A Usability Evaluation Method for SOC Tools Using a Simulated Operational Environment

Yukina Okazawa
Toho University

Akira Kanaoka
Toho University
akira.kanaoka@is.sci.toho-u.ac.jp

Takumi Yamamoto
Mitsubishi Electric Corporation

***Abstract*—Security Operation Centers (SOCs) rely on security monitoring tools such as SIEM systems and IDSs, yet the usability of these tools remains insufficiently examined despite their essential role in analysts' daily workflows. Prior research has highlighted operational burdens including overwhelming alert volume, high false positive rates, and analyst fatigue. However, existing efforts have focused mainly on technical alert reduction rather than evaluating how effectively SOC tools support analysts' decision making in practice. This gap indicates the need for a structured and SOC specific usability evaluation methodology. This paper introduces a methodology for evaluating the usability of SOC tools that combines a heuristic walkthrough with eleven evaluation criteria derived from empirical studies of SOC operations. These criteria capture usability factors that general purpose techniques often overlook, such as context dependent interpretation, escalation reasoning, and reliance on environmental knowledge. To support controlled and reproducible evaluations, we also present a simulated operational environment that produces realistic sequences of alerts, benign events, and false positives based on representative attack scenarios. We apply the method to an open source SIEM, Prelude OSS, and demonstrate how the framework identifies recurring usability challenges such as limited contextual support, inconsistent workflow guidance, and difficulties in handling realistic alert volumes. These challenges align with previously reported issues in SOC practice, indicating that the proposed method can systematically expose usability problems inherent to many SOC tools rather than issues specific to a single system. Together, the methodology and simulated environment provide a foundation for rigorous and repeatable usability evaluations of SOC tools, complementing existing technical approaches to alert reduction and offering concrete directions for improving tool design.**

I. Introduction

A Security Operation Center (SOC) is an organization that continuously monitors networks and systems to detect security incidents, assess their impact, and determine appropriate responses. SOC analysts are responsible for monitoring alerts, identifying potential incidents, and escalating or responding to them as necessary. These responsibilities are typically divided among Tier 1, Tier 2, and Tier 3 analysts. Tier 1 analysts perform alert monitoring and initial triage, Tier 2 analysts conduct in depth investigation and incident response, and Tier 3 analysts provide advanced analysis and threat hunting.

Challenges in SOC operations are most evident at Tier 1, where analysts often face an overwhelming number of alerts and a high prevalence of false positives. Since commonly used SOC tools, including SIEM systems, IDSs, and log management platforms, tend to generate alerts for a wide range of suspicious activity, Tier 1 analysts must manually verify large volumes of alerts in their daily work. This workload, often handled by analysts with limited experience, has been identified as a major issue in SOC environments[1].

Research on SOC operations has grown rapidly, with numerous studies addressing the excessive volume of logs and alerts, the prevalence of false positives, and the resulting burden on SOC analysts[2], [3]. Technical approaches have focused on reducing alert volume or prioritizing alerts through improved detection algorithms, machine learning techniques, and more recently, large language models[2], [3], [4], [5], [6], [7], [8], [9]. In parallel, several studies have examined SOC analysts themselves, documenting the challenges of alert verification, the cognitive demands of continuous monitoring, and even the risk of fatigue and burnout within SOC teams[1], [10], [11].

Although numerous studies have examined operational challenges in SOCs, prior work has primarily focused on technical mechanisms for reducing alert volume or improving the precision of detection. In contrast, significantly fewer studies have investigated how effectively SOC tools support analysts' sensemaking, workflow navigation, and escalation decisions. Existing usability evaluation techniques provide valuable insights but are not tailored to the characteristics of SOC operations, which require analysts to interpret alerts in context, connect fragmented pieces of information, and make time sensitive judgments under uncertainty. As a result, there is currently no established methodology for evaluating the usability of SOC tools in a manner that reflects the realities of SOC environments.

To address this gap, we aim to develop a structured and SOC specific usability evaluation method grounded in empirical findings from SOC research. Our goal is to create an evaluation framework that captures usability dimensions not reflected in general purpose heuristics, supports reproducible assessments through a simulated operational environment, and provides actionable insights into tool design. Based on this objective, we define the following research questions.

Workshop on Security Operation Center Operations and Construction (WOSOC) 2026
23 February 2026, San Diego, CA, USA
ISBN 978-1-970672-03-9
https://dx.doi.org/10.14722/wosoc.2026.23012
www.ndss-symposium.org

This work is guided by the following research questions.

