Toward Intelligent SOCs: Agentic AI for Cyber Threat Triage Using LLMs and Threat Intelligence

PI: Dr. Xusheng Xiao, Associate Professor, xusheng.xiao@asu.edu
School of Computing and Augmented Intelligence, Arizona State University

1. Introduction

In recent years, enterprises have increasingly faced cyber threats aimed at penetrating their networks to compromise security [1–3]. According to the 2024 Report on the Cybersecurity Posture of the United States [4], there was a 22% increase in reported ransomware incidents compared to 2022, alongside a 74% rise in the associated costs. Additionally, the CrowdStrike 2024 Global Threat Report highlights a 75% increase in cloud intrusions, noting that the fastest recorded cyber attack was executed in just 2 minutes [5]. Both reports underscore that the advent of generative AI has equipped threat actors with more sophisticated tools for social engineering attacks, thereby exacerbating the complexity and scale of cyber threats. To combat these ever-increasing threats, enterprise Security Operations Centers (SOCs) are increasingly adopting Threat Detection Systems (TDS) such as intrusion detection systems and Security Information and Event Management (SIEM) platforms like Zeek [6] and Dragos [7]. These systems enable real-time monitoring and alerting across organizational networks. As a result, alert triage has emerged as a critical process for interpreting and prioritizing these alerts, helping security teams allocate attention and resources to the most severe and actionable incidents.

Despite their effectiveness, these automated systems are also well-known for generating an excessive number of false positives [8]. A FireEye report highlights that the average organization receives 17,000 alerts per week, with over 51% identified as false positives and just 4% receiving adequate investigation [9]. This limitation arises because modern cyber attacks, such as Advanced Persistent Threats (APTs) [10], often exploit legitimate system privileges or zero-day vulnerabilities to stealthily advance toward high-value targets. These attacks typically unfold over multiple stages, generating numerous alerts, yet current systems lack the capability to reconstruct the complex causal relationships between these attack steps [11, 12]. For instance, TDS and SIEM systems may detect suspicious network connections, unknown files, and the receipt of new emails by users. Although they can assign risk scores to each individual event, they lack the ability to correlate only the suspicious events and assess whether these events are related or warrant escalation for further investigation. Making matters worse, a recent study [13] reveals that although organizations actively collect and share Cyber Threat Intelligence (CTI) [14], only 35% of organizations believe they fully understand various threat actors and their associated TTPs, and 68% express the need for enhanced awareness of the evolving threat landscape. Consequently, many enterprises still lack the necessary knowledge to effectively investigate and defend against even known threats.

To address these fundamental challenges faced by SOCs, we propose an agentic-AI framework, Spectra (Security Provenance and CTI-driven Triage Automation), which leverages Large Language Models (LLM) [15] to create AI agents that autonomously conduct alert triage by analyzing system activity and incorporating expert insights from CTI and reconstruct attack scenarios. Unlike existing agentic AI systems for cyber threat triage, which primarily adopt LLM-based chatbots to explain individual alerts, Spectra is equipped with four innovative designs: (1) Many alerts represent same system activities, such as an unscheduled upgrade operation that repetitively adds and deletes numerous files. Thus, Spectra will first group the received alerts based on their commonly accesses resources and attribute similarities, and employ AI agents powered by LLMs to investigate each group of alerts. (2) In addition to standard system logs and user activity data collected by TDS/SIEM systems (e.g., network traffic, firewall, and command logs), Spectra incorporates system auditing logs and performs provenance analysis (PA) [16] to correlate groups of alerts through shared system entities such as files and IP addresses. (3) Spectra will incorporate structured threat intelligence using a knowledge graph CtiKG derived from CTI sources including security bulletins and technical blogs, which will be built upon our lab's latest research [17]. (4) Finally, Spectra will introduce an agent-based planning mechanism that directs AI agents to retrieve relevant logs and query CtiKG to assess the presence of known attack patterns. This process will be repeated until Spectra can determine whether the evidence supports escalation for manual review.

2. Preliminary Work

Our lab has substantial experiences in developing PA on system audit logs for attack investigation [2, 18–23]. PA leverages system monitoring tools, such as Sysdig [24] and Linux Audit [25], to collect system call auditing events from the kernel. It then constructs a provenance graph (PG), in which nodes represent system entities—such as processes, files, and IP addresses—and edges capture control and data flows between these entities. Provenance analysis is usually applied on a Point-of-Interest (POI) event (e.g., creating a suspicious file) to generate a PG that reveals the contextual events of the POI event [16].

