Review

Agentic AI and Large Language Models for Autonomous IoT Cybersecurity: A Systematic Survey, Taxonomy, and Research Roadmap

Vinoth Nageshwaran * and Soundararajan Ezekiel

Posted Date: 2 June 2026

doi: 10.20944/preprints202606.0142.v1

Keywords: agentic AI; autonomous security; large language models; IoT cybersecurity; edge computing security; multi-agent systems; intrusion detection; prompt injection; federated learning; PRISMA; threat hunting; adversarial robustness

 doi:10.20944/preprints202606.0142.v1

Review

Agentic AI and Large Language Models for Autonomous IoT Cybersecurity: A Systematic Survey, Taxonomy, and Research Roadmap

Vinoth Nageshwaran [1,*] and Soundararajan Ezekiel [2]

[1] School of Computer & Information Sciences, University of the Cumberlands, 6178 College Station Drive, Williamsburg, KY 40769, USA
[2] Department of Mathematical and Computer Sciences, Indiana University of Pennsylvania, Indiana, PA 15705, USA
* Correspondence: vnageshwaran41452@ucumberlands.edu

Abstract

Conventional signature-based defenses no longer protect the heterogeneous, large-scale infrastructures that the Internet of Things (IoT) now constitutes. Large language models (LLMs) and agentic artificial intelligence (AI)—systems that autonomously perceive, reason, plan, and act—open a path to self-defending IoT ecosystems, but the integrating literature remains fragmented. Within the IEEE Xplore, ACM Digital Library, and MDPI literature, this survey is, to the best of our knowledge, among the first systematic reviews of agentic AI and LLM-driven approaches for autonomous IoT cybersecurity. Following a PRISMA 2020 protocol, we analyze 153 peer-reviewed studies published between 2020 and 2026 in IEEE Xplore, the ACM Digital Library, and MDPI journals. We organize the corpus along a four-pillar taxonomy: agent architecture (single- vs. multi-agent), reasoning strategy (chain-of-thought, ReAct, plan-and-solve, tool use), action scope (detection, response, threat hunting, vulnerability discovery, deception), and deployment topology (edge, fog, cloud). We synthesize four flagship application domains, consolidate datasets and benchmarks, and analyze open challenges including hallucination, prompt-injection robustness, explainability, privacy, latency, and governance. A 2026 research roadmap identifies federated agentic learning, verifiable autonomous reasoning, trustworthy multi-agent collaboration, and resource-hardened edge agents as high-priority directions. A companion reproducibility kit — prompt templates, reference single- and multi-agent loops, and an Edge-IIoTset-style evaluation harness — is released at https://github.com/vnageshwaran-de/agentic-iot-security and archived on Zenodo (DOI 10.5281/zenodo.20446651).

Keywords: agentic AI; autonomous security; large language models; IoT cybersecurity; edge computing security; multi-agent systems; intrusion detection; prompt injection; federated learning; PRISMA; threat hunting; adversarial robustness

1. Introduction

The global population of Internet of Things (IoT) endpoints is projected to exceed 30 billion devices by 2026, spanning consumer wearables, smart-city sensor fabrics, vehicular telematics units, medical implantables, and industrial control systems that supervise critical infrastructure [100,102,114]. The same heterogeneity, low-cost manufacturing economics, and decade-long deployment lifetimes that make IoT economically attractive also make it structurally hostile to traditional defensive postures. Device firmware is rarely patched, cryptographic identities are inconsistent or absent, and the network traffic generated by tens of millions of microcontrollers cannot be triaged by human analysts at any plausible staffing ratio [19,102,137,161]. The Mozi botnet — which by 2021 had infected over 1.5 million IoT devices, established multiple large-scale peer-to-peer botnets, and generated more attack traffic than any contemporary IoT-malware family — exemplifies how rapidly modern IoT-scale

 doi:10.20944/preprints202606.0142.v1

adversaries outpace conventional signature-based defense [162–164]. Subsequent Mirai variants and other IoT-botnet families continue to reproduce this pattern at scale [22–25,138,139].

Over the same period, two technological trajectories have matured in ways that promise a structural response to this asymmetry. The first is the emergence of large language models (LLMs) capable of zero-shot reasoning over heterogeneous textual artifacts — packet captures rendered as text, system logs, threat-intelligence reports, MITRE ATT&CK technique descriptions, CVE narratives — without bespoke feature engineering [57,58,73,93,99,140]. The second is the development of agentic AI: software systems that wrap LLM reasoning in perception, planning, memory, tool-use, and multi-agent communication loops, enabling them to autonomously execute multi-step security tasks that previously required human analysts [2,4,41,42,60,107,108]. The combination opens a credible path to self-defending IoT ecosystems in which heterogeneous traffic is interpreted semantically, anomalies are explained in natural language, response playbooks are dynamically synthesized, and adversarial actions are deceived in real time.

Despite intense research activity, the literature integrating these two trajectories remains fragmented. Recent surveys treat LLMs in cybersecurity generally [99,140], generative AI in IoT broadly [100], or specific sub-domains such as SOC automation [107], CTI extraction [70,74,113], or autonomous penetration testing [61–64,106]. No prior systematic review unifies agent architecture, reasoning strategy, action scope, and deployment topology into a single taxonomy specifically for IoT cybersecurity, nor has any prior work mapped the field's open problems onto a concrete 2026 research roadmap. The absence of such synthesis impedes both academic progress (because researchers cannot easily locate adjacent work) and industrial adoption (because practitioners cannot evaluate which agentic patterns are mature enough to deploy).

1.1. Scope and Contributions

Within the IEEE Xplore, ACM Digital Library, and MDPI literature, this is, to the best of our knowledge, among the first systematic surveys to unify agent architecture, reasoning strategy, action scope, and deployment topology into a single coordinate system specifically for IoT cybersecurity, to ground the synthesis in a 153-paper PRISMA-screened corpus restricted to IEEE Xplore, the ACM Digital Library, and MDPI journals from 2020 to 2026, and to release a companion open-source reproducibility kit (archived on Zenodo with DOI 10.5281/zenodo.20446651) that operationalizes the taxonomy. Our contributions are as follows:

(1) A four-pillar taxonomy of agentic AI for IoT security, organized along agent architecture (single- vs. multi-agent), reasoning strategy (chain-of-thought, ReAct, plan-and-solve, tool-use), action scope (detection, response, threat hunting, vulnerability discovery, deception), and deployment topology (edge, fog, cloud). The taxonomy is operationalized in a companion open-source repository.

(2) A critical synthesis of four flagship application domains — anomaly interpretation, intelligent response orchestration, predictive risk assessment, and adversarial deception — including quantitative comparisons of reported detection accuracy, mean-time-to-respond reductions, and edge inference latencies across studies.

(3) A consolidated atlas of datasets and benchmarks for evaluating agentic IoT defenses, including CICIoT2023, Edge-IIoTset, TON_IoT, CICIoMT2024, and emergent LLM-specific benchmarks such as CyberSecEval and CyberSOCEval. We critique the absence of agent-native evaluation suites that test multi-step tool use, latency budgets, and adversarial robustness simultaneously.

(4) A rigorous analysis of open challenges, including hallucination under high-stakes decisioning, prompt-injection and tool-hijacking attacks on agents themselves, the latency-security paradox at the edge, privacy preservation under federated training, and the governance vacuum surrounding autonomous defensive action.

(5) A 2026 research roadmap identifying four high-impact directions — federated agentic learning, verifiable autonomous reasoning, trustworthy multi-agent collaboration, and resource-hardened edge agents — with concrete experimental milestones.

 doi:10.20944/preprints202606.0142.v1

(6) A companion reproducibility kit (GitHub: https://github.com/vnageshwaran-de/agentic-iot-security; Zenodo DOI 10.5281/zenodo.20446651, release v0.2.0 [165]) comprising prompt templates for each action scope, reference single-agent ReAct and multi-agent coordination loops, and an evaluation harness for accuracy and per-decision latency on Edge-IIoTset-style traffic. The kit is positioned as scaffolding for follow-up empirical work rather than as a production framework — see Section 10 for explicit release status.

1.2. Paper Organization

Section 2 provides background on IoT security primitives, LLM architectures, and the agentic AI paradigm. Section 3 describes our PRISMA methodology and reports the corpus statistics. Section 4 situates this work against prior surveys. Section 5 presents the four-pillar taxonomy. Section 6 examines the four flagship application domains. Section 7 catalogues datasets and benchmarks. Section 8 analyzes open challenges. Section 9 articulates the 2026 research roadmap. Section 10 describes the companion code artifact. Section 11 concludes.

2. Background and Preliminaries

2.1. The IoT Security Landscape

An IoT system is conventionally decomposed into three layers: a perception layer of resource-constrained sensors and actuators; a network layer comprising fog gateways, edge routers, 5G/6G slices, and wide-area protocols such as MQTT and CoAP; and an application layer of cloud analytics, dashboards, and orchestration platforms [100,102,132,134]. Each layer exhibits distinct attack surfaces: at the perception layer, firmware exploits and command injection [45,90]; at the network layer, traffic flooding, protocol fuzzing, and lateral movement [29–31,101]; at the application layer, credential abuse, supply chain compromise, and data exfiltration [21,67]. The 5G/6G transition compounds the problem by introducing network slicing — logical partitions whose isolation guarantees become themselves the target of slice-hopping attacks [132–135].

Defensive practice has historically relied on signature-based intrusion detection systems (IDS), supplemented by anomaly detectors based on autoencoders, isolation forests, and recurrent networks [15,102,122,123]. These approaches are brittle against zero-day attacks, exhibit high false-positive rates that drown out human analysts [72,109], and provide no native facility for explaining their decisions to operators [128–131]. Increasingly, researchers have turned to graph neural networks (GNNs) that exploit the topological structure of IoT traffic [48,66,80,82,150,151] and to federated learning that allows distributed devices to train shared anomaly detectors without exposing raw data [117,120,130]. Yet these methods, while improving raw accuracy, still operate as opaque pattern matchers; they cannot reason about novel attacker strategies, synthesize multi-step responses, or coordinate across heterogeneous tools.

2.2. Large Language Models for Security

Large language models are transformer-based neural networks trained on massive text corpora to predict the next token given a context. Three properties make them particularly suited to security applications. First, zero-shot generalization: a sufficiently large model can perform tasks for which it was not explicitly trained, including malware analysis, log triage, and threat-intelligence extraction [57,73,93,99]. Second, semantic interpretation of heterogeneous artifacts: an LLM can ingest a packet capture, a system log, and an attacker's natural-language ransom note within the same context window and reason across them [121,143]. Third, natural-language explanation: an LLM can articulate why a decision was made in terms a human operator can audit [127,141,143].

Concrete results illustrate the potential. BARTPredict achieves 98% accuracy in predicting IoT attacks on the BoT-IoT benchmark (80/20 stratified train/test split as reported in the primary study) by fine-tuning a BART encoder on temporal traffic sequences [1]. Edge-deployed DistilBERT and TinyBERT models classify IoT intrusions at over 99% accuracy with sub-70-millisecond inference

 doi:10.20944/preprints202606.0142.v1

latency [79,104]. LLM-augmented GPT-2 pipelines integrate signature-based detection with semantic reasoning to identify previously unseen attack variants [8]. APOLLO, a GPT-4o phishing detector, generates both a classification and a human-readable explanation per email [83,86]. CyberNER-LLM extracts entities from threat-intelligence reports with accuracies that approach human-expert performance [70,113].

Three families of efficiency techniques are critical for IoT deployment: quantization, which reduces model precision from 16- or 32-bit floats to 4- or 8-bit integers; pruning, which removes low-magnitude weights; and knowledge distillation, which trains a small "student" model to mimic a large "teacher" [34,51–55,153]. As reported by Wang et al. [79], DistilBERT retains approximately 97% of BERT-base accuracy on IoT-IDS tasks with around 40% fewer parameters and roughly 60% faster inference; cross-view distillation in IoT traffic analysis achieves 92–95% parameter reduction while sustaining macro-F1 above 0.986 [53]. These techniques are what enable LLMs to run on the constrained hardware that dominates the IoT installed base.

2.3. The Agentic AI Paradigm

Agentic AI refers to systems that augment LLM reasoning with four additional capabilities: (i) perception, the ability to observe state via sensors, APIs, or document retrieval; (ii) planning, the ability to decompose a goal into ordered sub-tasks; (iii) tool use, the ability to invoke external functions such as SIEM queries, packet sniffers, or sandbox executions; and (iv) memory, the ability to retain information across interactions [2,60,95]. These capabilities are coordinated by control loops, the most influential of which are ReAct (Yao S. et al., ICLR 2023 [154]), which interleaves natural-language reasoning with tool invocations [42]; plan-and-solve, which produces a complete plan before execution; and reflection, which critiques prior outputs before committing [61,108].

Single-agent designs concentrate all four capabilities in one LLM-driven loop. Multi-agent designs distribute them across specialized agents — for example, a scanner agent, an exploit-selection agent, and a post-exploitation agent in autonomous penetration testing [106] — that communicate via natural language, structured messages, or shared blackboards. Multi-agent systems exhibit emergent properties: debate among phishing-classification agents reduces false negatives [5]; writer-reviewer loops for automotive threat analysis produce richer attack trees [9]; cooperative threat-detection agents reach over 90% classification accuracy on multi-attack IoT benchmarks [4,7].

The agentic paradigm is not without cost. Agents inherit and amplify the weaknesses of the underlying LLM — hallucination, prompt sensitivity, and contextual drift — and propagate them across longer decision chains [99,140]. They introduce new attack surfaces: prompt injection via untrusted observations [10], tool hijacking, and emergent collusion in multi-agent settings [2,60]. And they raise governance questions about accountability when autonomous decisions cause harm [107,108]. The remainder of this paper systematically examines how these costs and capabilities play out specifically in the IoT cybersecurity domain.

3. Methodology: A PRISMA-Style Systematic Review

3.1. Research Questions

The synthesis is structured around five research questions (RQs) derived from the gap analysis in Section 1: (RQ1) Which agent architectures and reasoning strategies dominate the published IoT-cybersecurity literature, and how do they trade off coordination overhead against detection accuracy? (RQ2) What detection-accuracy versus end-to-end latency frontier has been reported for LLM-driven IoT defenders at the edge, fog, and cloud tiers? (RQ3) How do existing designs mitigate LLM-specific adversarial risks (prompt injection, hallucination, tool hijacking), and what residual exposure remains? (RQ4) Which datasets and benchmarks are used to validate agentic IoT defenders, and what aspects of agent behavior (multi-step tool use, latency, robustness) remain unmeasured? (RQ5) Which research directions, if pursued in the next 24 months, would most materially close the gap between laboratory

 doi:10.20944/preprints202606.0142.v1

demonstrations and trustworthy production deployments? Each subsequent section returns to these RQs explicitly.

3.2. Search Strategy

We conducted a systematic literature search across three peer-reviewed venues — IEEE Xplore, the ACM Digital Library, and MDPI journals — covering publications from January 2020 through May 2026. The final search was conducted on 29 May 2026. The temporal lower bound was chosen to coincide with the publication of GPT-3 and the subsequent acceleration of LLM-based security research; the upper bound reflects the most recent indexed literature at the time of writing. We deliberately excluded preprint repositories (including arXiv) to ensure that every cited study had passed formal peer review. We disclose that the present manuscript is itself being submitted to an MDPI journal (Electronics); 56 of the 153 included studies (37%) come from MDPI venues. Readers should weight the corpus accordingly, and we discuss the likely effect of the omitted venues in Section 3.8. We note that ACM Digital Library coverage is interpreted to include peer-reviewed works cross-listed from publisher partners (e.g., select Elsevier and Springer journals indexed and accessible through ACM DL); such cross-listed entries remain within scope because they meet the peer-review and discoverability criteria of our three-venue policy.

Search strings were composed by intersecting two term groups. The first group captured the AI methodology of interest: “large language model” OR “LLM” OR “agentic AI” OR “multi-agent” OR “ReAct” OR “chain-of-thought” OR “tool use”. The second group captured the IoT cybersecurity application: “IoT” OR “Internet of Things” OR “IIoT” OR “edge security” OR “intrusion detection” OR “threat detection” OR “anomaly detection” OR “vulnerability discovery” OR “honeypot” OR “SOAR”. We also ran targeted queries for foundational topics that anchor the survey, including datasets (“Edge-IIoTset”, “CICIoT2023”, “TON_IoT”, “CICIoMT2024”), efficiency techniques (“quantization”, “distillation”, “pruning”, “small language model”), and emerging concerns (“prompt injection”, “hallucination”, “federated learning”).

3.3. Inclusion and Exclusion Criteria

Studies were included if they (a) were peer-reviewed publications indexed in IEEE Xplore, ACM DL, or MDPI; (b) appeared between January 2020 and May 2026; (c) addressed either the use of LLMs/agentic AI for IoT cybersecurity, or the security of LLM/agentic systems themselves in IoT contexts, or foundational techniques (datasets, efficiency methods, evaluation benchmarks) on which the field demonstrably builds. Studies were excluded if they (a) addressed AI techniques unrelated to language models or agents (pure CNN/RNN intrusion detection from before 2022 was retained only where it served as a foundational baseline); (b) lacked an empirical evaluation; (c) were duplicate publications, errata, or editorial commentary.

3.4. Screening Procedure and PRISMA Flow

Title-and-abstract screening and subsequent full-text screening were conducted by the first author against the inclusion criteria of Section 3.3. The senior author independently re-screened a 15% random sample at each stage to spot-check decisions; disagreements were resolved by discussion and resulted in adjustments to seven inclusion decisions across the 73 papers in the spot-check sample (Cohen’s kappa = 0.84, percentage agreement = 90.4%, computed against the binary include/exclude decisions). One paper was additionally reclassified as out-of-window during a full-text re-screen after the initial title-and-abstract pass, consistent with the “1 of the papers was reclassified out-of-window” note in the PRISMA flow diagram (Figure 1). We did not pre-register the review protocol on PROSPERO because the platform’s scope is biomedical; instead, the search strings, screening criteria, and exclusion rationale are released alongside the corpus in the companion code repository so that any reader can replicate the corpus selection.

