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

Siddhant Srinivas (siddhant.srinivas1528@coyote.csusb.edu), Brandon Kirk (brandon.kirk8304@coyote.csusb.edu), Julissa Zendejas (julissa.zendejas6692@coyote.csusb.edu), Michael Espino (michael.espino5607@coyote.csusb.edu), Matthew Boskovich (matthew.boskovich3084@coyote.csusb.edu), Abdul Bari (abdul.bari8019@coyote.csusb.edu), Khalil Dajani (khalil.dajani@csusb.edu), Nabeel Alzahrani (alzahran@csusb.edu)
School of Computer Science & Engineering,
California State University, San Bernardino, USA

Abstract

The increasing volume, velocity, and sophistication of cyber threats has placed immense pressure on modern Security Operations Centers (SOCs), where traditional rule-based and manual processes are proving insufficient. These limitations result in alert fatigue, delayed responses, high false-positive rates, analyst dependency, and escalating operational costs. Building upon recent advancements, Artificial Intelligence (AI) offers new opportunities to transform SOC workflows through automation and augmentation. In particular, Large Language Models (LLMs) and autonomous AI agents have demonstrated 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 offer improvements in detection accuracy, response time, and analyst support, critical challenges persist. These include issues of model interpretability, adversarial robustness, integration with legacy systems, and the risk of hallucinations, or data leakage. A detailed capability-maturity model based on both LLMs and AI Agents is introduced to outline the levels of integration with SOC tasks. This paper addresses which combinations of LLMs and AI agents are effective for automating SOC tasks, and identifies the types of models necessary for their integration into SOC workflows. This survey synthesizes trends, identifies persistent limitations, and outlines future directions to increase the level of autonomy for trustworthy, explainable, and safe AI integration in SOC environments. Clearly identifying the tools and knowledge to increase the accuracy and decrease the response time to improve SOC environments.

Index Terms- Security Operation Center, Large Language Models, AI Agent, Cybersecurity Automation, Human-AI Collaboration

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. However, despite these promising developments, significant gaps remain in existing literature and implementations. Prior studies often focus on isolated SOC aspects or specific AI methods, lacking a comprehensive vision for Human-AI collaboration across full cybersecurity operations. The optimal balance between automation and human oversight remains underexplored, often assuming static autonomy settings that neglect varied task complexity and risk [3]. Persistent challenges include model interpretability ("black boxes"), data quality issues [1], and integration with legacy systems [3], hallucinations [5], privacy leakage concerns [6], and susceptibility to adversarial attacks [2]. Earlier surveys exhibit several limitations: incomplete end-to-end LLM coverage, inadequate safety mitigation, limited examination of collaborative mechanisms, insufficient treatment of technical and architectural challenges, and inconsistent benchmarking methodologies. This survey offers a taxonomy of LLM and AI agent applications in SOCs, introduces a capability-maturity model to measure automation, performs a comprehensive strengths and limitations analysis [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 500 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. On automating all 8 SOC tasks, peer-reviewed and preprint sources were selected, covering a range of recent publications from 2022 to 2025, since practical SOC applications of LLMs and autonomous agents has rapidly increased [3], pre-2022 literature is largely outdated for this topic. Sources were collected from the reputable databases IEEE Xplore [7], arXiv [8], and the ACM Digital Library [9] as we believe they are currently the three most reputable databases for AI research. 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. 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 PRISMA for transparency, standardization, and replicability in systematic reviews or structured evidence syntheses. One independent reviewer conducted screening from abstract and four independent reviewers completed the selection based on the four criteria mentioned above where 100 papers were selected.

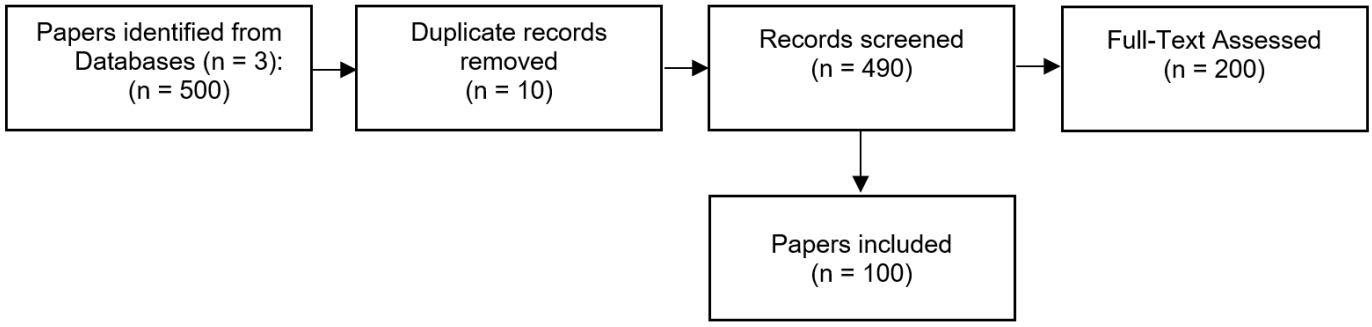

Fig. 1. PRISMA-style literature-selection flow used in this survey.

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. 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 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]

A. 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 uses 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 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 seconds with over 100% reduction in LLM costs [62]. 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 [63]. 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 [64]. Agentic frameworks extend LLM capabilities with tool use, memory, and planning addressing limitations like context window constraints and hallucinations [65]. 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].

B. 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], [66]. 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 [67], [68]. 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%) [69]. 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 [63]. 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 hours to 90 minutes, while also increasing automated ticketing from 10% to 75% [70]. 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].

C. 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 [71]. 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 3,591 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 [72]. LLM-Assisted Proactive Threat Intelligence integrates LLMs and RAG systems with continuous threat intelligence feeds to enhance real-time cybersecurity detection and response capabilities [73]. Despite strong benchmark performance, CTI systems using LLMs require real-world validation to address risks like hallucination, data leakage, and inconsistent generalization [20], [29].

D. 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 don't 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 1,600 Search Processing Language (SPL) rules; therefore, it observed an average reduction of 30% in alerts [38], [74]. 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 [75]. 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.

