Poster: Using LLMs to Automate Threat Intelligence Analysis Workflows in Security Operation Centers

Pei-Yu Tseng*, Lan Zhang†, Anoop Singhal‡, ZihDwo Yeh*, Xushu Dai*, Xiaoyan Sun§, Peng Liu*
*The Pennsylvania State University, PA, USA
†Northern Arizona University, AZ, USA
‡National Institute of Standards and Technology, MD, USA
§Worcester Polytechnic Institute, MA, USA
Email: jerry950909@gmail.com

***Index Terms*—Large language models(LLMs), AI agent, Retrieval-Augmented Generation(RAG), Security Information and Event Management(SIEM), Security Operations Center(SOC), Cyber Threat Intelligence(CTI), Indicators of Compromise (IOCs)**

I. Introduction

At the forefront of the industry, rule-based approaches remain the most reliable and predominantly used methods for detecting attacks, with SIEM being widely deployed across most organizations' and enterprises' SOC. Generally, SIEM systems gather logs from all devices and applications within an internal network. This includes logs from sources like Windows event logs, firewall logs, or logs from Apache and Nginx that record HTTP accesses, errors, and session data. These logs provide comprehensive records of all activities and events occurring on those devices or applications. Taking Windows as an example, the operating system logs a variety of events such as Process Creation, Network Connections, and File Creation, each with its own set of attributes. These attributes describe, among other details, the file path that spawned or created the main process (referred to as ParentImage), the arguments passed to the executable associated with the parent process (denoted as ParentCommandLine), and the current directory of the process, which is essentially the path without the image name (labeled as CurrentDirectory). This comprehensive logging ensures that each event is thoroughly documented, offering invaluable insights for system monitoring and security analysis. SIEM uses predefined rules composed of multiple fields to determine whether an event is abnormal. For example, a rule designed to detect Process Create events may include fields related to the command line or file paths, where the values within these fields are specified as regular expressions. When one or more attributes in an event match the corresponding regular expression defined in the rule, an alert is triggered, indicating a potential issue that requires further investigation. Since SIEM must rapidly identify text patterns within vast volumes of log data, regular expressions serve as the fundamental component of the detection rules.

When creating rules to detect specific attacks, security analysts begin by reviewing relevant CTI reports. These reports may originate from renowned cybersecurity companies such as FireEye and CrowdStrike, from experts on platforms like X (formerly Twitter) or Telegram, or from free security platforms like MITRE ATT&CK. By reading these reports, analysts gain insights into which techniques an attacker might use and how these techniques are implemented in practice. For example, a CTI report might indicate that an attacker modifies a registry key using specific commands—these commands and the targeted registry keys act as IOCs. The security analyst then generates regular expressions based on these IOCs and populates the corresponding fields in the detection rules. To summarize, when generating rules, the process generally involves three main steps: first, reading CTI reports; second, determining which elements qualify as IOCs; and third, generating the corresponding regular expressions. These steps require a substantial amount of manual effort and can take several hours to complete. While this approach may have been feasible in the past, the increasing frequency of global attacks over the past five years and the explosive monthly growth in CTI reports [1] have made heavy reliance on manual labor unsustainable.

Therefore, we propose an LLM-based AI agent[2] that, with just the text of a CTI report as input, automatically extracts IOCs, determines which ones are relevant, and generates the corresponding regular expressions. This approach significantly shortens the time required to develop rules, giving analysts the opportunity to be freed from labor-intensive and highly repetitive tasks.

II. Motivation example

In figure1, the upper section illustrates the traditional SOC workflow. In a conventional process, once a security analyst receives a CTI report, they might spend around 2 hours reading through it. Next, they would spend an additional 2 hours(using Google searches or relying on their own knowledge) to determine which information are IOCs. After that, they typically devote about 2 hours to building complex

regular expressions, often using regex debugging tools such as regex101 to iteratively debug and refine the regex until it meets compliance requirements. In contrast, our AI agent automates these processes. Depending on the CTI report and the number of IOCs, the entire workflow can be completed in under a few minutes. This efficiency is achieved by leveraging the LLM's built-in knowledge along with our AI agent's proprietary regex generation methodology.