Besides developing PA, our lab also develops LLM-based techniques for cyber threat intelligence [17, 26]. Specifically, our recent research, CtiKG, utilizes prompt engineering to efficiently build a security-oriented knowledge graph from CTI articles based on LLMs. In particular, to mitigate the challenges of LLMs in randomness, hallucinations and tokens limitation, CtiKG divides an article into segments and employs multiple LLM agents with dual memory design to (1) process each text segment separately and (2) summarize the results of the text segments to generate more accurate results. Our results show that CtiKG achieves 86.88% precision in building security-oriented knowledge graphs, achieving at least 30% improvements over the state-of-the-art techniques.

3. Proposed Research

With recent advances in LLMs, security chatbots [27] have been introduced to support analysts in delivering threat intelligence. Nonetheless, their adoption in SOCs has been limited, largely due to insufficient automation of operational tasks. To address this gap, industry efforts have begun focusing on building agentic AI systems that aim to automate security workflows [28]. Yet, these solutions still fall short in correlating multiple alerts and reconstructing attack scenarios. Meanwhile, academic efforts such as PA-based alert triage [29] attempt to address alert correlation, but they lack integration with threat intelligence and fall short of providing verdicts for alert escalation. In this proposal, our agentic AI framework, Spectra, will be the first to integrate both PA and CTI knowledge to correlate alerts detected by TDS and seek evidence to make verdicts for grouped alerts.

Specifically, our proposed research is organized as the following research tasks.

- *T1: Alert Deduplication and Grouping* We will build upon our latest research [8] to perform alert deduplication. Alarm deduplication is based on the insight that a considerable number of false alarms represent the same behaviors. For example, a continuous file transfers will cause a series of network links for a period time, and an upgrade will cause many files to add or delete. Also, some alerts come from similar executed commands with only slightly different command arguments or operation objects, and they usually represent same system activities. Thus, we will adopt a time window w_g and all the alerts in w that have same IP sources and destinations or similar command arguments/operation objects will be merged. Furthermore, many alerts are describing same suspicious system activities from different perspectives, and we will compute text embedding of the alert attributes and group them based on the similarities.
- *T2: Provenance Analysis for Alert Correlation* Recent studies [16, 29], along with our own work [2, 23], have demonstrated that provenance analysis leveraging system auditing can effectively uncover dependencies among attack steps, thereby enabling the reconstruction of complex attack scenarios. Thus, we will build upon our PA techniques to construct a PG that represent the dependencies among system entities (e.g., files, processes, and IP channels) for a given Point-Of-Interest (POI) event described in an

alert. Particularly, PA often generates lots of irrelevant dependencies due to background system activities, and our latest algorithms [2, 30] can be applied to effectively filter out irrelevant dependencies. These identified dependencies can then be used to identify the correlated system entities and further group the alerts. For example, an attack scenario where a web server is compromised to run a Java program that then downloads a malicious script could generate multiple alerts, covering suspicious network connections and file downloads. By applying PA, these alerts can be correlated through the causal chain: the web server spawning the Java process, the Java process initiating the download, and the resulting creation of the downloaded file.

- *T3: Integration of Cyber Threat Intelligence* CTI [14] knowledge provides valuable context that enables security analysts to judge whether reported system activities represent malicious behaviors previously identified in other attacks. CTI is primarily distributed through natural language reports found in official sources like CVE/NVD, as well as in security bulletins and online forums. Even though LLMs are well-suited to analyze these articles, simply providing them with all articles and instructing them to find supporting evidence for detected alerts remains highly challenging. The sheer volume of text can easily lead to hallucinations or cause the LLM to exceed token limits. To address this challenge, we will build upon our latest research, CtiKG [17], that employs a multi-agent approach to summarize CTI articles as a security-oriented knowledge graph. This structured representation greatly reduces irrelevant information described in these CTI articles, and can be easily consumed by LLMs to find evidence for specific system activities reported in the correlated alerts. Specifically, Spectra will generate query subgraphs from each group of alerts and applies neural graph submatching [31] to locate similar subgraphs in CtiKG. We will subsequently utilize the original text associated with these CtiKG subgraphs to interpret and describe the attack steps.
- *T4: Agentic AI for Alert Triage* We will build the agentic AI framework Spectra by utilizing the techniques developed in T1, T2, and T3. Specifically, Spectra will devise a set of LLM-based agents to accomplish the automatic alert triage: grouper, PA-executer, searcher, and planner.
  - *Grouper*: this agent will employ the techniques described in T1 to group received alerts.
  - *PA-executer*: this agent will employ the techniques described in T2 to find dependencies of the representative POI event (e.g., file downloads or network links) detected in each group of alerts.
  - *Searcher*: this agent will employ the techniques developed in T3 to find evidence from CtiKG.
  - *Planner*: this agent will continuously collects evidence and makes verdicts for the grouped alerts. Specifically, it will maintain a time window w_p, and instruct the grouper to group the received alerts in w_p. It then will instruct the PA-executor to correlated the alerts, and employ the searcher to find evidence from CTI articles. Once all these steps are done, it will make a verdict for each correlated groups of alerts and determine whether to escalate them for further manual review. If not, following our recent work [29], alerts with lower risky scores will expire first, and alerts that are correlated to latest alerts will get their expiration times extended. Finally, new alerts will be put into w_p for another round of analysis.

