Journal of *Cybersecurity and Privacy*

MDPI

Systematic Review

AI-Augmented SOC: A Survey of LLMs and Agents for Security Automation

Siddhant Srinivas *, Brandon Kirk, Julissa Zendejas, Michael Espino, Matthew Boskovich, Abdul Bari, Khalil Dajani and Nabeel Alzahrani

School of Computer Science & Engineering, California State University, San Bernardino, 5500 University Parkway, San Bernardino, CA 92407, USA; brandon.kirk8304@coyote.csusb.edu (B.K.); julissa.zendejas6692@coyote.csusb.edu (J.Z.); michael.espino5607@coyote.csusb.edu (M.E.); matthew.boskovich3084@coyote.csusb.edu (M.B.); abdul.bari8019@coyote.csusb.edu (A.B.); khalil.dajani@csusb.edu (K.D.); alzahran@csusb.edu (N.A.)
* Correspondence: siddhant.srinivas1528@coyote.csusb.edu

Abstract

The increasing volume, velocity, and sophistication of cyber threats have placed immense pressure on modern Security Operations Centers (SOCs). Traditional rule-based and manual processes are proving insufficient, leading to alert fatigue, delayed responses, high false-positive rates, analyst dependency, and escalating operational costs. Recent advancements in Artificial Intelligence (AI) offer new opportunities to transform SOC workflows through automation and augmentation. Large Language Models (LLMs) and autonomous AI agents have shown strong potential in enhancing capabilities such as log summarization, alert triage, threat intelligence, incident response, report generation, asset discovery, and vulnerability management. This paper reviews recent developments in the application of LLMs and AI agents across these SOC functions, introducing a taxonomy that organizes their roles and capabilities within operational pipelines. While these technologies improve detection accuracy, response time, and analyst support, challenges persist, including model interpretability, adversarial robustness, integration with legacy systems, and the risk of hallucinations or data leakage. A detailed capability-maturity model outlines the levels of integration with SOC tasks. This survey synthesizes trends, identifies persistent limitations, and outlines future directions for trustworthy, explainable, and safe AI integration in SOC environments.

Keywords: security operation center; Large Language Models; AI agent; cybersecurity automation; human-AI collaboration

Academic Editor: Carson K. Leung

Received: 13 September 2025
Revised: 20 October 2025
Accepted: 3 November 2025
Published: 5 November 2025

Citation: Srinivas, S.; Kirk, B.; Zendejas, J.; Espino, M.; Boskovich, M.; Bari, A.; Dajani, K.; Alzahrani, N. AI-Augmented SOC: A Survey of LLMs and Agents for Security Automation. *J. Cybersecur. Priv.* **2025**, *5*, 95. https://doi.org/10.3390/jcp5040095

1. Introduction

The digital landscape faces a rapid escalation in both the frequency and sophistication of cyber threats, straining modern SOCs. Traditional SOCs, relying on manual, rule-based, or signature-driven processes, prove inadequate against increasingly dynamic and sophisticated cyberattacks. This burden leads to significant financial losses, analyst alert fatigue, high false-positive rates, and delayed critical threat responses. To overcome these limitations, AI integration is essential for rapid, accurate, and scalable threat detection and response [1]. This paper focuses on using recent AI agent and LLM advancements to automate and augment these eight SOC tasks: log summarization, alert triage, threat intelligence, incident response, report generation, asset discovery, and vulnerability management. AI agents represent another transformative technology. AI Agents, autonomous systems

executing complex multi-step tasks, enable a transition toward proactive cybersecurity in next-generation SOCs [2,3]. These agents autonomously manage security operations with minimal human intervention, detecting, classifying, and responding to threats [4]. AI agents show promise in root cause analysis, automated security audits, and autonomous cyber defense (ACD). Complementing AI agents, LLMs, an advanced subset of generative AI, represent powerful tools for SOC task automation and augmentation due to their contextual understanding and analytical capabilities. LLMs process unstructured security data, analyze logs, summarize incidents, assist decision-making, and enable natural language interactions for security intelligence queries. Their proficiency in language comprehension and generation makes them well-suited at analyzing textual alerts and generating reports. Persistent challenges include model interpretability ("black boxes"), data quality issues [1], integration with legacy systems [3], hallucinations [5], privacy leakage concerns [6], and susceptibility to adversarial attacks [2]. However, despite these promising developments, significant gaps remain in existing literature and implementations. Several recent surveys have reviewed the application of LLMs in cybersecurity domains such as cyber defense [2], threat intelligence, and anomaly detection. However, these works remain largely domain-specific. This paper consolidates and extends those efforts by connecting them through the operational lens of the Security Operations Center (SOC). We provide an integrated taxonomy covering eight SOC functions and introduce a capability-maturity framework that situates AI and LLM integration across progressive autonomy levels. Thus, our contribution lies not in being the first survey on LLMs for cyber defense, but in offering the first unified SOC-centered synthesis that connects prior fragmented studies into a coherent, operationally grounded vision. This survey offers a taxonomy of LLM and AI agent applications in SOCs, introduces a capability-maturity model to measure automation, performs an analysis of strengths and limitations [3], addresses critical safety concerns [5], highlights augmentation performance improvements over the traditional methods, and outlines future research directions in explainable AI and human-AI collaboration [1].

2. Methodology

This survey was conducted through a systematic literature review aimed at exploring how LLMs and autonomous AI agents are being applied to augment core tasks within SOCs. This initial pool consisted of over 600 papers selected based on four criteria: relevance to the research topic, publication date (2022 or later), experimental research paper, and peer-review status or preprint credibility. Publication bias was not assessed because the review included both peer-reviewed and preprint sources to mitigate selection effects. On automating all 8 SOC tasks, peer-reviewed and preprint sources were selected, covering a range of recent publications from 2022 to 2025 (last searched 15 July 2025), since practical SOC applications of LLMs and autonomous agents has rapidly increased [3], pre-2022 literature is largely outdated for this topic as AI is advancing at a rapid pace. We systematically searched bibliographic databases ([7], arXiv [8], and the ACM Digital Library [9]) and official conference proceedings archives for peer-reviewed security venues (NDSS, USENIX Security, ACM CCS, and IEEE Symposium on Security and Privacy) for papers published from 2022 to 2025. For each conference, the accepted-papers lists were retrieved from the official symposium websites, and metadata were manually screened for relevance to AI agent or LLM-based security automation. The selection process began with keyword-driven searches targeting combinations of terms such as "LLMs," "AI agents," "automation," "augmentation," and specific SOC tasks such as "log summarization," "alert triage," "threat intelligence," "ticket handling," "incident response," "report generation," "asset discovery and management," and "vulnerability management". After initial filtering based on relevance and abstract screening, full-text analysis was performed to extract

methodological approaches, use-case contexts, model architectures, and evaluation strategies. Disagreements were resolved by consensus among reviewers. To ensure balanced coverage, studies were categorized based on task type, LLM/AI Agent application mode (e.g., autonomous vs. human-in-the-loop), and integration depth with existing SOC workflows. This process can be seen visually in Figure 1 using Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA). Our survey paper utilizes the completed PRISMA checklist for transparency, standardization, and replicability in systematic reviews or structured evidence syntheses, but has not been currently registered. The PRISMA 2020 Checklist is provided in Table S1 (see Supplementary Materials). Throughout this survey, performance metrics such as accuracy, precision, recall, and F1-score are reported as originally presented in the primary studies unless explicitly stated otherwise. When the same metrics are discussed comparatively or synthesized across studies, these represent aggregated interpretations by the authors for consistency and clarity. After preliminary screening 510 papers relevant to AI or LLM applications in SOC tasks, where 300 eliminated papers were discarded due to lack of SOC relevance, a second-stage qualitative assessment was conducted to ensure methodological rigor and thematic relevance. Certainty in findings was qualitatively estimated based on study credibility and consistency across multiple sources. Each of the 105 papers was independently reviewed by four authors using the following criteria: The work addressed at least one of the eight core SOC functions, the paper described model type, dataset, or evaluation metrics sufficient for cross-comparison, was published 2022 or later in a peer-reviewed venue or high-credibility preprint (arXiv, IEEE, ACM), and had explicit integration of LLMs, multi-agent frameworks, or human-AI collaboration mechanisms within cybersecurity contexts. This systematic review was conducted in accordance with the PRISMA 2020 guidelines to ensure methodological transparency, reproducibility, and reporting rigor. This paper adheres to all major PRISMA checklist items relevant to qualitative systematic reviews. This systematic review was preregistered with the Open Science Framework (OSF) under the registration DOI https://doi.org/10.17605/OSF.IO/9QC7V, in accordance with PRISMA 2020 guidelines to ensure methodological transparency and reproducibility.

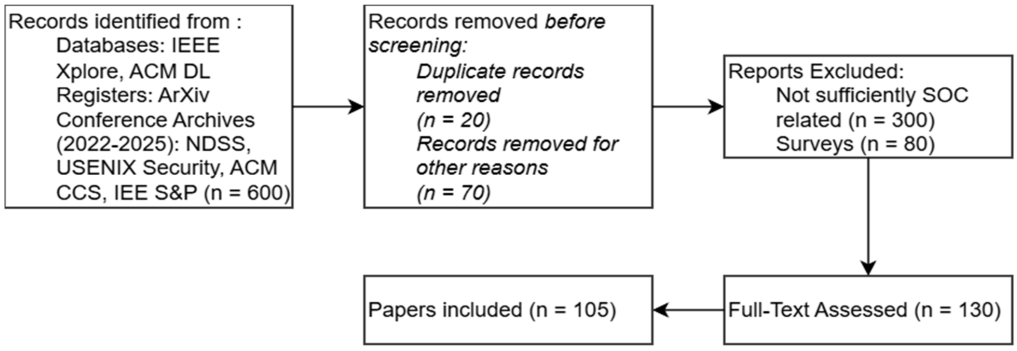

Figure 1. PRISMA-style literature-selection flow used in this survey.

3. Integration of AI Agents and LLMs in Security Operations Center Tasks

The evolution of SOCs toward AI-augmented environments represents a fundamental transformation in cybersecurity operations. This comprehensive analysis examines the integration of AI agents and LLMs across eight critical SOC tasks, highlighting both their capabilities and inherent limitations in modern threat landscapes. These SOC tasks are the functions the SOC implements; the LLM/AI Agent techniques listed in Table 1 are the strategies used to automate the SOC tasks.

Table 1. AI Augmented SOC Techniques. Note: This table highlights key features only and does not include all details from the referenced works.

