5.

The predominance of L1 codes is tied to the evaluation datasets most commonly used. NSL-KDD and CICIDS, the two most frequent datasets across the study set (Ahmed et al., 2022; Ashraf et al., 2022; Chatterjee and Ahmed, 2022; Farooq et al., 2022; Khonde and Ulagamuthalvi, 2022; Shahraki et al., 2022), provide pre-extracted flow-level feature vectors in which protocol-typed fields have been reduced to statistical summaries: mean packet length, inter-arrival time, byte counts, and similar aggregates. A detection node trained on these vectors has no access to protocol field semantics regardless of architectural sophistication. A transformer operating on NSL-KDD flow vectors is L1; the attention mechanism operates over statistical features, not protocol states.

The two L3 studies differ from the remainder in both methodology and dataset. Mirdula and Roopa (2023) implement a Software-Defined Networking controller that classifies devices against Manufacturer Usage Description (MUD) profiles. The controller tracks the expected protocol FSM for each device type; deviations

from the expected state sequence trigger an alert, and the blockchain layer logs MUD profile assignments and FSM violation events for tamper-evident audit. Evaluation uses BoT-IoT with a custom SDN testbed and achieves 99% detection accuracy. Shahraki et al. (2022) model the expected communication graph of IoT devices at the protocol session level. Graph anomalies serve as FSM violation signals; evaluation uses CICIDS and reports accuracy above 99%. Neither study provides a direct comparison against an L1 baseline on the same dataset, so no performance differential attributable to the protocol-awareness level transition can be established from the available evidence.

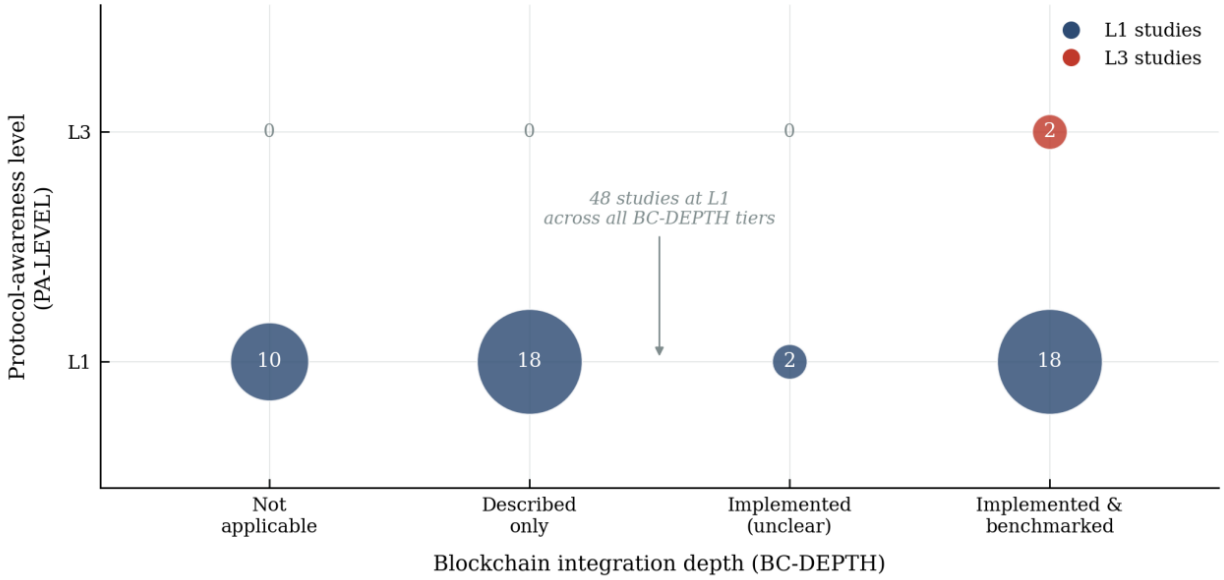

Figure 2: Joint distribution of protocol-awareness level (PA-LEVEL) and blockchain integration depth (BC-DEPTH) across the 50 included studies. Bubble area is proportional to study count. The L1 row dominates all BC-DEPTH tiers; the two L3 studies both reside in the Implemented and benchmarked tier.

PA-LEVEL coding was assigned from reported feature descriptions and could not always be verified against raw data or code. Where a study described features in general terms without listing field names, L1 was assigned as the conservative coding. The two L3 studies are unambiguous cases; the possibility of L2 or L2.5 studies miscoded as L1 due to incomplete feature reporting cannot be excluded.

3.3. Detection Performance

Performance metrics were extracted from each study's primary evaluation table or summary statement. Where a study reported multiple metrics, accuracy was preferred; where accuracy was not reported, F1-score was used. Cross-study comparison is methodologically warranted only for studies sharing a common evaluation dataset. Heterogeneous dataset use across the 50 included studies precludes aggregate comparison; accuracy on NSL-KDD, BoT-IoT, and ToN-IoT are not commensurate figures. Quantitative reporting is therefore restricted to the 8 Strong-rated studies, all of which used IoT-specific datasets or physical testbeds. Two of the eight, Babar et al. (2026) and Kolsur and

Hosmani (2025), are coded BC-DEPTH = Described only; their performance figures reflect the detection layer only and do not incorporate blockchain overhead.

Table 3: Detection performance for Strong-rated studies (n = 8).

Metrics are drawn from each study's primary evaluation result. Cross-row comparison is valid only within the same dataset. PA = protocol-awareness level. For "Described only" entries, reported figures cover the detection layer only.

Study	PA	Dataset	Architecture	Key metric	Value
Babar et al. (2026)	L1	ToN-IoT / IoT-HC	CNN, BiLSTM (FL)	Accuracy	99.95%
Kolsur & Hosmani (2025)	L1	ToN-IoT	CNN-LSTM	Accuracy	91.1%
Liu et al. (2022)	L1	BoT-IoT	BiLSTM, CNN	F1 (avg.)	>95%+
Mahendran et al. (2025)	L1	IIoT testbed	CasDL (custom)	Accuracy	98.5%
Mirdula & Roopa (2023)	L3	BoT-IoT (SDN)	RF + NDAE	Accuracy	99.0%
Odeh & Abu Taleb (2025)	L1	ToN-IoT	MLP (FL)	Accuracy / F1	92% / 0.91
Ullah et al. (2025)	L1	BoT-IoT	CNN, BiLSTM	Accuracy	99.84%
Villegas-Ch et al. (2025)	L1	UNSW-NB15	MLP, Autoencoder	F1 (avg.)	88.7–95.2%+

Accuracy figures for the BoT-IoT and ToN-IoT evaluations cluster above 99% (Liu et al., 2022; Mirdula and Roopa, 2023; Ullah et al., 2025; Babar et al., 2026). The UNSW-NB15 result from Villegas-Ch et al. (2025) and the ToN-IoT result from Kolsur and Hosmani (2025) return lower figures of 88–95%. Both datasets are less severely class-imbalanced than BoT-IoT and ToN-IoT in the partitions used by these studies. Whether the high-accuracy figures on BoT-IoT and ToN-IoT reflect genuine detection capability or are an artefact of class distribution cannot be determined from the reported results alone; none of the eight studies reports per-class precision

and recall at the attack subcategory level. The performance figures in Table 3 are therefore treated as dataset-specific benchmarks rather than deployment-ready accuracy estimates. The question of whether these benchmarks transfer to live IoT environments is examined in Section 5.