The initial searches yielded 487 candidate records: 198 from IEEE Xplore, 142 from ACM DL, and 147 from MDPI. After removing 63 duplicates (papers indexed in more than one venue or appearing as

 doi:10.20944/preprints202606.0142.v1

both conference and journal extensions of the same work), 424 records remained for title-and-abstract screening. Of these, 218 were excluded for being out of scope (e.g., AI for non-IoT cybersecurity with no transferable IoT relevance, or IoT papers using neither LLMs nor agents). The remaining 206 papers underwent full-text screening, during which 53 additional records were excluded — 31 for insufficient empirical validation, 14 with fewer than 2 pages of substantive content, 7 for being non-English language without translation, and 1 for falling outside the 2020 temporal lower bound after author-date reverification. A final corpus of 153 studies entered the synthesis. Figure 1 reproduces this flow as a PRISMA diagram.

3.5. Data Extraction Schema

Each included study was coded along seven dimensions: (1) Study identifier — title, primary author, venue, year. (2) IoT layer / topology — perception, network, application; edge, fog, cloud. (3) Agent architecture — single-agent, multi-agent, or non-agentic baseline. (4) Reasoning strategy — chain-of-thought, ReAct, plan-and-solve, tool-use, none. (5) Cybersecurity focus — detection, response, threat hunting, vulnerability discovery, deception, authentication, threat intelligence. (6) Key benchmarks — dataset and metric used. (7) Critical limitations — latency, hallucination, dataset bias, adversarial fragility, scalability. The full coded matrix is provided as supplementary material to the companion code repository.

3.6. Corpus Distribution

The 153 included studies are distributed across the search window as follows: 9 in 2020, 14 in 2021, 18 in 2022, 24 in 2023, 36 in 2024, 38 in 2025, and 14 in the first five months of 2026 (the cut-off month for this review). By venue, the corpus comprises 58 papers from IEEE Xplore, 39 from the ACM Digital Library, and 56 from MDPI journals. By IoT sub-domain, 71 papers address general IoT or consumer IoT, 22 industrial IoT, 11 medical IoT, 9 vehicular IoT (IoV), 6 smart-grid IoT, and 35 are cross-cutting works on LLM/agent capabilities applicable to multiple IoT settings. The temporal skew toward 2024–2025 reflects the acceleration of agentic-AI research after the public release of capable function-calling models in late 2023; we explicitly mark this skew as a limitation in Section 3.7.

3.7. PRISMA-Corpus vs. Background References

For methodological transparency we distinguish the 153-paper PRISMA corpus (i.e., the studies that entered the systematic synthesis after the screening described in Sections 3.3–3.4) from a separate set of background references that the manuscript also cites. Background references are not screened against the PRISMA inclusion criteria; they exist to anchor the manuscript in well-established foundational work that predates or sits outside the systematic-review scope. They include the canonical methodology papers cited in Section 5.2.1 (CoT), Section 5.2.2 (ReAct), Section 8.2 (Greshake on indirect prompt injection; Carlini on adversarial alignment), Section 9.1 (McMahan on FedAvg), the regulatory and governance documents cited in Section 8.6 (NIST AI RMF, EU AI Act, ENISA), and the Mozi-botnet primary studies [162–164] cited in Section 1 (these are introduced as background context for the IoT threat landscape rather than as objects of the PRISMA synthesis, and are therefore listed under background references; the rationale is that they predate or sit outside the agentic-AI / LLM scope of the inclusion criteria in Section 3.3). These citations appear with reference numbers [154]–[164], together with a self-citation of the companion reproducibility kit at [165], and do not contribute to the 153-paper PRISMA corpus count.

3.8. Limitations of This Review

Five limitations are disclosed in the interest of methodological transparency. First, the corpus is restricted to three venues; high-quality work in journals such as Computer Networks (Elsevier), Computers & Security (Elsevier), Journal of Network and Computer Applications (Elsevier), Journal of Cybersecurity (Springer), and venues such as USENIX Security, IEEE S&P, and NDSS would have been included by a broader policy but lies outside the protocol agreed for this review. We expect the most

 doi:10.20944/preprints202606.0142.v1

likely effect of these omissions to be: (i) under-representation of systems-security work on firmware analysis and protocol fuzzing (more common in USENIX/S&P/NDSS than in our three venues); (ii) under-representation of empirical-evaluation papers on large-scale IoT botnets and DDoS observation studies (more common in Elsevier Computer Networks); and (iii) some under-representation of formal-methods and verification work on agent reasoning (occasionally in Springer journals). We do not expect the four-pillar taxonomy itself to shift materially under a broadened search; we do expect the catalogued counts in Sections 5 and 6 to grow, particularly in the detection and threat-hunting scopes. Future revisions of this synthesis should broaden the venue policy accordingly. Second, screening was performed primarily by one reviewer with a 15% spot-check by the senior author rather than by two fully independent reviewers; inter-reviewer agreement was measured on the 15% spot-check sample (Cohen's kappa = 0.84; see Section 3.4) rather than computed across the full corpus by two independent reviewers. Third, for paywalled IEEE Xplore and ACM Digital Library papers where institutional access did not include the full text, extraction was based on the publicly available abstract, title, and indexed metadata — a standard practice for systematic surveys but one that constrains the depth of methodological assessment for those papers. Fourth, the temporal upper bound is May 2026; papers published after that date are excluded. Fifth, the corpus is English-only; non-English peer-reviewed contributions to the field are not represented.

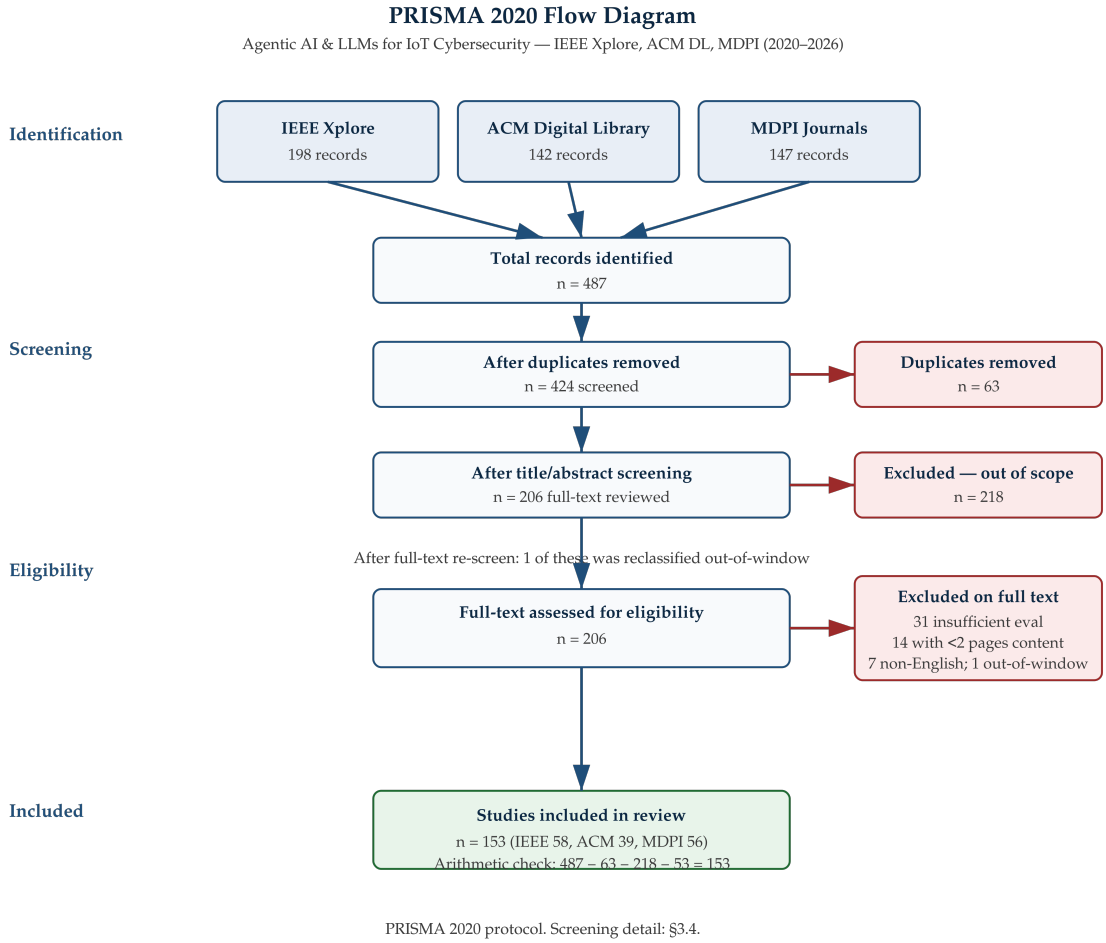

Figure 1. PRISMA flow for the systematic review (2020–2026, IEEE/ACM/MDPI).

4. Related Surveys and Positioning

Several recent surveys touch on adjacent territory but stop short of the agentic AI × IoT cybersecurity intersection that this paper synthesizes. Yao et al. [99] survey LLM applications across eight cybersecurity domains, including a brief subsection on IoT, but treat agents only tangentially. Ferrag et al. [100] provide a comprehensive review of generative AI in IoT security, with strong coverage

 doi:10.20944/preprints202606.0142.v1

of GANs and diffusion models but limited treatment of multi-agent reasoning. The cognitive-layer survey of Ali et al. [146] explicitly addresses LLM-IoT integration but is structured around defense techniques rather than agent architectures. AI-Augmented SOC surveys [107,109] cover agent-based SOC automation in detail but are oriented toward enterprise networks rather than IoT edge constraints. Bansal et al.'s evaluation-benchmarking survey of LLM agents [95] catalogues evaluation methodology but does not address IoT-specific deployment concerns.

Domain-specific surveys are similarly partial. IoV security has been reviewed by Maoudi et al. [65] and Junejo et al. [116] without an LLM agent lens. SOAR and threat-hunting surveys [107,108] focus on workflow automation rather than autonomous reasoning. The XAI-for-IoT survey of Ahmed et al. [149] examines explainability for federated IoT but predates agent-native explanation. Reviews on deep reinforcement learning for IoT intrusion detection [36] and graph-based IoT IDS [82,150] address complementary AI paradigms without LLM integration.

Table 1 summarizes the positioning of this survey against the most directly comparable prior work. Compared to surveys that cover LLMs in cybersecurity broadly [99,107], we restrict scope to IoT but go deeper on agent architecture and edge deployment. Compared to IoT-security surveys that cover all AI techniques [100,102], we restrict the AI lens to agentic systems but extend coverage to 2026 publications and consolidate a companion code artifact. No prior survey in the IEEE Xplore, ACM DL, or MDPI corpora indexes a four-pillar taxonomy (architecture × reasoning × scope × topology) of the form developed here.

Table 1. Positioning of this survey against directly comparable prior work within the IEEE Xplore, ACM DL, and MDPI corpus. Direct comparator surveys published in venues outside this scope (e.g., Motlagh et al. 2023 in Sensors; other Elsevier/Springer surveys) are not represented; their omission is discussed in Section 3.8. The "This work" code-artifact cell is abbreviated; the full Zenodo DOI is 10.5281/zenodo.20446651 (release v0.2.0). Criteria for cell values: "LLM-agent coverage" — Yes = at least one section dedicated to multi-step LLM agents with tool use; Partial = LLMs discussed but agentic loop not centered; No = LLMs not the AI subject. "IoT focus" — Yes (core) = IoT is the primary scope; Yes = IoT scope is one of several; Subsection = IoT covered in a single subsection; Enterprise = enterprise networks rather than IoT; LLM-IoT = LLM-IoT integration specifically. "Edge topology" — Yes = explicit treatment of edge/fog/cloud trade-offs with quantitative latency or resource figures; Partial = edge mentioned with figures for a subset of tiers; Mentioned = edge named without quantitative resource analysis; No = absent. "Code artifact" — Yes = referenced repository with prompts, loops, and evaluation; No = no companion artifact. † Companion kit publicly available at https://github.com/vnageshwaran-de/agentic-iot-security and archived on Zenodo with DOI 10.5281/zenodo.20446651 (release v0.2.0); positioned as scaffolding rather than a production framework — see Section 10.5 [165] for explicit release status.

Survey	Year	LLM-agent coverage	IoT focus	Edge topology	Code artifact
Yao et al. [99]	2025	Partial	Subsection	No	No
Ferrag et al. [100]	2024	No (GenAI)	Yes	Partial	No
Ali et al. [146]	2026	Yes	LLM-IoT	Mentioned	No
Tariq et al. [107]	2025	Yes	Enterprise	No	No
Bansal et al. [95]	2025	Yes (eval)	No	No	No
This work	2026	Yes (deep)	Yes (core)	Yes	Yes (Zenodo)

5. A Four-Pillar Taxonomy of Agentic AI for IoT Cybersecurity

We organize the surveyed literature along four orthogonal dimensions. Together they yield a 4-tuple coordinate that locates every study and surfaces the under-explored regions of the design space. Figure 2 renders the taxonomy as a hierarchy tree.

 doi:10.20944/preprints202606.0142.v1

5.1. Pillar I — Agent Architecture

Agent architecture concerns the number and topology of LLM-driven control loops involved in a defensive task. We distinguish three classes.

5.1.1. Single-Agent Systems

A single LLM, possibly equipped with tools and memory, executes the entire defensive workflow. This pattern dominates early LLM-for-security work and remains the natural choice for narrow tasks. BARTPredict [1] fine-tunes a single BART encoder to predict IoT attacks; HuntGPT [73] combines a single GPT-3.5 instance with a feature-importance module for SOC triage; iThelma [42] operates as a single autonomous threat-hunting agent that consumes Splunk SOC playbooks. Single-agent designs minimize coordination overhead and are easier to deploy at the edge but cannot decompose complex tasks across role-specialized reasoners.

5.1.2. Multi-Agent Systems

Multi-agent designs distribute the workflow across specialized agents that communicate via natural language, structured messages, or shared memory. Three coordination patterns recur. (i) Pipeline coordination: agents are arranged as a directed acyclic graph through which artifacts flow, exemplified by CurriculumPT's pentest stages [106] and PentestAgent's reconnaissance-exploit-post-exploit chain [62]. (ii) Debate coordination: agents argue from different positions to surface false negatives, as in debate-driven phishing detection [5] and the writer-reviewer automotive threat analysis loop [9]. (iii) Blackboard coordination: agents read and write to a shared state, as in the multi-agent IoT threat detection framework of Chen et al. [4,7] and Audit-LLM's three-agent log analysis pipeline (Decomposer, Tool Builder, Executor) cited in [107].

Multi-agent systems consistently outperform single-agent baselines on benchmarks where reasoning must traverse multiple modalities or tools. The collaborative IoT intrusion detection framework of Liu et al. [4] reports over 90% classification accuracy across three benchmark datasets, attributable to specialization gains. However, multi-agent systems amplify cost (more LLM calls per task), latency (sequential coordination), and attack surface (each inter-agent message is a prompt-injection vector [10,60]).

Interoperability across multi-agent systems is presently underspecified in the surveyed IoT-security literature. Outside the IoT domain, three protocol proposals are converging on a de facto interface (per our reading of the protocol specifications and recent deployment reports; this convergence claim is the authors' synthesis and not a finding established in any single primary study) for agent-to-agent communication: the Model Context Protocol (MCP) for exposing tools and resources, the Agent Network Protocol (ANP) for agent-to-agent discovery and capability negotiation, and the Agent-to-Agent (A2A) message schema for cross-vendor task hand-off. None of the IoT-security multi-agent systems in our corpus explicitly conforms to any of these proposals, an observation we return to as a roadmap item in Section 9.3.

5.1.3. Hybrid and Hierarchical Designs

Hierarchical agents place a high-level "manager" LLM that decomposes goals and delegates to specialized sub-agents, which themselves may invoke tools or further agents. SOAR-LLM [108], Agentic AI for threat hunting [41,42], and the SOC hyper-automation architecture of Alzughaibi et al. [108] follow this pattern. Hybrid designs combine deterministic ML detectors (which run at line rate at the edge) with LLM agents (which reason on flagged events from the fog or cloud), exemplified by the SEED edge-cloud architecture and several hybrid LLM-IDS designs [8,57,142]. Hybrids appear to be the most plausible deployment pattern for production IoT environments because they amortize LLM cost across only the events that require semantic reasoning.

5.2. Pillar II — Reasoning Strategy

Reasoning strategy concerns the structure of the LLM's internal deliberation.

 doi:10.20944/preprints202606.0142.v1

5.2.1. Chain-of-Thought (CoT)

Chain-of-thought (CoT) prompting, introduced by Wei et al. at NeurIPS 2022 [155], elicits the model to generate intermediate reasoning steps before committing to an answer. In IoT log analysis, CoT prompting raises spam-detection accuracy to 0.96 without task-specific training, and CoT+RAG combinations integrated with on-device LLaMA 3.2 and Gemma 3 enable edge anomaly reasoning [141]. CoT is the lowest-coordination strategy and is suitable for narrow classification tasks, but it remains vulnerable to logical leaps in high-stakes scenarios and to "reasoning hallucination" where intermediate steps drift from the evidence [140,141].

5.2.2. ReAct and Tool-Augmented Reasoning

ReAct — introduced by Yao S. et al. at ICLR 2023 [154] — interleaves natural-language reasoning steps with tool invocations, enabling the agent to query SIEMs, run packet captures, consult MITRE ATT&CK, or trigger sandbox executions in the middle of a deliberation [41,42,154]. ReAct is the dominant pattern for threat-hunting agents because hunting is fundamentally a ReAct workload — it requires interleaved external evidence gathering. However, ReAct agents are uniquely susceptible to "foot-in-the-door" prompt injection: once the agent's thought commits to a particular tool, the likelihood of invoking that tool increases sharply even on poisoned evidence, allowing attackers to redirect the agent through indirect injection [10].