SOC Task	LLM/AI Agent Techniques	Model Type	Evaluation Method	Dataset
Log Summarization	Log Parsing [10,11], Fine-tuning [12], Domain-Specific Processing [13], RAG [14]	GPT (3.5,4,4o) [15], LLaMA (7B, 2, 3.1) [16], Zephyr [17], CodeT5 [12], LogPrompt [18], Cygent [12]	F1 Score, Precision, Recall [18], Efficiency/Scalability [19], Accuracy [20]	Loghub-2.0 [21], Real-world [18], BGL, Spirit, Thunderbird [22], HDFS dataset [15]
Alert Triage	NLP [2], RAG [23], LLM Based Agents [24], In Context Learning (ICL) [25]	GPT (3.5,4o,4o mini) [2], LLaMA [26], GeminiPRO [25], SecureBERT [13], HuntGPT [27], CyLens [13]	Accuracy [20], Precision, F1 Score [28], Area Under the Curve (AUC) [29], BERTScore [13]	IoT Traffic [15], LogPub [10], CTI Reports [30]
Threat Intelligence	IoC extraction, TTP mapping [2,31], RAG [32], Multi-Agent Systems [33]	GPT (3.5,4,4o), LLaMA-3 [31,34], Claude [35], Transformer-based models [3]	Simulated Environments [3,33], Human-in-the-Loop Validation [36], Standardized benchmarks [27,37]	Real-world data, synthetic data [1], MITRE-CVE [4], NVD [13]
Ticket Handling	Unified AI-driven architecture, Context based Operations [38], Automated Root Cause Analysis [39]	Flow of Action [39], RCA Copilot [40], Aid-AI, Tickit [41], Ticket-BERT [42]	Accuracy, Precision, Recall, F1 Score [43], Mean Time to Repair [44], Refuse Rate [24]	National Vulnerability Database [13], ExtractFix [45], Vul4J [46]
Incident Response	RAG [3], XAI [5,26], Multiagent systems [47,48]	GPT-4, LLaMA [3], Claude, AidAI [41], GenDFIR [26], IRCopilot [47]	F1 score [1], accuracy precision [35]	NSL-KDD [4], KDD99 [27]
Report Generation	Prompt Engineering, RAG, multiagent [3]	GPT- 4 [3], LLaMA, Gemini [17]	Precision, Recall, F1 [49]	Thunderbird [50], BGL, Spirit [31]
Asset Discovery and Management	Asset Categorization [51], Data Normalization [52], IoT Agent, Work Order Agent [53], Multi-Agent [3]	GPT (3.5-turbo, 40), LLaMA (2-7b-chat-hf), Qwen, Prometheus [54], AI-Avatar [3], AssetsOpsBench [53]	Accuracy [55], F1-score [45], Detection Rate, False Positive Rate [56]	AssetOpsBench dataset [53], NSL-KDD, CICIDS2017 [37]
Vulnerability Management	NLP, LLM Code Analysis [31,55], Knowledge Graphs, Agentic AI [5,50], RAG, Explainable AI [57,58]	GPT, Llama, Gemini, Mistral, Zephyr, BERT, SciBERT, CyBERT [34,35,59] CyLens, Audit-LLM, ChatNVD [59,60]	Accuracy, Precision, Recall, F1-scores [15]	SARD dataset, PatchDB, CVEFixes, ExtractFix, IoT datasets [45,56], NVD, LLM Vulnerability Database (LVD), CTIBench datasets [57,61]

3.1. Log Summarization

Log summarization is a critical advancement in IT Operations (ITOps), enabling the transformation of large volumes of event log data and technical reports into concise, human-readable summaries. Traditional log analysis is time-consuming, requires expert intervention, and often produces results that lack interpretability [12,18]. At the core of this process is log parsing, which extracts static templates and dynamic parameters from raw log messages, forming the basis for downstream tasks such as anomaly detection [10,62]. AI-agent frameworks like CYGENT have achieved over 97% BERTscore, significantly improving log comprehension. LogAssist complements this by reducing log events requiring review by 99% and shortening analysis time by 40% [12,15]. LLM-driven frameworks like LILAC use in-context learning and adaptive parsing [21]; LibreLog, built on open-source LLMs, adds self-reflection for improved privacy and speed outperforming LILAC in GA by 5% and by up to 40 times in speed [14]. LogBatcher, available for real-world deployment, further improves on both, achieving a GA of 0.972, message-level accuracy of 0.895, and processing 3.6 million log messages in 569.6 s with a 100% reduction in LLM costs [62]. Systems like LogPrécis demonstrate that LLMs can distill hundreds of thousands of raw log sessions into a few thousand structured attack fingerprints, enabling efficient large-scale log summarization [63]. In hybrid systems, AI agents orchestrate log data collection from multiple sources, filtering and preprocessing data before feeding it to LLMs for summarization and anomaly detection [64]. The LLMs' output, including summarized logs and flagged suspicious events, can then be presented to analysts or trigger further automated actions by other AI agents within the SOC workflow. LLMs such as GPT-3.5, GPT-4, Claude, and Llama now serve as the cognitive backbone of SOC log summarization, leveraging advanced NLU, NLG, and pattern recognition for scalable, real-time insight extraction [65]. Agentic frameworks extend LLM capabilities with tool use, memory, and planning addressing limitations like context window constraints and hallucinations [66]. Despite these advances, LLMs may still struggle with cybersecurity-specific language and verbose threat reports, though fine-tuning and RAG can help mitigate these issues [15,31].

3.2. Alert Triage

Alert triage is a core Tier-1 SOC process where alerts are assessed for legitimacy and severity using contextual data such as system logs, network traffic, and threat intelligence. SOCs often process over 10,000 daily alerts, more than half of which are false positives overwhelming analysts and contributing to alert fatigue, especially given the limited adaptability of traditional SIEMs [1,3]. Human-AI teaming enables AI agents to assist Tier-1 analysts with alert correlation and prioritization, while analysts provide contextual validation to reduce false positives [29,67]. Alert prioritization criteria encompass alert type, severity, affected systems, potential organizational impact, and specific contextual details. Complex or sophisticated threats exhibiting intricate patterns are escalated to higher-tier analysts for in-depth investigations [68,69]. For example, Microsoft Copilot for Security Guided Response (CGR) (available for real-world deployment), an AI agent orchestrated tool, achieved an average macro-F1 score of 0.87, with 0.87 precision and 0.86 recall in triaging incidents, with critical misclassifications (true positives as benign or false positives) being rare (2.4%) [70]. In addition, Context2Vector addresses alert fatigue by leveraging context representation learning to improve event triage processes, reportedly doubling the attacker recall rate in certain scenarios [64]. Advancing beyond classification-only models, recent systems integrate LLMs with behavioral analytics to support more adaptive and interactive triage workflows. For example, HuntGPT combines anomaly detection with a GPT-3.5 conversational agent for adaptive threat triage and has demonstrated a success rate between 72% and 82.5% on standardized cybersecurity certification exams [27]. CyberAlly,

an LLM agent tested in simulated SOC environments, ingests and filters real-time events, performing classification, prioritization, and severity scoring, significantly reducing false positives and improving response times. For instance, CyberAlly's AI-driven triage halved false positives from 70% to 35% and reduced Mean Time To Respond (MTTR) from 8 h to 90 min, while also increasing automated ticketing from 10% to 75% [71]. LLMs, AI agents, and hybrid triage systems have significantly enhanced alert management by improving precision, reducing false positives, and accelerating response times through intelligent prioritization and contextual analysis; however, models like Llama-3.3-70B remain limited in dynamic environments due to high false positive rates, inconsistent generalization, and challenges in aligning automated outputs with evolving threat contexts [43].

3.3. Threat Intelligence

Cyber Threat Intelligence (CTI) is a specialized area of cyber defense focused on identifying, evaluating, and analyzing threats to organizational systems. Effective CTI collects threat data, extracts actionable insights, and integrates them into security operations. LLMs, such as GPT models, enhance Open-Source Intelligence (OSINT) by automating the analysis of historical cyber incident reports, improving both intelligence accuracy and threat forecasting. LOCALINTEL exemplifies the use of LLMs for generating organization-specific threat intelligence by contextualizing global threat repositories with local knowledge databases [2,54]. Multiple tools now integrate LLMs with knowledge graph generation, such as LLM-TikG, CTINEXUS, and AGIR, to structure unstructured data and automate reporting via STIX or CSKG formats, leading to more accurate anticipation and mitigation of cyberattacks [28,49]. For CTI to be actionable, it must be relevant, timely, accurate, complete, and easily ingestible by recipient systems [31]. LLM-based systems designed for CTI delivery, such as IntellBot, have demonstrated high performance in these areas, achieving a Context Precision of 0.934 and Context Recall of 0.933 for vulnerability-related queries, with overall RAGAS evaluation metrics consistently above 0.77 [72]. CyLens utilizes agentic LLMs to redefine CTI, encompassing tasks like attribution, behavior analysis, prioritization, and countermeasure development, along with curating CVE-centric threat reports [13]. This system incorporates knowledge from 271,570 threat reports and consistently outperforms industry-leading LLMs and state-of-the-art cybersecurity agents, achieving, for instance, 83.74% accuracy for threat actor attribution and 90.03% BERTScore for threat impact descriptions [13]. CTINEXUS achieved overall F1-scores of 87.65% in cybersecurity triplet extraction and notably increased the F1-score by 25.36% over EXTRACTOR for this task, with its entire experiment costing less than $0.30 [28]. IntelEX focuses on extracting attack-level threat intelligence Tactics, Techniques, and Procedures (TTPs) with contextual insights, employing chunking mechanisms, tailored prompts, and external vector databases for MITRE ATT&CK techniques. This approach has been shown to identify 3591 techniques and achieved an average F1 score of 0.792 for technique identification, substantially outperforming state-of-the-art AttackKG by 1.34x, with some instances reaching an F1 score of up to 0.902 in real-world applications [73]. Fine-tuned LLM frameworks like AECR automatically extract attack technique intelligence from CTI reports, improving accuracy and F1-score by over 60% compared to traditional NLP-based extraction [74]. LLM-Assisted Proactive Threat Intelligence integrates LLMs and RAG systems with continuous threat intelligence feeds to enhance real-time cybersecurity detection and response capabilities [75]. Despite strong benchmark performance, CTI systems using LLMs require real-world validation to address risks like hallucination, data leakage, and inconsistent generalization [20,29].

3.4. Ticket Handling

Ticket handling, or incident management, is available for real-world deployment and is essential in SOCs for maintaining service quality by meeting Service Level Agreements (SLA), reducing manual effort, prioritizing tickets, and recommending resolutions. Traditional methods such as ticket correlation and rule-based problem solution mapping often face limitations in efficiency and scalability. For security applications, the focus typically centers on prioritizing tickets based on incident type and severity, whereas in IT Service Management (ITSM) scenarios, clustering is performed based on issue root cause and solution similarity. In practical ITSM implementations, semantic similarity in Natural Language Processing (NLP) must be augmented with spatial and temporal factors, including device topology, timings, data source, and dynamic cluster size, to achieve high fidelity in ticket grouping [38]. AidAI, AI orchestrated agent, streamlines incident diagnosis in cloud environments by generating tickets, building hierarchical taxonomies, and using historical databases to identify recurring failure patterns resolving 51.4% of incidents [41]. TickIt uses LLMs for automated escalation, reducing alerts by 30% and decreasing manual workload enhancing both efficiency and user satisfaction. Ticket-BERT utilized a curated dataset of 76,000 raw tickets from Microsoft Kusto to label incident management tickets, involving comprehensive cleaning and processing of text data to handle diverse incidents effectively. Ticket-BERT, LLM-driven, demonstrated nearly 90% accuracy on a set of hard-to-identify tickets, which are difficult for human annotators to label quickly because they do not express specific incident issues [42]. The integration of LLMs and AI agents in ticket handling has been actively explored, with generative AI technologies successfully integrated into ticket management systems to streamline processes, offering capabilities like clustering, prioritizing, and providing resolution recommendations. These architectures integrate with platforms like ServiceNow and Splunk via robust APIs and microservices, leveraging resources such as over 1600 Search Processing Language (SPL) rules; therefore, an average reduction of 30% in alerts was observed [38,76]. LLexus represents an agent-based AI system specifically designed to automate the execution of Troubleshooting Guides (TSGs) for incident mitigation, using LLMs to generate plans from documents. It assumes the processing of 500 KB of data per incident, which corresponds to approximately 50,000 tokens, at a cost of about $0.5 [77]. Systems like AidAI and TickIt have demonstrated strong results, resolving over 50% of incidents and reducing alert volume by 30%. AI-driven systems shift ticket handling from reactive workflows to proactive, scalable operations. Strengths of these AI-driven approaches include substantial reductions in alert volumes, enhanced ticket prioritization, and automation of root cause analysis, leading to improved response times and customer satisfaction. However, limitations persist in terms of data privacy concerns, model interpretability, and the challenges of maintaining high performance across heterogeneous environments and incident types.