RQ1 How can the usability of SOC tools be systematically evaluated in a manner that reflects real operational conditions?

RQ2 What kinds of usability issues emerge when SOC tools are evaluated using a structured expert review, and to what extent are these issues specific to a particular tool or common across SOC tools in general?

Using the proposed method and evaluation environment, we identified both strengths and limitations of Prelude OSS, which is an open-source SIEM system. The evaluators rated the system positively for the richness and clarity of displayed information, the interactive exploration of alert details, and the ability to gather evidence needed for justifying escalation decisions. The overall volume of alerts generated in our simulated scenarios aligned with levels reported in prior studies, suggesting that the evaluation environment successfully captured realistic operational conditions. At the same time, several usability issues were observed. These included the difficulty of handling large volumes of alerts even when typical of real SOCs, the lack of clear differentiation between benign false positives and meaningful signals such as repeated login failures, and the necessity of relying on external knowledge about the monitored environment to complete escalation decisions.

These findings indicate that the limitations observed in Prelude OSS are not restricted to a single system but reflect broader usability challenges common across SOC tools, as suggested in previous SOC research. This highlights the need not only for techniques that reduce alert volume but also for systematic improvements to the usability of SOC tools themselves. The contributions of this work are as follows.

- We propose a structured usability evaluation method tailored to SOC tools, grounded in insights from prior research.
- We design and implement a realistic simulated environment that enables controlled and reproducible usability evaluations.
- We demonstrate the effectiveness of the method through a detailed evaluation of Prelude OSS, revealing usability issues with implications for SOC tools more broadly.

Unlike prior studies that focus on alert reduction or analyst behavior, this work provides the first structured methodology for evaluating the usability of SOC tools themselves. By constructing a reproducible simulated environment and deriving SOC-specific evaluation criteria grounded in empirical studies, our approach enables systematic and repeatable assessments that were previously difficult to conduct.

II. Related Work

A. Research Trends in SOC Operations and Alert Management

Research on Security Operation Centers (SOCs) has grown rapidly in recent years, as reflected in multiple survey studies that summarize common challenges and technological trends in SOC operations[2], [3]. A central and recurring issue identified across this literature is the excessive number of alerts generated by security tools. Accordingly, a large body of work has focused on alert reduction, prioritization, and automation, including approaches based on machine learning and large language models[4], [5], [12]. These studies have contributed to improving detection and triage capabilities, but they primarily emphasize algorithmic performance and accuracy.

B. Human Factors in SOC Operations

Beyond technical solutions, prior studies have highlighted the human-centered challenges faced by SOC analysts. Alahmadi et al.[1] conducted an in-depth interview study with SOC practitioners and showed that alert verification is a central and labor-intensive task, requiring analysts to manually classify alerts into actionable events, false positives, and benign triggers. They further reported that many alerts provide insufficient context, increasing cognitive load and verification time. More recently, Thimmaraju et al.[11] examined mental health issues among SOC analysts and identified burnout as a growing concern. Together, these studies indicate that SOC effectiveness is strongly influenced by how analysts interact with the tools that support daily operations.

C. Gaps in Usability-Oriented Evaluation of SOC Tools

While prior research has extensively examined alert management and analyst workload, the usability of SOC tools themselves has received limited attention. Existing studies largely assume that improving detection accuracy or reducing alert volume will directly alleviate analysts' burden, without systematically evaluating how tool interfaces support complex, time-sensitive decision making. Moreover, established usability evaluation methods have rarely been adapted to the unique operational context of SOCs. This gap motivates the need for structured usability evaluation approaches that explicitly consider SOC workflows and analyst cognition. Our work addresses this gap by proposing an expert-based usability evaluation method tailored to SOC tools, supported by a simulated operational environment.

III. SOC Tool Usability Evaluation Method

A. Motivation and Positioning of the Evaluation Method

Prior SOC research has examined operational challenges from organizational, procedural, and human perspectives. While these efforts have improved detection and triage techniques, the usability of tools used in daily SOC operations, such as SIEM systems and log analysis platforms, has received limited attention. Given the lack of established methodologies for evaluating SOC tool usability, we adopt an expert-based evaluation approach as a practical starting point.

B. Selection of an Expert Review Approach

We adopt the heuristic walkthrough as the core evaluation method. This approach combines cognitive walkthrough and heuristic evaluation, enabling both task-oriented inspection and

systematic assessment of interface elements. Prior security-focused usability studies have successfully applied this method to evaluate complex security tools[13], [14]. For SOC tools, which require analysts to interpret alerts while navigating complex interfaces under time pressure, this combined perspective is particularly suitable.