4. Evaluation Plan

We plan to build the provenance analysis based on the developed infrastructure [2, 20, 21, 23] in our lab, which uses Sysdig [24] to collect system audit logs. We will build CtiKG upon OpenAI Library [32] to achieve chat interactions with LLM. We will deploy our developed system in our lab servers, and perform various attacks in the deployed environment, and applied the developed system to analyze and predict these attacks. Our attacks will include attacks that are used as test cases in prior work [18], and also the Advanced Persistent Threat (APT) attacks used in our previous work [20, 21] and industrial settings [10]. Additionally, we will use the dataset released by the DARPA TC program [33] by converting their data formats to our databases, and apply our system to investigate their attacks.

5. Expected Outcome

Our proposed research will produce a agentic-AI system based on LLMs that can integrate system auditing and cyber threat intelligence to correlate alerts detected by TDS/SIEM and perform alert triage, which will greatly reduce the securty analysts' efforts in SOC. We will open source our implementation, release our experimental datasets, and write research papers to share our research findings.

References

[1] Eric M Hutchins, Michael J Cloppert, and Rohan M Amin. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research*, 1:80, 2011.

[2] DepImpact Project Website, 2021. https://github.com/usenixsub/DepImpact.

[3] New York Times. Target data breach incident, 2014. http://www.nytimes.com/2014/02/27/business/target-reports-on-fourth-quarter-earnings.html?_r=1.

[4] Office of the National Cyber Director. 2024 report on the cybersecurity posture of the united states, 2024. URL https://www.whitehouse.gov/wp-content/uploads/2024/05/2024-Report-on-the-Cybersecurity-Posture-of-the-United-States.pdf.

[5] CrowdStrike. Crowdstrike 2024 global threat report, 2024. URL https://www.crowdstrike.com/global-threat-report/.

[6] The Zeek Project. Zeek - Network Security Monitor. https://zeek.org/, 2024. Accessed: 2025-03-20.

[7] Dragos Inc. Dragos: Safeguarding Civilization. https://www.dragos.com/, 2024. Accessed: 2025-03-20.

[8] Feng Dong, Liu Wang, Xu Nie, Fei Shao, Haoyu Wang, Ding Li, Xiapu Luo, and **Xusheng Xiao**. {DISTDET}: A {Cost-Effective} distributed cyber threat detection system. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 6575–6592, 2023.

[9] FireEye & IDC. The numbers game: How many alerts is too many to handle? White Paper, FireEye, January 2015. Special Report.

[10] Fireeye. Anatomy of Advanced Persistent Threats, 2017. https://www.fireeye.com/current-threats/anatomy-of-a-cyber-attack.html.

[11] Zhenyuan Li, Runqing Yang, Qi Alfred Chen, and Yan Chen. Mimic the whole attack chain: A first look at evasion against provenance graph based detection. 12 2020. doi: 10.13140/RG.2.2.20941.87528.

[12] J. Dinal Herath, Ping Yang, and Guanhua Yan. Real-Time evasion attacks against deep learning-based anomaly detection from distributed system logs. In *Proceedings of the ACM Conference on Data and Application Security and Privacy (CODASPY)*, page 29–40, 2021.

[13] Mandiant. Global perspectives on threat intelligence, 2024. URL https://assets.starlinkme.net/gitex-vendor-assets/mandiant/Global%20Perspectives%20on%20Threat%20Intelligence.pdf.

[14] Rob McMillan. Open threat intelligence, 2013. https://www.gartner.com/doc/2487216/definition-threat-intelligence.

[15] Y. Wang, Y. Zhang, Y. Li, and X. Liu. A bibliometric review of large language models research from 2017 to 2023. *arXiv preprint arXiv:2304.02020*, 2023.

[16] Samuel T. King and Peter M. Chen. Backtracking intrusions. In *ACM Symposium on Operating systems principles (SOSP)*, pages 223–236. ACM, 2003.