3.5. Incident Response

Incident Response (IR) is a foundational cybersecurity process involving detection, response, and recovery to minimize cyber attack impact and restore operations. Achieving efficient IR necessitates timely decision-making, cross-functional collaboration, and rapid adaptation to evolving threats. Traditional IR methods, which depend on manual protocols and expert input, struggle with modern threat complexity challenges that AI addresses through real-time detection, predictive analytics, and automation [64]. AI applications in IR include optimizing the handling of security breaches and automating key response tasks [78]. These processes are vital given that organizations, on average, take 204 days to identify a breach and an additional 73 days to contain it, totaling 277 days [3]. The AI agent AidAI streamlines AI incident diagnosis and reporting in cloud environments, generating

incident tickets with initial investigations for unresolved issues and building domain-specific knowledge bases from historical records, achieving an average Micro F1 score of 0.854 and a resolution rate up to 86.3% for incidents [41]. LLM-powered incident response tools like AttackGen can automatically generate incident response scenarios based on industry type, attack vectors, and organization size, helping organizations prepare for and prevent external threats by providing incident reports and playbooks for user training [65]. IRCopilot represents a novel LLM-driven framework that mimics the dynamic phases of real-world incident response teams using collaborative LLM-based session components, specifically designed to reduce issues like hallucinations and context loss [47]. It has demonstrated significantly better performance than baseline LLMs, achieving sub-task completion rates up to 150% higher than directly applying GPT-4, and successfully resolving the vast majority of incident response tasks in real-world challenges [47]. Multi-agent collaboration frameworks leverage LLMs to simulate human-like agents that coordinate investigations, identify attack patterns, and recommend effective countermeasures [48,79]. TrioXpert represents an end-to-end incident management framework for microservice systems that uses LLM collaboration for multimodal data preprocessing, multi-dimensional system status representation, and collaborative reasoning in anomaly detection, failure triage, and root cause localization, achieving performance improvements of 4.7% to 57.7% in anomaly detection, 2.1% to 40.6% in failure triage, and 1.6% to 163.1% in root cause localization, with an average end-to-end diagnosis completed within 15 s [80]. An evaluation of LLM and AI agent assisted incident response in SOC workflows found that, while autonomous LLMs often omit or hallucinate details, human–AI collaboration significantly improves report readability and operational efficiency [81]. However, limitations persist in the widespread adoption of AI-driven IR systems including depending heavily on training data quality, making them vulnerable in dynamic environments [64,82] and requiring human oversight.

3.6. Report Generation

Report generation involves the automated creation of structured or human-readable outputs, a task significantly enhanced in cybersecurity by LLMs, which produce diverse content including detailed reports [65]. Their strength in summarization and contextual understanding makes LLMs well-suited for automated report generation in cybersecurity. The AI agent, AGIR, an NLG tool for automating cyber threat intelligence reporting, effectively creates cybersecurity reports from structured data using STIX graphs and LLMs. It achieves a recall of 0.993 and 1.000 precision (indicating no hallucinations) [49], and significantly reduces report writing time by 42.6% [49]. On the other hand, LLM-powered tools like AttackGen can automatically generate incident response scenarios and comprehensive threat intelligence reports for organizations, including playbooks for user training [65]. In a case study, human experts evaluated its generated plans, which received scores of 3 out of 5 for clarity and specificity, indicating that while plans were coherent, they sometimes lacked the specific, detailed instructions for humans to follow [65]. Studies indicate that LLMs, specifically GPT models, can generate cybersecurity policies that outperform human-generated ones in terms of completeness, effectiveness, and efficiency [65]. The summarization module in CyLens, designed to generate high-level, human-readable briefings from complex threat reports [29], demonstrated strong quantitative performance. When generating "threat impact descriptions" on historical threats, CyLens-8B achieved a BERTScore of 90.03%, outperforming CyLens-70B which scored 87.33% [13]. Similarly, LLM-BSCVM, a vulnerability management framework, can generate detailed repair suggestions and corresponding contract code, considerably shortening the generation time compared to manual audit reports [57]. Microsoft's enterprise CTI framework integrates tools like Copilot for Security (available for real-world deployment) and Azure Logic Apps, reducing

report generation time from 8 h to under 2 h, with 90.2% IoC extraction accuracy and 85.7% for APT identification [30]. These frameworks can produce comprehensive reports that include sections on Metadata and Overview, MITRE Summary Tables, Data Extraction, Tools and Malware, Defense Recommendations, References, and Tags [30]. GenDFIR, a framework combining Rule-Based AI (R-BAI) algorithms with Large Language Models (LLMs) to automate cyber incident timeline analysis and generate detailed incident reports, demonstrated strong performance in evaluations. It achieved an overall accuracy of 97.51% across various metrics, including 95.52% accuracy in report facts, 94.51% relevance, 100% exact match, and 100% Top-K evidence retrieval [26]. The consistently high F1 scores, precision, recall, and accuracy metrics such as IntelEX's peak F1 score of 0.902, CyLens's 90.03% BERTScore, and IntellBot's Context Precision of 0.934 underscore the strong potential of LLM-based CTI systems to enhance automated threat detection, attribution, and reporting pipelines at scale. LLMs and AI Agents significantly advance SOC operations by enabling context-aware, detailed reporting. However, challenges remain regarding the accuracy and reliability of automatically generated reports, particularly concerning the potential for hallucinations and the need for human validation of critical information.

3.7. Asset Discovery and Management

Asset discovery and management involve continuously identifying, monitoring, and securing valuable organizational resources across their lifecycle [3,29]. Accurate asset discovery and management are essential to maintaining situational awareness in SOCs, especially as AI integration increasingly relies on real-time visibility into dynamic Information Technology/Operational Technology (IT/OT) systems [64]. ReliaQuest's GreyMatter, which is available for real-world deployment, integrates agentic AI for alert triage, asset visibility, and response automation processing alerts 20 times faster than traditional methods, automating 98% of alerts, and reducing containment time to under 5 min [3]. Accurate asset discovery and management is essential to maintaining situational awareness in SOCs, especially as AI integration increasingly relies on real-time visibility into dynamic IT/OT systems achieves 97.5% detection accuracy with a 30% improvement in response time for data poisoning attacks, and over 90% accuracy with 1–2.5% false positives for ransomware analysis [15]. In hybrid systems, AI agents would continuously monitor networks for new or changing assets, feeding this data to LLMs for analysis. LLMs can analyze unstructured data like configuration files or traffic logs to classify assets, assess criticality, and generate security policies [52,55]. AssetOpsBench envisions AI agents autonomously managing industrial asset operations and maintenance, including condition monitoring and maintenance planning, which inherently involves managing asset data and configurations through LLMs [53]. This combined approach would create dynamic and proactive asset management systems, enhancing overall security postures, though direct application in large-scale, dynamic asset discovery remains an area of nascent research within the current technological landscape. LLMs and AI agents offer promising capabilities for asset discovery and management through automated analysis, classification, and policy generation. Despite promising use cases, LLMs are not yet optimized for real-time asset discovery in dynamic environments, often struggling with topology interpretation, ambiguous identifiers, and stateful analysis.

3.8. Vulnerability Management

Vulnerability management is a core cyber defense process that identifies, prioritizes, and mitigates system weaknesses before they can be exploited. It typically uses automated and manual scans to detect known issues like weak passwords or unpatched software. The output of vulnerability assessments includes detailed reports outlining vulnerability

severity, potential impact, and recommended remediation actions, guiding security teams in making informed decisions. The AI agent-orchestrated LLM-BSCVM constitutes a significant advancement as an end-to-end vulnerability management framework designed for smart contracts, leveraging a multi-agent collaborative approach for vulnerability detection, root cause analysis, repair recommendations, risk assessment, and audit reporting [57]. It achieved a vulnerability detection accuracy and F1 score exceeding 91% on benchmark datasets. It reduced the false positive rate from 7.2% in state-of-the-art (SOTA) methods to 5.1%, significantly decreasing error alarms and improving the precision and feasibility of vulnerability repair [57]. LLMs are increasingly being applied in this domain to assist in software code evaluations, effectively identifying security vulnerabilities [2,83]. LLM-based tools like DefectHunter and others use attention models, semantic reward, configuration validation, and reinforcement learning to patch vulnerable code [84]. Their effectiveness can be enhanced for specific domains such as IoT using datasets like QEMU, Pongo-70B, and CWE-754 [65]. The LLM-driven LProtector demonstrates the effectiveness of integrating GPT-4o with RAG and Chain-of-Thought (CoT) reasoning for vulnerability detection. It achieved 89.68% accuracy and 33.49% F1 scores on 5000 balanced Big-Vul samples, outperforming established tools [85,86]. ProphetFuzz introduces an LLM-based fuzzing system that predicts high-risk option combinations using only software documentation, achieving 32.85% higher vulnerability discovery than traditional fuzzers [87]. CASEY, a hybrid AI agent that leverages LLM, automates the identification of Common Weakness Enumerations (CWEs) of security bugs and assesses their severity, employing prompt engineering techniques and incorporating contextual information at varying levels of granularity to streamline the bug triaging process. CASEY achieved a CWE identification accuracy of 68% and a severity identification accuracy of 73.6%. Its combined accuracy for identifying both CWE and severity was 51.2% [59]. The integration of LLMs and AI agents represents a rapidly developing area in vulnerability management, with several realized and proposed hybrid solutions. LLMs support tasks such as vulnerability detection, behavior analysis, and synthetic data generation [3,65]. Agent-based tools, which use coordinated AI agents to systematically explore and exploit potential vulnerabilities, are being developed to test hybrid systems and identify novel attack vectors at the interfaces between AI and non-AI components [33]. These integrations of AI and LLMs are transforming vulnerability management into more automated, efficient, and proactive defense mechanisms against evolving cyber threats, while empirical data further shows that collaborative, LLM-supported remediation processes enhance user engagement and reduce remediation duration in real-world SOC environments [46,88,89]. Despite promising advancements, current LLM and agent-based vulnerability management systems face challenges in maintaining consistent performance across heterogeneous environments, where variations in code structure, context depth, and real-world unpredictability often hinder generalization and lead to overlooked edge-case vulnerabilities. The LLM/AI Agents methods explored in our survey paper listed in Table 2 utilize these techniques. The workflow of AI-Augmented SOC is displayed in Figure 2.

Table 2. Comparison of Conventional Approaches vs. AI-Augmented Methods. Note: This table highlights key features only and does not include all details from the referenced works.

SOC Task	Conventional Approaches	LLM Methods	AI Agent Methods	Key Quantitative Metric	Key Qualitative Finding
Log Summarization	Manual Review [18], Log Parsing [21], Rule-based systems, Source-code based methods [62], Manual regex patterns [90]	CYGENT (GPT-3.5, GPT-3 Davinci) [12], LogPrompt [18], LibreLog [14], LogParser-LLM [10]	CYGENT (as conversational agent) [12]	LogParser-LLM required only 272.5 LLM calls for 3.6M logs, GPT-3 Davinci outperformed other LLMs [10]	LLMs outperform manual analysis; LibreLog reduces LLM query load; CYGENT [10] showed data generalization issues
Alert Triage	Manual triage [76], Rule-based correlation [3], SIEM systems (80% false positives) [1]	LLMs for NIDS rule labeling [35], incident summarization [32], prioritization [1]	ReliaQuest agent [91], ContextBuddy [82], multi-agent triage systems [39]	ReliaQuest: 20×faster, 98% alert automation, 5 min containment, 30% improved detection [91]	Reduced alert fatigue and manual burden [1], enhanced contextual understanding [52]
Threat Intelligence	Manual analysis from diverse sources [92], traditional NLP [22], rule-based systems [93]	CTI extraction [2], IntellBot [72], CTINexus [28], LANCE [36], IntelEX [73], LocalIntel [54]	CyLens [13], IntellBot agents [72], LANCE engine [36], Multi-agent CTI extractors [31]	IntelEX F1 up to 0.902 [69], IntellBot: BERT >0.8 [72], CTINexus recall/precision ↑10% [28]	LLMs reduce CTI creation time by 75–87.5%, hallucination [30], low precision in decoder-only models still challenges [23]
Ticket Handling	Manual categorization [1] and resolution, rule-based mapping [3]	LLMs for grouping [1,3], prioritization, Ticket-BERT for fine-grained labeling [42]	Unified microservice agent architectures [38,83]	Rand score 0.96 for clustering [38], Ticket-BERT outperforms baselines [42]	LLMs reduce delay [94], traditional methods inefficient under volume [38,95]
Incident Response	Manual response protocols [1], AIOps with limited scope [4], isolated management [64]	GenDFIR [26], IRCopilot [47], LLexus [77], LLM-BSCVM [57]	AidAI [41], AutoBnB, Audit-LLM [79], Multi-agent IRCopilot [47]	6 faster detection/mitigation; task completion time ↓30.69% (IT Admins) [4]	LLMs enhance planning, IRCopilot has hallucination/context issues [47], human oversight remains essential
Report Generation	Manual CTI report writing [1], data aggregation, prone to errors [30]	GPT models for CTI summary [2], AGIR [49], Microsoft Copilot [96], LLM-BSCVM [57]	AidAI [41], multi-agent CTI generators [49], autonomous audit agents [57]	AGIR recall: 0.99, report time ↓42.6%, CTI effort ↓75–87.5% [49]	AI reduces manual workload [1], outputs need review for consistency [2], TTP accuracy still lower than human reports [30]
Asset Discovery and Management	Manual monitoring, planning and interventions [64]	LLMs [94]	AssetOps agent + specialized IoT/maintenance agents [53]	gpt-4.1 scored 100% in FMSR, llama-4-maverick excelled in WO tasks [53]	Enables end-to-end lifecycle automation, WO tasks still depend on structured comprehension [53]
Vulnerability Management	Manual bug triaging [38], static analysis tools [45]	LLMs for prediction and CWE/severity assessment (CASEY) [59]	ATAG agent [97], multi-agent IaC analyzers [88], LLM-BSCVM [57]	CASEY: CWE accuracy 68%, severity 73.6%, combined 51.2% [59]	LLMs outperform static tools [88], documentation still emerging [97], privacy concerns persist [26]