C. Five-point Rating Scale

During the heuristic walkthrough, evaluators record free-form observations and also assign a score to each evaluation criterion. To enable reliability analysis of multi-rater assessments, we use the following five-point rating scale:

- Score 5: Fully satisfies the evaluation criterion
- Score 4: Mostly satisfies the evaluation criterion
- Score 3: Neutral
- Score 2: Mostly does not satisfy the evaluation criterion
- Score 1: Does not satisfy the evaluation criterion

This scoring scheme promotes consistent interpretation among evaluators and enables the assessment of inter-rater agreement using established reliability metrics.

D. Development of Evaluation Criteria

We developed evaluation criteria grounded in empirical findings from prior SOC research. We began with Alahmadi et al.[1], whose qualitative study provides detailed insights into alert triage, information overload, and reliance on analysts' implicit knowledge. From these findings and subsequent SOC studies, we extracted factors that plausibly relate to tool usability, such as information presentation, alert interpretation, and workflow support. Organizational and managerial factors were excluded.

Through this abstraction process, we formulated eleven evaluation criteria that reflect usability-relevant dimensions of SOC operations. These criteria guide the heuristic walkthrough to focus on both task-oriented activities and interface-level characteristics. Table I lists the criteria, with detailed explanations provided in the supplementary materials.

All supplementary materials, including detailed explanations of eleven evaluation criteria are available at our public repository on Zenodo[15].

IV. Requirements for an Environment Used to Evaluate SOC Tools

Evaluating the usability of SOC tools requires an environment that reflects realistic SOC operations. Since tools such as SIEM systems and IDSs support analysts by presenting alerts and contextual information during daily operations, the usability of these tools is strongly influenced by the characteristics of the data they process. Prior studies have shown that SOC analysts routinely face large volumes of alerts, false positives, and difficulties in prioritization, indicating that usability evaluations must be conducted under conditions that mirror these operational realities.

To meaningfully assess whether SOC analysts can detect attacks and make appropriate escalation decisions, the evaluation environment must include not only attack-related data but also benign events and false positives. This mixture of operational noise and genuine threats is essential for reproducing the complexity of real SOC workflows.

Based on these considerations, we identify three essential components that must be designed when constructing an evaluation environment:

1) the evaluation network,
2) the attack scenarios,
3) the logs and alerts displayed in the SOC tool.

The evaluation network should represent a realistic operational architecture to ensure validity and reproducibility. Attack scenarios must capture plausible threats that SOC analysts are likely to encounter, avoiding both overly simplistic and unrealistically advanced cases. Finally, the logs and alerts presented by the SOC tool must reflect realistic distributions of benign events, false positives, and attack-related activity. An imbalance in these elements may bias the evaluation or obscure important usability issues.

V. Implementation of the Evaluation Environment

One approach to constructing an evaluation environment is to deploy a real world network, generate live traffic, and aggregate logs from physical hosts into the SOC tool. However, such an approach is costly, difficult to scale, and unsuitable for controlled usability evaluations.

Instead, we focus on characteristics shared across many SOC tools. Most SOC platforms support ingestion of Syslog data, which serves as a common format for integrating logs from heterogeneous security devices. Leveraging this property enables the construction of a simulated yet operationally plausible evaluation environment.

Based on this observation, we designed an environment in which synthesized Syslog entries are supplied to the SOC tool. For each attack scenario, we defined the sequence of attacker actions, the hosts involved, the components expected to generate logs, and the corresponding timestamps. Syslog entries were then generated from predefined templates using a custom script, allowing precise control over log content and timing without relying on physical devices.

In addition to attack related logs, we generated benign logs and false positive alerts to reflect realistic SOC operating conditions. False positive alerts were manually authored based on common alert structures, while benign logs were generated with the assistance of a large language model using descriptions of the environment and log formats. These logs were reviewed by multiple authors with expertise in enterprise network operations and SOC workflows.

To deliver logs to the SOC tool, a Syslog server was deployed as the log source. A separate log generation machine sent synthesized logs to the server while dynamically adjusting host identifiers and timestamps according to the defined scenarios. This setup enabled the SOC tool to process and display logs as if they originated from actual hosts in an operational environment.