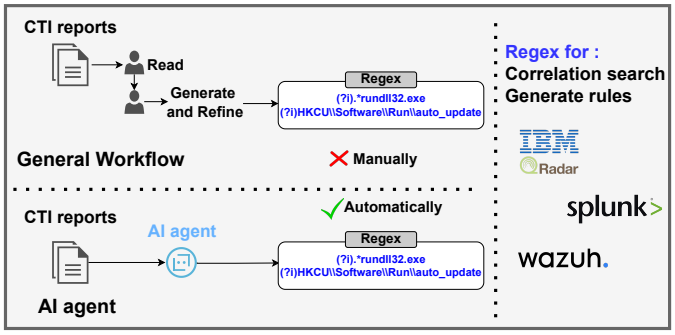


Fig. 1: Motivating example

III. System Design

In this research, we designed an AI agent capable of automatically extracting IOCs from CTI reports and generating regex. The input for this AI agent can be any article, paragraph, or even a short text related to threat intelligence. The output will be regex corresponding to the IOCs found in the given text. Generally, this AI agent can be divided into three steps. In the Suspicious IOC extraction phase, the AI agent automatically collects strings that are potential IOCs from the given text. The AI agent leverages multiple different LLM models to analyze the same text multiple times. The goal is to minimize the likelihood of missing any strings that could potentially be IOCs, ensuring that the AI agent captures all relevant candidates for subsequent analysis. Next, in the Capture Group Extraction and Filtering phase, the AI agent utilizes a pre-established graph database to perform retrieval searching. Based on a proprietary algorithm we developed, the agent will distinguish between capture groups and non-capture groups within the extracted strings. Then, it will filter out strings that do not contain any identified capture groups, as these are considered false positives resulting from the Suspicious IOC extraction phase. Finally, in the Regex generation phase, the AI agent employs a reasoning-based approach, leveraging LLM to iteratively generate regex. This iterative process helps refine the patterns, ensuring the regex is both accurate and effective in capturing relevant IOCs.

IV. Evaluation

In this experiment, we utilized OpenAI's GPT-4o to perform IOC extraction and regular expression generation, while the graph database was implemented using Neo4j. Additionally, all datasets and intermediate process data were stored in JSON format. The entire AI agent is built with over 5,000 lines of Python code.

1) Dataset: We collected references from MITRE ATT&CK as sources for the AI agent to generate regex. By crawling the content of MITRE ATT&CK, we ultimately collected 3,156 CTI reports as input. To validate whether the regex generated by our AI agent can be effectively applied in real-world environments, we utilized data from the MITRE ATT&CK Evaluation[3]. The MITRE ATT&CK Evaluation is an independent and rigorous assessment designed to test whether cybersecurity vendors' products can detect and report various types of attacks. A total of 19 well-known cybersecurity contractors participated, including Bitdefender, Palo Alto Networks, Trend Micro, among others. In this evaluation, 10 well-known attack scenarios (such as APT29, LockBit, etc.) are replicated. These vendors provide SIEM screenshots as evidence that they have successfully detected the execution of relevant TTPs. We manually collected all the vendor screenshots and extracted the strings that related to file paths, registry keys, and command lines from each one. In total, we collected over 2400 strings as a test set.

2) Result: In this experiment, we extracted over 15,000 IOCs and generated regular expressions that matched 99% of the test set. Additionally, the false positive rate of the generated regular expressions was only 3%, meaning that just 3% of them incorrectly matched strings from an unrelated test dataset.

V. Disclaimer

Commercial products are identified in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the identified products are necessarily the best available for the purpose.

References

[1] T. business report company, "Threat intelligence global market report." [Online]. Available: https://www.thebusinessresearchcompany.com/report/threat-intelligence-global-market-report#:~:text=The%20threat%20intelligence%20market%20size%20is%20expected%20to%20see%20rapid,intelligence%2C%20focus%20on%20cloud%20security.