(↑) means increase and (↓) means decrease.

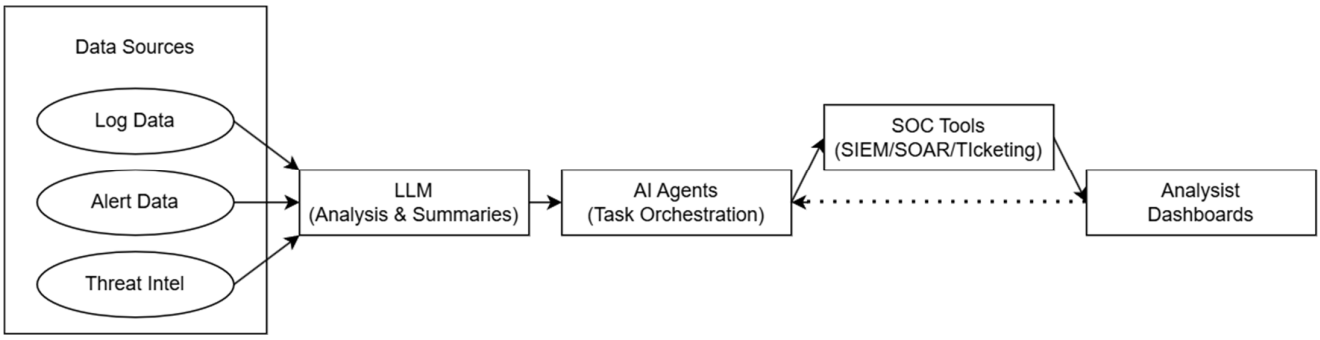

Figure 2. End-to-end AI-augmented SOC dataflow. Solid arrows represent automatic data movement. The dotted arrow represents analyst-driven data movement.

4. Capability Maturity Model

The deployment of AI agents and LLMs within SOCs can be classified into a five-level capability-maturity framework, illustrating progression from fully manual to fully autonomous operations. At Level 0 (Manual Operations), human analysts rely entirely on predefined rule sets and manually manage security alerts without AI or LLM involvement [26]. Level 1 (AI/LLM-Assisted Operations) introduces AI-driven decision support, such as alert prioritization and initial triage suggestions, with analysts maintaining complete control and verification. Examples include Microsoft's Copilot for incident classification [70], and LocalIntel for generating organization-specific threat intelligence [54]. Level 2 (Semi-Autonomous Operations) features AI systems integrated with Security Orchestration, Automation, and Response (SOAR) tools or LLMs automating routine tasks like alert filtering and ticket generation, with explicit human approval required for critical or ambiguous cases. Representative examples include LogGPT, which processes raw logs [15]. Level 3 (Conditionally Autonomous Operations) sees AI and LLMs independently managing complex analyses, attack path reconstructions, and comprehensive reporting, with analysts intervening primarily for review and approval of critical actions, aligning with Human-on-the-Loop (HoTL) models. Examples are CYGENT for automated reporting [12], and TickIt for ticket prioritization [44]. Level 4 (Fully Autonomous Operations) involves highly autonomous AI and LLM systems managing the complete incident lifecycle with minimal human involvement, shifting human roles to governance and strategic oversight. While AssetOps Agent illustrates this concept through global coordination [53], fully autonomous LLM-based solutions currently remain theoretical, emphasizing the need for further research into robust governance and ethical standards [3]. This adaptive autonomy spectrum utilizes Human-in-the-Loop (HITL), Human-on-the-Loop (HoTL), and Human-out-of-the-Loop (HoOTL) models, visually depicted in Figure 3.

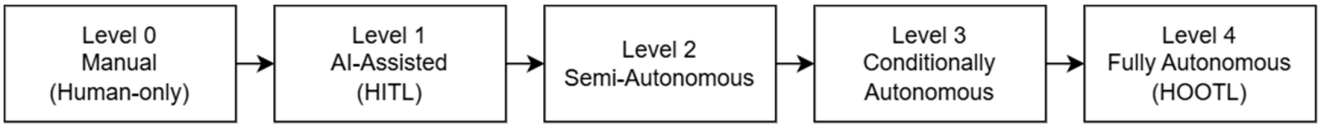

Figure 3. Five-level autonomy ladder for AI-enabled SOCs.

4.1. Relevance Compared to SOC-CMM

The proposed AI-Augmented SOC Capability Maturity Model draws conceptually from established frameworks such as SOC-CMM, which assess organizational maturity across Business, People, Process, Technology, and Services domains. SOC-CMM provides a general baseline for process and governance maturity but does not explicitly account

for AI/LLM-enabled autonomy, human-AI teaming dynamics, or adaptive automation levels. Our model therefore acts as a domain-specific extension to SOC-CMM, introducing an autonomy dimension that characterizes the integration depth of AI agents and LLMs within SOC workflows. Each of our five levels (Manual → Assisted → Semi-autonomous → Conditionally autonomous → Fully autonomous) can be mapped to SOC-CMM process-maturity levels by aligning Level 0–1 ↔ SOC-CMM Maturity 1–2 (Initial/Defined processes), Level 2–3 ↔ SOC-CMM Maturity 3–4 (Managed and Quantitatively Managed operations), and Level 4 ↔ SOC-CMM Maturity 5 (Optimizing) while adding AI-specific criteria such as model governance, data provenance, explainability, and autonomy-safety controls. Consequently, the proposed model should be interpreted not as a replacement for SOC-CMM but as an AI-focused overlay that quantifies automation maturity and trust calibration within next-generation, hybrid SOCs. It also quantifies how those capabilities are executed (manual → assisted → semi-autonomous → conditionally autonomous → fully autonomous).

4.2. Validity of Capability Maturity Model

To ensure that the proposed AI-Augmented SOC Capability Maturity Model is both credible and practically applicable, future work will focus on a multi-phase validation process. Content validity can be established through expert elicitation and alignment with established frameworks such as SOC-CMM, SIM3, and NIST CSF to verify the conceptual soundness of the autonomy levels and evaluation dimensions. Construct validity will be empirically assessed by applying the model to both simulated SOC environments (e.g., Wazuh, Security Onion, Elastic Stack) and operational settings, examining correlations between maturity scores and measurable performance indicators such as Mean Time to Detect (MTTD), Mean Time to Respond (MTTR), and false-positive rates. Reliability can be confirmed through inter-rater assessments to ensure consistent scoring across evaluators. Longer-term validation may include longitudinal studies to determine whether increases in maturity levels predict sustained improvements in SOC performance and human-AI collaboration effectiveness. The AI-augmented task-specific implementations are classified in Table 3.

Table 3. Classification of AI-Augmented SOC Maturity Model. Note: This table highlights key features only and does not include all details from the referenced works.

Category	Agent/System	Topology	Autonomy Level	Primary Data Source
Log Summarization	CYGENT [12]	AI Agent	1	Uploaded Log Files
	LibreLog [14]	LLM	3	LogHub-2.0 dataset
	LogBatcher [62]	LLM	3	Public Software Log
Alert Triage	Microsoft Copilot for Security Guided Response (CGR) [70]	AI Agent	1	Microsoft Defender alerts and telemetry
	CyberAlly [3]	AI Agent	2	Network telemetry and endpoint event data
	sHuntGPT [27]	LLM + AI Agent	3	Anomaly detection engine outputs
Threat Intelligence	LocalIntel [54]	AI Agent	1	CTI Reports
	CyLens [13]	LLMs	2	Event Logs
	CtiNexus [28]	LLMs	3	CTI Reports
Ticket Handling	LLexus [77]	AI Agent	3	Generated incidents
	AidAI [41]	LLMs	3	Historic data
	TickIT [44]	LLMs + CoT	3	Customer Support tickets and dialogue

Table 3. *Cont.*

Category	Agent/System	Topology	Autonomy Level	Primary Data Source
Incident Response	AidAI [41]	AI Agent LLMs + CoT	3	Historical Ticket Content
	IRCopilot [47]	LLMs	3	User Activity Logs
	TrioXpert [83]	Multi-agent LLM	3	Event Logs, D1 and D2 datasets
Report Generation	AGIR [49]	LLM + NLG	3	Intelligence sources
	GenDFIR [26]	LLM + RAG	3	Incident events
	AttackGen [65]	LLM	3	Threat intelligence data
Asset Discovery and Management	AssetOps Agent [53]	Multi-Agent + Global Coordinator	2	Multi-modal data
	GreyMatter [91]	AI Agent	3	Security alerts and incident response data
	SYNAPSE [98]	Multi-layer toolset AI Agents	1	Raw events and evidence
Vulnerability Management	LLM-BSCVM [57]	Multi-agent + RAG	2	TrustLLM and Dappscan
	LProtector [25]	LLM	2	National Vulnerability Database (NVD)
	CASEY [59]	LLM + AI Agent	2	Augmented NVD

5. Challenges

Despite significant advancements in integrating AI and LLMs into SOCs to augment human capabilities, several formidable challenges and open research issues persist across integration, operational, model, and data domains. Addressing these is crucial for realizing the full potential of AI-driven SOCs. These challenges include the lack of standardized interfaces for seamless AI-human collaboration, and the difficulty of aligning AI outputs with the nuanced decision-making processes of seasoned analysts. Moreover, concerns around data privacy, adversarial robustness, and the explainability of LLM-driven decisions further complicate their trustworthy adoption in mission-critical SOC environments. Table 4 showcases the strengths and limitations of AI automation within SOC tasks.

Table 4. Strengths and Limitations of AI Agents and LLMs in automating SOC Tasks. Note: This table highlights key features only and does not include all details from the referenced works.