	Evaluation Item
1	The number of displayed alerts is small.
2	The amount of displayed alerts is appropriate.
3	The quality of displayed alerts is good.
4	Appropriate measures are taken for benign triggers.
5	No prior knowledge of the monitored environment is required.
6	The displayed information is sufficiently comprehensive.
7	There are no factors that cause redundant or duplicate operations.
8	Help and reference documentation are available and sufficiently detailed.
9	When an alert indicating a potential threat is displayed, the entire process from detection to escalation can be completed using only the information shown on the screen.
10	It is possible to explain, with valid reasoning, why an alert should be addressed or can be safely ignored.
11	From an interface design perspective, it is possible to distinguish between alerts that require attention and those that can be ignored.

TABLE I
THE ELEVEN EVALUATION CRITERIA DERIVED FROM PRIOR SOC RESEARCH AND USED IN THE HEURISTIC WALKTHROUGH

VI. USABILITY EVALUATION ENVIRONMENT

A. Network Topology

We designed a network topology that represents a typical small-to-medium enterprise environment, including externally facing services, internal business systems, and end-user devices. The topology reflects common SOC monitoring targets such as web services, file servers, and employee workstations, while incorporating network security components including a firewall and an intrusion detection system. The resulting topology is shown in Figure 1.

B. Attack Scenarios

Although reported security incidents vary widely in scale and impact, many exhibit common structural patterns in attacker behavior and progression. Across public incident reports, threat scenario taxonomies, and widely used training and exercise materials, two incident classes appear repeatedly: externally initiated malware-based compromises and unauthorized activities involving valid credentials.

These patterns capture fundamental attack structures that SOC tools are expected to support during detection and analysis. Based on this observation, we selected two representative scenarios for the evaluation environment: a malware driven compromise scenario and an insider related unauthorized activity scenario. Together, these scenarios cover distinct but commonly encountered classes of incidents and provide a suitable foundation for usability evaluation.

1) Attack Scenario 1: Malware Infection: The sequence of events in this scenario is as follows.

1) A user inside the organization executes an email attachment that triggers malware infection, enabling remote control of the internal machine by an attacker.
2) Using the remotely controlled machine, the attacker performs network reconnaissance and discovers that the firewall and an internal file server are in operation.
3) The attacker accesses the firewall and modifies its configuration to evade detection.
4) The attacker searches for sensitive information on the file server and exfiltrates data.

Examples of logs and alerts generated by the attacker's communications are listed in chronological order in Table II.

2) Attack Scenario Related to an Insider Threat: This scenario begins with the attacker already possessing valid login credentials for the organization's web server. The sequence of events is as follows.

1) The attacker uses the obtained credentials to perform SSH login to the web server during off-hours (for example, late at night).
2) The attacker then performs SSH login to a desktop PC used by an internal user.
3) The attacker searches for sensitive information on the file server and exfiltrates data.

Examples of logs and alerts generated by the attacker's communications are listed in chronological order in Table III.

C. Displayed Alerts and Their Generation Process

To generate the logs and alerts used in the evaluation, we first collected real examples from the actual tools used in the evaluation network or from publicly available manuals. Based on these examples, we created templates and modified fields such as source and destination IP addresses, port numbers, and timestamps so that each entry reflected the details of the defined attack scenarios.

Many SIEM systems support both native log transfer mechanisms and Syslog ingestion. In this evaluation, we selected the Syslog format. A single Raspberry Pi acted as the log and alert sender, and the machine running the SIEM was configured as an external Syslog server. The Raspberry Pi transmitted synthesized entries to the Syslog server using the logger command, with hostnames and timestamps adjusted according to the chronological flow of each attack scenario. On the SIEM side, we configured mappings so that the received entries were presented appropriately within the interface, thereby reproducing a realistic incident analysis experience.

VII. USABILITY EVALUATION PROCEDURE

A. Selection of the Evaluation Target

In this study, we focus on SIEM systems as a representative class of SOC tools. Although many SIEM products exist, including commercial platforms and customized deployments operated by individual organizations, these systems present challenges for academic usability evaluation. Their internal correlation logic, configuration rules, and alert-processing