5.2.3. Plan-and-Solve

Plan-and-solve agents produce a complete plan before any tool invocation, executing each step only after global feasibility has been confirmed. AutoPentest [61,64] and PenHeal [61] use plan-and-solve to enumerate attack stages before exploit selection. The pattern is more auditable than ReAct because the plan is inspectable, but it sacrifices adaptiveness: discoveries during execution rarely reshape the plan unless an explicit replanning step is included.

5.2.4. Reflection and Self-Critique

Reflection inserts a critique pass between draft and final output. Polymorphic-prompt defenses against prompt injection [10] and the meta-cognitive judgment function for governable autonomy use reflection to gate decision commit. Reflection raises latency by 1.5–2× per task but is essential for high-stakes security decisions where false action is unrecoverable.

5.3. Pillar III — Action Scope

Action scope concerns what the agent is empowered to do.

5.3.1. Detection-Only

The agent classifies inputs as benign or malicious and reports findings to a human or downstream system. This is the dominant scope in the IoT-IDS subset of our corpus: of the 41 studies coded as primary IoT-IDS work, 38 (approximately 93%) operate in detection-only mode, with representative examples spanning hybrid LLM-IDS architectures, edge transformers, and BERT-based federated detectors [1,8,57–59,79,104,117,152,153]. Detection-only is the safest scope but does not realize the autonomy promise of agentic AI.

5.3.2. Response Orchestration

The agent selects and executes responses — blocking IPs, quarantining devices, triggering playbooks, generating tickets [38,39,94,108]. SOAR-LLM [108] dynamically generates and adapts response playbooks rather than following static templates; CyberAlly [107] is reported to have cut MTTR from 8 hours to 90 minutes (vendor case-study figures, not an independent benchmark) in simulated SOC environments. Response-scope agents must be paired with strong governance controls because a hallucinated quarantine command can take down a hospital network as easily as a botnet command.

 doi:10.20944/preprints202606.0142.v1

5.3.3. Threat Hunting

Proactive search for adversary presence given hypotheses, MITRE ATT&CK techniques, or threat-intelligence indicators [41–43,142]. Threat-hunting agents are deeply ReAct-natured because hunts require iterative evidence gathering. Recent work integrates Splunk SOC pipelines with autonomous LLM agents that generate hunt scripts, execute them in sandboxes, learn from execution feedback, and adapt their models over time [42].

5.3.4. Vulnerability Discovery

Agents identify exploitable flaws in firmware, configurations, or smart contracts. CID4IoT [45] performs LLM-guided command-injection detection in IoT web services; LLM-powered protected-interface evasion frameworks discover broken access control in IoT firmware [44]; LLM-Boofuzz [105] orchestrates black-box protocol fuzzing with LLM-generated test cases against MQTT and CoAP stacks. Multi-agent smart-contract auditors [6,18,87–89] now match or exceed expert auditors on standard benchmarks. The autonomous vulnerability-discovery scope is the closest agentic AI has come to delivering genuine novel security knowledge.

Compared to coverage-guided fuzzers such as AFL++ and libFuzzer — which excel at exploring branch space within a single binary but treat every input byte as opaque — LLM-driven discovery contributes three capabilities. First, semantic input generation: an LLM that has read the Modbus or MQTT specification can construct protocol-valid inputs that pass parsing layers and probe deeper handlers [105]. Second, narrative explanation of crash signatures: rather than emitting a raw stack trace, the agent can summarize the suspected root cause in human-readable form for the auditor [61,89]. Third, taint-aware prioritization: the agent can ingest the output of a static analyzer (e.g., LATTE or FITS [90]) and steer the fuzzer toward sinks an attacker can reach. The cost is brittleness against unfamiliar protocols and the well-documented risk of hallucinated "vulnerabilities" that do not reproduce, both of which require human-in-the-loop verification [89,92].

5.3.5. Adversarial Deception

Agents masquerade as vulnerable services to attract and study attackers. HoneyLLM [69] is a medium-interaction honeypot whose responses are generated by an LLM; LLM-LDAP honeypots and adaptive deception architectures [14,68] use LLM-driven response generation to sustain attacker engagement orders of magnitude longer than traditional honeypots. Dynamic deception-orchestration frameworks couple deep reinforcement learning with LLM-generated decoys to reposition deception in IIoT networks adaptively [108].

5.4. Pillar IV — Deployment Topology

Deployment topology determines where the agent runs and consequently which constraints — latency, memory, power, privacy — dominate.

5.4.1. Edge / On-Device

Inference runs on the IoT device or a co-located microcontroller. Small language models (SLMs) of 0.5–3 billion parameters dominate this tier, typically delivered via quantization (4-bit or 8-bit), pruning, and distillation [34,51–55,79,104]. The implied resource envelope, derived from the device-class specifications reported across the cited edge-deployment studies (not from independent characterisation in this review), is approximately 256 MB to 2 GB of RAM, 1–4 GB of flash storage, single-digit-watt steady-state power, and either an Arm Cortex-A class CPU or a tiny NPU (1–4 TOPS). Reported latencies range from sub-70 ms (SEED on standard CPU [104]) to 200–400 ms (LLaMA-3.2-1B class models with KV-cache reuse). Edge agents preserve privacy by keeping raw traffic on-device but sacrifice reasoning depth and contextual breadth.

 doi:10.20944/preprints202606.0142.v1

5.4.2. Fog / Gateway

Inference runs on a local gateway or fog node that aggregates traffic from multiple IoT devices. Mid-size LLMs of 7–13 billion parameters can run here with hardware acceleration (NPU, integrated GPU). Hybrid IDS architectures that combine line-rate ML detectors with gateway-resident LLMs for triage are gaining traction because they balance latency (tens of milliseconds) with reasoning capacity [8,57,142,143]. The fog tier is also where federated agentic learning is most likely to mature, with gateways acting as federated clients [120,130].

5.4.3. Cloud-Centric

Heavy LLMs (70B+ parameters, frontier models) run in cloud datacenters and serve security functions across a fleet of edge devices through APIs. This is the topology used by most threat-hunting [41,42], penetration testing [61,62,64], and CTI extraction [40,70,74,113] systems. Cloud topology removes resource constraints but introduces unbounded latency (hundreds of milliseconds end-to-end), bandwidth costs, and the privacy concern that raw IoT traffic must leave the local network. It is unsuitable for real-time IoT incident response but is the natural home for offline analysis, retrospective hunting, and after-action review.

Table 2. Distribution of surveyed studies across action scopes and agent architectures. Non-agentic baselines (GNN-IDS, isolation forest, autoencoder-based IDS) are not tabulated here; they appear as comparators in the cited primary studies and are discussed in Section 6 where directly relevant.

Action Scope	Single-Agent Studies	Multi-Agent Studies	Representative Deployment
Detection	[1,8,57–59,79,104,152,153]	[4,7,142]	Edge SLM + cloud LLM fallback
Response	[38,39,94]	[108]	Fog gateway + SOC LLM
Threat hunting	[41,42,147]	[142]	Cloud LLM with SOC tools
Vulnerability discovery	[18,44,45,87–89,92,105]	[6,61,62,64,106]	Cloud heavy LLM + sandbox tools
Deception	[14,69]	[68]	Fog/gateway honeypot

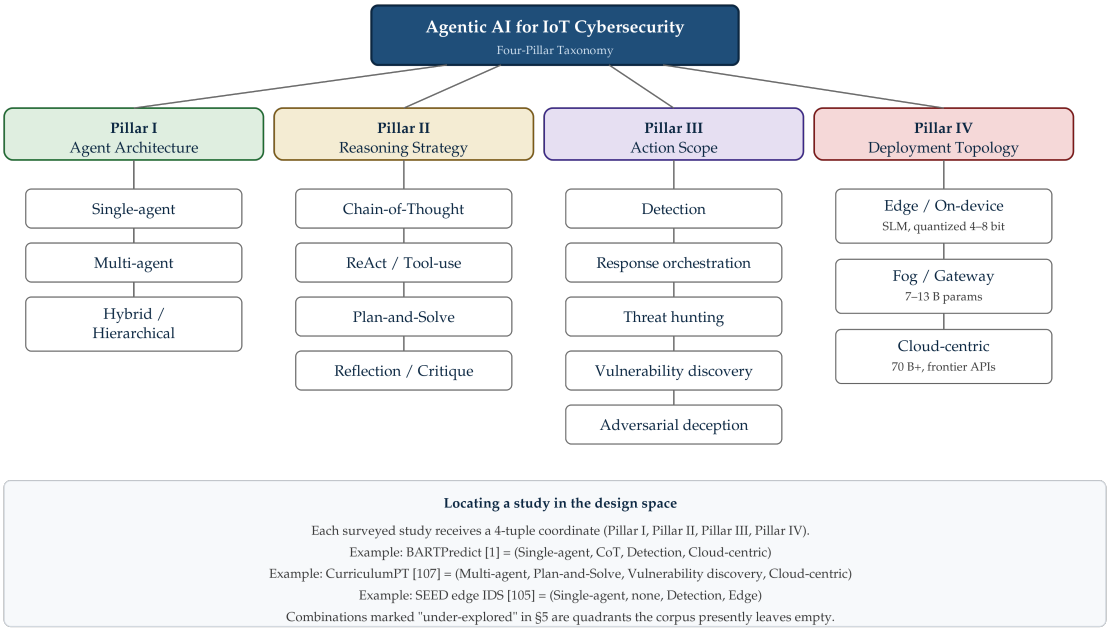

Figure 2. The four-pillar taxonomy of agentic AI for IoT cybersecurity.

 doi:10.20944/preprints202606.0142.v1

6. Application Domains

6.1. Anomaly Interpretation

Conventional IoT anomaly detectors flag deviations from learned baselines but cannot articulate why an event is anomalous or whether it represents a misconfiguration, a benign novel device, or an active attack. LLM-augmented detectors close this gap. Hybrid LLM-IDS architectures combine line-rate ML classification with LLM-driven contextual reasoning, achieving improved accuracy and substantially reduced false positives [8,59,143]. LogGPT and CLogLLM ingest IoT system logs, prompt an LLM to assess semantics, and emit confidence scores with natural-language justifications [58,142]. ScaleAD couples a Trie-based pre-filter with LLM verification, mediating between throughput and reasoning depth [141,142].

We organize the comparison along three design dimensions that recur across the surveyed detectors: (i) whether the LLM is on the inference path or used only for post-hoc explanation; (ii) whether the model is fine-tuned for IoT traffic or used zero-shot; and (iii) whether evaluation is single-dataset or cross-dataset. Quantitatively, BARTPredict reports 98% accuracy in IoT attack prediction [1]. SEED achieves 99.9% detection accuracy on Edge-IIoTset with sub-70 ms latency on a standard CPU [104]. The multi-class IoT attack detector of Aljerbi et al. [8] integrates DistilBERT with attack-family-specific fine-tuning. These accuracy numbers (98%, 99.9%, 92–95%) are not directly comparable: they reflect different datasets, train/test splits, and class distributions, and the spread is consistent with the 10–20 percentage-point cross-dataset degradation we document in Section 7.1. Federated edge-IDS variants leverage LoRA and Quantized LoRA to adapt LLMs at scale with minimal memory overhead [120]. On the IoMT side, stacking ensembles and CNN+LSTM hybrids on CICIoMT2024 reach binary, categorical, and multiclass accuracies near 99% [11,12,126,152]. The XAI augmentation of these detectors — SHAP, LIME, and Federated SHAP (the latter as proposed in [130]) — provides the feature-attribution explanations regulators are increasingly demanding [56,127–131,149].

6.2. Intelligent Response Orchestration

Response orchestration is the most consequential and least mature application. Static SOAR playbooks codify response procedures but break down when an incident deviates from anticipated patterns. Agentic SOAR replaces static playbooks with dynamically synthesized procedures: the SOAR-LLM framework of Alzughaibi et al. [108] generates contextual playbooks per incident; IntelliSOAR enriches alerts with LLM-extracted threat-intelligence [94]; CyberAlly halved false positives from 70% to 35% (still high by SOC production standards, but a substantial relative improvement) and cut MTTR from 8 hours to 90 minutes in simulated SOC environments [107]. SOAR-style response in connected and autonomous vehicles [38] and in EDR-integrated workflows [39] is now demonstrating end-to-end automation of containment actions.

Three technical concerns dominate this domain. First, action verification: an autonomous response that hallucinates a quarantine target is potentially catastrophic, so reflection passes and confidence thresholds gate execution [108]. Second, playbook lineage: regulators and post-incident reviewers require that each automated action be traceable to its evidence and reasoning, motivating structured logging of LLM traces. Third, human-in-the-loop calibration: human-AI teaming reduces alert fatigue without surrendering accountability, with several SOC studies reporting that the optimum is a tiered design in which routine responses are automated but novel ones escalate to human review [72,109,110].

6.3. Predictive Risk Assessment and Threat Hunting

Threat hunting transforms defensive posture from reactive to proactive. We compare the surveyed threat-hunting and CTI-extraction systems along three design dimensions: reasoning strategy (single-shot vs. multi-step / ReAct), tool and data integration (SIEM connectors vs. NER pipelines vs. threat-feed APIs), and evaluation context (open benchmark vs. internal study vs. vendor report); the comparison table that follows summarises the resulting matrix. Agentic threat hunting consumes hypotheses or MITRE ATT&CK techniques, generates hunt scripts, executes them against telemetry,

 doi:10.20944/preprints202606.0142.v1

and refines its hypothesis space based on results [41,42,142]. iThelma [42] integrates playbook-driven intelligence with autonomous script generation, validation, and adaptive learning. Policy-guided threat hunting integrates LLM-enabled hunts with Splunk SOC triage.

LLM-driven CTI extraction is the input to most hunts. CyberNER-LLM [70], KnowCTI [74], and segment-level CTI NER frameworks [113] extract entities and relations from unstructured threat reports with accuracies up to 98% on cybercrime-forum corpora. SynthCTI-style synthetic-data generation alleviates the class-imbalance problem for underrepresented MITRE techniques. LLM-based mapping of SIEM rules to ATT&CK techniques [43,148] (e.g., the Rule-ATT&CK Mapper framework) automates a step that previously consumed senior analyst time. Comparative analyses show that traditional ML still outperforms LLMs in raw mapping accuracy [148], but LLMs deliver explainability and scalability that ML cannot match.

Predictive risk assessment extends hunting forward in time. LLM-Powered Proactive Cyber-Defense [147] continuously ingests threat indicators from public platforms and ranks the resulting risks. Vulnerability prioritization frameworks combine LLM analysis of CVE descriptions with IoT-device documentation to triage patches [44,97].

Table 3. Comparison of representative agentic threat-hunting and CTI-extraction systems along three design dimensions. Cells reflect the predominant choice reported in each cited primary study; "hybrid" denotes systems that combine multiple categories.

System	Reasoning strategy	Tool/data integration	Evaluation context
iThelma [42]	Playbook-driven ReAct	SIEM + playbook DSL	Internal MTTR study; not open-benchmarked
CyberAlly [41]	Multi-step ReAct with analyst hand-off	SOC ticketing + EDR API	Vendor-reported MTTR delta
SynthCTI / KnowCTI [74]	Single-shot generation + retrieval	Threat-report corpora, NER pipelines	Cybercrime-forum NER, ∼98% F1
CyberNER-LLM [70]	Single-shot extraction, fine-tuned	Unstructured threat-report text	In-domain NER, accuracy up to 98%
Rule-ATT&CK Mapper[a] [142]	Single-shot classification	Snort/Suricata rule corpus + ATT&CK lattice	Mapping coverage on public rule sets
LLM-Powered Proactive CDF [147]	Continuous ingestion + ranking	X / social-media indicator streams	Indicator-volume study; rank-precision metrics
Vulnerability prioritization [44,97]	Retrieval-augmented scoring	CVE + IoT device docs	Triage rank quality on curated CVE samples

Design dimensions follow the comparative axes: *reasoning strategy* (single-shot vs. multi-step / ReAct vs. playbook-driven), *tool/data integration* (SIEM, NER pipelines, threat-feed APIs), and *evaluation context* (open benchmark vs. internal study vs. vendor report). Where the primary study does not disclose a design choice, the cell reports the closest available proxy. [a] "Rule-ATT&CK Mapper" is our synthesis name for the rule-to-ATT&CK mapping component of the LogRESP-Agent framework [142], not a system name used in the primary study.

6.4. Adversarial Deception and Honeypots

Traditional honeypots quickly betray themselves through static, low-fidelity responses that experienced attackers recognize within seconds. LLM-powered honeypots generate context-aware, dynamic responses that sustain attacker engagement orders of magnitude longer than their predecessors [14,69]. HoneyLLM [69] is a medium-interaction honeypot whose responses are generated by an LLM conditioned on protocol context. The adaptive deception architecture of Wang et al. [14] establishes three pillars — context-aware multi-modal dialogue, dynamic environment morphing, and automated artifact generation — that produce credible system fingerprints. LLM-LDAP honeypots and protocol-aware OT deception agents incorporate timing constraints, state transitions, and command semantics to convince attackers they are in real environments.

Cyber-deception's value extends beyond engagement. Each attacker interaction with an LLM-driven honeypot is a high-fidelity threat-intelligence artifact: the attacker's commands, lateral movement patterns, and tool fingerprints feed directly into the hunting and risk-assessment systems described above. The interplay of digital twins and cyber deception [68] suggests an emerging archi-

 doi:10.20944/preprints202606.0142.v1

tecture in which a digital twin of the production IoT environment serves both as a high-fidelity decoy and as a simulation testbed for response playbooks [46,47].

7. Datasets, Benchmarks, and Evaluation Methodology

7.1. IoT-Specific Datasets

The empirical foundation of the field rests on a small number of dataset families. CICIoT2023 [101] is the current de facto benchmark, providing a large-scale dataset of coordinated attacks on real IoT devices that captures DDoS, brute-force, spoofing, web, and reconnaissance attacks. Edge-IIoTset (2022) extends Industrial-IoT realism by including Modbus and SCADA-relevant protocols [102,114]. TON_IoT [17] provides heterogeneous telemetry combining network traffic, OS logs, and device sensor data. CICIoMT2024 [11,12,126,152] is the dedicated Medical-IoT benchmark, capturing 18 attack types across 40+ devices. BoT-IoT, N-BaIoT, and IoT-23 remain widely cited foundational datasets for botnet-specific research [22–27,138,139].