Evaluation	Strengths of AI Agents	Limitations of AI Agents	Strengths of LLMs	Limitations of LLMs
Scalability	Adapt well to increasing data volumes and SOC complexity [57].	Scalability can be hindered by growing communication overhead [99].	Generalize across domains and scale efficiently [2].	Require high compute for deployment and fine-tuning [2].
Interpretability	Providing context-aware, human-comprehensible insights and audit opinions via CoT reasoning [3,5]	Limited due to the "black-box" nature of some models and the potential for cascading errors from hallucinations [1,3]	Enabling natural language interactions for insights and summarized reports, increase accuracy using CoT prompting and RAG [37,49,65]	Susceptibility to hallucinations and output variability [30,100,101]
Latency & Efficiency	Efficient processing, multi-agent collaboration, high volume cybersecurity operations [1,3,4]	High computational resources, scalability of the number of AI Agents, the need of high-quality training data [3,64,91]	Efficient processing and analyzing vast diverse data, automating complex tasks, generating structured insights [1,64,69]	High computational resources, probabilistic nature leading to output variability [2,64,93]

Table 4. *Cont.*

Evaluation	Strengths of AI Agents	Limitations of AI Agents	Strengths of LLMs	Limitations of LLMs
SOC Integration	Streamline SOC workflows by autonomously managing tasks and coordinating responses across security tools [3]	Complex dynamics [33], emergent behaviors, communication overhead, and hindering predictability [97]	Enhance integration through natural language understanding [3], automated reporting, and context-aware insights [95], streamlining information [6]	Demand high computational resources and produce variable outputs, requiring human oversight and extensive fine-tuning for precision [52]
Human-AI Teaming	Can autonomously detect, classify, and respond to threats in real time, significantly reducing TTD [29].	Vulnerable to attacks like prompt injection and memory poisoning [4].	Can process vast amounts of unstructured security data [82].	Can suffer from factual errors or "hallucinations" [3].
Privacy & Security	AI agents automate tasks in secure, private data environments [99]	Multi-agent systems introduce complex vulnerabilities [33], risking sensitive data exfiltration [97]	Enhance security via threat detection [2], incident response, and secure code generation [6]	Risk sensitive data exposure, hallucinations, and adversarial attacks [2,101]

5.1. Integration Challenges

The successful implementation of AI agents and LLMs in SOCs is often hampered by difficulties in integrating new technologies with existing infrastructure and workflows, as well as managing the dynamic operational environment [66], legacy system incompatibility and fragmented SIEM interoperability necessitating substantial effort in developing new APIs, middleware, or components, which incurs considerable cost and complexity [38]. A significant challenge lies in ensuring that automated processes can adapt to constantly changing incident response procedures and security toolchains, and a lack of coordination between technical and non-technical personnel can hinder effective integration [96,102]. The hybrid integration of LLMs and AI agents into cloud security operations, often relies on frameworks such as SOAR, EDR, XDR, LangChain, AutoGen, and interoperability protocols like the Model Context Protocol (MCP), Agent-to-Agent Protocol (A2A), and Agent Communication Protocol (ACP) [3,64]. These challenges are further compounded by the architectural complexity and ambiguity inherent in multi-agent LLM systems, the difficulty in managing consistent context and alignment across free-form communication protocols, and overcoming semantic ambiguities or functional overlaps that arise when integrating disparate data sources and APIs [99,103]. Such integrations can introduce scalability bottlenecks due to communication overhead, and present novel security risks, including Agent-in-the-Middle (AiTM) attacks, data leakage, and malicious prompt injections, requiring standardization and secure communication protocols to manage complex deployments in dynamic cybersecurity environments [52]. To solve SOC integration challenges, training environments for autonomous cyber defense agents, such as Cyberwheel, are designed to allow agents to ingest alerts from existing detectors and align with popular cyber-detection tools like host and network intrusion detection systems, requiring only a translation layer for real-world deployment [104].

5.2. Operational Challenges

Scalability remains a significant hurdle, as the immense volume and complexity of data generated in modern IT environments challenge both AI models and traditional SIEM systems. Multi-approach and ensemble methods show promise, but scaling AI

agents in live cloud environments remains difficult [93]. LLMs require significant computational resources for large-scale data processing, such as knowledge graph construction and link prediction [60]. Adaptive parsing caches help improve efficiency by reducing duplicate LLM queries and overhead [21,62]. The constantly evolving nature of cyber threats necessitates that AI models are regularly updated and retrained to maintain their effectiveness. Rapid decision-making and coordinated responses are critical for incident response, which traditional manual methods struggle to provide against fast-moving, complex attack vectors, due to dynamic network topologies limiting real-time performance [6,13,83]. Achieving a balance between AI autonomy and human oversight is critical, requiring dynamic adjustments to autonomy based on task complexity to avoid automation complacency and maintain human accountability. Human factors, including alert fatigue, burnout, and skepticism toward automation, significantly influence SOC efficiency. Research into optimal human-AI collaboration strategies and personalized trust models is still needed, as LLMs may exhibit non-deterministic behavior and biases, requiring human verification and management of outputs [41,102].

5.3. Model-Related Challenges

Model related challenges, such as "black box" problems, inherent to the AI and LLM models themselves significantly impact their reliability, performance, and trustworthiness in SOC applications. The limited explainability undermines trust and accountability, as analysts need to justify actions based on AI recommendations. While XAI techniques aim to provide transparency by capturing contributing factors and providing justifications, their acceptance and practical application by incident responders remain limited due to complexity [83]. This necessitates continuous updating and retraining of models and raises critical concerns about the robustness of AI-driven systems against such attacks, with some studies indicating that current AI agent frameworks in cloud security have not yet fully addressed these specific threats. LLMs exhibit critical reliability vulnerabilities, being simultaneously susceptible to adversarial attacks, such as covert message exchanges (secret collusion) or subtle lexical manipulations, and prone to factual inaccuracies or "hallucinations" that generate misleading or fabricated information. These dual weaknesses highlight the need for continuous self-verification mechanisms and robust defense strategies extending beyond basic paraphrasing or prompt filtering [93]. Such inaccuracies are unacceptable in security operations, often necessitating human oversight to verify outputs, which is particularly challenging for automated CTI analysis where errors can have severe consequences [13]. Strategies like structured prompt engineering with "Role, Goal, Constraints, Instructions, Example" principles, explicit instructions to respond with "I don't know" when uncertain, and employing an "LLM as a judgment" module are being explored to mitigate hallucinations and improve factual accuracy [75,83]. While LLMs show great potential, domain-specific AI models, including LLMs, have demonstrated inadequacies in generalizing across diverse security contexts. Customization and fine-tuning are often required to achieve desired performance for specific cybersecurity tasks, such as complex system diagnosis or network-specific problem-solving, as generic embeddings from models trained on non-network specific data may not suffice [90,95].

5.4. Data-Related Challenges

Challenges primarily related to the acquisition, quality, privacy, and volume of data are critical for training and operating AI and LLM models in cybersecurity. The scarcity of accurately labeled data, stemming from privacy concerns, variability of cyber threats, and labeling complexity, significantly hampers AI training [15,105]. The inherent heterogeneity of security data, spanning structured logs, semi-structured telemetry, and unstructured

threat reports, creates significant challenges for AI model ingestion, integration, and analysis [38,52]. Variability in formats, data quality, and evolving threat contexts further undermines model generalization, underscoring the need for adaptive AI solutions capable of handling inconsistent and dynamically changing data inputs [16]. Low-quality or inconsistent data, such as unclear incident descriptions, also reduces AI effectiveness and necessitates manual review [41,102]. Privacy risks, including unintended exposure of sensitive information by generative AI, are pressing concerns, spurring research into privacy-preserving AI methods and open-source alternatives [15,16]. The overwhelming volume and complexity of security data, especially lengthy CTI reports and verbose LLM outputs, exacerbate information overload and complicate incident response [83,106]. Data poisoning and model manipulation risks challenge LLMs for web security defense in SOCs [107]. Approaches involving string similarity and dynamic data structuring are being explored to address these inconsistencies [15].

5.5. Integrated and Critical Discussion of Challenges

The four categories of challenges discussed above, integration, operational, model-related, and data-related, are highly interdependent rather than isolated technical issues. Integration difficulties, such as inconsistent APIs and legacy architectures, limit the availability of reliable and timely data for model training, thereby aggravating data-quality and heterogeneity problems. Poor or inconsistent data in turn amplify model-related issues such as hallucinations, bias, and unstable performance, which undermine analysts' confidence in AI-generated outputs. When model behavior is opaque or unpredictable, operational decision-making becomes more cautious or inconsistent, reinforcing a cycle of human mistrust and reduced reliance on automation. Conversely, excessive automation without interpretability can lead to over trust, complacency, and reduced situational awareness. These feedback loops create a socio-technical fragility within the AI-augmented SOC, where deficiencies in one domain (e.g., data quality) cascade into others (e.g., model transparency and human oversight). The erosion of trust caused by these interdependencies directly affects human, AI collaboration. Analysts confronted with non-deterministic or unexplainable AI behavior hesitate to delegate critical decisions, while AI systems deprived of accurate feedback fail to learn and calibrate appropriately. The absence of standardized human-in-the-loop protocols, coupled with fragmented accountability frameworks, further deepens this trust gap. Therefore, addressing these challenges requires an integrated strategy that jointly advances technical robustness (e.g., explainability, adversarial resilience), process maturity (e.g., clear human-AI interaction protocols), and organizational readiness (e.g., AI literacy and governance). Only through coordinated improvements across these layers can SOCs achieve calibrated trust and effective human–AI teaming.

6. Future Directions

The future landscape of cybersecurity will be significantly shaped by advanced integration of LLMs and AI agents, fostering a shift from reactive to proactive defense and enabling more sophisticated human-machine collaboration. Future research will focus on enhancing model interpretability, ensuring robustness against adversarial threats, and scaling these technologies across complex environments, while refining trust calibration between human analysts and AI systems. In the immediate term, several academic-industry collaborations and pilot deployments are already underway to implement LLM-driven summarization, agent-based alert triage, and autonomous ticket routing within simulated SOC environments, providing empirical validation of these models under real-time constraints. In log summarization and analysis, ongoing work aims to improve LLMs' ability to provide interpretable anomaly explanations, as demonstrated by LogPrompt, which offers

human-readable justifications for detected issues [18,43]. Future methodologies will explore adaptive LLMs that can handle evolving log formats without constant retraining, building on approaches that achieved up to 0.96 parsing accuracy with LogParser-LLM, and further reducing false positives, with multi-agent systems like Audit-LLM already showing a 40% reduction in false positives for insider threat detection [15,16]. Similarly, advancements in alert triage and incident response will emphasize augmented human-AI collaboration through XAI and Reinforcement Learning from Human Feedback (RLHF) [3,29]. AI agents are projected to significantly reduce MTTD and MTTM, with simulations showing up to six times faster response than human intervention, as exemplified by Microsoft Copilot for Security's (2024-25 release) integration with SIEM and XDR tools to enhance automated response [4,70]. For ticket handling, research will refine LLM capabilities for optimizing grouping, prioritization, and resolution recommendations, leveraging AI-driven architectures that have achieved a Rand score of 0.96 in ticket clustering by incorporating spatial and temporal factors [38]. In CTI and report generation, the focus will be on autonomous data extraction and contextualization using RAG and multi-agent systems [23,98]. Projects like LocalIntel already demonstrate 93% accuracy in contextualizing threat intelligence at an organizational level, while AGIR has achieved a 42.6% reduction in report writing time with a 0.99 recall and no hallucinations [49,54]. Further work will explore fine-tuning LLMs for detailed TTP extraction, with TTPHunter already achieving over 90% F1 scores for various attack techniques [35]. For asset discovery and management, AI agents will aim to automate the full lifecycle management of industrial assets, as envisioned by frameworks like AssetOpsBench, which includes over 140 scenarios for evaluation [53]. In vulnerability management, research will investigate multi-agent, AI-driven strategies leveraging LLMs and RAG for automated detection and remediation in IaC, with reported detection rates of 85% [88]. Frameworks like LLM-BSCVM have achieved 91% accuracy in vulnerability detection and an F1 score, reducing false positives from 7.2% to 5.1%, necessitating future efforts in integrating symbolic execution and formal verification for higher precision [57]. The Agent Security Bench (ASB) framework highlights the growing need for rigorous benchmarking of adversarial scenarios, suggesting that future research on automating SOC tasks with AI agents and LLMs must incorporate standardized evaluations of both attack resilience and defense strategies to ensure operational robustness in real-world deployments [98]. Organizations implementing LLMs and AI agents in SOCs must address key real-world deployment considerations. These include the complexity of managing AI-driven systems such as integration with existing security tools and ensuring interoperability as well as demands for real-time threat detection and swift action, which challenge latency constraints. The integration of AI agents and LLMs into SOCs is actively addressing its challenges. Human-AI collaboration frameworks are being developed to balance automation with human oversight, thereby reducing alert fatigue and fostering trust [1,88]. To tackle transparency limitations and factual errors like hallucinations, XAI and RAG are crucial, providing understandable reasoning and grounding LLM responses in real-time, domain-specific knowledge [108]. Additionally, modular and adaptive AI architectures help overcome compatibility issues with legacy systems and ensure continuous learning and updates against evolving cyber threats and data variability [37,99]. Additionally, workforce issues like continuous training and balancing human oversight with automation must be considered, alongside the critical responsibility of maintaining regulatory compliance and data privacy. These advancements collectively underscore a trajectory toward increasingly intelligent, autonomous, and human-collaborative cybersecurity operations.