E. 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. [63]. AI applications in IR include optimizing the handling of security breaches and automating key response tasks [76]. 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 [64]. 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], [77]. 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 seconds [78]. 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 [79], [63] and requiring human oversight.

F. 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 [64]. 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 [64]. 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 [64]. Studies indicate that LLMs, specifically GPT models, can generate cybersecurity policies that outperform human-generated ones in terms of completeness, effectiveness, and efficiency [64]. 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 hours to under 2 hours, 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.

G. Asset Discovery and Management

Asset discovery and management involves continuously identifying, monitoring, and securing valuable organizational resources across their lifecycle [3], [29]. 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 Information Technology/ Operational Technology (IT/OT) systems [63]. 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 minutes [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.

H. 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], [80]. LLM-based tools like DefectHunter and others use attention models, semantic reward, configuration validation, and reinforcement learning to patch vulnerable code [81]. Their effectiveness can be enhanced for specific domains such as IoT using datasets like QEMU, Pongo-70B, and CWE-754. [64]. 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 5,000 balanced Big-Vul samples, outperforming established tools [82], [83]. 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], [64]. 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, though careful consideration of security implications and human oversight remains essential [46], [59], [84]. 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.

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 [85]	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 [74], Rule-based correlation [3],	LLMs for NIDS rule labeling [35], incident	ReliaQuest agent [86], ContextBuddy	ReliaQuest: 20x faster, 98% alert automation,	Reduced alert fatigue and manual burden [1],

	SIEM systems (80% false positives) [1]	summarization [32], prioritization [1]	[79], multi-agent triage systems [39]	5-min containment, 30% improved detection [86]	enhanced contextual understanding [52]
Threat Intelligence	Manual analysis from diverse sources [87], traditional NLP [22], rule-based systems [88]	CTI extraction [2], IntellBot [71], CTINexus [28], LANCE [36], IntelEX [72], LocalIntel [54]	CyLens [13], IntellBot agents [71], LANCE engine [36], Multi-agent CTI extractors [31]	IntelEX F1 up to 0.902 [68], IntellBot: BERT >0.8 [71], 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], [78]	Rand score 0.96 for clustering [38] Ticket-BERT outperforms baselines [42]	LLMs reduce delay [89], traditional methods inefficient under volume [38], [90]
Incident Response	Manual response protocols [1], AIOps with limited scope [4], isolated management [63]	GenDFIR [26], IRCopilot [47], LLexus [75], LLM-BSCVM [57]	AidAI [41], AutoBnB, Audit-LLM [92], Multi-agent IRCopilot [47]	6x 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 [91], 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 [63]	LLMs[89]	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 [92], multi-agent IaC analyzers [84], LLM-BSCVM [57]	CASEY: CWE accuracy 68%, severity 73.6%, combined 51.2% [59]	LLMs outperform static tools [84], documentation still emerging [92], privacy concerns persist [26]

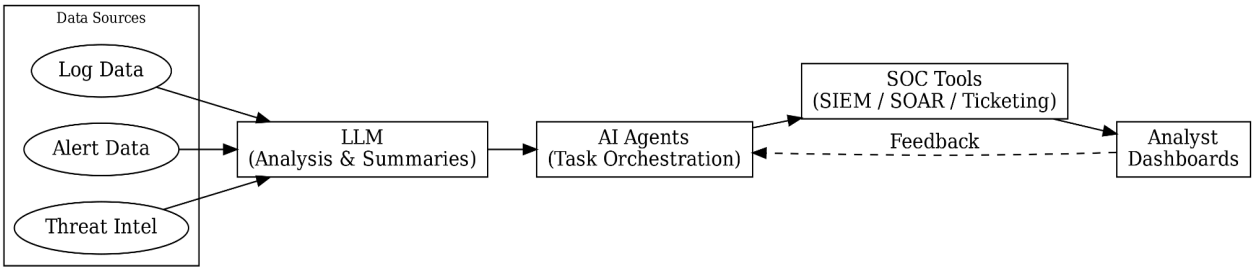

Fig. 2. End-to-end AI-augmented SOC dataflow

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 [69], 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 and used for evaluation in Table 3.

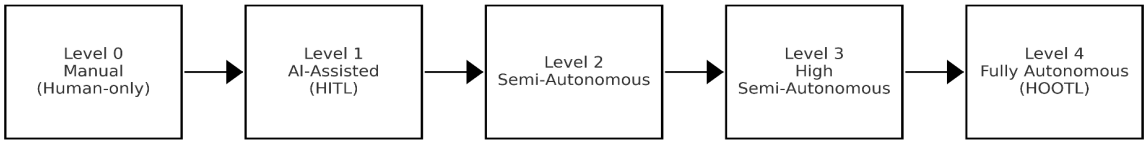

Fig. 3. Five-level autonomy ladder for AI-enabled SOCs.

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) [69]	AI Agent	1	Microsoft Defender alerts and telemetry
	CyberAlly [3]	AI Agent	2	Network telemetry and endpoint event data
	HuntGPT [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 [75]	AI Agent	3	Generated incidents
	AidAI [41]	LLMs	3	Historic data
	TickIT [44]	LLMs + CoT	3	Customer Support tickets and dialogue
Incident Response	AidAI [41]	AI Agent LLMs + CoT	3	Historical Ticket Content
	IRCopilot [47]	LLMs	3	User Activity Logs
	TrioXpert [78]	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 [64]	LLM	3	Threat intelligence data
Asset Discovery and Management	AssetOps Agent [53]	Multi-Agent + Global Coordinator	2	Multi-modal data
	GreyMatter [86]	AI Agent	3	Security alerts and incident response data
	SYNAPSE [93]	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 [94].	Generalize across domains and scale efficiently [2].	Require high compute for deployment and fine-tuning [2].
Interpretability	Providing context-aware, human-comprehensible	Limited due to the "black-box" nature of some models and the	Enabling natural language interactions for insights and summarized reports, increase	Susceptibility to hallucinations and output variability [30], [95], [96]

	insights and audit opinions via CoT reasoning [3], [5]	potential for cascading errors from hallucinations [1], [3]	accuracy using CoT prompting and RAG [37], [49], [64]	
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], [63], [86]	Efficient processing and analyzing vast diverse data, automating complex tasks, generating structured insights [1], [63], [68]	High computational resources, probabilistic nature leading to output variability [2], [63], [88]
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 [92]	Enhance integration through natural language understanding [3], automated reporting, and context-aware insights [90], 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 [79].	Can suffer from factual errors or "hallucinations" [3].
Privacy & Security	AI agents automate tasks in secure, private data environments [94]	Multi-agent systems introduce complex vulnerabilities [33], risking sensitive data exfiltration [92]	Enhance security via threat detection [2], incident response, and secure code generation [6]	Risk sensitive data exposure, hallucinations, and adversarial attacks [2], [96]