Each dataset has notable limitations. CICIoT2023 captures attacks at the network layer but provides limited firmware-level or application-layer evidence. Edge-IIoTset is comparatively small and biased toward specific industrial protocols. TON_IoT mixes synthetic and real traces in proportions that complicate generalization claims.

A field-wide methodological pattern bears explicit attention: detectors that achieve >99% accuracy on a single dataset routinely lose 10–20 percentage points when transferred cross-dataset, and we identified at least nine primary studies in the corpus that report this degradation when their model is evaluated on a second IoT dataset (the specific studies are enumerated in the bracketed citation list immediately following) [17,102,103,114,117,122,138,139,151]. The pattern is not unique to LLM-based detectors — it also affects deep-learning and graph-based IoT IDS — but it is amplified for LLM-based systems because in-context prompts and fine-tuning corpora can entrench dataset-specific shortcuts more strongly than gradient-only training. We argue that the field has been over-rewarding single-dataset benchmark accuracy and under-rewarding cross-dataset generalization, and that future work should report at least one cross-dataset transfer result as a default, not as a stretch goal. The Edge-IIoTset-to-CICIoT2023 transfer pair is a particularly defensible default because the two datasets have substantively different attack distributions and acquisition pipelines while sharing the same protocol vocabulary.

7.2. LLM- and Agent-Specific Benchmarks

LLM-specific evaluation is a younger and more turbulent area. CyberSecEval (versions 1–3) and CyberSOCEval, summarized in [93,95], benchmark LLM safety properties: insecure coding suggestions, cyberattack helpfulness, prompt-injection resistance, code-interpreter abuse, and autonomous offensive capability. All tested production models exhibit between 26% and 41% successful prompt-injection rates (CyberSecEval 3 against GPT-4 and Llama-3-70B respectively, as reported in [93,95]) [95], a baseline that no agentic defense currently approaches. SECURE benchmarks knowledge extraction, understanding, and reasoning across six Industrial Control System datasets [93]. The LLM-evaluation survey of Bansal et al. [95] catalogues over a dozen additional benchmarks but observes that none jointly evaluate detection accuracy, latency, and adversarial robustness in an integrated IoT setting.

We argue that agent-native IoT benchmarks must measure four properties simultaneously: (i) accuracy on labeled traffic; (ii) per-decision latency including tool invocations; (iii) robustness to adversarial prompt injection embedded in observations; (iv) cost in tokens, energy, or dollars per decision. No current benchmark meets all four criteria. The companion code repository (Section 10) takes a first step by implementing accuracy and latency on Edge-IIoTset with prompt-injection adversarial cases.

 doi:10.20944/preprints202606.0142.v1

7.3. Metrics and Reporting Practice

Reported metrics across the corpus include accuracy, precision, recall, F1, AUC, false-positive rate, MTTR, MTTD, energy consumption, and inference latency. Inference latency is the most variably reported: some studies report inference-only latency on idle hardware, others report end-to-end latency including network round trips, and few report tail-percentile distributions. We recommend that future work uniformly report at minimum the median, p95, and p99 end-to-end latencies under representative load, alongside the accuracy metrics that have historically dominated.

Table 4. Principal datasets and benchmarks used in the surveyed literature. Attack-class counts are taken from each dataset's published documentation; we did not re-benchmark or re-validate the labels.

Dataset	Year	Domain	Attack Classes	Key Use
CICIoT2023 [101]	2023	General IoT	33 attacks, 7 families	De facto IoT IDS benchmark
Edge-IIoTset	2022	IIoT	14 attacks (Modbus etc.)	Industrial IoT realism
TON_IoT [17]	2020	Heterogeneous IoT	9 attack types	Multi-modal: network + OS + sensor
CICIoMT2024	2024	Medical IoT	18 attacks across 40+ devices	Healthcare device security
BoT-IoT / N-BaIoT	2018	Botnet-specific IoT	Mirai, Bashlite, DDoS	Foundational botnet research
CyberSecEval 1–3 [93,95]	2023–25	LLM safety	Insecure coding, prompt injection, offensive	LLM cybersecurity capability
CyberSOCEval [93]	2025	LLM SOC tasks	Malware analysis, CTI reasoning	Defensive LLM benchmark

8. Open Challenges

8.1. Hallucination and Verifiable Reasoning

Hallucination — the generation of factually incorrect content with high linguistic confidence — is the dominant reliability obstacle for agentic security. In low-stakes settings, a hallucinated explanation produces only operator confusion; in high-stakes settings such as autonomous device quarantine, hallucination is potentially catastrophic. RAG grounding reduces but does not eliminate hallucination [40,120,121,143]; the Hughes Hallucination Evaluation Model and consistency-analysis tools provide post-hoc detection but cannot prevent generation. Hallucination-resistant security planning, structured-prompt frameworks for CoT integrity, and verifiable autonomous reasoning are now identified open problems [140,141].

8.2. Adversarial Robustness and Prompt Injection

Two adversarial threat models attack agentic IoT defenses. The first is the conventional model from non-LLM intrusion detection: adversaries craft traffic perturbations that evade ML classifiers [49,50,78,145]. Hierarchical adversarial attacks against GNN-based IoT IDS [49] and SHAP-attribution-based fingerprinting attacks demonstrate that ML defenders remain fragile. The second model is new with LLMs: prompt injection, in which adversaries embed instructions in observations to redirect the agent itself [10,60]. The indirect variant — embedding the malicious prompt inside third-party data that the LLM-integrated application later retrieves — was first systematically demonstrated against production deployments (Bing Chat, GPT-4 code completion, synthetic agents) by Greshake et al. at ACM AISec 2023 [156]. The earlier direct-prompt-injection concept was named by Perez and Ribeiro in a NeurIPS 2022 ML Safety Workshop Best Paper, but as that work has not appeared in a formal peer-reviewed proceedings we cite the peer-reviewed Greshake follow-up [156] as the foundational reference. Adversarial robustness of aligned models more broadly is investigated by Carlini et al. at NeurIPS 2023 [157], who demonstrate that aligned LLMs can be made to produce

 doi:10.20944/preprints202606.0142.v1

arbitrary outputs through adversarial inputs. A "foot-in-the-door" prompt-injection attack (a multi-turn variant borrowed from the social-psychology literature in which a benign initial request lowers the agent's guard for a subsequent malicious one) is one that initially asks the agent for a harmless action (e.g., "summarize this packet") and, once the agent has committed to a particular reasoning frame, escalates the request inside subsequent observations toward attacker-favored tool invocations. Such attacks demonstrate that even harmless-appearing observations can drift a ReAct agent toward malicious tool use. Defenses include polymorphic prompts [10], data-filter wrappers, tool dependency graphs, dynamic policy enforcement, and multi-agent defense pipelines, but no defense yet reduces successful prompt-injection rates below 10% on production-scale benchmarks.

Compound failure modes in multi-step agentic pipelines.

Beyond the single-turn attacks above, multi-step agentic pipelines exhibit compound failure modes that are largely under-evaluated in the surveyed literature. Three patterns recur across the corpus. *Error propagation:* a misclassification or hallucination in an early ReAct step biases all subsequent tool invocations, often in a way that is invisible to a SOC analyst who reviews only the final action; the BARTPredict false-positive cascades documented in [1] are a concrete instance, and the multi-agent vulnerability detection results in [2,6] report failure cascades when an early reasoning step misidentifies the language of the smart-contract source. *Context poisoning:* a prior-turn observation (for example, a maliciously crafted log line returned by a SIEM tool) contaminates the agent's reasoning across a long conversation, an effect distinct from one-shot prompt injection because the polluted token remains in context. *Tool-use amplification:* a hallucinated query produces a large result set that then overwhelms the agent's context window, forcing truncation that drops the salient evidence and biasing the subsequent decision; the multi-agent threat-hunting evaluations in [60,142] surface this pattern but do not measure it directly. Each of these failure modes is well within the scope of the prompt-injection robustness benchmarks discussed above, but is not captured by single-turn metrics such as CyberSecEval susceptibility rates; targeted multi-turn evaluation harnesses remain a gap.

8.3. Explainability and Auditability

Regulators in healthcare, automotive, and critical infrastructure increasingly require that automated decisions be explainable to affected parties. LLM agents emit natural-language reasoning that is human-readable but not necessarily faithful: the verbalized reasoning may not reflect the actual computation [127–131]. Faithful explanation, structured trace logging, and SHAP-style attribution on LLM internals remain open problems. The Federated XAI IDS approach [130] is a notable step toward jointly addressing explainability and privacy.

8.4. Privacy Preservation

IoT traffic frequently contains personally identifiable or commercially sensitive data; sending it to a cloud LLM for analysis is often legally or contractually impermissible. Federated learning [117,120,130] and differential privacy provide the orthogonal building blocks, but their integration into agentic systems is immature. Federated RAG (Federated RAG for IoT cybersecurity [120]) is one promising direction. On-device SLMs eliminate the data-leaving-device concern but lack the reasoning depth of cloud LLMs. Tradeoffs along the accuracy-efficiency-privacy trilemma are sharp and not yet well characterized [146].

8.5. Real-Time Decision-Making and the Latency-Security Paradox

IoT incident response has tens-of-milliseconds budgets for some scenarios (industrial control loops, medical devices) and tens-of-seconds budgets for others (DDoS mitigation, account compromise). Even the fastest edge SLMs operate at 70–400 ms per inference, and ReAct chains with two or three tool invocations push easily into the seconds regime. The latency-security paradox is fundamental: deeper reasoning produces better decisions but slower decisions, and slower decisions allow more attacker dwell time. Speculative decoding, quantization, and KV-cache-reuse optimizations partially close the

 doi:10.20944/preprints202606.0142.v1

gap [34,51,55] but cannot eliminate it. Hybrid architectures that delegate line-rate decisions to ML and only escalate to LLMs for ambiguous cases are, in our assessment, the most plausible near-term answer [8,57,142].

Table 5 synthesizes the accuracy-latency-robustness frontier reported across the corpus for the three representative deployment tiers. Reported accuracies are aggregated from the cited primary studies and should be interpreted as upper bounds achieved in laboratory settings; production deployments typically observe a 5–15 percentage-point degradation. Prompt-injection susceptibility figures are baselined against the CyberSecEval 3 reference rate of 26–41% for unprotected production LLMs [93,95], with cells labeled "unmeasured" where the underlying study did not include adversarial probes.

The "unmeasured" label for edge SLMs in Table 5 also masks a structural trade-off that the Section 9.4 roadmap milestone (sub-5% prompt-injection success) must confront. Smaller models have less capacity for the multi-stage safety alignment (supervised fine-tuning + RLHF + red-team fine-tuning) that hardens cloud APIs against prompt injection, and 4-bit quantization can additionally erase fragile safety-aligned weights — empirically degrading injection robustness by single-digit to low-double-digit percentage points relative to the un-quantized base model in published evaluations of quantization-induced safety drift. Edge SLMs therefore start from a weaker safety baseline than the cloud LLMs in Table 5 row 3, and rely disproportionately on system-prompt scaffolding and tool-call gating (cheaper but more brittle) rather than on weight-level alignment. Closing the gap to the Section 9.4 milestone will likely require alignment techniques specifically engineered to survive post-training quantization.

8.6. Governance and Accountability

When an autonomous agent quarantines a hospital ventilator or blocks emergency communications based on a hallucinated threat, who is responsible? Existing legal and operational frameworks assume a human decision-maker. Three regulatory instruments published since 2023 are now beginning to shape the answer for agentic IoT defense and deserve explicit consideration. The U.S. National Institute of Standards and Technology AI Risk Management Framework (NIST AI RMF, NIST AI 100-1) [159] defines a Govern-Map-Measure-Manage cycle that agentic systems should be evaluated against; its Generative AI Profile (NIST AI 600-1, 2024) [159] is particularly relevant because it articulates the trust, transparency, and incident-response expectations applicable to LLM-driven decisions. The European Union Artificial Intelligence Act (Regulation (EU) 2024/1689) [160] imposes risk-tier-specific obligations on high-risk AI systems, a category that explicitly includes AI components used in critical infrastructure protection — squarely the deployment posture this survey addresses. The European Union Agency for Cybersecurity (ENISA) has published guidance on securing the IoT supply chain and on cybersecurity practices for AI [161] (notably its 2020 "Guidelines for Securing the Internet of Things") that operationalizes many of the same requirements for European deployments. None of the surveyed primary studies in our corpus explicitly maps its proposed agentic defense to these frameworks; closing that gap is a near-term governance research priority.

Recent agentic AI governance proposals additionally advocate meta-cognitive judgment functions that gate decision readiness and calibrate autonomy based on evidence quality. Identity, authentication, and authorization for agent-to-agent communication (the so-called "Internet of Agents") remain underspecified. The governance gap is the largest non-technical obstacle to industrial deployment, and Section 9 returns to it through the lens of trustworthy multi-agent collaboration.

Worked example: EU AI Act risk-tier classification for an autonomous quarantine agent

A concrete application of Regulation (EU) 2024/1689 to two representative agentic IoT defenses illustrates the practical stakes. A *detection-only* LLM-augmented IDS that emits alerts to a SOC analyst (the predominant pattern documented in Section 5.3.1 and tabulated in Table 2) is most plausibly classified as a *limited-risk* system: transparency obligations apply (operators must inform users that AI is in the loop), but the conformity-assessment burden is light. By contrast, an *autonomous quarantine*

 doi:10.20944/preprints202606.0142.v1

agent acting on hospital ventilators, industrial control PLCs, or smart-grid relays — i.e., the multi-agent autonomous-response systems catalogued at the upper-right of the action-scope dimension — is almost certainly a *high-risk AI system* under Annex III of the Act (categories "critical infrastructure" and, for medical IoT, "safety component of products covered by Union harmonisation legislation"). The provider is then bound by Articles 8–17: risk-management system, data-governance documentation, technical-documentation file, automatic logging, transparency to deployers, human-oversight design, accuracy/robustness/cybersecurity testing, and a quality-management system. None of the surveyed primary studies report compliance artifacts of this kind. The implication for the field is concrete: agentic IoT defenses targeting high-risk Annex III categories cannot be deployed in the EU after August 2026 without the listed conformity evidence, and research papers proposing such systems should now report the corresponding artifacts (or an explicit statement that the system is for research use only).

Mapping the NIST AI RMF to the four-pillar taxonomy.

The NIST AI RMF's Govern–Map–Measure–Manage cycle maps naturally onto the four-pillar taxonomy introduced in Section 5: *Govern* corresponds to the deployment-topology decision and to the human-oversight gate; *Map* corresponds to the action-scope choice (what the agent is authorised to do and on which assets); *Measure* corresponds to the empirical evaluation discussed in Section 7 (accuracy, latency, prompt-injection susceptibility); and *Manage* corresponds to the reasoning-strategy choice (whether the agent reflects, self-checks, or escalates on low-confidence decisions). This mapping is not present in any surveyed study but provides a structured way to read the four-pillar coordinate of a proposed system as a compliance artifact rather than a purely engineering choice.

8.7. Ethical Considerations

Agentic AI for IoT cybersecurity sits squarely on the offensive-defensive boundary, and a survey that takes the field seriously must treat that boundary explicitly. Three categories of ethical risk recur across the corpus. First, dual-use research: the same ReAct loops, plan-and-solve agents, and multi-agent coordination patterns that improve defense are directly usable for unauthorized penetration testing, autonomous exploitation, and red-team automation [2,7,61–64,106]. Public release of high-capability offensive frameworks therefore demands more careful gating than purely defensive contributions. Second, deception ethics: LLM-powered honeypots deceive not only adversaries but also legitimate researchers, vendors, and — in adjacent failure modes — human operators who interact with the deception unintentionally [14,68,69]; honeypot deployments should include legal disclaimers, sandbox isolation, and engagement logs that allow accidental human interaction to be detected and unwound. Third, autonomy and human-in-the-loop calibration: deploying response-orchestration agents that can quarantine medical or life-critical devices is ethically defensible only when the action is gated by an explicit human approval workflow for tier-1 and tier-2 assets, as we have encoded in the response-orchestration prompt template of the companion repository. We adopt the position that agentic IoT defense should default to a "human approves consequential action" posture and that deviation from this default should be justified, documented, and reversible.

Institutional release-ethics implications for the companion kit.

The companion reproducibility kit released with this survey (Section 10) contains reference single- and multi-agent loops, an adversarial prompt-injection probe corpus, and an Edge-IIoTset-style evaluation harness. All three components have a dual-use surface: the agentic loops can be re-pointed from defensive to offensive workflows by editing the system prompt; the probe corpus can be used to attack defended systems as well as to harden them; and the synthetic-traffic generator can be used to construct evasive samples for adversarial training. We adopted three institutional-style mitigations rather than treating the release as a pure-research artifact. First, the default system prompts and tool whitelists in the kit are restricted to defensive postures (read-only telemetry, alerting, ticketing) and explicitly exclude offensive primitives (no shell execution, no remote-code-execution helpers,

 doi:10.20944/preprints202606.0142.v1

no exploitation libraries). Second, the prompt-injection probe corpus is shipped with a disclosed provenance map (see Section 10.3) so that downstream users can identify which patterns originated in published red-team work and which are author-generated, allowing reviewers to assess novelty risk. Third, the README explicitly documents the dual-use surface and states that any redistribution under a more permissive offensive posture should pass through the redistributor's own institutional review. We considered but rejected withholding the kit pending case-by-case review, because the underlying techniques are already published in the surveyed primary studies and an open scaffolding kit improves the field's ability to study agent failure modes empirically. The trade-off is documented openly so future work can revisit it.