7. Threats to Validity

Despite the breadth and methodological rigor of this survey, several threats to validity may influence the interpretation and generalizability of its findings. We outline five key categories of concern: selection bias, ecological validity, measurement inconsistency, evolving baselines, and human-AI integration uncertainty. While this survey draws from a curated set of 105 peer-reviewed and preprint sources across IEEE, ACM, arXiv, and several conference proceedings archives (NDSS, USENIX Security, ACM CCS, and IEEE Symposium on Security and Privacy) (the filtering criteria particularly the emphasis on recency (post-2022), and English-language publications may introduce a selection bias. This heterogeneity reflects the multidisciplinary nature of SOC automation research, encompassing both network-level telemetry and higher-level textual intelligence sources. However, the lack of standardized datasets across studies introduces significant variability in task difficulty, labeling quality, and evaluation metrics, which complicates cross-comparison and synthesis of results. On the other hand, regional deployments, proprietary industrial case studies, and domain-specific implementations (e.g., military or Operational Technology-based SOCs) may be underrepresented, potentially narrowing the global applicability of conclusions. Many included studies evaluate LLMs and AI agents within simulated or benchmarked SOC environments using synthetic data, constrained adversarial scenarios, or offline testbeds. These conditions often lack the volatility, noise, and ambiguity present in real-world deployments, particularly where coordination between tools, analysts, and incident response protocols introduces temporal dependencies and adversarial uncertainty. As such, reported performance metrics may not fully extrapolate to operational SOCs operating under regulatory or resource constraints. The surveyed literature exhibits substantial heterogeneity in evaluation metrics ranging from F1-scores and recall for CTI systems to subjective human trust assessments and latency measurements. Metric heterogeneity across primary studies may also influence comparisons, as precision, recall, and F1-scores were drawn directly from original evaluations using varying datasets and benchmarks. This lack of standardized benchmarks complicates cross-study comparisons and may obscure subtle trade-offs between precision, transparency, and runtime performance. Future work should prioritize unified evaluation frameworks, particularly for safety-critical SOC tasks like threat triage and vulnerability remediation. Given the rapid progression of LLMs and autonomous agent architectures, some referenced tools or findings may become outdated soon after publication. Models such as GPT-4, Claude 3, and Gemini are continuously updated, and capabilities like context retention, multi-agent coordination, or reasoning interfaces may shift substantially with newer versions. Therefore, the findings in this survey should be interpreted as a snapshot of a fast-moving field, with an expectation of obsolescence in benchmarks and system architectures. While this survey introduces a capability-maturity model for LLM/agent autonomy in SOCs, few empirical studies evaluate human-AI collaboration at scale in real-time operations. Variables such as trust calibration, false positive fatigue, accountability for misaligned decisions, and the ethical implications of semi-autonomous escalation remain poorly explored. Without longitudinal studies or operational audits, claims of productivity gains or safety improvements may overstate real-world readiness. Recognizing these limitations is essential to responsibly interpret the current state of AI-augmented SOCs and to guide future research toward more robust, real-world-ready solutions.

8. Conclusions

Across all the SOC tasks, AI-augmented systems consistently improve detection accuracy, reduce false positives, and accelerate response times, demonstrating measurable operational benefits over traditional rule-based methods. The proposed AI-Augmented

SOC Capability Maturity Model extends existing frameworks like SOC-CMM by introducing an autonomy dimension that classifies SOCs from manual to fully autonomous operation. Current implementations largely remain at early to intermediate stages of automation (Levels 1–2), emphasizing human oversight and decision support. Achieving higher autonomy levels will require advances in interpretability, robust data governance, and secure multi-agent coordination. Despite strong empirical progress, adoption barriers persist, including integration complexity, model transparency, and the calibration of trust between human analysts and AI systems. These socio-technical dependencies underline that sustainable SOC evolution depends not on full automation but on adaptive human–AI collaboration. Future research should focus on explainable and privacy-preserving models, standardized evaluation benchmarks, and empirical testing in operational SOC environments to validate real-world resilience. By aligning scalable AI capabilities with human expertise, next-generation SOCs can evolve toward intelligent, transparent, and trustworthy cyber defense ecosystems.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/jcp5040095/s1, Table S1: PRISMA 2020 Checklist. Ref. [109] has been cited in the main text.

Author Contributions: Data curation, Formal analysis, Investigation, Methodology, Visualization, Writing—original draft (under guidance), S.S., B.K., J.Z., M.E., M.B. and A.B.; funding acquisition, K.D.; Conceptualization, Supervision, Methodology, Project administration, Writing—guidance & review, N.A. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by California State University, San Bernardino, School of Computer Science and Engineering.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data was created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Binbeshr, F.; Imam, M.; Ghaleb, M.; Hamdan, M.; Rahim, M.A.; Hammoudeh, M. The Rise of Cognitive SOCs: A Systematic Literature Review on AI Approaches. *IEEE Open J. Comput. Soc.* **2025**, *6*, 360–379. [CrossRef]
2. Hassanin, M.; Moustafa, N. A Comprehensive Overview of Large Language Models (LLMs) for Cyber Defences: Opportunities and Directions. *arXiv* **2024**, arXiv:2405.14487v1. [CrossRef]
3. Mohsin, A.; Janicke, H.; Ibrahim, A.; Sarker, I.H.; Camtepe, S. A Unified Framework for Human AI Collaboration in Security Operations Centers with Trusted Autonomy. *arXiv* **2025**, arXiv:2505.23397v2. [CrossRef]
4. Chigurupati, M.; Malviya, R.K.; Toorpu, A.R.; Anand, K. AI Agents for Cloud Reliability: Autonomous Threat Detection and Mitigation Aligned with Site Reliability Engineering Principles. In Proceedings of the 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), Houston, TX, USA, 5–7 February 2025; IEEE: Piscataway, NJ, USA, 2025. [CrossRef]
5. Song, C.; Ma, L.; Zheng, J.; Liao, J.; Kuang, H.; Yang, L. Audit-LLM: Multi-Agent Collaboration for Log-based Insider Threat Detection. *arXiv* **2024**, arXiv:2408.08902v1. [CrossRef]
6. Gupta, M.; Akiri, C.; Aryal, K.; Parker, E.; Praharaj, L. From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy. *IEEE Access* **2023**, *11*, 80218–80245. [CrossRef]
7. IEEE Xplore. *IEEE Xplore Digital Library*; IEEE: Piscataway, NJ, USA, 2025. Available online: https://ieeexplore.ieee.org/ (accessed on 4 June 2025).
8. arXiv. *arXiv.org e-Print Archive*; Cornell University: Ithaca, NY, USA, 2025. Available online: https://arxiv.org/ (accessed on 4 June 2025).
9. ACM Digital Library. ACM Digital Library, Association for Computing Machinery. Available online: https://dl.acm.org/ (accessed on 4 June 2025).

10. Zhong, A.; Mo, D.; Liu, G.; Liu, J.; Lu, Q.; Zhou, Q.; Wu, J.; Li, Q.; Wen, Q. LogParser-LLM: Advancing efficient log parsing with large language models. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), Barcelona, Spain, 25–29 August 2024. [CrossRef]
11. Huang, J.; Jiang, Z.; Chen, Z.; Lyu, M.R. LUNAR: Unsupervised LLM-based Log Parsing. *arXiv* **2024**, arXiv:2406.07174v2. [CrossRef]
12. Balasubramanian, P.; Seby, J.; Kostakos, P. CYGENT: A cybersecurity conversational agent with log summarization powered by GPT-3. *arXiv* **2024**, arXiv:2403.17160. [CrossRef]
13. Liu, X.; Liang, J.; Yan, Q.; Jang, J.; Mao, S.; Ye, M.; Jia, J.; Xi, Z. CyLens: Towards Reinventing Cyber Threat Intelligence in the Paradigm of Agentic Large Language Models. *arXiv* **2025**, arXiv:2502.20791v2. [CrossRef]
14. Ma, Z.; Kim, D.J.; Chen, T.-H.P. LibreLog: Accurate and efficient unsupervised log parsing using open-source large language models. *arXiv* **2024**, arXiv:2408.01585. [CrossRef]
15. Akhtar, S.; Khan, S.; Parkinson, S. LLM-based event log analysis techniques: A survey. *arXiv* **2025**, arXiv:2502.00677. [CrossRef]
16. Ma, Z.; Chen, A.R.; Kim, D.J.; Chen, T.-H.; Wang, S. LLMParser: An Exploratory Study on Using Large Language Models for Log Parsing. In Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, Lisbon, Portugal, 14–20 April 2024; ACM: New York, NY, USA, 2024; pp. 1–13. [CrossRef]
17. Fieblinger, R.; Alam, T.; Rastogi, N. Actionable Cyber Threat Intelligence using Knowledge Graphs and Large Language Models. *arXiv* **2024**, arXiv:2407.02528v1. [CrossRef]
18. Liu, Y.; Tao, S.; Meng, W.; Wang, J.; Ma, W.; Chen, Y.; Zhao, Y.; Yang, H.; Jiang, Y. Interpretable Online Log Analysis Using Large Language Models with Prompt Strategies. In Proceedings of the 32nd IEEE/ACM International Conference on Program Comprehension (ICPC '24), Lisbon, Portugal, 15–16 April 2024; pp. 35–46. [CrossRef]
19. Gupta, P.; Bhukar, K.; Kumar, H.; Nagar, S.; Mohapatra, P.; Kar, D. LogAn: An LLM-Based Log Analytics Tool with Causal Inferencing. In Proceedings of the 16th ACM/SPEC International Conference on Performance Engineering Companion (ICPE Companion), Toronto, ON, Canada, 5–9 May 2025; pp. 1–3. [CrossRef]
20. Al Siam, A.; Hassan, M.; Bhuiyan, T. Artificial Intelligence for Cybersecurity: A State of the Art. In Proceedings of the 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), Houston, TX, USA, 5–7 February 2025; IEEE: Piscataway, NJ, USA, 2025. [CrossRef]
21. Jiang, Z.; Liu, J.; Chen, Z.; Li, Y.; Huang, J.; Huo, Y.; He, P.; Gu, J.; Lyu, M.R. LILAC: Log Parsing using LLMs with Adaptive Parsing Cache. *arXiv* **2023**, arXiv:2310.01796v3. [CrossRef]
22. Karlsen, E.; Luo, X.; Zincir-Heywood, N.; Heywood, M. Heywood, Benchmarking Large Language Models for Log Analysis, Security, and Interpretation. *arXiv* **2023**, arXiv:2311.14519v1. [CrossRef]
23. Fayyazi, R.; Taghdimi, R.; Yang, S.J. Advancing TTP Analysis: Harnessing the Power of Large Language Models with Retrieval-Augmented Generation. *arXiv* **2024**, arXiv:2401.00280. [CrossRef]
24. Zhang, H.; Huang, J.; Mei, K.; Yao, Y.; Wang, Z.; Zhan, C.; Wang, H.; Zhang, Y. Agent Security Bench (ASB): Formalizing and Benchmarking Attacks and Defenses in LLM-based Agents. *arXiv* **2024**, arXiv:2410.02644v4. [CrossRef]
25. Liu, X.; Yu, F.; Li, X.; Yan, G.; Yang, P.; Xi, Z. Benchmarking LLMs in an Embodied Environment for Blue Team Threat Hunting. *arXiv* **2025**, arXiv:2505.11901v1. [CrossRef]
26. Loumachi, F.Y.; Ghanem, M.C.; Ferrag, M.A. GenDFIR: Advancing Cyber Incident Timeline Analysis Through Retrieval Augmented Generation and Large Language Models. *arXiv* **2024**, arXiv:2409.02572v4. [CrossRef]
27. Ali, T.; Kostakos, P. HuntGPT: Integrating Machine Learning-Based Anomaly Detection and Explainable AI with Large Language Models (LLMs). *arXiv* **2023**, arXiv:2309.16021. [CrossRef]
28. Cheng, Y.; Bajaber, O.; Tsegai, S.A.; Song, D.; Gao, P. CTINexus: Automatic Cyber Threat Intelligence Knowledge Graph Construction Using LLMs. *arXiv* **2025**, arXiv:2410.21060.
29. Jalalvand, F.; Chhetri, M.B.; Nepal, S.; Paris, C. Alert Prioritisation in Security Operations Centres: A Systematic Survey on Criteria and Methods. *ACM Comput. Surv.* **2024**, *57*, 1–36. [CrossRef]
30. Shah, S.; Parast, F.K. Parast, AI-Driven Cyber Threat Intelligence Automation. *arXiv* **2024**, arXiv:2410.20287v1. [CrossRef]
31. Cuong Nguyen, H.; Tariq, S.; Baruwal Chhetri, M.; Quoc Vo, B. Towards Effective Identification of Attack Techniques in Cyber Threat Intelligence Reports Using Large Language Models. In Proceedings of the Companion of the ACM on Web Conference 2025 (WWW Companion'25), Sydney, Australia, 28 April–2 May 2025. [CrossRef]
32. Jin, P.; Zhang, S.; Ma, M.; Li, H.; Kang, Y.; Li, L.; Liu, Y.; Qiao, B.; Zhang, C.; Zhao, P.; et al. Assess and Summarize: Improve Outage Understanding with Large Language Models. *arXiv* **2023**. [CrossRef]
33. de Witt, C.S. Open Challenges in Multi-Agent Security: Towards Secure Systems of Interacting AI Agents. *arXiv* **2025**, arXiv:2505.02077. [CrossRef]
34. Sharma, A.N.; Akbar, K.A.; Thuraisingham, B.; Khan, L. Enhancing Security Insights with KnowGen-RAG: Combining Knowledge Graphs, LLMs, and Multimodal Interpretability. In Proceedings of the 10th ACM International Workshop on Security and Privacy Analytics, Pittsburgh, PA, USA, 6 June 2025; pp. 2–12. [CrossRef]