A. 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 [65], SOCs frequently rely on legacy systems that may not be fully compatible with modern AI tools, necessitating substantial effort in developing new APIs, middleware, or components, which incurs considerable cost and complexity [38]. Few studies have comprehensively addressed the seamless integration of AI techniques into existing SOC workflows and the collaboration with human analysts [89]. 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 [91], [97]. The hybrid integration of LLMs and AI agents into cloud security operations, often relying 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), faces persistent challenges including seamless interoperability with diverse and often legacy security tooling and SIEM platforms [3], [63]. 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 [94], [98]. 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, underscoring the ongoing need for standardized frameworks and robust 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 [99].

B. 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 [88]. 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 [6]. Rapid decision-making and coordinated responses are essential but challenging due to dynamic network topologies limiting real-time performance [13], [78]. 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, underscoring the need for research into human-AI collaboration and personalized trust models. 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], [97]. The high volume of alerts, often false positives, highlights the necessity for AI solutions that reduce redundant escalations and improve operational efficiency.

C. 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. This lack of transparency 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 [78]. 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 themselves can be susceptible to adversarial attacks, including covert message exchange (secret collusion) and subtle shifts in word choice, requiring self-examination mechanisms and more robust defenses beyond simple paraphrasing [88]. A significant limitation of LLMs is their propensity for factual errors or "hallucinations" generating inaccurate or fabricated information. 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 [73], [78]. 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 [85], [90]. Ongoing research focuses on fine-tuning models with domain-specific data and continually evolving models to adapt to new threats and log formats.

D. 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], [100]. The heterogeneity of security data, originating from various sources and formats from structured logs to unstructured threat reports further complicates data ingestion, integration, and analysis by AI models [38], [52]. Low-quality or inconsistent data, such as unclear incident descriptions, also reduces AI effectiveness and necessitates manual review [41], [97]. 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 [78], [101]. Data poisoning and model manipulation risks challenge LLMs for web security defense in SOCs [102]. Furthermore, inconsistencies in log formats can impede the effectiveness of SIEM systems. The diverse sources and formats of security data contribute to data heterogeneity, complicating its integration and analysis by AI models. The variability and complexity of real-world threats can also affect the performance of AI agents trained on controlled datasets, highlighting the need for AI solutions that can adapt to evolving data structures and variations in user input [16]. Approaches involving string similarity and dynamic data structuring are being explored to address these inconsistencies [15].

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 integration with SIEM and XDR tools to enhance automated response [4], [69]. 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], [93]. 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% [84]. 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 [93]. 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], [84]. 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 [103]. 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], [94]. 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 100 peer-reviewed and preprint sources across IEEE, ACM, and arXiv, the filtering criteria particularly the emphasis on recency (post-2022), and English-language publications may introduce a selection bias. Consequently, 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. 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. Conclusion

This survey examined over 500 academic and preprint papers published between 2022 and 2025, narrowing the selection to 100 high-quality sources focused on the integration of LLMs and AI agents in SOCs. Our analysis shows that SOCs are steadily evolving from traditional, manual workflows toward hybrid, AI-augmented architectures that enhance key functions such as log summarization, alert triage, threat intelligence, ticket handling, incident response, report generation, asset discovery and

management, and vulnerability management. While these technologies offer promising improvements in detection speed, accuracy, and scalability, most real-world SOC implementations remain at early stages of automation, Level 1 or 2 in our proposed capability-maturity model far behind the sophistication of current cyber threats. Three primary challenges hinder further adoption: limited model interpretability, lack of robustness to adversarial inputs, and high integration friction with legacy systems. Addressing these issues will require both technical advancements, such as RAG, as well as methodological shifts that prioritize human-AI teaming, trust calibration, and longitudinal benchmarking beyond accuracy metrics. Organizational readiness, including AI literacy in analyst training and modular SOC infrastructure, will also be essential. Our findings suggest that augmentation, rather than full automation, yields the most practical and resilient path forward. By combining the pattern recognition and scalability of AI with the contextual judgment and adaptability of human analysts, SOCs can build flexible, autonomy-tuned workflows capable of responding to an increasingly complex and automated threat landscape.

References

[1] F. Binbeshr, M. Imam, M. Ghaleb, M. Hamdan, M. A. Rahim, and M. Hammoudeh, "The Rise of Cognitive SOCs: A Systematic Literature Review on AI Approaches," IEEE Open J. Comput. Soc., vol. 6, pp. 360–379, 2025, doi: 10.1109/ojcs.2025.3536800. Available: http://dx.doi.org/10.1109/OJCS.2025.3536800

[2] M. Hassanin and N. Moustafa, "A Comprehensive Overview of Large Language Models (LLMs) for Cyber Defences: Opportunities and Directions." arXiv, 2024. doi: 10.48550/ARXIV.2405.14487. Available: https://arxiv.org/abs/2405.14487

[3] A. Mohsin, H. Janicke, A. Ibrahim, I. H. Sarker, and S. Camtepe, "A Unified Framework for Human AI Collaboration in Security Operations Centers with Trusted Autonomy." arXiv, 2025. doi: 10.48550/ARXIV.2505.23397. Available: https://arxiv.org/abs/2505.23397

[4] M. Chigurupati, R. K. Malviya, A. R. Toorpu, and K. Anand, "AI Agents for Cloud Reliability: Autonomous Threat Detection and Mitigation Aligned with Site Reliability Engineering Principles," 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC). IEEE, pp. 1–4, Feb. 05, 2025. doi: 10.1109/icaic63015.2025.10849322. Available: http://dx.doi.org/10.1109/ICAIC63015.2025.10849322