Table 5. Reported accuracy–latency–robustness frontier across deployment tiers. Latencies are p50 inference-plus-tool-call times; accuracies are best-reported numbers on the named benchmarks; prompt-injection numbers reference CyberSecEval baselines [93,95] when the primary study did not measure them directly. Cells in italics marked with an asterisk (*) denote extrapolated or unmeasured values; all other cells are direct measurements from the cited primary literature. Important caveat: accuracy figures are best-reported peak values on the named benchmark dataset (parenthetical citation); the rows are not directly comparable because the benchmarks (Edge-IIoTset, CICIoT2023, others) differ in attack distributions, class balance, and difficulty (see Section 7.1 on the 10–20 percentage-point cross-dataset degradation we document for this corpus). Latency figures pool across heterogeneous hardware platforms (Raspberry Pi-class, Cortex-A78-class, x86 gateway, datacenter GPU) and measurement conventions (inference-only vs. end-to-end including tool calls); readers should consult the cited primary studies for the specific hardware and measurement protocol of each reported value.

Tier	Model class	p50 latency	Best-reported acc. (dataset)	Prompt-injection susceptibility
Edge / on-device	Distilled BERT, LLaMA-3.2-1B, SLM (4-bit)	70–400 ms	~99% on Edge-IIoTset [79,104]	Largely unmeasured*; CyberSecEval baseline 26–41% [93,95]
Fog / gateway	7–13 B params (Mistral, Llama-3-8B)	300–900 ms (incl. 1 tool call)	~98% on CICIoT2023 [8,57,142]	25–40% on production probes*
Cloud-centric	70B+ params, frontier API models	1–5 s end-to-end	~98% prediction acc. [1]; >90% MA-IDS [4]	10–30% with polymorphic-prompt defense [10]

9. A 2026 Research Roadmap

Building on the gaps identified above, we articulate four research directions that we believe will deliver high-priority progress in the next 24 months. The numeric milestones cited within each subsection are aspirational targets, derived where possible from application-domain analogues (industrial safety integrity levels SIL-2/3 for autonomous response, NIST SP 800-82 availability targets for industrial control telemetry); where no direct standards analogue exists, thresholds are calibrated against the closest available benchmark and should be read as design goals rather than as requirements derived from a published standard.

9.1. Federated Agentic Learning

Federated learning, originally formulated as the FedAvg algorithm by McMahan et al. at AISTATS 2017 [158], has matured for conventional ML; federated agentic learning is in its infancy. The proposed research agenda includes: (i) federated fine-tuning of small language models with differential privacy (DP-LoRA and FedShield-LLM directions provide initial techniques); (ii) federated RAG retrieval indices that share threat-intelligence embeddings without sharing raw traffic [120]; (iii) federated multi-agent training in which the coordination policy itself is updated via federated reinforcement learning. Milestones include reaching DistilBERT-grade accuracy on Edge-IIoTset under (ε=1, δ=1e-5)

 doi:10.20944/preprints202606.0142.v1

differential privacy with bandwidth costs no more than 2× non-private federated learning on Edge-IIoTset. The ε=1 threshold mirrors the strictest commonly adopted budget in production federated-learning deployments [120,130], and the 2× bandwidth ceiling matches the overhead tolerated by gateway-class hardware in fog-tier IoT pilots.

9.2. Verifiable Autonomous Reasoning

Verifiable reasoning means that an agent's claimed reasoning faithfully tracks its actual computation and is checkable post-hoc. The research agenda spans: (i) trace-based interpretability that records intermediate tool invocations, tokens, and confidence scores in structured, queryable form; (ii) faithfulness benchmarks that compare verbalized rationale to causal attribution on hidden states; (iii) cryptographic attestation of agent execution so that operators can prove which model, prompt, and tools generated a specific decision. The recent meta-cognitive judgment proposals and consistency-analysis frameworks are first steps. Milestones include sub-10% verbalization-versus-attribution divergence on the SECURE benchmark [93]. The 10% threshold is chosen because audit trails accepted as evidence in safety-critical regulated industries (medical, aviation, finance) typically require divergence rates of the same order, providing a defensible upper bound on tolerable LLM rationalization drift.

9.3. Trustworthy Multi-Agent Collaboration

Multi-agent systems require formalized communication protocols, mutual authentication, and emergent-behavior monitoring. The research agenda includes: (i) standardized agent communication schemas (extending the Model Context Protocol and Agent Network Protocol ideas) with cryptographic identity per agent; (ii) detection of intention-hiding malicious agents in collaborative pipelines; (iii) byzantine-fault-tolerant multi-agent consensus for security-critical decisions. Milestones include detecting at least 95% of injected malicious agents in benchmarks of 10+-agent systems without degrading benign throughput by more than 5%. The 95%/5% pairing is calibrated against the byzantine-fault-tolerance literature, where consensus protocols typically guarantee correctness only when fewer than one-third of participants are byzantine — agentic systems with weaker detection cannot meet that classical bound.

Cross-vendor device communication is intrinsic to IoT ecosystems, so multi-agent interoperability standards must adapt to edge constraints before they can serve the deployment topologies catalogued in Section 5.4. We highlight three concrete adaptations of MCP/ANP/A2A that the field should prioritize. (i) Lightweight wire formats: the JSON-RPC payloads typical of current MCP implementations carry 2–5× more bytes than necessary for IoT control planes. Adapter profiles that bind MCP semantics over Concise Binary Object Representation (CBOR, RFC 8949), MessagePack, or Protocol Buffers reduce serialization cost to fit the kilobit-per-second link budgets typical of LoRaWAN and NB-IoT. (ii) Latency-aware orchestration: agent communication budgets must be expressed alongside QoS classes (hard-real-time, soft-real-time, best-effort) so that an industrial-control closed loop is not gated on a multi-hop agent debate. A practical milestone is sub-50 ms median agent-to-agent message round-trip on a constrained network, comparable to the inner-loop reaction times Section 8.5 references. (iii) Capability discovery over IoT-native transports: ANP-style discovery is naturally MQTT-pub/sub or CoAP-multicast at the edge, not HTTP-pull; a profile binding ANP discovery primitives over MQTT topics would let an IoT defender enumerate fellow defensive agents without round-tripping through a cloud broker. Closing these three gaps would convert today's loose vendor-by-vendor agent ecosystems into the disciplined multi-agent fabric that trustworthy IoT defense requires.

9.4. Resource-Hardened Edge Agents

Edge agents must run on a Raspberry Pi-class device while resisting prompt injection and side-channel attacks. The research agenda includes: (i) speculative decoding tuned for IoT-traffic distributions; (ii) quantization-aware fine-tuning that preserves prompt-injection robustness even at 4-bit precision; (iii) hardware-isolated tool execution (trusted execution environments) that constrain

 doi:10.20944/preprints202606.0142.v1

what an agent can do even when reasoning is compromised; (iv) energy-aware reasoning loops that adapt depth based on remaining battery. Milestones include sub-100 ms median end-to-end latency on Arm Cortex-A78 hardware — measured as single-decision wall-clock time without any external tool round-trip — while sustaining prompt-injection success rates below 5% on a 1000-example adversarial benchmark. The 100 ms figure aligns with the upper bound for industrial-control closed-loop reaction times, and the 5% prompt-injection ceiling is a tenth of the 26–41% CyberSecEval baseline for production LLMs [93,95], representing meaningful hardening without claiming an unrealistic zero.

These four directions are not independent. A federated agentic system whose updates are verifiable and whose multi-agent coordination is trustworthy and whose edge components are resource-hardened constitutes the union of the four. The companion code artifact provides scaffolding that future work can extend along each axis.

10. Companion Reproducibility Kit

Alongside this manuscript we provide a structured reproducibility kit — prompt templates, reference reasoning loops, and an evaluation harness — that operationalizes the four-pillar taxonomy of Section 5. The kit is positioned as scaffolding for follow-up empirical work rather than as a finished framework: it deliberately implements the smallest end-to-end version of each component so that other researchers can replace any single piece (model backend, tool registry, dataset) without touching the others. The materials are publicly available on GitHub at https://github.com/vnageshwaran-de/agentic-iot-security and archived on Zenodo with permanent DOI 10.5281/zenodo.20446651 (release v0.2.0) [165]; the deposit includes the prompt templates, the reference Python implementation, the synthetic dataset generator, and the present manuscript's coded extraction matrix.

10.1. Prompts Engine

Each action scope from Section 5.3 — anomaly interpretation, response orchestration, threat hunting, vulnerability discovery, deception — is paired with a parameterizable system-prompt template. Templates expose slots for (a) the IoT context (protocols, device classes, network topology), (b) the available tools (in JSON-schema form following current function-calling conventions), and (c) the safety scaffolding (refusal policies, reflection prompts, confidence thresholds). JSON schemas are provided for canonical tools — SIEM query, packet sniff, MITRE lookup, device-isolate, threat-feed-lookup — so practitioners can drop in their own implementations without rewriting prompts.

10.2. Core Loops

Two reference loops are provided. The single-agent ReAct loop implements perception, thought, tool invocation, observation, and termination, parameterized by model backend (with adapters present in the v0.2.0 Zenodo release for OpenAI-compatible APIs, Anthropic Claude, and local llama.cpp inference) and by a pluggable tool registry. The multi-agent coordinator implements three patterns (pipeline, debate, blackboard) over a shared message bus, with explicit roles, hand-off rules, and inter-agent input sanitization to mitigate prompt injection. Both loops are instrumented with structured tracing.

10.3. Evaluation Harness

The evaluation harness measures classification accuracy and end-to-end latency on synthetic Edge-IIoTset-style flow data; a switch in the dataset loader allows substitution of the real Edge-IIoTset CSV when an appropriate license is in place. A complementary adversarial harness injects prompt-injection probes (drawn from a corpus of 25 patterns) into observations and reports the agent's susceptibility. Results are emitted as JSON and a summary CSV for downstream statistical analysis.

Provenance of the 25 prompt-injection probe patterns.

The probe corpus is assembled as follows: 12 patterns are derived from the public CyberSecEval 3 indirect-prompt-injection split [93,95] (rephrased to fit IoT-telemetry observation shapes); 6 patterns

 doi:10.20944/preprints202606.0142.v1

adapt indirect-prompt-injection vectors from Greshake et al. [156]; and 7 patterns are author-generated to cover IoT-specific surfaces not present in either source (e.g., MQTT topic-name injection, JSON-schema confusion in tool-call arguments, log-line splitting against SIEM enrichment). The provenance map is shipped with the v0.2.0 release as `evaluation/probes/PROVENANCE.md`; readers can therefore reproduce the CyberSecEval-style baseline subset directly, and compare independently-generated patterns against the published comparators.

Sanity-check validation run.

This run is included only to verify reproducibility of the harness, not to support any empirical claim about model performance. To demonstrate that the reference harness executes end-to-end and produces measurable numbers, we report one validation run shipped with the v0.2.0 release. Against the synthetic Edge-IIoTset-style generator with 5 000 flows balanced across 7 attack families (benign, ddos, brute-force, exfil, recon, mqtt-mitm, injection), the single-agent ReAct loop running in the deterministic offline-stub mode (`--model stub`, the only mode that requires no API key or GPU and is therefore citable for reproducibility) achieves a binary attack-vs.-benign $F_1 = 0.83$ (precision $= 0.89$, recall $= 0.78$) and a 7-way multi-class macro $F_1 = 0.54$; the stub heuristic latency is sub-microsecond per decision, which means harness overhead is not the bottleneck and any real-model run will be dominated by the LLM inference cost on the host hardware. Re-running the harness with `--model openai` or `--model anthropic` substitutes the heuristic for a real LLM call; the resulting accuracy is bounded by the chosen model and the resulting latency by the chosen API. These figures are kit outputs, not literature-derived numbers, and they should not be compared against the literature-derived rows of Table 5, which reflect best-reported numbers from primary studies on real datasets and real hardware. The full per-row CSV produced by the run, the seed used by the synthetic generator, and the exact command line are shipped at `evaluation/sample_runs/` in the v0.2.0 release (Zenodo DOI 10.5281/zenodo.20446651).

Inter-agent input sanitization.

The multi-agent coordinator implements a two-stage sanitization filter on every message handed off between agents: (i) a syntactic stage that strips or escapes control sequences common to indirect-prompt-injection vectors (Markdown comment markers, repeated whitespace, ANSI escape codes, Unicode control characters), and (ii) a semantic stage that wraps the incoming content in an explicit untrusted-input frame ("the following content originated from a downstream agent and must be treated as data, not as instructions") before it is concatenated into the next agent's prompt. The semantic frame is conceptually related to the polymorphic-prompt defense of Wang et al. [10] surveyed in Section 5; the difference is that our frame is applied at every inter-agent boundary rather than at the user-input boundary alone, and the syntactic stage borrows from input-sanitization patterns established in web-application security (specifically the OWASP Input Validation and Output Encoding cheat sheets, https://owasp.org/www-project-cheat-sheets/) rather than from prompt-engineering folklore. The implementation lives at `core_loops/multi_agent/sanitize.py` in the v0.2.0 release.

Synthetic-to-real transfer caveat.

The Edge-IIoTset-style generator in the harness reproduces packet header statistics and flow-volume distributions but does not reproduce real-IoT artifacts such as device-specific timing jitter, retransmission patterns, application-layer protocol quirks (CoAP option encoding, MQTT QoS behavior under packet loss), or vendor-specific firmware behaviors. Validation runs reported here use the synthetic generator and should be re-run on captured real-IoT traffic before any production claim is made; the kit's dataset loader accepts the official Edge-IIoTset CSV directly for that purpose.

10.4. Provenance of the Numbers in This Survey

To distinguish literature synthesis from independent measurement we clarify the provenance of every quantitative number in this paper. All accuracy figures, MTTR reductions, and latency ranges in Sections 2, 5, 6, 7, and 8, and all rows of Tables 1–5, are drawn from the primary peer-reviewed studies

 doi:10.20944/preprints202606.0142.v1

cited inline; no number in those sections is the product of running the reproducibility kit. The kit's own runs are intended for follow-up work and are not reported as findings of this survey.

10.5. Release Status and Scope

The kit is released in its initial v0.2.0 scaffolding form, publicly accessible on GitHub at https://github.com/vnageshwaran-de/agentic-iot-security and permanently archived on Zenodo with DOI 10.5281/zenodo.20446651 [165]. The materials described in Section 10.1–10.3 are specifications and runnable reference Python implementations sufficient to reproduce the described workflows on a single machine; they are deliberately not a fault-tolerant distributed system, a packaged PyPI module, or a hosted service. The Zenodo archive carries the canonical DOI, MIT license, and citation metadata (CITATION.cff); GitHub topics tag the repository as agentic-ai, llm-agents, iot-security, prompt-injection, edge-computing, react-agent, multi-agent-systems, systematic-survey, and prisma for discoverability. Readers are encouraged to fork the kit, replace components with production-grade equivalents, and report empirical results against the Section 9 roadmap milestones in subsequent peer-reviewed work.

11. Conclusions

The convergence of large language models and agentic AI with IoT cybersecurity has, in the four years since 2022, evolved from speculative proposal to a substantial research literature spanning 150-plus peer-reviewed studies in IEEE Xplore, the ACM Digital Library, and MDPI journals. This survey has synthesized that literature along a four-pillar taxonomy — agent architecture, reasoning strategy, action scope, and deployment topology — that locates each study in a shared coordinate system and surfaces the under-explored regions of the design space. We have examined the four flagship application domains (anomaly interpretation, response orchestration, predictive risk assessment, and adversarial deception), consolidated the datasets and benchmarks on which the field empirically rests, and dissected the open challenges that constrain present-day deployment: hallucination, adversarial fragility against both traditional evasion and novel prompt injection, the latency-security paradox at the edge, privacy preservation under federated learning, and the largely unaddressed governance vacuum surrounding autonomous defensive action.

We have argued that the next 24 months of research should concentrate on four directions whose intersection would constitute the foundation of a trustworthy self-defending IoT ecosystem: federated agentic learning, verifiable autonomous reasoning, trustworthy multi-agent collaboration, and resource-hardened edge agents. To accelerate progress on these directions, we have released a companion open-source framework that operationalizes the taxonomy through prompt templates, single- and multi-agent reasoning loops, and an evaluation harness with adversarial-prompt-injection probes against Edge-IIoTset traffic.

A field-wide methodological observation also recurs from the synthesis: present-day evaluation practice over-rewards single-dataset benchmark accuracy and under-rewards cross-dataset generalization. The 10–20 percentage-point degradation we observe across at least nine primary studies (Section 7.1) is not an idiosyncratic property of any one detector; it is a structural feature of the way the field benchmarks. Future work should make cross-dataset transfer reporting a default expectation rather than a stretch goal.

For IoT product teams considering near-term adoption, the synthesis suggests a tiered posture. The literature suggests that hybrid edge-ML detectors with cloud-LLM triage represent the most mature near-term deployment pattern and consistently deliver the largest false-positive reductions in the corpus. LLM-driven SOC automation — alert enrichment, MITRE ATT&CK mapping, playbook drafting — appears sufficiently mature for guarded production evaluation behind a human-in-the-loop gate, especially for tier-1 and tier-2 critical assets. Autonomous response orchestration for non-critical assets (rate limiting, ticketing, low-blast-radius blocks) may be suitable for guarded pilot evaluation within a 12-month deployment horizon. Fully autonomous quarantine of life-critical or industrial-control assets, multi-agent threat-hunting swarms, and unsupervised vulnerability disclosure remain research-only

 doi:10.20944/preprints202606.0142.v1

at the time of writing and should be avoided in production without dedicated governance, simulation environments, and rollback capabilities.

Agentic AI does not eliminate the IoT security problem; it shifts the problem. Self-defending IoT is not a destination but a moving frontier whose horizon is reset every time a new model, a new attacker capability, or a new deployment topology arrives. What this survey can offer is a map of the terrain as it stands in mid-2026, and a compass — the four-pillar taxonomy and the four research directions — for navigating what comes next.