35. Daniel, N.; Kaiser, F.K.; Giladi, S.; Sharabi, S.; Moyal, R.; Shpolyansky, S.; Murilllo, A.; Elyashar, A.; Puzis, R. Labeling NIDS Rules with MITRE ATT&CK Techniques: Machine Learning vs. Large Language Models. *arXiv* **2024**, arXiv:2412.10978. [CrossRef]
36. Froudakis, E.; Avgetidis, A.; Frankum, S.T.; Perdisci, R.; Antonakakis, M.; Keromytis, A. Uncovering Reliable Indicators: Improving IoC Extraction from Threat Reports. *arXiv* **2025**, arXiv:2506.11325. [CrossRef]
37. Alnahdi, A.; Narain, S. Towards Transparent Intrusion Detection: A Coherence-Based Framework in Explainable AI Integrating Large Language Models. In Proceedings of the 2024 IEEE 6th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA), Washington, DC, USA, 28–31 October 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 87–96. [CrossRef]
38. Jain, S.; Gupta, A.; Neha, K. AI Enhanced Ticket Management System for Optimized Support. In Proceedings of the 4th International Conference on AI-ML Systems (AIMLSystems 2024), Baton Rouge, LA, USA, 8–11 October 2024; pp. 1–7. [CrossRef]
39. Pei, C.; Wang, Z.; Liu, F.; Li, Z.; Liu, Y.; He, X.; Kang, R.; Zhang, T.; Chen, J.; Li, J.; et al. Flow-of-Action: SOP Enhanced LLM-Based Multi-Agent System for Root Cause Analysis. In Proceedings of the Companion ACM Web Conf. 2025 (WWW Companion'25), Sydney, NSW, Australia, 28 April–2 May 2025; pp. 1–10. [CrossRef]
40. Chen, Y.; Xie, H.; Ma, M.; Kang, Y.; Gao, X.; Shi, L.; Cao, Y.; Gao, X.; Fan, H.; Wen, M.; et al. Automatic Root Cause Analysis via Large Language Models for Cloud Incidents. *arXiv* **2023**, arXiv:2305.15778v4. [CrossRef]
41. Yang, Y.; Deng, Y.; Xiong, Y.; Li, B.; Xu, H.; Cheng, P. AidAI: Automated Incident Diagnosis for AI Workloads in the Cloud. *arXiv* **2025**, arXiv:2506.01481. [CrossRef]
42. Liu, Z.; Benge, C.; Jiang, S. Ticket-BERT: Labeling incident management tickets with language models. *arXiv* **2023**, arXiv:2307.00108. [CrossRef]
43. Li, C.; Zhu, Z.; He, J.; Zhang, X. RedChronos: A Large Language Model-Based Log Analysis System for Insider Threat Detection in Enterprises. *arXiv* **2025**, arXiv:2503.02702.
44. Liu, F.; He, X.; Zhang, T.; Chen, J.; Li, Y.; Yi, L.; Zhang, H.; Wu, G.; Shi, R. TickIt: Leveraging Large Language Models for Automated Ticket Escalation. *arXiv* **2025**. [CrossRef]
45. Nong, Y.; Yang, H.; Cheng, L.; Hu, H.; Cai, H. APPATCH: Automated Adaptive Prompting Large Language Models for Real-World Software Vulnerability Patching. In Proceedings of the 2025 Network and Distributed System Security Symposium (NDSS 2025), San Diego, CA, USA, 23–26 February 2025; Internet Society: Fredericksburg, VA, USA, 2025; pp. 1–15.
46. Lin, J.; Mohaisen, D. Evaluating Large Language Models in Vulnerability Detection Under Variable Context Windows. *arXiv* **2025**. [CrossRef]
47. Lin, X.; Zhang, J.; Deng, G.; Liu, T.; Zhang, T.; Guo, Q.; Chen, R. IRCopilot: Automated Incident Response with Large Language Models. *arXiv* **2025**. [CrossRef]
48. Liu, Z. Multi-Agent Collaboration in Incident Response with Large Language Models. *arXiv* **2024**. [CrossRef]
49. Perrina, F.; Marchiori, F.; Conti, M.; Verde, N.V. AGIR: Automating Cyber Threat Intelligence Reporting with Natural Language Generation. *arXiv* **2023**, arXiv:2310.02655. [CrossRef]
50. Wudali, P.N.; Kravchik, M.; Malul, E.; Gandhi, P.A.; Elovici, Y.; Shabtai, A. Rule-ATT&CK Mapper (RAM): Mapping SIEM Rules to TTPs Using LLMs. *arXiv* **2025**, arXiv:2502.02337.
51. Goel, D.; Husain, F.; Singh, A.; Ghosh, S.; Parayil, A.; Bansal, C.; Zhang, X.; Rajmohan, S. X-lifecycle Learning for Cloud Incident Management using LLMs. *arXiv* **2024**. [CrossRef]
52. Albanese, M.; Ou, X.; Lybarger, K.; Lende, D.; Goldgof, D. Towards AI-driven human-machine co-teaming for adaptive and agile cyber security operation centers. *arXiv* **2025**, arXiv:2505.06394.
53. Patel, D.; Lin, S.; Rayfield, J.; Zhou, N.; Vaculin, R.; Martinez, N.; O'donncha, F.; Kalagnanam, J. AssetOpsBench: Benchmarking AI Agents for Task Automation in Industrial Asset Operations and Maintenance. *arXiv* **2025**, arXiv:2506.03828.
54. Mitra, S.; Neupane, S.; Chakraborty, T.; Mittal, S.; Piplai, A.; Gaur, M.; Rahimi, S. LocalIntel: Generating organizational threat intelligence from global and local cyber knowledge. *arXiv* **2025**, arXiv:2401.10036.
55. Chopra, S.; Ahmad, H.; Goel, D.; Szabo, C. ChatNVD: Advancing Cybersecurity Vulnerability Assessment with Large Language Models. *arXiv* **2025**, arXiv:2412.04756.
56. CRondanini, C.; Carminati, B.; Ferrari, E.; Kundu, A.; Gaudiano, A. Malware Detection at the Edge with Lightweight LLMs: A Performance Evaluation. *arXiv* **2025**. [CrossRef]
57. Jin, Y.; Li, C.; Fan, P.; Liu, P.; Li, X.; Liu, C.; Qiu, W. LLM-BSCVM: An LLM-based blockchain smart contract vulnerability management framework. *arXiv* **2025**, arXiv:2505.17416.
58. Pasca, E.M.; Delinschi, D.; Erdei, R.; Matei, O. LLM-Driven, Self-Improving Framework for Security Test Automation: Leveraging Karate DSL for Augmented API Resilience. *IEEE Access* **2025**, *13*, 56861–56886. [CrossRef]
59. Torkamani, M.J.; Ng, J.; Mehrotra, N.; Chandramohan, M.; Krishnan, P.; Purandare, R. Streamlining security vulnerability triage with large language models. *arXiv* **2025**, arXiv:2501.18908. [CrossRef]