[5] C. [Song, L. Ma, J. Zheng, J. Liao, H. Kuang, and L. Yang, "Audit-LLM: Multi-Agent Collaboration for Log-based Insider Threat Detection." arXiv, 2024. doi: 10.48550/ARXIV.2408.08902. Available: https://arxiv.org/abs/2408.08902

[6] M. Gupta, C. Akiri, K. Aryal, E. Parker, and L. Praharaj, "From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy," IEEE Access, vol. 11, pp. 80218–80245, 2023, doi: 10.1109/access.2023.3300381. Available: http://dx.doi.org/10.1109/ACCESS.2023.3300381

[7] IEEE Xplore, "IEEE Xplore Digital Library," IEEE, Available: https://ieeexplore.ieee.org /

[8] arXiv, "arXiv.org e-Print archive," Cornell University,. Available: https://arxiv.org/.

[9] ACM Digital Library, "ACM Digital Library," Association for Computing Machinery, Available: https://dl.acm.org/

[10] A. Zhong, D. Mo, G. Liu, J. Liu, Q. Lu, Q. Zhou, J. Wu, Q. Li, and Q. Wen, "LogParser-LLM: Advancing efficient log parsing with large language models," in Proc. 30th ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD '24), Barcelona, Spain, Aug. 2024. DOI: 10.1145/3637528.3671810. Available: https://doi.org/10.1145/3637528.3671810.

[11] J. Huang, Z. Jiang, Z. Chen, and M. R. Lyu, "LUNAR: Unsupervised LLM-based Log Parsing." arXiv, 2024. doi: 10.48550/ARXIV.2406.07174. Available: https://arxiv.org/abs/2406.07174

[12] P. Balasubramanian, J. Seby, and P. Kostakos, "CYGENT: A cybersecurity conversational agent with log summarization powered by GPT-3," arXiv preprint arXiv:2403.17160, Mar. 2024. Available: https://arxiv.org/abs/2403.17160

[13] X. Liu et al., "CyLens: Towards Reinventing Cyber Threat Intelligence in the Paradigm of Agentic Large Language Models." arXiv, 2025. doi: 10.48550/ARXIV.2502.20791. Available: https://arxiv.org/abs/2502.20791

[14] Z. Ma et al., "LibreLog: Accurate and efficient unsupervised log parsing using open-source large language models," arXiv preprint arXiv:2408.01585, Nov. 2024.. Available: https://arxiv.org/abs/2408.01585

[15] S. Akhtar, S. Khan, and S. Parkinson, "LLM-based event log analysis techniques: A survey." arXiv, 2025. doi: 10.48550/ARXIV.2502.00677. Available: https://arxiv.org/abs/2502.00677

[16] Z. Ma, A. R. Chen, D. J. Kim, T.-H. Chen, and S. Wang, "LLMParser: An Exploratory Study on Using Large Language Models for Log Parsing," Proceedings of the IEEE/ACM 46th International Conference on Software Engineering. ACM, pp. 1–13, Apr. 12, 2024. doi: 10.1145/3597503.3639150. Available: http://dx.doi.org/10.1145/3597503.3639150

[17] R. Fieblinger, M. T. Alam, and N. Rastogi, "Actionable Cyber Threat Intelligence using Knowledge Graphs and Large Language Models." arXiv, 2024. doi: 10.48550/ARXIV.2407.02528. Available: https://arxiv.org/abs/2407.02528

[18] Y. Liu et al., "Interpretable Online Log Analysis Using Large Language Models With Prompt Strategies," in Proc. 32nd IEEE/ACM Int. Conf. on Program Comprehension (ICPC '24), Lisbon, Portugal, Apr. 2024, pp. 35–46, doi: 10.1145/3643916.3644408. Available: https://doi.org/10.1145/3643916.3644408.

[19] P. Gupta, K. Bhukar, H. Kumar, S. Nagar, P. Mohapatra, and D. Kar, "LogAn: An LLM-Based Log Analytics Tool with Causal Inferencing," in Proc. 16th ACM/SPEC Int. Conf. on Performance Engineering Companion (ICPE Companion), Toronto, ON, Canada, May 2025, pp. 1–3, doi: 10.1145/3680256.3721246. Available: https://doi.org/10.1145/3680256.3721246.

[20] A. A. Siam, M. M. Hassan, and T. Bhuiyan, "Artificial Intelligence for Cybersecurity: A State of the Art," 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC). IEEE, pp. 1–7, Feb. 05, 2025. doi: 10.1109/icaic63015.2025.10848980. Available: http://dx.doi.org/10.1109/ICAIC63015.2025.10848980

[21] Z. Jiang et al., "LILAC: Log Parsing using LLMs with Adaptive Parsing Cache." arXiv, 2023. doi: 10.48550/ARXIV.2310.01796. Available: https://arxiv.org/abs/2310.01796

[22] E. Karlsen, X. Luo, N. Zincir-Heywood, and M. Heywood, "Benchmarking Large Language Models for Log Analysis, Security, and Interpretation." arXiv, 2023. doi: 10.48550/ARXIV.2311.14519. Available: https://arxiv.org/abs/2311.14519

[23] R. Fayyazi, R. Taghdimi, and S. J. Yang, "Advancing TTP Analysis: Harnessing the Power of Large Language Models with Retrieval-Augmented Generation," arXiv preprint arXiv:2401.00280, 2024. Available: https://arxiv.org/abs/2401.00280

[24] H. Zhang et al., "Agent Security Bench (ASB): Formalizing and Benchmarking Attacks and Defenses in LLM-based Agents." arXiv, 2024. doi: 10.48550/ARXIV.2410.02644. Available: https://arxiv.org/abs/2410.02644

[25] X. Liu, F. Yu, X. Li, G. Yan, P. Yang, and Z. Xi, "Benchmarking LLMs in an Embodied Environment for Blue Team Threat Hunting." arXiv, 2025. doi: 10.48550/ARXIV.2505.11901. Available: https://arxiv.org/abs/2505.11901