4. Blockchain Integration: Depth, Role, and Evaluation

4.1. Integration Depth across the Included Studies

BC-DEPTH was assigned by reading the evaluation section of each study and determining whether at least one quantitative measurement was reported for the blockchain component itself. Measurements counted toward *Implemented and benchmarked* included transaction throughput, consensus latency, block propagation time, storage overhead, and smart contract execution cost. Studies in which a blockchain component was described in the architecture but absent from the evaluation protocol were coded *Described only*. Studies where a blockchain component was implemented but evaluation metrics could not be unambiguously attributed to it were coded *Implemented with benchmarking unclear*.

Out of 50 included studies, 20 are coded *Implemented and benchmarked*, 18 *Described only*, 10 *Not applicable*, and 2 *Implemented with benchmarking unclear*.

Among the 18 *Described only* studies, 8 target IIoT or Industrial CPS environments (Ahakonye et al., 2025; Alruwaili et al., 2024; Dineshbabu and Vigenesh, 2025; Lilhore et al., 2026; Luo, Zeng, Ma and Wei, 2025; Rathee et al., 2022; Salim, Comivi, Nurbek, Park and Park, 2022; Vargas, Lozano-Garzon, Montoya and Donoso, 2021), 6 target general IoT (Alabdan et al., 2025; Alamro et al., 2025; Alkhammash, 2024; Kokila and Reddy, 2024; Kumar et al., 2021; Manzano, Zaman, Upadhyay, Goel and Sampalli, 2025), and 4 target healthcare IoT (Alatawi, 2025; Ashraf et al., 2022; Babar et al., 2026; Kolsur and Hosmani, 2025). IIoT studies account for the largest single share.

Two studies rated Strong on detection criteria, Babar et al. (2026) and Kolsur and Hosmani (2025), are coded BC-DEPTH = *Described only*. Both provide detection evaluation on IoT-specific datasets with ablation analysis; neither reports any measurement of its blockchain component. Their detection performance figures, reported in Table 3, therefore reflect the detection layer in isolation.

BC-DEPTH coding is conservative: a study was coded *Implemented and benchmarked* only when the blockchain measurement was unambiguous and clearly separated from detection metrics. Where reporting was ambiguous the lower tier was assigned. It is possible that some studies coded *Described only* conducted blockchain evaluation that was not reported in the primary publication; this cannot be verified from the available evidence.

4.2. Consensus Mechanisms and Platforms

BC-CONSENSUS was coded from the system design or implementation section of each study. Where a study named a specific consensus mechanism, that mechanism was recorded directly. Where a study described a blockchain platform that subsumes consensus (Hyperledger Fabric), the platform name was recorded as the consensus code. Where a study claimed a blockchain component but named neither a mechanism nor a platform, the code *Ethereum (consensus not specified)* was applied for the one study where this occurred (Kumar et al., 2021). Studies coded BC-DEPTH *Not applicable* received BC-CONSENSUS = *Not applicable*.

Out of the 40 studies with an active blockchain component, consensus mechanism distribution is as follows:

- **PoS** in 10 studies (Ahmed et al., 2022; Alabdan et al., 2025; Alamro et al., 2025; Alkhammash, 2024; Kokila and Reddy, 2024; Kolsur and Hosmani, 2025; Nguyen, Diro, Chilamkurti, Heyne and Phan, 2025; Rathee et al., 2022; Srinivasan and Senthilkumar, 2025; Vargas et al., 2021)
- **PoA** in 9 studies (Abdelhady, Ghandoura, Motwakel and Alajmi, 2026; Ahakonye et al., 2025; Ibrahim, Ahakonye, Lee and Kim, 2026b;

Mahendran et al., 2025; Mukisa, Ahakonye, Kim and Lee, 2025; Ngo et al., 2024; Selvarajan, Srivastava, Khadidos, Khadidos, Baza, Alshehri and Lin, 2023; Shukla, Thakur, Hussain, Breslin and Jameel, 2021; Ullah et al., 2025)

- **PBFT** in 9 studies (Babar et al., 2026; Lekssays, Landa, Carminati and Ferrari, 2021; Lilhore et al., 2026; Luo et al., 2025; Manzano et al., 2025; Odeh and Abu Taleb, 2025; Parashar et al., 2026; Salim et al., 2022; Sengan et al., 2026)
- **Hyperledger Fabric** in 7 studies (Alatawi, 2025; Alkhonaini, Alohali, Aljebreen, Eltahir, Alanazi, Yafoz, Alsini and Khadidos, 2025; Dineshbabu and Vigenesh, 2025; Ibrahim, Ahakonye, Lee and Kim, 2026a; Khonde and Ulagamuthalvi, 2022; Liu et al., 2022; Villegas-Ch et al., 2025)
- **PoW** in 4 studies (Alruwaili et al., 2024; Arazzi, Nicolazzo and Nocera, 2023; Ashraf et al., 2022; Farooq et al., 2022)

PoA and PBFT are permissioned, deterministic mechanisms that offer bounded finality: a transaction is confirmed within a fixed number of rounds with no probabilistic uncertainty. These properties are compatible with low-latency detection pipelines in which alert generation and blockchain logging must occur within a bounded window. The 18 studies using PoA or PBFT account for the largest share of mechanistically specified deployments. PoS introduces variable finality depending on validator stake distribution; the 10 PoS studies do not report finality time measurements. PoW imposes the highest computational overhead and longest expected confirmation times; its use in 4 studies targeting IIoT environments is not accompanied by latency justification relative to detection pipeline requirements in any of the four. Hyperledger Fabric operates under an ordering-service model distinct from Nakamoto-style consensus; the 7 studies using it report transaction throughput figures but do not benchmark end-to-end detection-to-logging latency.

Consensus choice was coded from reported descriptions and cannot be verified against deployed configurations or source code. No study in the set provides a justification for its consensus selection relative to the latency or throughput requirements of its detection pipeline, so the operational suitability of each choice cannot be assessed from the available evidence.

4.3. Blockchain Benchmarking Gaps

Among the 20 studies coded *Implemented and benchmarked*, the scope of blockchain evaluation varies considerably. Benchmarking metrics were extracted from each study's results section and classified into five categories: transaction throughput (TPS or equivalent), consensus latency, end-to-end detection-to-logging latency, fault tolerance or Byzantine resilience testing, and gas or transaction cost.

Transaction throughput is the most frequently reported metric. It appears in 12 of the 20 studies (Ibrahim et al., 2026a,b; Khonde and Ulagamuthalvi, 2022; Lekssays et al., 2021; Mahendran et al., 2025; Ngo et al., 2024; Odeh and Abu Taleb, 2025; Selvarajan et al., 2023; Sengan et al., 2026; Shukla et al., 2021; Ullah et al., 2025; Villegas-Ch et al., 2025). Consensus latency or block commit time is reported in 9 studies (Alkhonaini et al., 2025; Arazzi et al., 2023; Ibrahim et al., 2026b; Lekssays et al., 2021; Parashar et al., 2026; Sengan et al., 2026; Shukla et al., 2021; Ullah et al., 2025; Villegas-Ch et al., 2025).