60. Applebaum, A.; Dennler, C.; Dwyer, P.; Moskowitz, M.; Nguyen, H.; Nichols, N.; Park, N.; Rachwalski, P.; Rau, F.; Webster, A.; et al. Bridging Automated to Autonomous Cyber Defense. In Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security, Los Angeles, CA, USA, 11 November 2022; ACM: New York, NY, USA, 2022; pp. 149–159. [CrossRef]
61. Alam, M.T.; Bhusal, D.; Nguyen, L.; Rastogi, N. CTIBench: A Benchmark for Evaluating LLMs in Cyber Threat Intelligence. *arXiv* **2024**. [CrossRef]
62. Xiao, Y.; Le, V.H.; Zhang, H. Stronger, Faster, and Cheaper Log Parsing with LLMs. *arXiv* **2024**, arXiv:2406.06156.
63. Boffa, M.; Drago, I.; Mellia, M.; Vassio, L.; Giordano, D.; Valentim, R.; Ben Houidi, Z. LogPrécis: Unleashing language models for automated malicious log analysis. *Comput. Secur.* **2024**, *141*, 103805. [CrossRef]
64. Khayat, M.; Barka, E.; Serhani, M.A.; Sallabi, F.; Shuaib, K.; Khater, H.M. Empowering Security Operation Center with Artificial Intelligence and Machine Learning—A Systematic Literature Review. *IEEE Access* **2025**, *13*, 19162–19194. [CrossRef]
65. Aung, Y.L.; Christian, I.; Dong, Y.; Ye, X.; Chattopadhyay, S.; Zhou, J.; Chattopadhyay, S.; Zhou, J. Generative AI for Internet of Things Security: Challenges and Opportunities. *arXiv* **2025**, arXiv:2502.08886. [CrossRef]
66. Tran, K.T.; Dao, D.; Nguyen, M.D.; Pham, Q.V.; O'Sullivan, B.; Nguyen, H.D. Nguyen, Multi-Agent Collaboration Mechanisms: A Survey of LLMs. *arXiv* **2025**. [CrossRef]
67. Jin, H.; Papadimitriou, G.; Raghavan, K.; Zuk, P.; Balaprakash, P.; Wang, C.; Mandal, A.; Deelman, E. Large Language Models for Anomaly Detection in Computational Workflows: From Supervised Fine-Tuning to In-Context Learning. *arXiv* **2024**. [CrossRef]
68. Wong, M.Y.; Valakuzhy, K.; Ahamad, M.; Blough, D.; Monrose, F. Understanding LLMs Ability to Aid Malware Analysts in Bypassing Evasion Techniques. In Proceedings of the Companion 26th International Conference on Multimodal Interaction, San Jose, Costa Rica, 4–8 November 2024; ACM: New York, NY, USA, 2024; pp. 36–40. [CrossRef]
69. Chhetri, M.B.; 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**, *24*, 22. [CrossRef]
70. Freitas, S.; Kalajdjieski, J.; Gharib, A.; McCann, R. AI-Driven Guided Response for Security Operation Centers with Microsoft Copilot for Security. In Proceedings of the Companion ACM Web Conference 2025 (WWW Companion'25), Sydney, NSW, Australia, 28 April–2 May 2025; pp. 1–10. [CrossRef]
71. Kim, M.; Wang, J.; Moore, K.; Goel, D.; Wang, D.; Mohsin, A.; Ibrahim, A.; Doss, R.; Camtepe, S.; Janicke, H. CyberAlly: Leveraging LLMs and Knowledge Graphs to Empower Cyber Defenders, Companion. In Proceedings of the ACM on Web Conference 2025, New York, NY, USA, 28 April–2 May 2025; ACM: New York, NY, USA, 2025; pp. 2851–2854. [CrossRef]
72. Arikkat, D.R.; Abhinav, M.; Binu, N.; Parvathi, M.; Biju, N.; Arunima, K.S.; Vinod, P.; Rafidha Rehiman, K.A.; Conti, M. IntellBot: Retrieval Augmented LLM Chatbot for Cyber Threat Knowledge Delivery. *arXiv* **2024**, arXiv:2411.05442. [CrossRef]
73. Xu, M.; Wang, H.; Liu, J.; Lin, Y.; Liu, C.X.Y.; Lim, H.W.; Dong, J.S. IntelEX: A LLM-driven attack-level threat intelligence extraction framework. *arXiv* **2024**, arXiv:2412.10872.
74. Chen, M.; Zhu, K.; Lu, B.; Li, D.; Yuan, Q.; Zhu, Y. AECR: Automatic attack technique intelligence extraction based on fine-tuned large language model. *Comput. Secur.* **2025**, *150*, 104213. [CrossRef]
75. Paul, S.; Alemi, F.; Macwan, R. LLM-assisted proactive threat intelligence for automated reasoning. *arXiv* **2025**, arXiv:2504.00428. [CrossRef]
76. Ghosh, S.K.; Gjomemo, R.; Venkatakrishnan, V.N. Citar: Cyberthreat Intelligence-driven Attack Reconstruction. In Proceedings of the Fifteenth ACM Conference on Data and Application Security and Privacy, Pittsburgh, PA, USA, 19 June 2024; ACM: New York, NY, USA, 2024; pp. 245–256. [CrossRef]
77. Las-Casas, P.; Kumbhare, A.G.; Fonseca, R.; Agarwal, S. LLexus: An AI agent system for incident management, SIGOPS Oper. *Syst. Rev.* **2024**, *58*, 23–36. [CrossRef]
78. Hays, S.; White, J. Employing LLMs for Incident-Response Planning and Review. *arXiv* **2024**, arXiv:2403.01271.
79. Liu, Z. AutoBnB: Multi-Agent Incident Response with Large Language Models. In Proceedings of the 2025 13th International Symposium on Digital Forensics and Security (ISDFS), Boston, MA, USA, 24–25 April 2025; IEEE: Piscataway, NJ, USA, 2025; pp. 1–6. [CrossRef]
80. Sun, Y.; Luo, Y.; Wen, X.; Yuan, Y.; Nie, X.; Zhang, S.; Liu, T.; Luo, X. TrioXpert: An Automated Incident Management Framework for Microservice Systems. *arXiv* **2025**, arXiv:2506.10043. [CrossRef]
81. Kramer, D.; Rosique, L.; Narotam, A.; Bursztein, E.; Kelley, P.G.; Thomas, K.; Woodruff, A. Integrating Large Language Models into Security Incident Response. In Proceedings of the Twenty-First Symposium on Usable Privacy and Security (SOUPS 2025), Seattle, WA, USA, 11–12 August 2025; USENIX Association: Berkeley, CA, USA, 2025; pp. 133–144.
82. Singh, R.; Chhetri, M.B.; Nepal, S.; Paris, C. ContextBuddy: AI-Enhanced Contextual Insights for Security Alert Investigation. *arXiv* **2025**, arXiv:2506.09365.
83. Jensen, R.I.T.; Tawosi, V.; Alamir, S. Software Vulnerability and Functionality Assessment using LLMs. *arXiv* **2024**. [CrossRef]
84. Lian, X.; Chen, Y.; Cheng, R.; Huang, J.; Thakkar, P.; Zhang, M.; Xu, T. Configuration Validation with Large Language Models. *arXiv* **2023**. [CrossRef]

85. Sheng, Z.; Wu, F.; Zuo, X.; Li, C.; Qiao, Y.; Hang, L. LProtector: An LLM-driven Vulnerability Detection System. *arXiv* **2024**. [CrossRef]
86. Xu, M.; Fan, J.; Huang, X.; Zhou, C.; Kang, J.; Niyato, D.; Mao, S.; Han, Z.; Shen, X.; Lam, K.Y.; et al. Forewarned is Forearmed: A Survey on Large Language Model-based Agents in Autonomous Cyberattacks. *arXiv* **2025**. [CrossRef]
87. Wang, D.; Zhou, G.; Chen, L.; Li, D.; Miao, Y. ProphetFuzz: Fully Automated Prediction and Fuzzing of High-Risk Option Combinations with Only Documentation via Large Language Model. In Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, Salt Lake City, UT, USA, 14–18 October 2024; ACM: New York, NY, USA, 2024; pp. 735–749. [CrossRef]
88. Toprani, D.; Madisetti, V.K. LLM Agentic Workflow for Automated Vulnerability Detection and Remediation in Infrastructure-as-Code. *IEEE Access* **2025**, *13*, 69175–69190. [CrossRef]
89. Wang, X.; Tian, Y.; Huang, K.; Liang, B. Practically implementing an LLM-supported collaborative vulnerability remediation process: A team-based approach. *Comput. Secur.* **2025**, *148*, 104113. [CrossRef]
90. Beck, V.; Landauer, M.; Wurzenberger, M.; Skopik, F.; Rauber, A. System Log Parsing with Large Language Models: A Review. *arXiv* **2025**, arXiv:2504.04877.
91. Kshetri, N.; Voas, J. Agentic Artificial Intelligence for Cyber Threat Management. *Computer* **2025**, *58*, 86–90. [CrossRef]
92. Massengale, S.; Huff, P. Linking Threat Agents to Targeted Organizations: A Pipeline for Enhanced Cybersecurity Risk Metrics. In Proceedings of the 2024 4th Intelligent Cybersecurity Conference (ICSC), Valencia, Spain, 17–20 September 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 132–141. [CrossRef]
93. Shukla, A.; Gandhi, P.A.; Elovici, Y.; Shabtai, A. RuleGenie: SIEM Detection Rule Set Optimization. *arXiv* **2025**. [CrossRef]
94. Fu, Y.; Yuan, X.; Wang, D. RAS-Eval: A Comprehensive Benchmark for Security Evaluation of LLM Agents in Real-World Environments. *arXiv* **2025**. [CrossRef]
95. Hamadanian, P.; Arzani, B.; Fouladi, S.; Kakarla, S.K.R.; Fonseca, R.; Billor, D.; Cheema, A.; Nkposong, E.; Chandra, R. A Holistic View of AI-Driven Network Incident Management. In Proceedings of the 22nd ACM Workshop on Hot Topics in Networks (HotNets '23), Cambridge, MA, USA, 28–29 November 2023; pp. 1–9. [CrossRef]
96. Bono, J.; Grana, J.; Xu, A. Generative AI and Security Operations Center Productivity: Evidence from Live Operations. *arXiv* **2024**. [CrossRef]
97. Gandhi, P.A.; Shukla, A.; Tayouri, D.; Ifland, B.; Elovici, Y.; Puzis, R.; Shabtai, A. ATAG: AI-Agent Application Threat Assessment with Attack Graphs. *arXiv* **2025**, arXiv:2506.02859. [CrossRef]
98. Bountakas, P.; Fysarakis, K.; Kyriakakis, T.; Karafotis, P.; Aristeidis, S.; Tasouli, M.; Alcaraz, C.; Alexandris, G.; Andronikou, V.; Koutsouri, T.; et al. SYNAPSE—An Integrated Cyber Security Risk & Resilience Management Platform, With Holistic Situational Awareness, Incident Response & Preparedness Capabilities: SYNAPSE. In Proceedings of the 19th International Conference on Availability, Reliability and Security, Vienna, Austria, 30 July–2 August 2024; ACM: New York, NY, USA, 2024; pp. 1–10. [CrossRef]
99. Sarkar, A.; Sarkar, S. Survey of LLM Agent Communication with MCP: A Software Design Pattern Centric Review. *arXiv* **2025**. [CrossRef]
100. Tseng, P.; Yeh, Z.; Dai, X.; Liu, P. Using LLMs to Automate Threat Intelligence Analysis Workflows in Security Operation Centers. *arXiv* **2024**. [CrossRef]
101. Ding, A.; Li, G.; Yi, X.; Lin, X.; Li, J.; Zhang, C. Generative AI for Software Security Analysis: Fundamentals, Applications, and Challenges. *IEEE Softw.* **2024**, *41*, 46–55. [CrossRef]
102. Castro, S.R.; Campbell, R.; Lau, N.; Villalobos, O.; Duan, J.; Cardenas, A.A. Large Language Models are Autonomous Cyber Defenders. *arXiv* **2025**. [CrossRef]
103. Saura, P.F.; Jayaram, K.R.; Isahagian, V.; Bernabé, J.B.; Skarmeta, A. On Automating Security Policies with Contemporary LLMs. *arXiv* **2025**, arXiv:2506.04838. [CrossRef]
104. Oesch, S.; Chaulagain, A.; Weber, B.; Dixson, M.; Sadovnik, A.; Roberson, B.; Watson, C.; Austria, P. Towards a High Fidelity Training Environment for Autonomous Cyber Defense Agents. In Proceedings of the 17th Cyber Security Experimentation and Test Workshop, Philadelphia, PA, USA, 13 August 2024; ACM: New York, NY, USA, 2024; pp. 91–99. [CrossRef]
105. Subramaniam, P.; Krishnan, S. DePLOI: Applying NL2SQL to Synthesize and Audit Database Access Control. *arXiv* **2024**. [CrossRef]
106. Roy, D.; Zhang, X.; Bhave, R.; Bansal, C.; Las-Casas, P.; Fonseca, R.; Rajmohan, S. Exploring LLM-based agents for root cause analysis. In Proceedings of the ACM International Conference on the Foundations of Software Engineering (FSE Companion), Porto de Galinhas, Brazil, 15–19 July 2024. [CrossRef]
107. Shah, S.P.; Deshpande, A.V. Addressing Data Poisoning and Model Manipulation Risks using LLM Models in Web Security. In Proceedings of the 2024 International Conference on Distributed Systems, Computer Networks and Cybersecurity (ICDSCNC), Bengaluru, India, 20–21 September 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [CrossRef]

108. Kalakoti, R.; Vaarandi, R.; Bahşi, H.; Nõmm, S. Evaluating Explainable AI for Deep Learning-Based Network Intrusion Detection System Alert Classification. *arXiv* **2025**, arXiv:2506.07882. [CrossRef]
109. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*, n71. [CrossRef]