[2] P. Tseng, Z. Yeh, X. Dai, and P. Liu, "Using llms to automate threat intelligence analysis workflows in security operation centers," 2024. [Online]. Available: https://arxiv.org/abs/2407.13093

[3] M. Corporation, "Mitre att&ck evaluation." [Online]. Available: https://attackevals.mitre-engenuity.org/

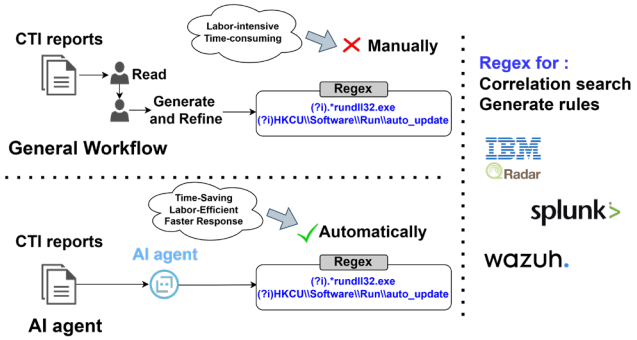
Using LLMs to Automate Threat Intelligence Analysis Workflows in Security Operation Centers

Pei-Yu Tseng[1], Lan Zhang[2], Anoop Singhal[3], ZihDwo Yeh[1], Xushu Dai[1], Xiaoyan Sun[4], Peng Liu[1]
[1]The Pennsylvania State University, [2]Northern Arizona University, [3]National Institute of Standards and Technology,
[4]Worcester Polytechnic Institute

Abstract

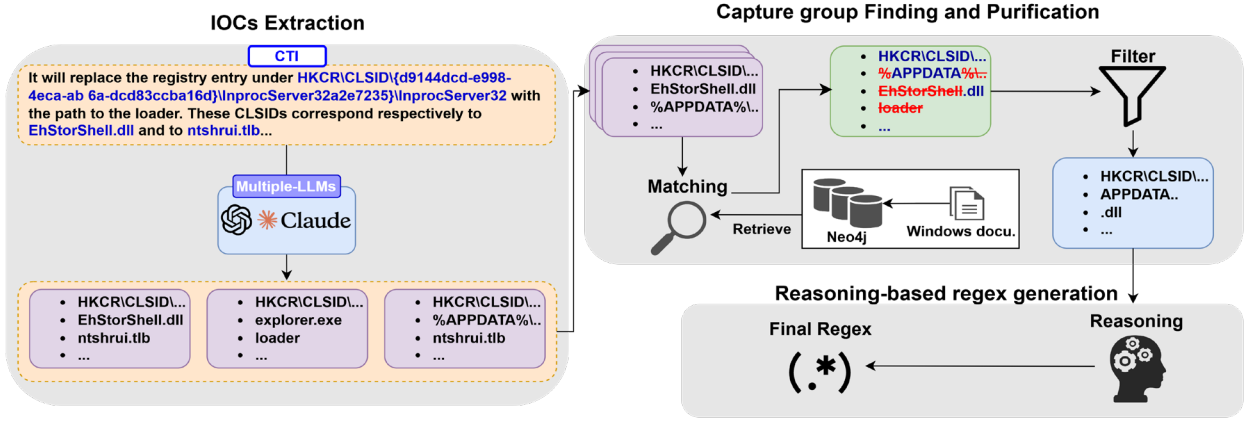
• **Cyber Threat Intelligence(CTI)** reports are essential for threat intelligence, with **Indicators of Compromise(IOCs)** forming the foundation for detection rules and forensic analysis because they provide concrete, observable artefacts that link suspicious or malicious activity back to specific threats.
• Creating regular expressions from IOCs enables the rapid identification of suspicious events within large volumes of logs.
• However, this task can consume significant human resources and cause security analysts to become fatigued, as the entire process is highly repetitive and perceived as low-value work.
• An AI agent is proposed to read CTI reports, extract IOCs, and generate regular expressions while reducing hallucination.