End-to-end latency from detection event to blockchain log confirmation, the metric most directly relevant to operational deployment, is reported in only 4 studies (Ibrahim et al., 2026a; Mahendran et al., 2025; Parashar et al., 2026; Ullah et al., 2025). Byzantine fault tolerance or malicious node resilience is tested in 5 studies (Ibrahim et al., 2026b; Khonde and Ulagamuthalvi, 2022; Parashar et al., 2026; Shukla et al., 2021; Ullah et al., 2025). Gas or transaction cost is reported in 1 study (Ngo et al., 2024).

Three studies coded *Implemented and benchmarked* carry explicit extraction flags indicating that no blockchain performance metrics were reported in the primary publication:

- Ahmed et al. (2022) implement a distributed P2P ledger for inter-agent secure data transmission in a multi-agent IDS; no consensus algorithm is named and no TPS, latency, or block-size figures appear in the results.
- Farooq et al. (2022) describe a four-layer private blockchain architecture using smart contracts for access control in a smart home network; the blockchain is architecturally specified but evaluated only through detection accuracy on the IDS component.
- Liu et al. (2022) combine Hyperledger Fabric smart contracts for transaction integrity with an autoencoder-MLP detection model in a smart transport system; blockchain performance, including TPS and latency, is explicitly excluded from the evaluation scope.

All three differ in deployment domain (general IoT, smart home, and healthcare IoT respectively) and in the role assigned to the blockchain component, but share the same evaluative gap: the blockchain layer is confirmed from the study text but not measured. Their BC-DEPTH coding therefore reflects architectural implementation rather than empirical evaluation.

Node counts used in blockchain benchmarking are consistently small. Where node counts are reported, they range from 3 to 80 nodes. No study evaluates its blockchain component at a node count representative of a large-scale IoT deployment. Sengan et al. (2026) test a 12-node PBFT network and report peak throughput of 698 TPS with consensus latency of 234–467 ms; Ullah et al. (2025) test a multi-tier sharded architecture and report detection-to-logging end-to-end latency but do not specify a node count for the full network tier.

Reported throughput and latency figures are not comparable across studies. Node counts, hardware configurations, network topologies, and dataset sizes differ in every case, and no shared benchmarking protocol or baseline exists across the set. The absence of a common evaluation substrate for blockchain performance means that the 20 benchmarked studies establish, in aggregate, that blockchain components can be implemented and measured, but do not establish what performance levels are achievable or required for operational IoT-IDS deployment.

5. Evaluation Substrate and Benchmark Reliability

5.1. Dataset Distribution

The primary evaluation dataset for each study was identified from the evaluation section or experimental setup. Where a study used multiple datasets, all were recorded; dataset counts therefore sum to more than 50. Datasets were classified as **IoT-specific** where their documented collection methodology drew exclusively from IoT or IIoT device traffic, and as **generic** where the capture environment included general-purpose enterprise or laboratory network traffic. UNSW-NB15 occupies a borderline position in this classification because it was generated in a controlled laboratory environment simulating both enterprise and IoT-adjacent traffic; it is treated here as generic.

BoT-IoT is the most frequently used dataset, appearing in 16 studies. CICIDS and its variants (CICIDS2017, CICDDoS2019, CICIoT2023) appear in 14 studies. NSL-KDD appears in 10 studies. ToN-IoT appears in 9 studies. UNSW-NB15 appears in 8 studies. Edge-IIoTset appears in 7 studies. N-BaIoT appears in 3 studies. KDD Cup 99 appears in 4 studies. Physical or custom testbeds are used in 4 studies, and one study uses a proprietary dataset.

Across the 50 included studies, 80 distinct dataset uses were recorded (counting multiple datasets per study separately). IoT-specific datasets (BoT-IoT, ToN-IoT, Edge-IIoTset, N-BaIoT) account for 35 of these uses; generic datasets (NSL-KDD, CICIDS variants, UNSW-NB15, KDD Cup 99) account for 36. Physical or custom testbeds contribute 4 uses, and one proprietary dataset contributes 1; the remaining 4 uses correspond to studies whose dataset records were not matched to any classified category during extraction.

The IoT-specific and generic categories appear in roughly equal proportions across the study set, but the distribution is uneven by domain: studies targeting healthcare IoT draw more frequently on

generic datasets, while studies targeting IIoT environments show higher use of BoT-IoT and Edge-IIoTset.

5.2. IoT Representativeness of Primary Datasets

IoT representativeness was assessed against three criteria applied to each dataset's documented collection methodology, not to the individual study under review:

- **Criterion R1:** Required that traffic be generated exclusively from IoT or IIoT devices.
- **Criterion R2:** Required that the dataset include protocols representative of IoT communication stacks: MQTT, CoAP, Zigbee, LoRaWAN, Modbus, or equivalent.
- **Criterion R3:** Required that the attack types represented include threats specific to the IoT threat model: firmware exploitation, protocol abuse, or device impersonation.

BoT-IoT, ToN-IoT, and Edge-IIoTset meet all three criteria. N-BaIoT meets R1 and R3 but not R2: it captures botnet traffic from IoT devices at the TCP/IP layer without IoT-specific protocol representation. NSL-KDD and KDD Cup 99 meet none of the three criteria; both were generated from a 1998 DARPA simulation of general enterprise network traffic and contain no IoT device participation. CICIDS and its variants meet R3 partially, as some attack categories overlap with IoT-relevant threats, but traffic was generated from general-purpose hosts and criterion R1 is not met. UNSW-NB15 meets R3 partially on the same basis.