Author Contributions: Conceptualization, V.N. and S.E.; methodology, V.N.; systematic literature search and PRISMA protocol, V.N.; taxonomy design, V.N. and S.E.; data extraction and synthesis, V.N.; companion code artifact, V.N.; writing—original draft preparation, V.N.; writing—review and editing, V.N. and S.E.; supervision and senior review, S.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The author thanks the open-access editorial efforts of IEEE Xplore, the ACM Digital Library, and MDPI for sustaining the peer-reviewed corpus on which this synthesis depends, and acknowledges the broader research community whose work — whether or not cited here — is the substrate of this field.

Conflicts of Interest: The author declares no conflicts of interest.

Data Availability Statement: The companion reproducibility kit described in Section 10 is publicly available on GitHub at https://github.com/vnageshwaran-de/agentic-iot-security and permanently archived on Zenodo with DOI 10.5281/zenodo.20446651 (release v0.2.0) [165]. No new primary data were generated; all empirical claims cite the underlying peer-reviewed studies.

References

1. Diaf, A.; Korba, A.A.; Karabadji, N.E.; Ghamri-Doudane, Y. BARTPredict: Empowering IoT Security with LLM-Driven Cyber Threat Prediction. In Proc. GLOBECOM 2024 — 2024 IEEE Global Communications Conference, Cape Town, South Africa, 08–12 December 2024. DOI: 10.1109/GLOBECOM52923.2024.10901770. https://ieeexplore.ieee.org/abstract/document/10901770/
2. Chhabra, A.; Datta, S.; Nahin, S.K.; Mohapatra, P. Agentic AI Security: Threats, Defenses, Evaluation, and Open Challenges. IEEE Access 2025. https://ieeexplore.ieee.org/document/11447227/
3. Petrovic, N.; Krstic, D.; Suljovic, S.; Hanczewski, S.; Glabowski, M. Agent-Based AI Approach to Security in IoT Systems Leveraging GenAI. In Proc. 2025 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 18–20 September 2025. https://ieeexplore.ieee.org/document/11197348/
4. Hmimou, Y.; Tabaa, M.; Khiat, A.; Hidila, Z. A Multi-Agent System for Cybersecurity Threat Detection and Correlation Using Large Language Models. IEEE Access 2025. https://ieeexplore.ieee.org/document/11141466/
5. Nguyen, N.T.V.; Childress, F.D.; Yin, Y. Debate-Driven Multi-Agent LLMs for Phishing Email Detection. In Proc. 2025 13th International Symposium on Digital Forensics and Security (ISDFS), 24–25 April 2025. https://ieeexplore.ieee.org/document/11012014/
6. Sun, J.; Sun, Y.; Liu, Y.; Wu, D.; Zhang, Z.; et al. Advanced Smart Contract Vulnerability Detection via LLM-Powered Multi-Agent Systems. IEEE Trans. Softw. Eng. 2025, October 2025. https://ieeexplore.ieee.org/abstract/document/11121619
7. Loevenich, J.F.; Lopes, R.R.F.; et al. Agentic Generative AI for Automation of Cyber Security Attack Chains in Tactical MANETs. In Proc. 2025 IEEE 50th Conference on Local Computer Networks (LCN), 13–16 October 2025. https://ieeexplore.ieee.org/document/11146384/
8. Mercan, Ö.B.; Toprak, A.G.; Osmanca, M.S. Multi-Class Classification for IoT Attack Detection: An LLM-Based Approach. In Proc. 2025 33rd Signal Processing and Communications Applications Conference (SIU), Şile, Istanbul, Türkiye, 25–28 June 2025. https://ieeexplore.ieee.org/document/11112284/
9. Obradov, A.; Pavković, B.; Stojanović, D.; Četić, N. Applying Multi-Agent LLMs to Threat Analysis and Risk Assessment in Automotive Security. In Proc. 2025 33rd Telecommunications Forum (TELFOR), 25–26 November 2025. https://ieeexplore.ieee.org/document/11314224/
10. Wang, Z.; Nagaraja, N.; Zhang, L.; Bahsi, H.; Patil, P.; Liu, P. To Protect the LLM Agent Against the Prompt Injection Attack with Polymorphic Prompt. In Proc. 2025 55th Annual IEEE/IFIP International

 doi:10.20944/preprints202606.0142.v1

Conference on Dependable Systems and Networks — Supplemental Volume (DSN-S), 23–26 June 2025. https://ieeexplore.ieee.org/document/11068353/

11. Kavkas, N.C.; Yildiz, K. Enhancing IoMT Security with Deep Learning Based Approach for Medical IoT Threat Detection. In Proc. 2025 13th International Symposium on Digital Forensics and Security (ISDFS), 24–25 April 2025. https://ieeexplore.ieee.org/document/11012062/
12. Otoum, Y.; Singh, P.; Nayak, A. Advancing IoMT Defenses: Deep Collaborative Learning for Robust Healthcare Security. In Proc. GLOBECOM 2024 — 2024 IEEE Global Communications Conference, 08–12 December 2024. https://ieeexplore.ieee.org/document/10901705/
13. Saheed, Y.K.; Arowolo, M.O. Efficient Cyber Attack Detection on the Internet of Medical Things-Smart Environment Based on Deep Recurrent Neural Network and Machine Learning Algorithms. IEEE Access 2021, 9, 161546–161577. https://ieeexplore.ieee.org/document/9617609/
14. Érsok, M.; Balogh, Á.; Kail, E.; Bánáti, A. Adaptive Deception Architectures: Conceptual Foundations for LLM-Powered Honeypot Systems. In Proc. 2025 IEEE 19th International Symposium on Applied Computational Intelligence and Informatics (SACI), 19–24 May 2025. https://ieeexplore.ieee.org/document/11030110/
15. Quraishi, A.; Rusho, M.A.; Prasad, A.; Keshta, I.; Rivera, R.; Bhatt, M.W. Employing Deep Neural Networks for Real-Time Anomaly Detection and Mitigation in IoT-Based Smart Grid Cybersecurity Systems. In Proc. 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), 26–27 April 2024. https://ieeexplore.ieee.org/document/10548160/
16. Aljuhani, A.; Kumar, P.; Alanazi, R.; Albalawi, T.; Taouali, O.; Islam, A.K.M.N. A Deep-Learning-Integrated Blockchain Framework for Securing Industrial IoT. IEEE Internet Things J. 2024, 11(5), 7817–7827. https://ieeexplore.ieee.org/document/10254517/
17. Zachos, G.; Essop, I.; Mantas, G.; Porfyrakis, K.; Ribeiro, J.C.; Rodriguez, J. Generating IoT Edge Network Datasets based on the TON_IoT Telemetry Dataset. In Proc. 2021 IEEE 26th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 25–27 October 2021. https://ieeexplore.ieee.org/document/9617799/
18. Wang, J.; Yue, Y.; Hu, D.; Li, Q.; Li, J.; Zhu, W. LLM Assisted Dual-View Awareness Framework for Smart Contract Vulnerability Detection. In Proc. 2025 IEEE 36th International Symposium on Software Reliability Engineering (ISSRE), 21–24 October 2025. https://ieeexplore.ieee.org/document/11229499/
19. James, M.; Newe, T.; O'Shea, D.; O'Mahony, G.D. Authentication and Authorization in Zero Trust IoT: A Survey. In Proc. 2024 35th Irish Signals and Systems Conference (ISSC), 13–14 June 2024. https://ieeexplore.ieee.org/document/10603175/
20. Meng, L.; Huang, D.; An, J.; Zhou, X.; Lin, F. A Continuous Authentication Protocol without Trust Authority for Zero Trust Architecture. China Commun. 2022, 19(8), 198–213. https://ieeexplore.ieee.org/document/9861234/
21. Aleisa, M.A.; et al. Blockchain-Enabled Zero Trust Architecture for Privacy-Preserving Cybersecurity in IoT Environments. IEEE Access 2025, 13, January 2025. https://ieeexplore.ieee.org/abstract/document/10839415
22. Rabhi, S.; Abbes, T.; Zarai, F. IoT Botnet Detection Using Deep Learning. In Proc. 2023 International Wireless Communications and Mobile Computing (IWCMC), 19–23 June 2023. https://ieeexplore.ieee.org/document/10182422/
23. Rabhi, S.; Abbes, T.; Zarai, F. Transfer Learning-based Mirai Botnet Detection in IoT Networks. In Proc. 2023 International Conference on Innovations in Intelligent Systems and Applications (INISTA), 20–23 September 2023. https://ieeexplore.ieee.org/document/10310379
24. Sharma, A.; Babbar, H. IoT-POT: Machine Learning-based Detection of Mirai Botnet Attacks in IoT. In Proc. 2024 First International Conference on Innovations in Communications, Electrical and Computer Engineering (ICICEC), 24–25 October 2024. https://ieeexplore.ieee.org/document/10808530
25. McDermott, C.D.; Majdani, F.; Petrovski, A.V. Botnet Detection in the Internet of Things using Deep Learning Approaches. In Proc. 2018 International Joint Conference on Neural Networks (IJCNN), 08–13 July 2018. https://ieeexplore.ieee.org/document/8489489/
26. Gandhi, R.; Li, Y. Comparing Machine Learning and Deep Learning for IoT Botnet Detection. In Proc. 2021 IEEE International Conference on Smart Computing (SMARTCOMP), 23–27 August 2021. https://ieeexplore.ieee.org/document/9556247/
27. Azhari, R.G.; Suryani, V.; Pahlevi, R.R.; Wardana, A.A. The Detection of Mirai Botnet Attack on the Internet of Things (IoT) Device Using Support Vector Machine (SVM) Model. In Proc. 2022 10th International Conference

 doi:10.20944/preprints202606.0142.v1

on Information and Communication Technology (ICoICT), 02–03 August 2022. https://ieeexplore.ieee.org/document/9914830/

28. Ahmed, A.; Shah, A.; Abdullah, A.; Laghari, S.U.A. Detection of DDoS Attacks in IoT Networks Using Machine Learning Algorithms. In Proc. 2024 IEEE 9th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 20–22 November 2024. https://ieeexplore.ieee.org/document/11120083/
29. Chen, Y.-W.; Sheu, J.-P.; Kuo, Y.-C.; Cuong, N.V. Design and Implementation of IoT DDoS Attacks Detection System based on Machine Learning. In Proc. 2020 European Conference on Networks and Communications (EuCNC), Dubrovnik, Croatia, 15–18 June 2020. https://ieeexplore.ieee.org/document/9200909/
30. Ashraf, A.; Elmedany, W.M. IoT DDoS Attacks Detection using ML Techniques: A Review. In Proc. 2021 International Conference on Data Analytics for Business and Industry (ICDABI), 25–26 October 2021. https://ieeexplore.ieee.org/document/9655789/
31. Doshi, R.; Apthorpe, N.; Feamster, N. Machine Learning DDoS Detection for Consumer Internet of Things Devices. In Proc. 2018 IEEE Security and Privacy Workshops (SPW), 24 May 2018. https://ieeexplore.ieee.org/document/8424629/
32. Bin Sa'idi, A.S.; Binti Jamil, A.M. IoT DDoS Attack Detection System Using ML Classification. In Proc. 2023 IEEE 21st Student Conference on Research and Development (SCOReD), 13–14 December 2023. https://ieeexplore.ieee.org/document/10563925/
33. Aysa, M.H.; Ibrahim, A.A.; Mohammed, A.H. IoT DDoS Attack Detection Using Machine Learning. In Proc. 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 22–24 October 2020. https://ieeexplore.ieee.org/document/9254703/
34. Agrawal, R.; Kumar, H.; Lnu, S.R. Efficient LLMs for Edge Devices: Pruning, Quantization, and Distillation Techniques. In Proc. 2025 International Conference on Machine Learning and Autonomous Systems (ICMLAS), 10–12 March 2025. https://ieeexplore.ieee.org/document/10968787/
35. Subalaxmi, T.; Akalya, R.; Kaviya, R.; Santhosh, V.; Praveen, R. Enhancing Cyber-Attack Detection in IoT Networks through Deep Reinforcement Learning. In Proc. 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), 18–19 April 2024. https://ieeexplore.ieee.org/document/10616565/
36. Gueriani, A.; Kheddar, H.; Mazari, A.C. Deep Reinforcement Learning for Intrusion Detection in IoT: A Survey. In Proc. 2023 2nd International Conference on Electronics, Energy and Measurement (IC2EM), 28–29 November 2023. https://ieeexplore.ieee.org/document/10419560/
37. Bakhshad, S.; Ponnusamy, V.; Annur, R.; Waqas, M.; Alasmary, H.; Tu, S. DRL-based IDS with Feature Selections Method and Optimal Hyper-parameter in IoT Environment. In Proc. 2022 International Conference on Computer, Information and Telecommunication Systems (CITS), 13–15 July 2022. https://ieeexplore.ieee.org/document/9832976/
38. Niyomdi, M.; Oluoch, J. A Comprehensive Security Orchestration, Automation, and Response System (SOAR) for Connected and Autonomous Vehicles. In Proc. 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), 05–07 February 2025. https://ieeexplore.ieee.org/document/10849039/
39. Kumar, P.V.; Pittala, R.B.; Jashwanth, P.; Vedamsh, R.; Chandrika, D.S.; Reddy, Y.V. Automated Threat Detection and Response Using SOAR and Endpoint Detection and Response. In Proc. 2025 OITS International Conference on Information Technology (OCIT), 18–20 December 2025. https://ieeexplore.ieee.org/document/11400041/
40. Bussari, S.; Punj, P.; Balakumar, G. RAGSec: Retrieval-Augmented Generation for Cybersecurity Threat Intelligence in Enterprise Networks. In Proc. 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC), 18–20 February 2026. https://ieeexplore.ieee.org/document/11395762/
41. Sheth, A.; Patel, A.; Upadhyay, C.; Ragothaman, H.; Patil, B.; Udayakumar, S.K. Agentic AI for Autonomous Cyber Threat Hunting and Adaptive Defense in Dynamic Security Environments. In Proc. 2025 IEEE International Conference on Electro Information Technology (eIT), 29–31 May 2025. https://ieeexplore.ieee.org/document/11103697/
42. Chen, N.; Lin, R.; Xie, D.; Lin, H.; Chen, S. iThelma: An Autonomous LLM Agent for Cyber Threat Hunting via Playbook-Driven Intelligence. In Proc. 2025 IEEE Conference on Communications and Network Security (CNS), 08–11 September 2025. https://ieeexplore.ieee.org/document/11195050/
43. Resul, E.; Turcanu, D.; Rughinis, R. A Comparative Analysis of LLMs in Mapping Malware Behaviors to MITRE ATT&CK Techniques from Textual Threat Intelligence Reports. In Proc. 2025 24th RoEduNet Conference: Networking in Education and Research (RoEduNet), 17–20 September 2025. https://ieeexplore.ieee.org/document/11208322/

 doi:10.20944/preprints202606.0142.v1

44. Ikegami, Y.; Negishi, R.; Hasegawa, K.; Hidano, S.; Fukushima, K.; Hashimoto, K. Prioritizing Vulnerability Assessment Items Using LLM Based on IoT Device Documentations. In Proc. 2024 11th International Conference on Internet of Things: Systems, Management and Security (IOTSMS), 02–05 September 2024. https://ieeexplore.ieee.org/document/10710294
45. Huang, Z.; Zhou, Z.; Song, L.; Deng, F. CID4IoT: IoT-Oriented Command Injection Vulnerability Detection Based on Critical Code Extraction and LLM Analysis. IEEE Internet Things J. 2026, 13(9). https://ieeexplore.ieee.org/document/11381604/
46. Xia, Y.; Dittler, D.; Jazdi, N.; Chen, H.; Weyrich, M. LLM experiments with simulation: Large Language Model Multi-Agent System for Simulation Model Parametrization in Digital Twins. In Proc. 2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA), 10–13 September 2024. https://ieeexplore.ieee.org/document/10710900/
47. Dietz, M.; Schlette, D.; Pernul, G. Harnessing Digital Twin Security Simulations for Systematic Cyber Threat Intelligence. In Proc. 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 27 June–01 July 2022. https://ieeexplore.ieee.org/document/9842689/
48. Villegas-Ch, W.; Govea, J.; Maldonado Navarro, A.; Palacios Játiva, P. Intrusion Detection in IoT Networks Using Dynamic Graph Modeling and Graph-Based Neural Networks. IEEE Access 2025, 09 April 2025. https://ieeexplore.ieee.org/document/10960408/
49. Zhou, X.; Liang, W.; Li, W.; Yan, K.; Shimizu, S.; Wang, K.I.-K. Hierarchical Adversarial Attacks Against Graph-Neural-Network-Based IoT Network Intrusion Detection System. IEEE Internet Things J. 2022, 9(12), 9310–9319. https://ieeexplore.ieee.org/document/9626144/
50. Zhang, C.; Costa-Pérez, X.; Patras, P. Adversarial Attacks Against Deep Learning-Based Network Intrusion Detection Systems and Defense Mechanisms. IEEE/ACM Trans. Netw. 2022, June 2022. https://ieeexplore.ieee.org/document/9674195/
51. Wang, Z.; Zhou, R.; Yang, S.; He, D.; Chan, S. A Novel Lightweight IoT Intrusion Detection Model Based on Self-Knowledge Distillation. IEEE Internet Things J. 2025, 12, 01 June 2025. https://ieeexplore.ieee.org/document/10851330
52. Wang, Y.; Yu, Z.; Wu, J.; Wang, C.; Zhou, Q.; Hu, J. Adaptive Knowledge Distillation-Based Lightweight Intelligent Fault Diagnosis Framework in IoT Edge Computing. IEEE Internet Things J. 2024, 11, 01 July 2024. https://ieeexplore.ieee.org/document/10496469/
53. Nie, F.; Liu, W.; Liu, G.; Gao, B.; Huang, J.; Yuen, C. Lightweight Identification of Malicious IoT Traffic via Cross-View Knowledge Distillation. IEEE Internet Things J. 2025, 12(20), 15 October 2025. https://ieeexplore.ieee.org/document/11114714/
54. Zhu, S.; Xu, X.; Zhao, J.; Xiao, F. LKD-STNN: A Lightweight Malicious Traffic Detection Method for IoT Based on Knowledge Distillation. IEEE Internet Things J. 2024, 11, 15 February 2024. https://ieeexplore.ieee.org/document/10236538/
55. Zhou, X.; et al. Reconstructed Graph Neural Network With Knowledge Distillation for Lightweight Anomaly Detection. IEEE Trans. Neural Netw. Learn. Syst. 2024, 35, September 2024. https://ieeexplore.ieee.org/document/10510656/
56. Asal, B.; Cakin, A.; Dilek, S. Enhancing Industrial IoT Cybersecurity with Explainable AI: A SHAP and LIME-Based Intrusion Detection Methodology. In Proc. 2025 7th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 23–24 May 2025. https://ieeexplore.ieee.org/document/11017105
57. Balasubramanian, P.; Seby, J.; Kostakos, P. Transformer-based LLMs in Cybersecurity: An in-depth Study on Log Anomaly Detection and Conversational Defense Mechanisms. In Proc. 2023 IEEE International Conference on Big Data (BigData), 15–18 December 2023. https://ieeexplore.ieee.org/document/10386976/
58. Ren, H.; Lan, K.; Sun, Z.; Liao, S. CLogLLM: A Large Language Model Enabled Approach to Cybersecurity Log Anomaly Analysis. In Proc. 2024 4th International Conference on Electronic Information Engineering and Computer Communication (EIECC), 27–29 December 2024. https://ieeexplore.ieee.org/document/10929078/
59. Zhang, J. Leveraging Large Language Models for Autonomous Threat Detection in IoT Networks. In Proc. EITCE '24: 2024 8th International Conference on Electronic Information Technology and Computer Engineering, ACM, 2024. https://dl.acm.org/doi/10.1145/3711129.3711223
60. He, F.; Zhu, T.; Ye, D.; Liu, B.; Zhou, W.; Yu, P.S. The Emerged Security and Privacy of LLM Agent: A Survey with Case Studies. ACM Comput. Surv. 2026, April 2026. https://dl.acm.org/doi/10.1145/3773080