[26] F. Y. Loumachi, M. C. Ghanem, and M. A. Ferrag, "GenDFIR: Advancing Cyber Incident Timeline Analysis Through Retrieval Augmented Generation and Large Language Models." arXiv, 2024. doi: 10.48550/ARXIV.2409.02572. Available: https://arxiv.org/abs/2409.02572

[27] T. Ali and P. Kostakos, "HuntGPT: Integrating Machine Learning-Based Anomaly Detection and Explainable AI With Large Language Models (LLMs)," arXiv preprint arXiv:2309.16021, Sept. 2023. Available: https://arxiv.org/abs/2309.16021

[28] Y. Cheng, O. Bajaber, S. A. Tsegai, D. Song, and P. Gao, “CTINexus: Automatic Cyber Threat Intelligence Knowledge Graph Construction Using LLMs,” arXiv preprint arXiv:2410.21060, 2025. Available: https://arxiv.org/abs/2410.21060
[29] F. Jalalvand, M. Baruwal Chhetri, S. Nepal, and C. Paris, “Alert Prioritisation in Security Operations Centres: A Systematic Survey on Criteria and Methods,” ACM Comput. Surv., vol. 57, no. 2, pp. 1–36, Nov. 2024, doi: 10.1145/3695462. Available: http://dx.doi.org/10.1145/3695462
[30] S. Shah and F. K. Parast, “AI-Driven Cyber Threat Intelligence Automation.” arXiv, 2024. doi: 10.48550/ARXIV.2410.20287. Available: https://arxiv.org/abs/2410.20287
[31] H. C. Nguyen, S. Tariq, M. B. Chhetri, and B. Q. Vo, “Towards Effective Identification of Attack Techniques in Cyber Threat Intelligence Reports Using Large Language Models,” in Companion Proc. ACM Web Conf. 2025 (WWW Companion ’25), Sydney, Australia, Apr.–May 2025, pp. xxx–xxx, doi: 10.1145/3701716.3715469.. Available: https://doi.org/10.1145/3701716.3715469.
[32] P. Jin et al., “Assess and Summarize: Improve Outage Understanding with Large Language Models.” arXiv, 2023. doi: 10.48550/ARXIV.2305.18084. Available: https://arxiv.org/abs/2305.18084
[33] C. Schroeder de Witt, “Open Challenges in Multi-Agent Security: Towards Secure Systems of Interacting AI Agents,” arXiv preprint arXiv:2505.02077, May 2025. Available: https://arxiv.org/abs/2505.02077
[34] A. N. Sharma, K. A. Akbar, B. Thuraisingham, and L. Khan, “Enhancing Security Insights with KnowGen-RAG: Combining Knowledge Graphs, LLMs, and Multimodal Interpretability,” Proceedings of the 10th ACM International Workshop on Security and Privacy Analytics. ACM, pp. 2–12, Jun. 04, 2025. doi: 10.1145/3716815.3729012. Available: http://dx.doi.org/10.1145/3716815.3729012
[35] N. Daniel, F. K. Kaiser, S. Giladi, S. Sharabi, R. Moyal, S. Shpolyansky et al., “Labeling NIDS Rules with MITRE ATT&CK Techniques: Machine Learning vs. Large Language Models,” arXiv preprint arXiv:2412.10978, Dec. 2024. [Online]. Available: https://arxiv.org/abs/2412.10978
[36] E. Froudakis, A. Avgetidis, S. T. Frankum, R. Perdisci, M. Antonakakis, and A. Keromytis, “Uncovering Reliable Indicators: Improving IoC Extraction from Threat Reports,” arXiv preprint arXiv:2506.11325, Jun. 2025. Available: https://arxiv.org/abs/2506.11325
[37] A. Alnahdi and S. Narain, “Towards Transparent Intrusion Detection: A Coherence-Based Framework in Explainable AI Integrating Large Language Models,” 2024 IEEE 6th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA). IEEE, pp. 87–96, Oct. 28, 2024. doi: 10.1109/tps-isa62245.2024.00020. Available: http://dx.doi.org/10.1109/TPS-ISA62245.2024.00020
[38] S. Jain, A. Gupta, and K. Neha, “AI Enhanced Ticket Management System for Optimized Support,” in Proc. 4th Int. Conf. on AI-ML Systems (AIMLSystems 2024), Baton Rouge, LA, USA, Oct. 2024, pp. 1–7, doi: 10.1145/3703412.3703433. Available: https://doi.org/10.1145/3703412.3703433.
[39] C. Pei, Z. Wang, F. Liu, Z. Li, Y. Liu, X. He et al., “Flow-of-Action: SOP Enhanced LLM-Based Multi-Agent System for Root Cause Analysis,” in Companion Proc. ACM Web Conf. 2025 (WWW Companion ’25), Sydney, NSW, Australia, Apr.–May 2025, pp. 1–10, doi: 10.1145/3701716.3715225. . Available: https://doi.org/10.1145/3701716.3715225.
[40] Y. Chen et al., “Automatic Root Cause Analysis via Large Language Models for Cloud Incidents.” arXiv, 2023. doi: 10.48550/ARXIV.2305.15778. Available: https://arxiv.org/abs/2305.15778
[41] Y. Yang, L. Wang, H. Chen, and M. Wu, “AidAI: Automated Incident Diagnosis for AI Workloads in the Cloud,” arXiv preprint arXiv:2506.01481, 2025. Available: https://arxiv.org/abs/2506.01481