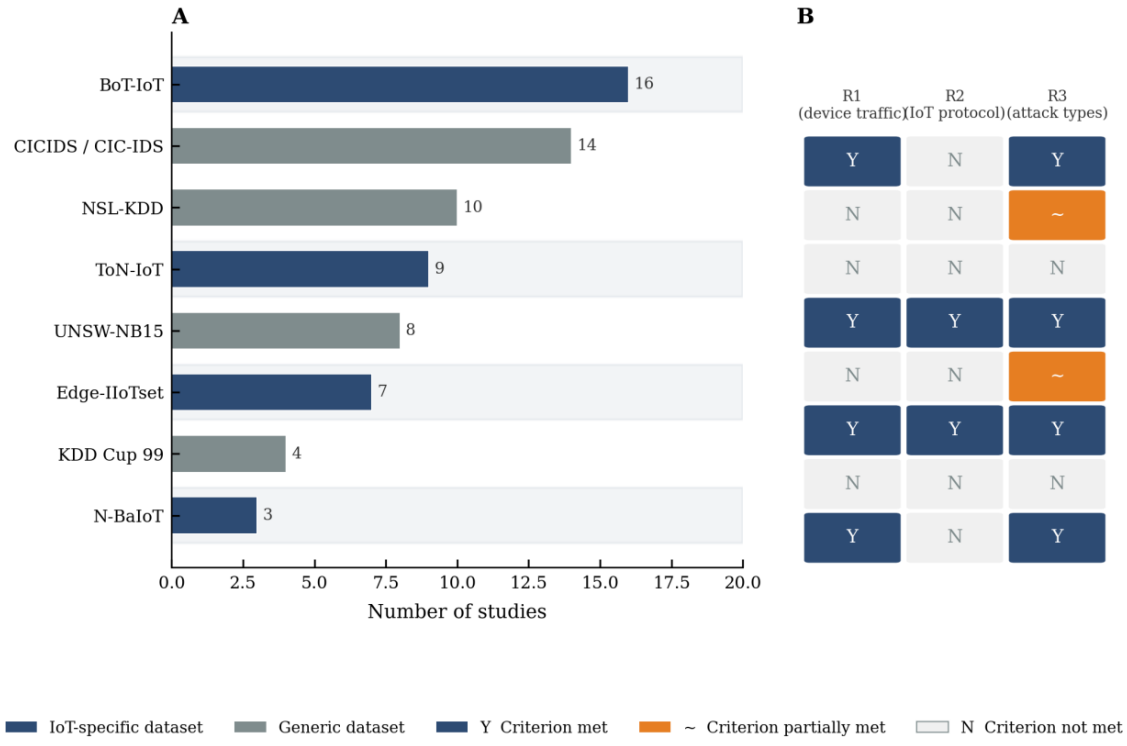

Figure 3: Dataset usage and IoT representativeness across the 50 included studies. Panel A shows the number of studies using each dataset (navy = IoT-specific; grey = generic). Panel B shows representativeness against three criteria: R1 (IoT device tra-c only), R2 (IoT protocol representation), and R3 (IoT-

specific attack types). Y = criterion met; ~ = partially met; N = not met.

Of the 50 included studies, 8 use exclusively IoT-specific datasets; 16 use a mixture of IoT-specific and generic datasets; 11 use exclusively generic datasets; and 2 use only custom or physical testbeds. The remaining 13 studies use datasets that were not matched to any of the above categories in the extraction records.

Detection performance reported by the 11 exclusively generic-dataset studies cannot be attributed to IoT threat detection capability under the representativeness criteria applied here. NSL-KDD, the most common among them, contains no IoT device traffic, no IoT protocol representation, and no IoT-specific attack types. A detection accuracy of 99% on NSL-KDD establishes only that the model can classify the pre-extracted flow features in that benchmark; it does not establish whether the same model would detect threats in a live IoT deployment. This limitation applies equally to studies that use NSL-KDD as one of several datasets alongside an IoT-specific benchmark, where the two accuracy figures are not commensurate and should not be aggregated.

5.3. Federated Learning and Distributed Detection

FL use was coded from each study's architecture description. A study was coded *FL = Yes* only where a federated training protocol was explicitly implemented; studies proposing FL as future work or describing a simulated aggregation step without a genuine multi-party training loop were coded *FL = No*.

Of the 50 included studies, 18 implement FL. IoT and Industrial CPS deployments account for 11 of the 18; healthcare IoT accounts for 5. On consensus mechanism, PoA is the most common among FL studies (5 studies), followed by Hyperledger Fabric (4) and PBFT (3). Four FL studies are coded BC-DEPTH = *Not applicable*, indicating that their federated architecture does not include a blockchain component. FL and blockchain integration are therefore not universal co-occurrences in this study set.

Of the 18 studies implementing FL, 17 are coded L1 on PA-LEVEL. The single exception is Shahraki et al. (2022), which models protocol session graphs in a federated setting and is coded L3. FL operates at the model aggregation layer: it governs how locally trained parameters are combined across distributed nodes, not which features the local detection node inspects. A federated CNN trained on flow-level statistics at each edge node aggregates L1 representations, not protocol-typed features. FL does not alter the PA-LEVEL of the detection node and does not address the protocol-awareness limitation identified in Section 3.2.

FL coding was binary and does not capture variation in federation fidelity. Studies differ in the number of participating nodes (ranging from 3 to over 100 in reported configurations), the aggregation algorithm used (FedAvg, FedProx, or custom variants), and whether non-IID data distribution across nodes was modelled. These differences affect the reliability of reported FL accuracy figures but cannot be assessed systematically from the primary publications.

6. Discussion and Conclusion

6.1. Overview of the Analysis

This analysis set out to characterise how deep learning-based intrusion detection and blockchain trust mechanisms are combined in IoT deployments, and to assess whether the resulting systems are evaluated in ways that support the claims made for them. The 50 studies examined span five years of publication, four deployment domains, and the full range of consensus mechanisms currently proposed for IoT security architectures, and three structural features are consistent across all 50 studies.

Detection nodes operate almost exclusively on flow-level statistical features, without reference to the protocol structure of the traffic they monitor. Blockchain components are more frequently described than measured, and where measured, the metrics selected rarely address the operational requirements of a detection pipeline. Among the 50 included studies, none examines whether the detection depth and the trust scope of the blockchain layer are architecturally aligned.

These three features are not independent: they are connected by a shared assumption that

detection accuracy and trust integration can be evaluated separately, and that benchmark performance on available datasets is an adequate proxy for deployment validity. The discussion that follows argues that this assumption is incorrect and that correcting it requires a different research design from the one that dominates the current literature.

6.2. The Detection-Evaluation Disconnect

The choice of evaluation dataset is not a neutral methodological decision. It determines which threat classes can be detected, which protocol environments are represented, and whether the resulting accuracy figures have any bearing on the deployment context the study claims to address. The findings in Sections 5.1 and 5.2 show that a substantial portion of the literature evaluates IoT intrusion detection systems on traffic that contains no IoT devices, no IoT protocols, and no IoT-specific attack types. NSL-KDD appears in ten studies and was generated from a 1998 simulation of enterprise network activity; it predates the IoT threat environment by over a decade. A detection system trained and evaluated exclusively on NSL-KDD has not been tested against the threat categories it is deployed to detect.

The three gap clusters identified in this analysis are classified using the seven-type gap taxonomy of Miles (2017), which was developed to characterise knowledge gaps in systematic reviews across applied research domains. The taxonomy distinguishes gaps by their epistemic character: whether knowledge is absent because the right measurement has not been applied (Methodological Gap), because an existing instrument has not been used (Empirical Gap), because the relevant conceptual framework does not yet exist (Knowledge Gap), or because the problem has not been recognised as a research target (Research Agenda Gap), among others. The three types applied here are used as defined in the original taxonomy without modification.

Under the taxonomy of Miles (2017), the first cluster is a **Methodological Gap**. The field has not varied its evaluation instrument class as an experimental variable. Generic and IoT-specific datasets are used side by side across the study set, but among the 50 included studies none tests whether accuracy figures differ by substrate class for the same model on the same threat categories. The structural reason is tractable: generic datasets carry established baselines, are freely available, and enable comparison with prior work. These are legitimate considerations, but they are considerations of research economy, not research validity. They explain why the pattern exists; they do not justify it.

The implication extends beyond the detection layer. A blockchain trust mechanism appended to a detector that has not been validated on IoT traffic inherits that validation gap. The trust layer may be correctly implemented and correctly measured, yet it secures the outputs of a detection node whose IoT-environment performance is unknown. Resolving this gap requires evaluation on datasets that meet the three representativeness criteria established in Section 5.2: IoT device traffic, IoT protocol representation, and IoT-specific attack types. BoT-IoT, ToN-IoT, and Edge-IIoTset meet all three; eight studies in the set use one of them exclusively, demonstrating that this is feasible within existing research infrastructure.