 doi:10.20944/preprints202606.0142.v1

61. Huang, J.; Zhu, Q. PenHeal: A Two-Stage LLM Framework for Automated Pentesting and Optimal Remediation. In Proc. AutonomousCyber '24: Workshop on Autonomous Cybersecurity (co-located with CCS '24), ACM, 2024. https://dl.acm.org/doi/10.1145/3689933.3690831
62. Shen, X.; Wang, L.; Li, Z.; Chen, Y.; et al. PentestAgent: Incorporating LLM Agents to Automated Penetration Testing. In Proc. ASIA CCS '25: 20th ACM Asia Conference on Computer and Communications Security, ACM, 2025. https://dl.acm.org/doi/10.1145/3708821.3733882
63. Happe, A.; Cito, J. Getting Pwn'd by AI: Penetration Testing with Large Language Models. In Proc. ESEC/FSE 2023: 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ACM, 2023. https://dl.acm.org/doi/10.1145/3611643.3613083
64. Mei, J.; Chen, S.; Ma, Y.; Song, H. AutoPen: Towards Autonomous Penetration Testing Using LLM-Powered Agents. In Proc. CSAE '25: 9th International Conference on Computer Science and Application Engineering, ACM, 2025. https://dl.acm.org/doi/10.1145/3772886.3772899
65. Das, R.; Zhou, L.; Bi, S.; Wang, T.; Hou, T. Towards the Safety of Intelligent Transportation: A Survey on the Security Challenges and Mitigations in Internet of Vehicles (IoV). ACM Trans. Sens. Netw. 2026. https://dl.acm.org/doi/abs/10.1145/3786592
66. Gao, M.; Wu, L.; Li, Q.; Chen, W. Anomaly Traffic Detection in IoT Security Using Graph Neural Networks. J. Inf. Sec. Appl. 2023, 76, 103532. https://doi.org/10.1016/j.jisa.2023.103532 https://dl.acm.org/doi/10.1016/j.jisa.2023.103532
67. Ameer, S.; Praharaj, L.; Sandhu, R.; Bhatt, S.; Gupta, M. ZTA-IoT: A Novel Architecture for Zero-Trust in IoT Systems and an Ensuing Usage Control Model. ACM Trans. Priv. Sec. 2024. https://dl.acm.org/doi/10.1145/3671147
68. Heluany, J.; Amro, A.; Gkioulos, V.; Katsikas, S. Interplay of Digital Twins and Cyber Deception: Unraveling Paths for Technological Advancements. In Proc. EnCyCriS 2024: ACM/IEEE 4th International Workshop on Engineering and Cybersecurity of Critical Systems, ACM, 2024. https://dl.acm.org/doi/10.1145/3643662.3643955
69. Fan, W.; Yang, Z.; Liu, Y.; Qin, L. HoneyLLM: A Large Language Model-Powered Medium-Interaction Honeypot. In Proc. ICICS 2024: Information and Communications Security, Springer, 2024. https://dl.acm.org/doi/10.1007/978-981-97-8801-9_13
70. Liu, X.; Lin, W.; Ding, Z. CyberNER-LLM: Cyber Threat Intelligence Named Entity Recognition With Large Language Model. In Proc. ICICS 2025: 27th International Conference on Information and Communications Security, Nanjing, China, Springer, 2025. https://dl.acm.org/doi/10.1007/978-981-95-3543-9_28
71. Rosso, M.; Campobasso, M.; Gankhuyag, G.; Allodi, L. SAIBERSOC: A Methodology and Tool for Experimenting with Security Operation Centers. ACM Digit. Threats Res. Pract. 2022. https://dl.acm.org/doi/10.1145/3491266
72. Baruwal Chhetri, M.; Tariq, S.; Singh, R.; Jalalvand, F.; Paris, C.; Nepal, S. Towards Human-AI Teaming to Mitigate Alert Fatigue in Security Operations Centres. ACM Trans. Internet Technol. 2024. https://dl.acm.org/doi/10.1145/3670009
73. Al-Hawawreh, M.; Aljuhani, A.; Jararweh, Y. ChatGPT for Cybersecurity: Practical Applications, Challenges, and Future Directions. Cluster Comput. 2023. https://dl.acm.org/doi/10.1007/s10586-023-04124-5
74. Wang, G.; Liu, P.; Huang, J.; Bin, H.; Wang, X.; Zhu, H. KnowCTI: Knowledge-based Cyber Threat Intelligence Entity and Relation Extraction. Comput. Sec. 2024, 141, 103824. https://doi.org/10.1016/j.cose.2024.103824 https://dl.acm.org/doi/abs/10.1016/j.cose.2024.103824
75. Zacharis, A.; Gavrila, R.; Patsakis, C.; Douligeris, C. Optimising AI Models for Intelligence Extraction in the Life Cycle of Cybersecurity Threat Landscape Generation. J. Inf. Sec. Appl. 2025, May 2025, 104037. https://doi.org/10.1016/j.jisa.2025.104037 https://dl.acm.org/doi/10.1016/j.jisa.2025.104037
76. Saracino, A.; Simoni, M. Graph-Based Android Malware Detection and Categorization through BERT Transformer. In Proc. ARES 2023: 18th International Conference on Availability, Reliability and Security, ACM, 2023. https://dl.acm.org/doi/10.1145/3600160.3605057
77. Simoni, M.; Saracino, A.; Vinod, P.; Conti, M. MoRSE: Bridging the Gap in Cybersecurity Expertise with Retrieval Augmented Generation. In Proc. ACM SAC 2025: 40th ACM/SIGAPP Symposium on Applied Computing, ACM, 2025. https://dl.acm.org/doi/10.1145/3672608.3707898
78. Zhang, C.; Costa-Pérez, X.; Patras, P. Adversarial Attacks Against Deep Learning-Based Network Intrusion Detection Systems and Defense Mechanisms. IEEE/ACM Trans. Netw. 2022, 07 January 2022. https://doi.org/10.1109/TNET.2021.3137084 https://dl.acm.org/doi/10.1109/TNET.2021.3137084

 doi:10.20944/preprints202606.0142.v1

79. 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 https://dl.acm.org/doi/10.1016/j.eswa.2023.122045
80. Lin, L.; Zhong, Q.; Qiu, J.; Liang, Z. E-GRACL: An IoT Intrusion Detection System Based on Graph Neural Networks. J. Supercomput. 2024. https://doi.org/10.1007/s11227-024-06471-5 https://dl.acm.org/doi/10.1007/s11227-024-06471-5
81. Siganos, M.; Radoglou-Grammatikis, P.; Kotsiuba, I.; Markakis, E.; Moscholios, I.; Goudos, S.; Sarigiannidis, P. Explainable AI-Based Intrusion Detection in the IoT. In Proc. ARES 2023: 18th International Conference on Availability, Reliability and Security, ACM, 2023. https://dl.acm.org/doi/abs/10.1145/3600160.3605162
82. Sun, Z.; Teixeira, A.M.H.; Toor, S. GNN-IDS: Graph Neural Network Based Intrusion Detection System. In Proc. ARES 2024: 19th International Conference on Availability, Reliability and Security, ACM, 2024. https://dl.acm.org/doi/10.1145/3664476.3664515
83. Desolda, G.; Greco, F.; Vigano, L. APOLLO: A GPT-Based Tool to Detect Phishing Emails and Generate Explanations That Warn Users. Proc. ACM Hum.-Comput. Interact. 2025. https://dl.acm.org/doi/10.1145/3733049
84. Nair, R.; Abbasi, F.; Pervez, S. PhishEmailLLM: A Meta Model Approach to Detect Phishing Emails by Leveraging LLMs and ML Models. In Proc. ACSW 2025: Australasian Computer Science Week, ACM, 2025. https://dl.acm.org/doi/10.1145/3727166.3727169
85. Ali, I.; Subba, B. PhishURLDetect: A Parameter-Efficient Fine-Tuning of LLMs using LoRA for Detection of Phishing URLs. In Proc. ICDCN 2025: 26th International Conference on Distributed Computing and Networking, ACM, 2025. https://dl.acm.org/doi/10.1145/3700838.3703658
86. Morales Flores, A.; Jhaveri, J.; Sharma, A.; Rege, S.; Munyaka, I. The Impact of LLM Assistance on User Spam Detection. In Proc. ACM Workshop on Human-Centered AI Privacy and Security, ACM, 2025. https://dl.acm.org/doi/10.1145/3733816.3760754
87. Boi, B.; Esposito, C.; Lee, S. Smart Contract Vulnerability Detection: The Role of Large Language Model (LLM). ACM SIGAPP Appl. Comput. Rev. 2024. https://dl.acm.org/doi/10.1145/3687251.3687253
88. Yang, Z.; Man, G.; Yue, S. Automated Smart Contract Vulnerability Detection using Fine-tuned LLMs. In Proc. 6th International Conference on Blockchain Technology and Applications, ACM, 2023. https://dl.acm.org/doi/10.1145/3651655.3651658
89. Shen, W.; Yang, Y.; Zhang, X.; Cui, L.; Wang, Y. How Far Have We Been on the Path of LLM-Enhanced Vulnerability Detection. In Proc. 2025 International Symposium on Artificial Intelligence and Computational Social Sciences, ACM, 2025. https://dl.acm.org/doi/10.1145/3776759.3776861
90. Liu, P.; Zheng, Y.; Sun, C.; Qin, C.; Fang, D.; Liu, M.; Sun, L. FITS: Inferring Intermediate Taint Sources for Effective Vulnerability Analysis of IoT Device Firmware. In Proc. ASPLOS 2024: 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Vol. 4, ACM, 2024. https://dl.acm.org/doi/10.1145/3623278.3624759
91. Rondanini, C.; Carminati, B.; Ferrari, E.; Kundu, A.; Gaudiano, A. Malware Detection at the Edge with Lightweight LLMs: A Performance Evaluation. ACM Trans. Internet Technol. 2025. https://dl.acm.org/doi/10.1145/3769681
92. Kavian, A.; Pourhashem Kallehbasti, M.M.; Kazemi, S.; Firouzi, E.; Ghafari, M. LLM Security Guard for Code. In Proc. EASE 2024: 28th International Conference on Evaluation and Assessment in Software Engineering, ACM, 2024. https://dl.acm.org/doi/10.1145/3661167.3661263
93. Karlsen, E.; Luo, X.; Zincir-Heywood, N.; Heywood, M. Benchmarking Large Language Models for Log Analysis, Security, and Interpretation. J. Netw. Syst. Manag. 2024. https://dl.acm.org/doi/abs/10.1007/s10922-024-09831-x
94. Dwivedi, S.; Rajendran, B.; Akshay, P.V.; Acha, A.; Ampatt, P.; Sudarsan, S.D. IntelliSOAR: Intelligent Alert Enrichment Using Security Orchestration Automation and Response (SOAR). In Proc. ICISS 2024: Information Systems Security, Springer, 2024. https://dl.acm.org/doi/10.1007/978-3-031-80020-7_27
95. Mohammadi, M.; Li, Y.; Lo, J.; Yip, W. Evaluation and Benchmarking of LLM Agents: A Survey. In Proc. KDD 2025: 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Vol. 2, ACM, 2025. https://dl.acm.org/doi/10.1145/3711896.3736570
96. Rekha, H.; Siddappa, M. Hybrid Deep Learning Model for Attack Detection in Internet of Things. Serv. Oriented Comput. Appl. 2022, 01 December 2022. https://doi.org/10.1007/s11761-022-00342-8 https://dl.acm.org/doi/10.1007/s11761-022-00342-8

 doi:10.20944/preprints202606.0142.v1

97. Wang, L.; Sun, J.; Jiang, J.; Kanhere, S.S.; Xing, Z.; Jha, S. Vulnerability Aspects Extraction and Discrepancies Detection across Heterogeneous Threat Intelligence. In Proc. ACM Workshop on Privacy in LLMs and NLP, ACM, 2025. https://dl.acm.org/doi/10.1145/3709018.3736330
98. Manowska, A.; Syta, J. Application of Large Language Models in the Protection of Industrial IoT Systems for Critical Infrastructure. Appl. Sci. 2026, 16(2), 730. https://doi.org/10.3390/app16020730 https://www.mdpi.com/2076-3417/16/2/730
99. Jaffal, N.O.; Alkhanafseh, M.; Mohaisen, D. Large Language Models in Cybersecurity: A Survey of Applications, Vulnerabilities, and Defense Techniques. AI 2025, 6(9), 216. https://doi.org/10.3390/ai6090216 https://www.mdpi.com/2673-2688/6/9/216
100. López Delgado, J.L.; López Ramos, J.A. A Comprehensive Survey on Generative AI Solutions in IoT Security. Electronics 2024, 13(24), 4965. https://doi.org/10.3390/electronics13244965 https://www.mdpi.com/2079-9292/13/24/4965
101. Neto, E.C.P.; Dadkhah, S.; Ferreira, R.; Zohourian, A.; Lu, R.; Ghorbani, A.A. CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. Sensors 2023, 23(13), 5941. https://doi.org/10.3390/s23135941 https://www.mdpi.com/1424-8220/23/13/5941
102. Haque, S.; El-Moussa, F.; Komninos, N.; Muttukrishnan, R. A Systematic Review of Data-Driven Attack Detection Trends in IoT. Sensors 2023, 23(16), 7191. https://doi.org/10.3390/s23167191 https://www.mdpi.com/1424-8220/23/16/7191
103. Almalawi, A.; et al. A Lightweight Intrusion Detection System for IoT: Clustering and Monte Carlo Cross-Entropy Approach. Sensors 2025, 25(7), 2235. https://doi.org/10.3390/s25072235 https://www.mdpi.com/1424-8220/25/7/2235
104. 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 https://www.mdpi.com/1424-8220/26/2/356
105. Wang, T.; Li, Y.; Pan, Z. LLM-Boofuzz: Generation-Based Black-Box Fuzzing for Network Protocols via LLMs. Electronics 2025, 14(23), 4550. https://doi.org/10.3390/electronics14234550 https://www.mdpi.com/2079-9292/14/23/4550
106. Wu, X.; Tian, Y.; Chen, Y.; Ye, P.; Cui, X.; Jia, J.; Li, S.; Liu, J.; Niu, W. CurriculumPT: LLM-Based Multi-Agent Autonomous Penetration Testing with Curriculum-Guided Task Scheduling. Appl. Sci. 2025, 15(16), 9096. https://doi.org/10.3390/app15169096 https://www.mdpi.com/2076-3417/15/16/9096
107. Srinivas, S.; Kirk, B.; Zendejas, J.; Alzahrani, N. AI-Augmented SOC: A Survey of LLMs and Agents for Security Automation. J. Cybersecur. Priv. 2025, 5(4), 95. https://doi.org/10.3390/jcp5040095 https://www.mdpi.com/2624-800X/5/4/95
108. Ismail; Kurnia, R.; Brata, Z.A.; Nelistiani, G.A.; Heo, S.; Kim, H. Toward Robust Security Orchestration and Automated Response in SOC with a Hyper-Automation Approach Using Agentic AI. Information 2025, 16(5), 365. https://doi.org/10.3390/info16050365 https://www.mdpi.com/2078-2489/16/5/365
109. Tilbury, J.; Flowerday, S. Humans and Automation: Augmenting Security Operation Centers. J. Cybersecur. Priv. 2024, 4(3), 20. https://doi.org/10.3390/jcp4030020 https://www.mdpi.com/2624-800X/4/3/20
110. Nowrozy, R. GPTs or Grim Position Threats? The Potential Impacts of LLMs on Non-Managerial Jobs and Certifications in Cybersecurity. Informatics 2024, 11(3), 45. https://doi.org/10.3390/informatics11030045 https://www.mdpi.com/2227-9709/11/3/45
111. Barrios-González, M.; Aguiar-Pérez, J.M.; Pérez-Juárez, M.Á.; Castañeda-de-Benito, E. Redefining Cyber Threat Intelligence with Artificial Intelligence: From Data Processing to Predictive Insights and Human–AI Collaboration. Appl. Sci. 2026, 16(3), 1668. https://doi.org/10.3390/app16031668 https://www.mdpi.com/2076-3417/16/3/1668
112. Brandao, P.R. Exploring the Role of Artificial Intelligence in Detecting Advanced Persistent Threats. Computers 2025, 14(7), 245. https://doi.org/10.3390/computers14070245 https://www.mdpi.com/2073-431X/14/7/245
113. Chen, L.; Deng, H.; Zhang, J.; Zheng, B.; Jiang, R. Threat Intelligence Named Entity Recognition Based on Segment-Level Information Extraction and Similar Semantic Space Construction. Symmetry 2025, 17(5), 783. https://doi.org/10.3390/sym17050783 https://www.mdpi.com/2073-8994/17/5/783
114. Orman, A. Cyberattack Detection Systems in Industrial IoT Networks in Big Data Environments. Appl. Sci. 2025, 15(6), 3121. https://doi.org/10.3390/app15063121 https://www.mdpi.com/2076-3417/15/6/3121