[42] Z. Liu, C. Benge, and S. Jiang, “Ticket-BERT: Labeling incident management tickets with language models,” arXiv preprint arXiv:2307.00108, Jun. 2023. Available: https://arxiv.org/abs/2307.00108
[43] C. Li, Z. Zhu, J. He, and X. Zhang, “RedChronos: A Large Language Model-Based Log Analysis System for Insider Threat Detection in Enterprises,” arXiv preprint arXiv:2503.02702, 2025. Available: https://arxiv.org/abs/2503.02702
[44] F. Liu et al., “TickIt: Leveraging Large Language Models for Automated Ticket Escalation,” arXiv, 2025, doi: 10.48550/ARXIV.2504.08475. Available: https://arxiv.org/abs/2504.08475
[45] Y. Nong, H. Yang, L. Cheng, H. Hu, and H. Cai, “APPATCH: Automated Adaptive Prompting Large Language Models for Real-World Software Vulnerability Patching.” arXiv, 2024. doi: 10.48550/ARXIV.2408.13597. Available: https://arxiv.org/abs/2408.13597
[46] J. Lin and D. Mohaisen, “Evaluating Large Language Models in Vulnerability Detection Under Variable Context Windows.” arXiv, 2025. doi: 10.48550/ARXIV.2502.00064. Available: https://arxiv.org/abs/2502.00064
[47] X. Lin et al., “IRCopilot: Automated Incident Response with Large Language Models.” arXiv, 2025. doi: 10.48550/ARXIV.2505.20945. Available: https://arxiv.org/abs/2505.20945
[48] Z. Liu, “Multi-Agent Collaboration in Incident Response with Large Language Models.” arXiv, 2024. doi: 10.48550/ARXIV.2412.00652. Available: https://arxiv.org/abs/2412.00652
[49] F. Perrina, F. Marchiori, M. Conti, and N. V. Verde, “AGIR: Automating Cyber Threat Intelligence Reporting with Natural Language Generation,” arXiv preprint arXiv:2310.02655, 2023. Available: https://arxiv.org/abs/2310.02655
[50] P. N. Wudali, M. Kravchik, E. Malul, P. A. Gandhi, Y. Elovici, and A. Shabtai, “Rule-ATT&CK Mapper (RAM): Mapping SIEM Rules to TTPs Using LLMs,” arXiv preprint arXiv:2502.02337, Feb. 2025. Available: https://arxiv.org/abs/2502.02337
[51] D. Goel et al., “X-lifecycle Learning for Cloud Incident Management using LLMs.” arXiv, 2024. doi: 10.48550/ARXIV.2404.03662. Available: https://arxiv.org/abs/2404.03662
[52] M. Albanese, X. Ou, K. Lybarger, D. Lende, and D. Goldgof, “Towards AI-driven human-machine co-teaming for adaptive and agile cyber security operation centers,” arXiv preprint arXiv:2505.06394, Jun. 2025. Available: https://arxiv.org/abs/2505.06394
[53] D. Patel, S. Lin, J. Rayfield, N. Zhou, R. Vaculin, N. Martinez, F. O’Donncha, and J. Kalagnanam, “AssetOpsBench: Benchmarking AI Agents for Task Automation in Industrial Asset Operations and Maintenance,” arXiv preprint arXiv:2506.03828, 2025. Available: https://arxiv.org/abs/2506.03828
[54] S. Mitra, S. Neupane, T. Chakraborty, S. Mittal, A. Piplai, M. Gaur, and S. Rahimi, “LocalIntel: Generating organizational threat intelligence from global and local cyber knowledge,” arXiv preprint arXiv:2401.10036, Feb. 2025. Available: https://arxiv.org/abs/2401.10036
[55] S. Chopra, H. Ahmad, D. Goel, and C. Szabo, “ChatNVD: Advancing Cybersecurity Vulnerability Assessment with Large Language Models,” arXiv preprint arXiv:2412.04756, 2025. Available: https://arxiv.org/abs/2412.04756
[56] C. Rondanini, B. Carminati, E. Ferrari, A. Gaudiano, and A. Kundu, “Malware Detection at the Edge with Lightweight LLMs: A Performance Evaluation.” arXiv, 2025. doi: 10.48550/ARXIV.2503.04302. Available: https://arxiv.org/abs/2503.04302
[57] Y. Jin et al., “LLM-BSCVM: An LLM-based blockchain smart contract vulnerability management framework,” arXiv preprint arXiv:2505.17416, May 2025. Available: https://arxiv.org/abs/2505.17416
[58] E. Marian Pasca, D. Delinschi, R. Erdei, and O. Matei, “LLM-Driven, Self-Improving Framework for Security Test Automation: Leveraging Karate DSL for Augmented API Resilience,” IEEE Access, vol. 13, pp. 56861–56886, 2025, doi: 10.1109/access.2025.3554960. Available: http://dx.doi.org/10.1109/ACCESS.2025.3554960
[59] M. J. Torkamani, J. Ng, N. Mehrotra, M. Chandramohan, P. Krishnan, and R. Purandare, “Streamlining security vulnerability triage with large language models,” arXiv preprint arXiv:2501.18908, Jan. 2025. Available: https://arxiv.org/abs/2501.18908
[60] A. Applebaum et al., “Bridging Automated to Autonomous Cyber Defense,” Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security. ACM, pp. 149–159, Nov. 07, 2022. doi: 10.1145/3560830.3563732. Available: http://dx.doi.org/10.1145/3560830.3563732