6.3. The Trust-Assertion Problem

A security property that has not been measured is a design intention. The findings in Sections 4.1 and 4.3 show that across 18 studies, blockchain integration is described at the architecture level but absent from the evaluation protocol. The properties attributed to these blockchain layers—tamper evidence, decentralised verification, and Byzantine fault tolerance—appear in the system descriptions of each study. They do not appear in the results. This is not a matter of incomplete reporting; it is a matter of incomplete research design.

Under the taxonomy of Miles (2017), this is an **Empirical Gap**: propositions requiring empirical verification have not been verified. The distinction between a Methodological Gap and an Empirical Gap is precise here. Cluster A is a problem of instrument choice: the right type of measurement is missing. Cluster B is a problem of measurement omission: the instrument exists but was not applied. Four

studies in the set report end-to-end detection-to-logging latency (Ibrahim et al., 2026a; Mahendran et al., 2025; Parashar et al., 2026; Ullah et al., 2025); this establishes that measuring blockchain behaviour in an operational detection context is technically feasible. The gap is therefore not a consequence of measurement difficulty; it is a consequence of treating blockchain integration as an architectural given rather than an experimental variable.

The IIoT concentration sharpens the stakes. Eight of the 18 *Described only* studies target IIoT or Industrial CPS environments, where trust failure has physical consequences. A blockchain layer in an industrial control system that has not been tested for consensus latency under load, Byzantine resilience, or network partition behaviour is not a trust mechanism in any operational sense; it is an architectural claim. The studies in this group assert distributed trust in environments where the cost of trust failure is highest, on the basis of evidence that would not support the same claim in a less critical domain.

Resolving Cluster B requires treating the blockchain layer as an experimental variable with its own evaluation protocol, separate from and additional to the detection evaluation. The four studies reporting end-to-end latency provide a template: each measures both the detection accuracy and the time from detection event to confirmed blockchain log entry. Extending this pattern to Byzantine fault injection, node-count scaling, and network partition testing would produce a blockchain evaluation substrate comparable in rigour to what detection evaluation already achieves on its own layer.

6.4. The Co-Design Deficit

Clusters A and B identify problems that can, in principle, be resolved independently: one through dataset substitution, one through evaluation protocol extension. Cluster C identifies a problem that cannot be resolved by either fix alone. Among the 50 included studies, none asks whether the depth at which the detection node operates and the scope of the trust mechanism provided by the blockchain layer are architecturally aligned. The two components are designed and evaluated on separate axes: detection accuracy on one, blockchain properties on the other. The question of whether they are mutually consistent is not posed.

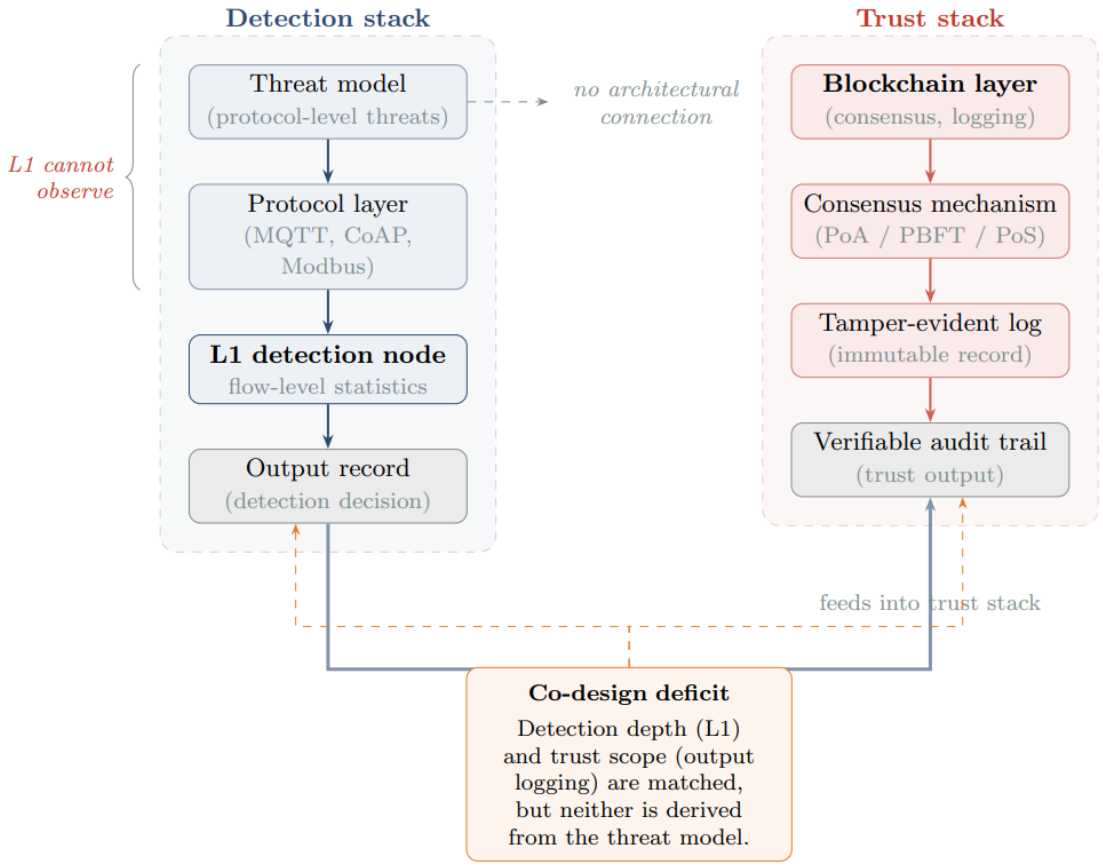

Figure 4: The detection-trust misalignment in blockchain-integrated IoT IDS. The detection stack (left) operates at L1 and cannot observe the protocol layer or the threat model. The trust stack (right) receives only the L1 output record. The two stacks are architecturally connected at a single point (output record → verifiable audit trail) rather than co-designed from the threat model downward. The dashed arrow between the threat model and the blockchain layer marks the absent connection that co-design would require.

The misalignment is structural: forty-eight of the 50 included studies employ detection nodes operating at L1, where the detection unit classifies traffic on flow-level statistical features with no reference to protocol state or protocol field semantics. The blockchain layer in these studies secures the output records of the L1 node; this produces tamper-evident logging of detection decisions. The function is well-defined, but it is not the same function as securing the detection process against protocol-level threats.

One alternative interpretation deserves acknowledgement: tamper-evident logging of L1 detection decisions may be sufficient for post-hoc forensic audit, where the goal is attribution and evidence preservation after a breach rather than real-time detection. For general enterprise environments this interpretation is defensible. For the IIoT and healthcare IoT deployments that account for the majority of the study set, however, post-hoc audit is insufficient: in industrial control systems and clinical monitoring environments, the consequence of an undetected protocol-level threat is not recoverable through forensic evidence alone. The argument for co-design is therefore strongest precisely in the deployment domains that dominate the reviewed literature.

A threat actor who exploits an MQTT topic injection, a CoAP amplification vector, or a Modbus function code abuse operates at the protocol semantic level. An L1 detection node cannot observe this class of threat, regardless of the trust properties of the blockchain layer

appended to it. The blockchain secures what the detector outputs; it does not extend what the detector can see.