Motivation



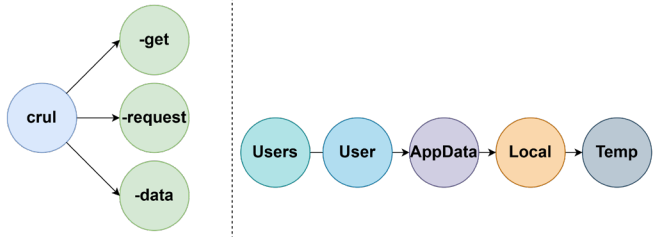
The rule-generation process typically comprises three main steps:
• Reading CTI reports
• Identifying which elements qualify as IOCs
• Generating the corresponding regular expressions

System design



• **Phase 1:** Employs several LLM passes to identify all possible IOC candidates.
• **Phase 2:** Utilizes a graph database and two algorithms to **distinguish between capture and non-capture groups**, filtering out false positives.
• **Phase 3:** Continuously refines and creates regex using a **reasoning-based method** to guarantee accuracy and efficiency.

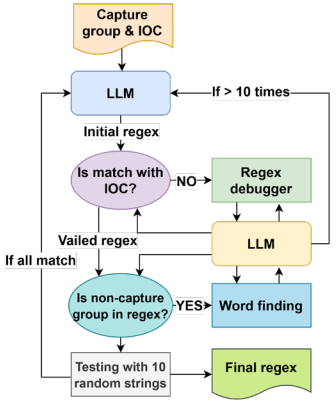
The graph structure of File paths, Registry keys, and CLI in the graph database



Examples of generated Regex

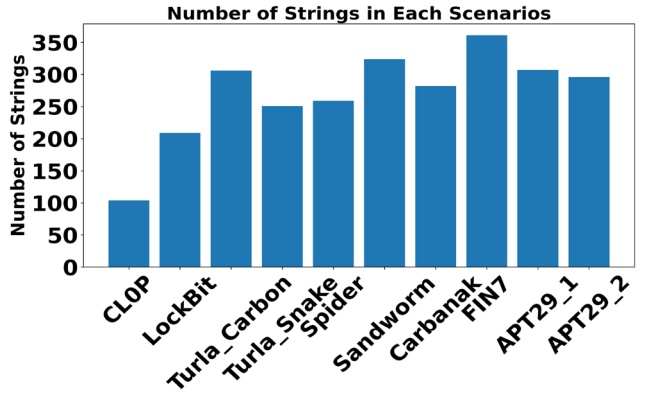
```
(?i).*vssadmin.*create.*shadow.*(/for=[A-Z]:|/for=.*).*
(?i).*icacls.*(\\/grant\\s+\\S+:F|\\/T|\\/C|\\/Q).*
```

Workflow of Reasoning base regex generation



• Confirm that the regex is **valid**.
• Verify that the regex does not **over-generalize**.
• Check that the regex **excludes non-capture groups**.

Evaluation & Dataset



• **False Negative Rate**: The number of ground truth strings that cannot be matched by the regexes generated from the collected IOCs.
  • **99%** can be matched.
• **False Positive Rate**: If a regex matches a ground-truth string that it shouldn't match, it is considered a false positive (FP).
  • The false positive rate was just **3%.**

Takeaway

Identifying IOCs in CTI reports and developing regex is a crucial step in forensic analysis, but it is both time-consuming and highly repetitive. We leveraged LLMs to automate IOC extraction and regex generation from CTI reports, achieving 90% IOC detection and 99% regex matching, freeing security analysts from repetitive work to focus on creative tasks.

Reference

[1] P. Tseng, Z. Yeh, X. Dai, and P. Liu, "Using llms to automate threatintelligence analysis workflows in security operation centers," 2024.[Online]. Available: https://arxiv.org/abs/2407.13093
[2] M. Corporation, "Mitre att&ck evaluation." [Online]. Available:https://attackevals.mitre-engenuity.org/