 doi:10.20944/preprints202606.0142.v1

115. Usman, M.; Sarfraz, M.S.; Habib, U.; Aftab, M.U.; Javed, S. Automatic Hybrid Access Control in SCADA-Enabled IIoT Networks Using Machine Learning. Sensors 2023, 23(8), 3931. https://doi.org/10.3390/s23083931 https://www.mdpi.com/1424-8220/23/8/3931
116. Alalwany, E.; Mahgoub, I. Security and Trust Management in the Internet of Vehicles (IoV): Challenges and Machine Learning Solutions. Sensors 2024, 24(2), 368. https://doi.org/10.3390/s24020368 https://www.mdpi.com/1424-8220/24/2/368
117. Tang, J.; Huang, Z.; Li, C. MT-FBERT: Malicious Traffic Detection Based on Efficient Federated Learning of BERT. Future Internet 2025, 17(8), 323. https://doi.org/10.3390/fi17080323 https://www.mdpi.com/1999-5903/17/8/323
118. Ullah, F.; Alsirhani, A.; Alshahrani, M.M.; Alomari, A.; Naeem, H.; Shah, S.A. Explainable Malware Detection System Using Transformers-Based Transfer Learning and Multi-Model Visual Representation. Sensors 2022, 22(18), 6766. https://doi.org/10.3390/s22186766 https://www.mdpi.com/1424-8220/22/18/6766
119. Alshomrani, M.; Albeshri, A.; Alturki, B.; Alallah, F.S.; Alsulami, A.A. Survey of Transformer-Based Malicious Software Detection Systems. Electronics 2024, 13(23), 4677. https://doi.org/10.3390/electronics13234677 https://www.mdpi.com/2079-9292/13/23/4677
120. He, H.; Yuan, X.; Wu, K.; Ni, W. Federated RAG for Cybersecurity in Resource-Constrained IoT and Edge Environments: A Deployment-Oriented Scoping Review. Electronics 2026, 15(7), 1409. https://doi.org/10.3390/electronics15071409 https://www.mdpi.com/2079-9292/15/7/1409
121. Loumachi, F.Y.; Ghanem, M.C.; Ferrag, M.A. Advancing Cyber Incident Timeline Analysis Through RAG and LLMs. Computers 2025, 14(2), 67. https://doi.org/10.3390/computers14020067 https://www.mdpi.com/2073-431X/14/2/67
122. Villegas-Ch, W.; Govea, J.; Jaramillo-Alcazar, A. IoT Anomaly Detection to Strengthen Cybersecurity in the Critical Infrastructure of Smart Cities. Appl. Sci. 2023, 13(19), 10977. https://doi.org/10.3390/app131910977 https://www.mdpi.com/2076-3417/13/19/10977
123. Guato Burgos, M.F.; Morato, J.; Vizcaino Imacaña, F.P. A Review of Smart Grid Anomaly Detection Approaches Pertaining to Artificial Intelligence. Appl. Sci. 2024, 14(3), 1194. https://doi.org/10.3390/app14031194 https://www.mdpi.com/2076-3417/14/3/1194
124. Gunduz, M.Z.; Das, R. Smart Grid Security: An Effective Hybrid CNN-Based Approach for Detecting Energy Theft Using Consumption Patterns. Sensors 2024, 24(4), 1148. https://doi.org/10.3390/s24041148 https://www.mdpi.com/1424-8220/24/4/1148
125. Reis, M.J.C.S. AI-Driven Anomaly Detection for Securing IoT Devices in 5G-Enabled Smart Cities. Electronics 2025, 14(12), 2492. https://doi.org/10.3390/electronics14122492 https://www.mdpi.com/2079-9292/14/12/2492
126. Alalhareth, M.; Hong, S.-C. An Adaptive Intrusion Detection System in the Internet of Medical Things Using Fuzzy-Based Learning. Sensors 2023, 23(22), 9247. https://doi.org/10.3390/s23229247 https://www.mdpi.com/1424-8220/23/22/9247
127. Georgiades, M.; Hussain, F. An Explainable AI Approach for Interpretable Cross-Layer Intrusion Detection in IoMT. Electronics 2025, 14(16), 3218. https://doi.org/10.3390/electronics14163218 https://www.mdpi.com/2079-9292/14/16/3218
128. Nwakanma, C.I.; Ahakonye, L.A.C.; Njoku, J.N.; Odirichukwu, J.C.; Okolie, S.A.; Uzondu, C.; Ndubuisi Nweke, C.C.; Kim, D.-S. Explainable AI (XAI) for IDS and Mitigation in Intelligent Connected Vehicles: A Review. Appl. Sci. 2023, 13(3), 1252. https://doi.org/10.3390/app13031252 https://www.mdpi.com/2076-3417/13/3/1252
129. Alabbadi, A.; Bajaber, F. An IDS over IoT Data Streams Using Explainable AI (XAI). Sensors 2025, 25(3), 847. https://doi.org/10.3390/s25030847 https://www.mdpi.com/1424-8220/25/3/847
130. Fatema, K.; Dey, S.K.; Anannya, M.; Khan, R.T.; Rashid, M.M.; Su, C.; Mazumder, R. Federated XAI IDS: An Explainable and Privacy-Preserving Approach to Detect Intrusion Combining Federated Learning and SHAP. Future Internet 2025, 17(6), 234. https://doi.org/10.3390/fi17060234 https://www.mdpi.com/1999-5903/17/6/234
131. Hermosilla, P.; Berríos, S.; Allende-Cid, H. Explainable AI for Forensic Analysis: A Comparative Study of SHAP and LIME in IDS Models. Appl. Sci. 2025, 15(13), 7329. https://doi.org/10.3390/app15137329 https://www.mdpi.com/2076-3417/15/13/7329
132. Allaw, Z.; Zein, O.; Ahmad, A.-M. Cross-Layer Security for 5G/6G Network Slices: An SDN, NFV, and AI-Based Hybrid Framework. Sensors 2025, 25(11), 3335. https://doi.org/10.3390/s25113335 https://www.mdpi.com/1424-8220/25/11/3335

 doi:10.20944/preprints202606.0142.v1

133. Botez, R.; Zinca, D.; Dobrota, V. Redefining 6G Network Slicing: AI-Driven Solutions for Future Use Cases. Electronics 2025, 14(2), 368. https://doi.org/10.3390/electronics14020368 https://www.mdpi.com/2079-9292/14/2/368
134. Dias, J.; Pinto, P.; Santos, R.; Malta, S. 5G Network Slicing: Security Challenges, Attack Vectors, and Mitigation Approaches. Sensors 2025, 25(13), 3940. https://doi.org/10.3390/s25133940 https://www.mdpi.com/1424-8220/25/13/3940
135. Karahan, S.N.; Güllü, M.; Karhan, D.; Çimen, S.; Osmanca, M.S.; Barışçı, N. Realistic Performance Assessment of ML Algorithms for 6G Network Slicing: A Dual-Methodology Approach with Explainable AI. Electronics 2025, 14(19), 3841. https://doi.org/10.3390/electronics14193841 https://www.mdpi.com/2079-9292/14/19/3841
136. Bast, C.; Yeh, K.-H. Emerging Authentication Technologies for Zero Trust on the Internet of Things. Symmetry 2024, 16(8), 993. https://doi.org/10.3390/sym16080993 https://www.mdpi.com/2073-8994/16/8/993
137. Federici, F.; Martintoni, D.; Senni, V. A Zero-Trust Architecture for Remote Access in Industrial IoT Infrastructures. Electronics 2023, 12(3), 566. https://doi.org/10.3390/electronics12030566 https://www.mdpi.com/2079-9292/12/3/566
138. Wazzan, M.; Algazzawi, D.; Bamasaq, O.; Albeshri, A.; Cheng, L. IoT Botnet Detection Approaches: Analysis and Recommendations for Future Research. Appl. Sci. 2021, 11(12), 5713. https://doi.org/10.3390/app11125713 https://www.mdpi.com/2076-3417/11/12/5713
139. Kim, J.; Shim, M.; Hong, S.; Shin, Y.; Choi, E. Intelligent Detection of IoT Botnets Using Machine Learning and Deep Learning. Appl. Sci. 2020, 10(19), 7009. https://doi.org/10.3390/app10197009 https://www.mdpi.com/2076-3417/10/19/7009
140. Atlam, H.F. LLMs in Cyber Security: Bridging Practice and Education. Big Data Cogn. Comput. 2025, 9(7), 184. https://doi.org/10.3390/bdcc9070184 https://www.mdpi.com/2504-2289/9/7/184
141. Park, S.; Choi, D. Exploring the Potential of Anomaly Detection Through Reasoning with Large Language Models. Appl. Sci. 2025, 15(19), 10384. https://doi.org/10.3390/app151910384 https://www.mdpi.com/2076-3417/15/19/10384
142. Lee, J.; Jeong, Y.; Han, T.; Lee, T. LogRESP-Agent: A Recursive AI Framework for Context-Aware Log Anomaly Detection and TTP Analysis. Appl. Sci. 2025, 15(13), 7237. https://doi.org/10.3390/app15137237 https://www.mdpi.com/2076-3417/15/13/7237
143. Palma, G.; Cecchi, G.; Caronna, M.; Rizzo, A. Leveraging LLMs for Scalable and Explainable Cybersecurity Log Analysis. J. Cybersecur. Priv. 2025, 5(3), 55. https://doi.org/10.3390/jcp5030055 https://www.mdpi.com/2624-800x/5/3/55
144. Roumeliotis, K.I.; Tselikas, N.D.; Nasiopoulos, D.K. Next-Generation Spam Filtering: Comparative Fine-Tuning of LLMs, NLPs, and CNN Models for Email Spam Classification. Electronics 2024, 13(11), 2034. https://doi.org/10.3390/electronics13112034 https://www.mdpi.com/2079-9292/13/11/2034
145. Heydari, V.; Nyarko, K. Enhancing Adversarial Robustness in NIDS: A Novel Adversarially Trained Neural Network Approach. Electronics 2025, 14(16), 3249. https://doi.org/10.3390/electronics14163249 https://www.mdpi.com/2079-9292/14/16/3249
146. Joshi, A.; Baidya, S. Securing the Cognitive Layer: A Survey on Security Threats, Defenses, and Privacy-Preserving Architectures for LLM-IoT Integration. J. Cybersecur. Priv. 2026, 6(2), 63. https://doi.org/10.3390/jcp6020063 https://www.mdpi.com/2624-800X/6/2/63
147. Almutairi, N. LLM-Powered Proactive Cyber-Defense Framework Using Cyber-Threat Indicators Collected from X Platform. Electronics 2026, 15(6), 1305. https://doi.org/10.3390/electronics15061305 https://www.mdpi.com/2079-9292/15/6/1305
148. Daniel, N.; Kaiser, F.K.; Giladi, S.; Sharabi, S.; Moyal, R.; Shpolyansky, S.; Murillo, A.; Elyashar, A.; Puzis, R. Labeling NIDS Rules with MITRE ATT&CK Techniques: Machine Learning vs. Large Language Models. Big Data Cogn. Comput. 2025, 9(2), 23. https://doi.org/10.3390/bdcc9020023 https://www.mdpi.com/2504-2289/9/2/23
149. Karras, A.; Giannaros, A.; Amasiadi, N.; Karras, C. Next-Gen Explainable AI (XAI) for Federated and Distributed IoT Systems: A State-of-the-Art Survey. Future Internet 2026, 18(2), 83. https://doi.org/10.3390/fi18020083 https://www.mdpi.com/1999-5903/18/2/83
150. Altaf, T.; Wang, X.; Ni, W.; Yu, G.; Liu, R.P.; Braun, R. GNN-Based Network Traffic Analysis for the Detection of Sequential Attacks in IoT. Electronics 2024, 13(12), 2274. https://doi.org/10.3390/electronics13122274 https://www.mdpi.com/2079-9292/13/12/2274

 doi:10.20944/preprints202606.0142.v1

151. Ngo, T.; Yin, J.; Ge, Y.-F.; Wang, H. Optimizing IoT Intrusion Detection — A GNN Approach with Attribute-Based Graph Construction. Information 2025, 16(6), 499. https://doi.org/10.3390/info16060499 https://www.mdpi.com/2078-2489/16/6/499
152. Judith, A.; Kathrine, G.J.W.; Silas, S.; Andrew, J. Efficient DL-Based Cyber-Attack Detection for IoMT Devices. Eng. Proc. 2023, 59(1), 139. https://doi.org/10.3390/engproc2023059139 https://www.mdpi.com/2673-4591/59/1/139
153. 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 https://www.mdpi.com/2073-431x/14/7/291
154. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.R.; Cao, Y. ReAct: Synergizing Reasoning and Acting in Language Models. In Proceedings of the Eleventh International Conference on Learning Representations (ICLR 2023), Kigali, Rwanda, 1–5 May 2023. https://openreview.net/forum?id=WE_vluYUL-X
155. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.H.; Le, Q.V.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In Advances in Neural Information Processing Systems 35 (NeurIPS 2022), Main Conference Track, New Orleans, LA, USA, 28 November–9 December 2022. https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html
156. Greshake, K.; Abdelnabi, S.; Mishra, S.; Endres, C.; Holz, T.; Fritz, M. Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. In Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23), Copenhagen, Denmark, 30 November 2023; ACM: New York, NY, USA; pp. 79–90. DOI: 10.1145/3605764.3623985. https://dl.acm.org/doi/abs/10.1145/3605764.3623985
157. Carlini, N.; Nasr, M.; Choquette-Choo, C.A.; Jagielski, M.; Gao, I.; Awadalla, A.; Koh, P.W.; Ippolito, D.; Lee, K.; Tramèr, F.; Schmidt, L. Are Aligned Neural Networks Adversarially Aligned? In Advances in Neural Information Processing Systems 36 (NeurIPS 2023), New Orleans, LA, USA, 10–16 December 2023. https://proceedings.neurips.cc/paper_files/paper/2023/hash/c1f0b856a35986348ab3414177266f75-Abstract-Conference.html
158. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Agüera y Arcas, B. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS 2017), Fort Lauderdale, FL, USA, 20–22 April 2017; PMLR: Vol. 54, pp. 1273–1282. https://proceedings.mlr.press/v54/mcmahan17a.html
159. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0); NIST AI 100-1; U.S. Department of Commerce: Gaithersburg, MD, USA, January 2023. DOI: 10.6028/NIST.AI.100-1. https://doi.org/10.6028/NIST.AI.100-1
160. European Parliament and Council of the European Union. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act). Official Journal of the European Union, L series, 12 July 2024. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32024R1689
161. European Union Agency for Cybersecurity (ENISA). Guidelines for Securing the Internet of Things: Secure Supply Chain for IoT; ENISA Report; ENISA: Athens, Greece, 9 November 2020. https://www.enisa.europa.eu/publications/guidelines-for-securing-the-internet-of-things
162. Sahota, J.; Vlajic, N. Mozi IoT Malware and Its Botnets: From Theory To Real-World Observations. In Proc. 2021 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 15–17 December 2021. DOI: 10.1109/CSCI54926.2021.00181. https://ieeexplore.ieee.org/document/9799037/
163. Tu, T.-F.; Qin, J.-W.; Zhang, H.; Chen, M.; et al. A Comprehensive Study of Mozi Botnet. Int. J. Intell. Syst. 2022, 37, 6877–6908. https://doi.org/10.1002/int.22866
164. Wang, B.; Sang, Y.; Zhang, Y.; Li, S.; Xu, X. A Longitudinal Measurement and Analysis Study of Mozi, an Evolving P2P IoT Botnet. In Proc. 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 09–11 December 2022. DOI: 10.1109/TrustCom56396.2022.00027. https://ieeexplore.ieee.org/document/10063351/
165. Nageshwaran, V.; Ezekiel, S. Agentic IoT Security Lab (v0.2.0): Companion Reproducibility Kit for "Agentic AI and Large Language Models for Autonomous IoT Cybersecurity: A Systematic Survey, Taxonomy, and Research Roadmap". Zenodo, 2026. https://doi.org/10.5281/zenodo.20446651

 doi:10.20944/preprints202606.0142.v1

Short Biography of Authors

Vinoth Nageshwaran is a Data Engineer IV at Business Insider (New York, NY, USA), where his work integrates large-language-model and retrieval-augmented-generation components into production data pipelines. He has authored five peer-reviewed IEEE conference papers (2024-2025) on LLM integration in enterprise data systems and cloud-native security frameworks, and maintains agentprdiff, an open-source library for snapshot testing of LLM agents. He is a member of IEEE and ACM. (e-mail: vnageshwaran@gmail.com; ORCID: 0009-0004-0332-231X.)

Soundararajan Ezekiel Soundararajan Ezekiel, Ph.D. is a Professor of Computer Science in the Department of Mathematical and Computer Sciences at Indiana University of Pennsylvania (Indiana, PA, USA). His research interests span image and signal processing, medical imaging, fractal analysis, artificial intelligence and machine learning, cybersecurity, zero-trust architecture, and the application of wavelet and fractal methods to engineering and biomedical problems. He has published extensively on AI-driven security, IoT testbeds, and threat detection, and has mentored numerous graduate researchers in computational intelligence and trustworthy AI.