This is a **Knowledge Gap** in the taxonomy of Miles (2017): the knowledge of what architecturally aligned detection-trust co-design looks like does not exist in the reviewed literature. The two L3 studies in the set, Mirdula and Roopa (2023) and Shahraki et al. (2022), are the closest approximation: both operate detection at the protocol session or FSM level and both include a blockchain component. Neither, however, examines whether the trust scope of the blockchain layer corresponds to the threat class that the L3 detection node is designed to surface. The co-design question is not answered by either study; it is not posed by either study.

The **Practical-Knowledge** dimension of this gap is equally precise. The dominant engineering practice across the study set is to select a DL architecture, train it on an available benchmark, and append a blockchain layer for trust. This sequence treats trust as an add-on rather than a design constraint. A co-design approach would begin from the threat model: what protocol-level threats must be detected, at what PA-LEVEL must the detection node operate to observe them, and what blockchain trust scope is required to make the resulting detection decisions credible to the receiving system? The study set contains no instance of this design sequence.

6.5. Dependency Structure and Research Agenda

The three gap clusters are not independent research problems. Cluster A constrains the validity of any detection performance claim made without IoT-representative evaluation. Cluster B constrains the validity of any trust claim made without blockchain evaluation. Cluster C identifies the absence of a design question that neither Cluster A resolution nor Cluster B resolution would, by itself, answer. The dependency runs in one direction: Cluster A must be resolved before detection performance on IoT substrates can be established; Cluster B must be resolved before trust properties can be asserted with empirical support; and Cluster C requires both resolved before the co-design question can even be posed in testable form. Addressing Cluster C without first addressing Clusters A and B would produce a co-designed system whose detection validity and trust properties are both unverified.

Three concrete research directions follow from this dependency structure:

1. **Addressing Cluster A:** Future studies evaluating blockchain-integrated IoT IDS should adopt at minimum one dataset meeting all three representativeness criteria established in Section 5.2: IoT device traffic, IoT protocol representation, and IoT-specific attack types. Where multiple datasets are used, accuracy figures should be reported disaggregated by dataset class rather than averaged. This change requires no new infrastructure; BoT-IoT, ToN-IoT, and Edge-IIoTset are all publicly available and have been used in studies already within the set.
2. **Addressing Cluster B:** Blockchain integration should be treated as an experimental variable with its own evaluation protocol. At minimum, this protocol should report end-to-end detection-to-logging latency, consensus latency under the node count representative of the target deployment, and Byzantine fault tolerance under at least one adversarial node configuration. The four studies that already report end-to-end latency provide a template. The node-count and fault tolerance components require a modest extension of existing testbed setups rather than new hardware.
3. **Addressing Cluster C (Most Demanding):** A co-design study should begin from an explicit threat model that specifies which protocol-level threats the system must detect, derive the minimum PA-LEVEL required to observe those threats, design a detection node operating at that level, and then derive the blockchain trust scope required to make the detection decisions credible to the receiving system. The resulting system would be the first in the reviewed literature to treat detection depth and trust scope as jointly constrained design variables rather than independent

component choices. Mirdula and Roopa (2023) and Shahraki et al. (2022) provide the closest existing foundation: both operate at L3 and both include a blockchain component. Neither examines the alignment between detection depth and trust scope, but both demonstrate that L3 detection with blockchain integration is implementable. A co-design extension of either study would constitute a direct response to Cluster C.

6.6. Conclusion

This analysis examined 50 blockchain-integrated deep learning IDS studies across IoT, IIoT, and healthcare IoT deployment domains, coding each against seven dimensions covering detection protocol-awareness, blockchain integration depth, consensus mechanism, federated learning use, deployment domain, evaluation dataset, and study quality. The coding instrument applied the L1–L3 protocol-awareness taxonomy developed for this analysis and the seven-type gap taxonomy of Miles (2017).

Three findings stand out:

1. Detection nodes operate at L1 in 48 of 50 studies: flow-level statistical features are the universal input representation, irrespective of deployment domain or architectural complexity.
2. Blockchain integration is described in 18 studies without evaluation and benchmarked in 20, but end-to-end detection-to-logging latency, the operationally critical metric, is reported in only 4.
3. Among the 50 included studies, none co-designs the detection depth and the blockchain trust scope as jointly constrained variables.

These findings map onto three gap types under Miles (2017). The evaluation substrate mismatch is a *Methodological Gap*: the field has not varied dataset class as an experimental variable. The blockchain trust-assertion problem is an *Empirical Gap*: trust properties have been asserted without empirical verification. The co-design deficit is a *Knowledge Gap* with a *Practical-Knowledge* dimension: the knowledge of what architecturally aligned detection-trust co-design requires does not exist in the reviewed literature, and the dominant engineering practice of appending blockchain to an existing L1 detector has proceeded without a theoretical account of whether the trust scope matches the threat model.

The three research directions proposed in Section 6.5 address these gaps in dependency order. Dataset substitution addresses Cluster A and requires no new infrastructure. Blockchain evaluation protocol extension addresses Cluster B and is demonstrated as feasible by four studies already in the set. Co-design from a threat-model-first sequence addresses Cluster C and requires both prior resolutions as a foundation. The field has the datasets, the architectures, and the blockchain platforms to make progress on all three; what is absent is the research design that treats detection depth and trust scope as a jointly optimised system rather than two independently selected components.

Declaration of Competing Interests

The author declares that there are no competing interests.

Data Availability

No new data were generated in this study. All studies included in the analysis are publicly available peer-reviewed publications; the coding dataset and extraction records are available from the corresponding author on reasonable request.

References

- Abdelhady, G., Ghandoura, A., Motwakel, A., Alajmi, A., 2026. Hybrid machine learning anomaly detection and lightweight zero trust authentication for LoRaWAN networks. IEEE Access 14, 18387-18407. doi:10.1109/ACCESS.2026.3660731.
- Ahad, A., Ali, Z., Mateen, A., Tahir, M., Hannan, A., Garcia, N.M., Pires, I.M., 2023. A comprehensive review on 5G-based smart healthcare network security: Taxonomy,

issues, solutions and future research directions. Array 18, 100290. doi:10.1016/j.array.2023.100290.