[61] M. T. Alam, D. Bhusal, L. Nguyen, and N. Rastogi, “CTIBench: A Benchmark for Evaluating LLMs in Cyber Threat Intelligence.” arXiv, 2024. doi: 10.48550/ARXIV.2406.07599. Available: https://arxiv.org/abs/2406.07599
[62] Y. Xiao, V.-H. Le, and H. Zhang, “Stronger, Faster, and Cheaper Log Parsing With LLMs,” arXiv preprint arXiv:2406.06156, Jun. 2024. Available: https://arxiv.org/abs/2406.06156
[63] M. Khayat et al., “Empowering Security Operation Center With Artificial Intelligence and Machine Learning—A Systematic Literature Review,” IEEE Access, vol. 13, pp. 19162–19194, 2025, doi: 10.1109/ACCESS.2025.3532951. Available: https://doi.org/10.1109/ACCESS.2025.3532951.
[64] Y. L. Aung, I. Christian, Y. Dong, X. Ye, S. Chattopadhyay, and J. Zhou, “Generative AI for Internet of Things Security: Challenges and Opportunities,” arXiv preprint arXiv:2502.08886, Feb. 2025. Available: https://arxiv.org/abs/2502.08886
[65] K.-T. Tran, D. Dao, M.-D. Nguyen, Q.-V. Pham, B. O’Sullivan, and H. D. Nguyen, “Multi-Agent Collaboration Mechanisms: A Survey of LLMs.” arXiv, 2025. doi: 10.48550/ARXIV.2501.06322. Available: https://arxiv.org/abs/2501.06322
[66] H. Jin et al., “Large Language Models for Anomaly Detection in Computational Workflows: from Supervised Fine-Tuning to In-Context Learning.” arXiv, 2024. doi: 10.48550/ARXIV.2407.17545. Available: https://arxiv.org/abs/2407.17545
[67] M. Yong Wong, K. Valakuzhy, M. Ahamad, D. Blough, and F. Monrose, “Understanding LLMs Ability to Aid Malware Analysts in Bypassing Evasion Techniques,” Companion Proceedings of the 26th International Conference on Multimodal Interaction. ACM, pp. 36–40, Nov. 04, 2024. doi: 10.1145/3686215.3690147. Available: http://dx.doi.org/10.1145/3686215.3690147
[68] M. B. Chhetri et al., “Towards Human-AI Teaming to Mitigate Alert Fatigue in Security Operations Centres,” ACM Trans. Internet Technol., vol. 24, no. 3, Art. 12, 22 pp., Jul. 2024, doi: 10.1145/3670009. Available: https://doi.org/10.1145/3670009.
[69] S. Freitas, J. Kalajdjieski, A. Gharib, and R. McCann, “AI-Driven Guided Response for Security Operation Centers with Microsoft Copilot for Security,” in Companion Proc. ACM Web Conf. 2025 (WWW Companion ’25), Sydney, NSW, Australia, Apr.–May 2025, pp. 1–10, doi: 10.1145/3701716.3715209. Available: https://doi.org/10.1145/3701716.3715209.
[70] M. Kim et al., “CyberAlly: Leveraging LLMs and Knowledge Graphs to Empower Cyber Defenders,” Companion Proceedings of the ACM on Web Conference 2025. ACM, pp. 2851–2854, May 08, 2025. doi: 10.1145/3701716.3715171. Available: http://dx.doi.org/10.1145/3701716.3715171
[71] D. R. Arikkat, A. M., N. Binu, P. M., N. Biju, K. S. Arunima et al., “IntellBot: Retrieval Augmented LLM Chatbot for Cyber Threat Knowledge Delivery,” arXiv preprint arXiv:2411.05442, Nov. 2024. . Available: https://arxiv.org/abs/2411.05442
[72] M. Xu et al., “IntelEX: A LLM-driven attack-level threat intelligence extraction framework,” arXiv preprint arXiv:2412.10872, Dec. 2024.Available: https://arxiv.org/abs/2412.10872
[73] S. Paul, F. Alemi, and R. Macwan, “LLM-assisted proactive threat intelligence for automated reasoning,” arXiv preprint arXiv:2504.00428, Apr. 2025. Available: https://arxiv.org/abs/2504.00428
[74] S. K. Ghosh, R. Gjomemo, and V. N. Venkatakrishnan, “Citar: Cyberthreat Intelligence-driven Attack Reconstruction,” Proceedings of the Fifteenth ACM Conference on Data and Application Security and Privacy. ACM, pp. 245–256, Jun. 19, 2024. doi: 10.1145/3714393.3726519. Available: http://dx.doi.org/10.1145/3714393.3726519
[75] P. Las-Casas, A. G. Kumbhare, R. Fonseca, and S. Agarwal, “LLexus: an AI agent system for incident management,” SIGOPS Oper. Syst. Rev., vol. 58, no. 1, pp. 23–36, Aug. 2024, doi: 10.1145/3689051.3689056. Available: http://dx.doi.org/10.1145/3689051.3689056
[76] S. Hays and J. White, “Employing LLMs for Incident-Response Planning and Review,” arXiv preprint arXiv:2403.01271, Mar. 2024. Available: https://arxiv.org/abs/2403.01271
[77] Z. Liu, “AutoBnB: Multi-Agent Incident Response with Large Language Models,” 2025 13th International Symposium on Digital Forensics and Security (ISDFS). IEEE, pp. 1–6, Apr. 24, 2025. doi: 10.1109/isdfs65363.2025.11012055. Available: http://dx.doi.org/10.1109/ISDFS65363.2025.11012055
[78] Y. Sun et al., “TrioXpert: An Automated Incident Management Framework for Microservice Systems,” arXiv preprint arXiv:2506.10043, Jun. 2025. Available: https://arxiv.org/abs/2506.10043
[79] R. Singh, M. B. Chhetri, S. Nepal, and C. Paris, “ContextBuddy: AI-Enhanced Contextual Insights for Security Alert Investigation,” arXiv preprint arXiv:2506.09365, 2025.Available: https://arxiv.org/abs/2506.09365
[80] R. I. T. Jensen, V. Tawosi, and S. Alamir, “Software Vulnerability and Functionality Assessment using LLMs.” arXiv, 2024. doi: 10.48550/ARXIV.2403.08429. Available: https://arxiv.org/abs/2403.08429
[81] X. Lian et al., “Configuration Validation with Large Language Models.” arXiv, 2023. doi: 10.48550/ARXIV.2310.09690. Available: https://arxiv.org/abs/2310.09690
[82] Z. Sheng, F. Wu, X. Zuo, C. Li, Y. Qiao, and L. Hang, “LProtector: An LLM-driven Vulnerability Detection System.” arXiv, 2024. doi: 10.48550/ARXIV.2411.06493. Available: https://arxiv.org/abs/2411.06493
[83] M. Xu et al., “Forewarned is Forearmed: A Survey on Large Language Model-based Agents in Autonomous Cyberattacks.” arXiv, 2025. doi: 10.48550/ARXIV.2505.12786. Available: https://arxiv.org/abs/2505.12786
[84] D. Toprani and V. K. Madisetti, “LLM Agentic Workflow for Automated Vulnerability Detection and Remediation in Infrastructure-as-Code,” IEEE Access, vol. 13, pp. 69 175-69 190, Apr. 2025, doi: 10.1109/ACCESS.2025.3560911 Available: https://doi.org/10.1109/ACCESS.2025.3560911.
[85] V. Beck, M. Landauer, M. Wurzenberger, F. Skopik, and A. Rauber, “System Log Parsing With Large Language Models: A Review,” arXiv preprint arXiv:2504.04877, May 2025. Available: https://arxiv.org/abs/2504.04877
[86] N. Kshetri and J. Voas, “Agentic Artificial Intelligence for Cyber Threat Management,” Computer, vol. 58, no. 5, pp. 86–90, May 2025, doi: 10.1109/MC.2025.3544797. Available: https://doi.org/10.1109/MC.2025.3544797.
[87] S. Massengale and P. Huff, “Linking Threat Agents to Targeted Organizations: A Pipeline for Enhanced Cybersecurity Risk Metrics,” 2024 4th Intelligent Cybersecurity Conference (ICSC). IEEE, pp. 132–141, Sep. 17, 2024. doi: 10.1109/icsc63108.2024.10895328. Available: http://dx.doi.org/10.1109/ICSC63108.2024.10895328
[88] A. Shukla, P. A. Gandhi, Y. Elovici, and A. Shabtai, “RuleGenie: SIEM Detection Rule Set Optimization.” arXiv, 2025. doi: 10.48550/ARXIV.2505.06701. Available: https://arxiv.org/abs/2505.06701
[89] Y. Fu, X. Yuan, and D. Wang, “RAS-Eval: A Comprehensive Benchmark for Security Evaluation of LLM Agents in Real-World Environments.” arXiv, 2025. doi: 10.48550/ARXIV.2506.15253. Available: https://arxiv.org/abs/2506.15253
[90] P. Hamadanian, B. Arzani, S. Fouladi, S. K. R. Kakarla, R. Fonseca, D. Billor et al., “A Holistic View of AI-Driven Network Incident Management,” in Proc. 22nd ACM Workshop on Hot Topics in Networks (HotNets ’23), Cambridge, MA, USA, Nov. 2023, pp. 1–9, doi: 10.1145/3626111.3628176. Available: https://doi.org/10.1145/3626111.3628176.
[91] J. Bono, J. Grana, and A. Xu, “Generative AI and Security Operations Center Productivity: Evidence from Live Operations.” arXiv, 2024. doi: 10.48550/ARXIV.2411.03116. Available: https://arxiv.org/abs/2411.03116
[92] P. A. Gandhi, A. Shukla, D. Tayouri, B. Ifland, Y. Elovici, R. Puzis, and A. Shabtai, “ATAG: AI-Agent Application Threat Assessment with Attack Graphs,” arXiv preprint arXiv:2506.02859, 2025. Available: https://arxiv.org/abs/2506.02859
[93] P. Bountakas et al., “SYNAPSE - An Integrated Cyber Security Risk & Resilience Management Platform, With Holistic Situational Awareness, Incident Response & Preparedness Capabilities: SYNAPSE,” Proceedings of the 19th International Conference on Availability, Reliability and Security. ACM, pp. 1–10, Jul. 30, 2024. doi: 10.1145/3664476.3669924. Available: http://dx.doi.org/10.1145/3664476.3669924