[17] Liangyi Huang and **Xusheng Xiao**. CTIKG: LLM-Powered Knowledge Graph Construction from Cyber Threat Intelligence. In *Proceedings of the First Conference on Language Modeling (COLM)*, 2024.

[18] Zhang Xu, Zhenyu Wu, Zhichun Li, Kangkook Jee, Junghwan Rhee, **Xusheng Xiao**, Fengyuan Xu,

Haining Wang, and Guofei Jiang. High fidelity data reduction for big data security dependency analyses. In *ACM Conference on Computer and Communications Security (CCS)*, pages 504–516, 2016.

[19] Yu Tao Tang, Ding Li, Zhi Chun Li, Mu Zhang, Kangkook Jee, Xu Sheng Xiao, Zhen Yu Wu, Junghwan Rhee, Feng Yuan Xu, and Qun Li. Nodemerge: Template based efficient data reduction for big-data causality analysis. In *ACM Conference on Computer and Communications Security (CCS)*, pages 1324–1337, 2018.

[20] Peng Gao, **Xusheng Xiao**, Zhichun Li, Fengyuan Xu, Sanjeev R. Kulkarni, and Prateek Mittal. AIQL: Enabling efficient attack investigation from system monitoring data. In *USENIX Annual Technical Conference (ATC)*, pages 113–126, 2018.

[21] Peng Gao, **Xusheng Xiao**, Ding Li, Zhichun Li, Kangkook Jee, Zhenyu Wu, Chung Hwan Kim, Sanjeev R. Kulkarni, and Prateek Mittal. SAQL: A stream-based query system for real-time abnormal system behavior detection. In *USENIX Security Symposium*, pages 639–656, 2018.

[22] Jiaping Gui, **Xusheng Xiao**, Ding Li, Chung Hwan Kim, and Haifeng Chen. Progressive processing of system behavioral query. In *Annual Computer Security Applications Conference (ACSAC)*, pages 378–389, 2019.

[23] Zhiqiang Xu, Pengcheng Fang, Changlin Liu, **Xusheng Xiao**, Yu Wen, and Dan Meng. Depcomm: Graph summarization on system audit logs for attack investigation. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 540–557. IEEE, 2022.

[24] Sysdig. Sysdig, 2017. https://sysdig.com/.

[25] Redhat. The linux audit framework, 2017. https://github.com/linux-audit/.

[26] Peng Gao, Fei Shao, Xiaoyuan Liu, **Xusheng Xiao**, Zheng Qin, Fengyuan Xu, Prateek Mittal, Sanjeev R. Kulkarni, and Dawn Song. Enabling efficient cyber threat hunting with cyber threat intelligence. In *Proceedings of the IEEE International Conference on Data Engineering (ICDE)*, pages 193–204, 2021.

[27] Dincy R. Arikkat, Abhinav M., Navya Binu, Parvathi M., Navya Biju, K. S. Arunima, Vinod P., Rafidha Rehiman K. A., and Mauro Conti. Intellbot: Retrieval augmented llm chatbot for cyber threat knowledge delivery, 2024. URL https://arxiv.org/abs/2411.05442.

[28] Payal Chakravarty and Vijay Ganti. The dawn of agentic ai in security operations at RSAC 2025. Google Cloud Blog, April 2025. URL https://cloud.google.com/blog/products/identity-security/the-dawn-of-agentic-ai-in-security-operations-at-rsac-2025.

[29] Wajih Ul Hassan, Shengjian Guo, Ding Li, Zhengzhang Chen, Kangkook Jee, Zhichun Li, and Adam Bates. Nodoze: Combatting threat alert fatigue with automated provenance triage. In *Network and Distributed System Security Symposium (NDSS)*, 2019.

[30] Shaofei Li, Feng Dong, **Xusheng Xiao**, Haoyu Wang, Fei Shao, Jiedong Chen, Yao Guo, Xiangqun Chen, and Ding Li. Nodlink: An online system for fine-grained apt attack detection and investigation. In *Proceedings of the 2024 Network and Distributed System Security Symposium (NDSS)*, 2024.

[31] Rex Ying, Andrew Wang, Jiaxuan You, Chengtao Wen, Arquimedes Canedo, and Jure Leskovec. Neural subgraph matching. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=LMslR3CTzE_. OpenReview preprint.

[32] OpenAI. The official python library for the openai api, 2023. URL https://github.com/openai/openai-python. Accessed: 2024-01-27.

[33] DARPA. Transparent computing engagement 3 data release. https://github.com/darpa-i2o/Transparent-Computing.