- Ahakonye, L.A.C., Nwakanma, C.I., Lee, J.M., Kim, D.S., 2025. Pure Chain-enhanced federated learning for dynamic fault tolerance and attack detection in distributed systems. High-Confidence Computing 6, 100354. doi:10.1016/j.hcc.2025.100354.
- Ahmed, M.A., Althubiti, S.A., Nageswara Rao, D., Lydia, E.L., Cho, W., Joshi, G.P., Kim, S.W., 2022. Blockchain assisted intrusion detection system using differential flower pollination model. Computers, Materials and Continua. doi:10.32604/cmc.2022.
- Alabdan, R., Alabduallah, B., Alruwais, N., Arasi, M.A., Asklany, S.A., Alghushairy, O., Alallah, F.S., Alshareef, A., 2025. Blockchain-assisted improved interval type-2 fuzzy deep learning-based attack detection on internet of things driven consumer electronics. Alexandria Engineering Journal 110, 153-. doi:10.1016/j.aej.2024.09.117.
- Alamro, H., Maray, M., Aljabri, J., Alahmari, S., Abdullah, M., Alqurni, J.S., Alotaibi, F.A., Mohamed, A.A., 2025. Mathematical modelling-based blockchain with attention deep learning model for cybersecurity in IoT-consumer electronics. Alexandria Engineering Journal 113, 366-. doi:10.1016/j.aej.2024.11.016.
- Alatawi, M.N., 2025. EdgeGuard-IoT: 6G-enabled edge intelligence for secure federated learning and adaptive anomaly detection in Industry 5.0. Computers, Materials and Continua. doi:10.32604/cmc.2025.066606, online first.
- Alkhammash, M., 2024. A metaheuristic approach to detecting and mitigating DDoS attacks in blockchain-integrated deep learning models for IoT applications. IEEE Access 13, 193184-. doi:10.1109/ACCESS.2024.3519132.
- Alkhonaini, M.A., Alohali, M.A., Aljebreen, M., Eltahir, M.M., Alanazi, M.H., Yafoz, A., Alsini, R., Khadidos, A.O., 2025. Sandpiper optimization with hybrid deep learning model for blockchain-assisted intrusion detection in IoT environment. Alexandria Engineering Journal 112, 49-61. doi:10.1016/j.aej.2024.10.032.
- Alruwaili, F.F., Alohali, M.A., Aljaffan, N., Alhashmi, A.A., Mahmud, A., Assiri, M., 2024. A decentralized approach to smart home security: blockchain with red-tailed hawk-enabled deep learning. IEEE Access 12, 14146. doi:10.1109/ACCESS.2024.3352502.
- Aouini, Z., Pekar, A., 2022. NFStream: A flexible network data analysis framework. Computer Networks 204, 108719. doi:10.1016/j.comnet.2021.108719.
- Arazzi, M., Nicolazzo, S., Nocera, A., 2023. A fully privacy-preserving solution for anomaly detection in IoT using federated learning and homomorphic encryption. Information Systems Frontiers. doi:10.1007/s10796-023-10443-0, online first.
- Ashraf, E., Areed, N.F.F., Salem, H., Abdelhay, E.H., Farouk, A., 2022. FIDChain: Federated intrusion detection system for blockchain-enabled IoT healthcare applications. Healthcare 10, 1110. doi:10.3390/healthcare10061110.
- Babar, M., Khan, Z., Kaleem, S., Qureshi, B., Boulila, W., 2026. TAWB-SFL: trust-aware blockchain-orchestrated split federated learning for intrusion detection in 6G healthcare IoT. IEEE Open Journal of the Communications Society. doi:10.1109/OJCOMS.2026.3667693, early access.
- Bizzarri, A., Yu, C.E., Jalaian, B., Riguzzi, F., Bastian, N.D., 2024. A synergistic approach in network intrusion detection by neurosymbolic AI. doi:10.48550/arXiv.2406.00938, arXiv:2406.00938.
- Chatterjee, A., Ahmed, B.S., 2022. IoT anomaly detection methods and applications: A survey. Internet of Things 19, 100568. doi:10.1016/j.iot.2022.100568.
- Chinnasamy, R., Subramanian, M., Veerappampalayam Easwaramoorthy, S.,

Cho, J., 2025. Deep learning-driven methods for network-based intrusion detection systems: A systematic review. ICT Express 11, 181-199. doi:10.1016/j.icte.2025.01.005.

- Dineshbabu, V., Vigenesh, M., 2025. Secure FLACF: secure federated learning access control framework with blockchain-infused intrusion detection system for IIoT. Journal of Mobile Multimedia 21, 939-966. doi:10.13052/jmm1550-4646.2155.
- Farooq, M.S., Khan, S., Rehman, A., Abbas, S., Khan, M.A., Hwang, S.O., 2022. Blockchain-based smart home networks security empowered with fused machine learning. Sensors 22, 4522. doi:10.3390/s22124522.
- Ferrag, M.A., Maglaras, L., Moschoyiannis, S., Janicke, H., 2020. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. Journal of Information Security and Applications 50, 102419. doi:10.1016/j.jisa.2019.102419.
- Friha, O., Ferrag, M.A., Shu, L., Maglaras, L., Choo, K.K.R., Nafaa, M., 2022. FELIDS: Federated learning-based intrusion detection system for agricultural Internet of Things. Journal of Parallel and Distributed Computing 165, 17-31. doi:10.1016/j.jpdc.2022.03.003.
- Hasan, M.M., Islam, R., Mamun, Q., Islam, M.Z., Gao, J., 2026. Adversarial attacks against network intrusion detection systems: Bridging the gap between theoretical vulnerabilities and practical constraints. Internet of Things 31, 101997. doi:10.1016/j.iot.2026.101997.
- He, Y., Chen, Z., Wu, B., Sood, K., Xiao, K., Yan, Y., Sun, L., 2026. Resource-based online orchestration for multi-domain collaborative analysis of network encrypted traffic. Cybersecurity 9, 9. doi:10.1186/s42400-026-00580-9.
- Ibrahim, H., Ahakonye, L.A.C., Lee, J.M., Kim, D.S., 2026a. PureChain closed loop intrusion detection and real-time recovery for industrial IoT. IEEE Internet of Things Journal. doi:10.1109/JIOT.2026.3672474, early access.
- Ibrahim, H., Ahakonye, L.A.C., Lee, J.M., Kim, D.S., 2026b. A unified AI-PureChain framework for verifiable intrusion prevention in industrial IoT systems. IEEE Internet of Things Journal 13, 12906-12919. doi:10.1109/JIOT.2026.3652250.
- Javed, A.R., Hassan, M.A., Shahzad, F., Ahmed, W., Singh, S., Baker, T., Gadekallu, T.R., 2022. Integration of blockchain technology and federated learning in vehicular (IoT) networks: A comprehensive survey. Sensors 22, 4394. doi:10.3390/s22124394.
- Khonde, S.R., Ulagamuthalvi, V., 2022. Hybrid intrusion detection system using blockchain framework. EURASIP Journal on Wireless Communications and Networking 2022, 97. doi:10.1186/s13638-022-02089-4.
- Kokila, M., Reddy, K.S., 2024. BlockDLO: blockchain computing with deep learning orchestration for secure data communication in IoT environment. IEEE Access 12, 134521. doi:10.1109/ACCESS.2024.3462735.
- Kolsur, S., Hosmani, S., 2025. Artificial intelligence-based healthcare cybersecurity system with blockchain: modified parallel convolutional neural network for attack detection. Medicine in Novel Technology and Devices 28, 100412. doi:10.1016/j.medntd.2025.100412.
- Kumar, P., Kumar, R., Gupta, G.P., Tripathi, R., 2021. A distributed framework for detecting DDoS attacks in smart contract-based Blockchain-IoT systems by leveraging fog computing. Transactions on Emerging Telecommunications Technologies 32, e4112. doi:10.1002/ett.4112.
- Lekssays, A., Landa, L., Carminati, B., Ferrari, E., 2021. PAutoBotCatcher: A blockchain-based privacy-preserving botnet detector for Internet of Things. Computer Networks 200, 108512. doi:10.1016/j.comnet.2021.108512.
- Liao, H., Murah, M.Z., Hasan, M.K., Aman, A.H.M., Fang, J., Hu, X., Khan, A.U.R., 2024. A survey of deep learning technologies for intrusion detection in Internet of Things.