[94] A. Sarkar and S. Sarkar, “Survey of LLM Agent Communication with MCP: A Software Design Pattern Centric Review.” arXiv, 2025. doi: 10.48550/ARXIV.2506.05364. Available: https://arxiv.org/abs/2506.05364
[95] P. Tseng, Z. Yeh, X. Dai, and P. Liu, “Using LLMs to Automate Threat Intelligence Analysis Workflows in Security Operation Centers.” arXiv, 2024. doi: 10.48550/ARXIV.2407.13093. Available: https://arxiv.org/abs/2407.13093
[96] A. Ding et al., “Generative AI for Software Security Analysis: Fundamentals, Applications, and Challenges,” IEEE Software, vol. 41, no. 5, pp. 46–55, 2024, doi: 10.1109/MS.2024.3416036.. Available: https://doi.org/10.1109/MS.2024.3416036.
[97] S. R. Castro, R. Campbell, N. Lau, O. Villalobos, J. Duan, and A. A. Cardenas, “Large Language Models are Autonomous Cyber Defenders.” arXiv, 2025. doi: 10.48550/ARXIV.2505.04843. Available: https://arxiv.org/abs/2505.04843
[98] P. F. Saura, K. R. Jayaram, V. Isahagian, J. Bernal Bernabé, and A. Skarmeta, “On Automating Security Policies with Contemporary LLMs,” arXiv preprint arXiv:2506.04838, 2025. [Online]. Available: https://arxiv.org/abs/2506.04838
[99] S. Oesch et al., “Towards a High Fidelity Training Environment for Autonomous Cyber Defense Agents,” Proceedings of the 17th Cyber Security Experimentation and Test Workshop. ACM, pp. 91–99, Aug. 13, 2024. doi: 10.1145/3675741.3675752. Available: http://dx.doi.org/10.1145/3675741.3675752
[100] P. Subramaniam and S. Krishnan, “DePLOI: Applying NL2SQL to Synthesize and Audit Database Access Control.” arXiv, 2024. doi: 10.48550/ARXIV.2402.07332. Available: https://arxiv.org/abs/2402.07332
[101] D. Roy et al., “Exploring LLM-based agents for root cause analysis,” in Proc. ACM Int. Conf. on Foundations of Software Engineering (FSE Companion), Porto de Galinhas, Brazil, Jul. 2024. DOI: 10.1145/3663529.3663841. Available: https://doi.org/10.1145/3663529.3663841.
[102] S. P. Shah and A. V. Deshpande, “Addressing Data Poisoning and Model Manipulation Risks using LLM Models in Web Security,” 2024 International Conference on Distributed Systems, Computer Networks and Cybersecurity (ICDSCNC). IEEE, pp. 1–6, Sep. 20, 2024. doi: 10.1109/icdscnc62492.2024.10941696. Available: http://dx.doi.org/10.1109/ICDSCNC62492.2024.10941696
[103] R. Kalakoti, R. Vaarandi, H. Bahşi, and S. Nõmm, “Evaluating Explainable AI for Deep Learning-Based Network Intrusion Detection System Alert Classification,” arXiv preprint arXiv:2506.07882, 2025. Available: https://arxiv.org/abs/2506.07882