IEEE Access 12, 4745-4761. doi:10.1109/ACCESS.2023.3349287.

- Lilhore, U.K., Sharma, Y.K., Pradeep, S., Rubaiee, S., Alroobaea, R., Baqasah, A.M., Alsafyani, M., Tekeste, L.G., 2026. Blockchain-enabled federated learning framework with hybrid CNN-LSTM anomaly detection for secure edge IoT networks. Journal of Cloud Computing. doi:10.1186/s13677-026-00907-4, article in press.
- Liu, T., Sabrina, F., Jang-Jaccard, J., Xu, W., Wei, Y., 2022. Artificial intelligence-enabled DDoS detection for blockchain-based smart transport systems. Sensors 22, 32. doi:10.3390/s22010032.
- Liu, Y., Wang, J., Yan, Z., Wan, Z., Jäntti, R., 2023. A survey on blockchain-based trust management for Internet of Things. IEEE Internet of Things Journal 10, 5898-5922. doi:10.1109/JIOT.2023.3237893.
- Luo, S., Zeng, P., Ma, C., Wei, Y., 2025. Anomaly detection for industrial internet of things devices based on self-adaptive blockchain sharding and federated learning. Journal of Communications and Networks 27, 92-102. doi:10.23919/JCN.2025.000019.
- Mahendran, R.K., Khan, A., Ullah, F., Ali, F., AlZubi, A.A., 2025. PRISM-IIoT: a holistic approach for privacy preservation in industrial IoT using advanced cryptography and blockchain-enabled auditability framework. Alexandria Engineering Journal 128, 816. doi:10.1016/j.aej.2025.07.027.
- Manzano, R., Zaman, M., Upadhyay, D., Goel, N., Sampalli, S., 2025. Federated learning framework based on distributed storage and diffusion model for intrusion detection on IoT networks. IEEE Access 13, 79571-. doi:10.1109/ACCESS.2025.3565409.
- Meneghello, F., Calore, M., Zucchetto, D., Polese, M., Zanella, A., 2019. IoT: Internet of Threats? A survey of practical security vulnerabilities in real IoT devices. IEEE Internet of Things Journal 6, 8182-8201. doi:10.1109/JIOT.2019.2935189.
- Miles, D.A., 2017. A taxonomy of research gaps: Identifying and defining the seven research gaps, in: Doctoral Student Workshop: Finding Research Gaps - Research Methods and Strategies, Dallas, Texas.
- Mirdula, S., Roopa, M., 2023. MUD enabled deep learning framework for anomaly detection in IoT integrated smart building. e-Prime - Advances in Electrical Engineering, Electronics and Energy 5, 100186. doi:10.1016/j.prime.2023.100186.
- Mukisa, K.J., Ahakonye, L.A.C., Kim, D.S., Lee, J.M., 2025. Blockchain-augmented FL IDS for non-IID edge-IoT data using adaptive trimmed mean aggregation. IEEE Internet of Things Journal 12, 45150-45159. doi:10.1109/JIOT.2025.3598832.
- Ngo, D.M., Lightbody, D., Temko, A., Murphy, C.C., Popovici, E., 2024. A scalable security approach in IoT networks: smart contracts and anomaly-based IDS for gateways using hardware accelerators. IEEE Access 12, 159519-. doi:10.1109/ACCESS.2024.3486605.
- Nguyen, V.D., Diro, A., Chilamkurti, N., Heyne, W., Phan, K.T., 2025. A novel blockchain-enabled federated learning scheme for IoT anomaly detection. IEEE Transactions on Machine Learning in Communications and Networking. doi:10.1109/TMLCN.2025.3585842, early access.
- Odeh, A., Abu Taleb, A., 2025. Federated learning and blockchain framework for scalable and secure IoT access control. Computers, Materials and Continua. doi:10.32604/cmc.2025.065426, online first.
- Parashar, J., Hung, B.T., Upreti, K., Kshirsagar, P.R., Mahajan, S., Kadry, S., 2026. An enhanced hybrid framework for IoT healthcare security using blockchain-driven multimedia data analysis and cybersecurity techniques. Array 30, 100759. doi:10.1016/j.array.2026.100759.
- Rathee, G., Kerrache, C.A., Ferrag, M.A., 2022. A blockchain-based intrusion detection system using Viterbi algorithm and

indirect trust for IIoT systems. Journal of Sensor and Actuator Networks 11, 71. doi:10.3390/jsan11040071.

- Salim, M.M., Comivi, A.K., Nurbek, T., Park, H., Park, J.H., 2022. A blockchain-enabled secure digital twin framework for early botnet detection in IIoT environment. Sensors 22, 6133. doi:10.3390/s22166133.
- Selvarajan, S., Srivastava, G., Khadidos, A.O., Khadidos, A.O., Baza, M., Alshehri, A., Lin, J.C.W., 2023. An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems. Journal of Cloud Computing: Advances, Systems and Applications 12, 38. doi:10.1186/s13677-023-00412-y.
- Sengan, S., Shieh, C.S., Horng, M.F., 2026. A hybrid blockchain based deep learning model for multivector attack detection in internet of things enabled healthcare systems. Scientific Reports 16, 10060. doi:10.1038/s41598-026-40765-3.
- Shahraki, A., Abbasi, M., Taherkordi, A., Jurcut, A.D., 2022. Active learning for network traffic classification: A technical study. IEEE Transactions on Cognitive Communications and Networking 8, 422-439. doi:10.1109/TCCN.2021.3119062.
- Shukla, S., Thakur, S., Hussain, S., Breslin, J.G., Jameel, S.M., 2021. Identification and authentication in healthcare Internet-of-Things using integrated fog computing based blockchain model. Internet of Things 15, 100422. doi:10.1016/j.iot.2021.100422.
- Srinivasan, M., Senthilkumar, N.C., 2025. Intrusion detection and prevention system (IDPS) model for IoT environments using hybridized framework. IEEE Access 13, 26608-. doi:10.1109/ACCESS.2025.3538461.
- Ullah, S., Wu, J., Kamal, M.M., Mohamed, H.G., Sheraz, M., Chuah, T.C., 2025. MBID: a scalable multi-tier blockchain architecture with physics-informed neural networks for intrusion detection in large-scale IoT networks. Computer Modeling in Engineering and Sciences. doi:10.32604/cmes.2025.068849, online first.
- Vargas, H., Lozano-Garzon, C., Montoya, G.A., Donoso, Y., 2021. Detection of security attacks in industrial IoT networks: A blockchain and machine learning approach. Electronics 10, 2662. doi:10.3390/electronics10212662.
- Villegas-Ch, W., Govea, J., Gutierrez, R., Mera-Navarrete, A., 2025. Optimizing security in IoT ecosystems using hybrid artificial intelligence and blockchain models: a scalable and efficient approach for threat detection. IEEE Access 13, 16933-. doi:10.1109/ACCESS.2025.3532800.
- Zeng, H., Yunis, M., Khalil, A., Mirza, N., 2024. Towards a conceptual framework for AI-driven anomaly detection in smart city IoT networks for enhanced cybersecurity. Journal of Innovation & Knowledge 9, 100601. doi:10.1016/j.jik.2024.100601.