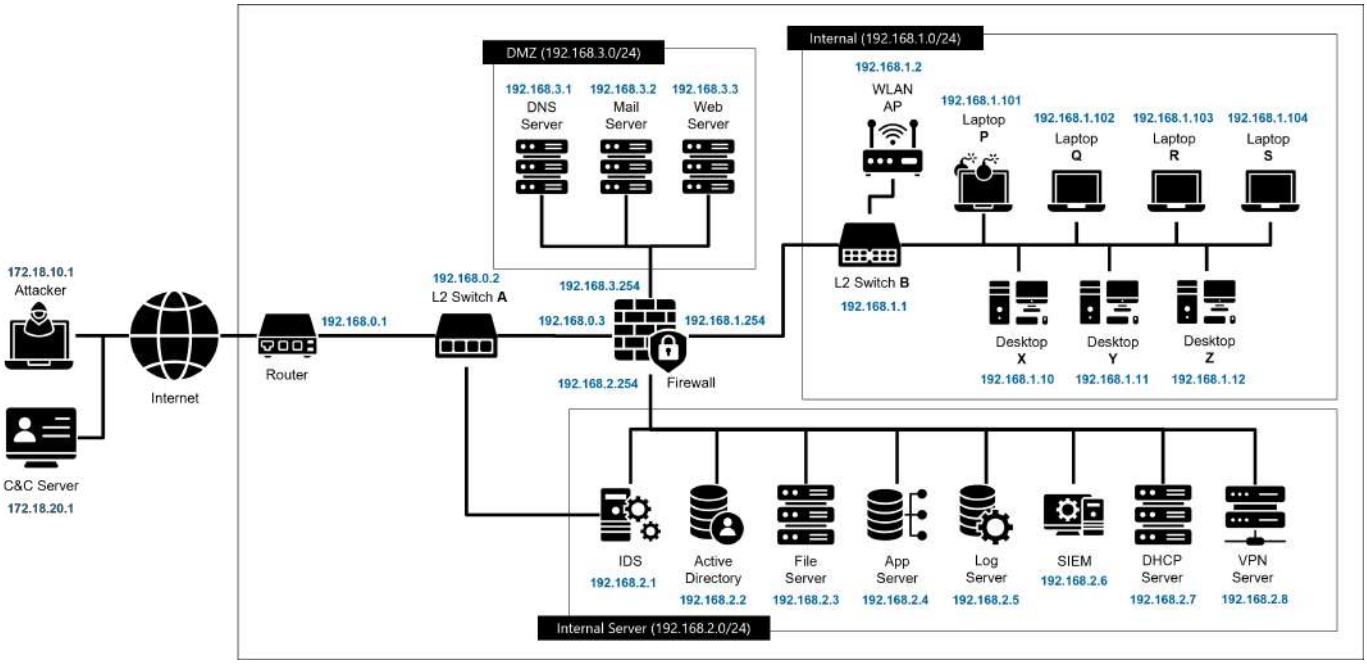

Fig. 1. Network topology of the simulated environment used in this evaluatioin. The model assumes a typical small-to-medium enterprise network.

TABLE II
CHRONOLOGY OF THE MALWARE INFECTION SCENARIO, ILLUSTRATING ALERT GENERATION DURING INFECTION, RECONNAISSANCE, AND SUBSEQUENT LOSS OF VISIBILITY AFTER FIREWALL CONFIGURATION CHANGES.

Time	Action	Log/Alert
14:26:16	Executing email attachment,	Proxy: 1730438776 6982 192.168.1.101 TCP_MISS/200 25682 GET http://malicious-site.com/malware.exe - HIER_DIRECT/172.18.20.1 application/octet-stream
14:26:19	Malware infection	Snort: [**] [1:53211:2] MALWARE-OTHER Win.Trojan.AZORult malicious executable download attempt [**] [Classification: A Network Trojan was Detected] [Priority :1] {TCP} 172.18.20.1:443 → 192.168.1.101:10125
14:31:24	Network reconnaissance	FW: logid=0000000011 type=traffic subtype=forward level=warning vd=root srcip=192.168.1.101 srcintf=wan dstip=192.168.1.1 dstintf=lan poluuid=64372cca-dba1-51ee-6d3f-8ce071356a3d sessionid=1266583 proto=1 action=ip-conn policyid=2 policytype=policy appcat=unscanned crscore=5 craction=262144 crlevel=low
14:47:52	Access to firewall	FW: logid=0100032001 type=event subtype=system level=information vd=root event-time=1730440196 logdesc=Admin login successful sn=1557771654 user=administrator ui=http(192.168.1.101) method=http srcip=192.168.1.101 dstip=192.168.1.254 action=login status=success reason=none profile=super_admin msg=Administrator administrator logged in successfully from http(192.168.1.101)
14:54:13	Firewall configuration changes	FW: eventtime=1730440571 tz=+0900 logid=0100044547 type=event subtype=system level=information vd=root logdesc=Object attribute configured user=administrator ui=GUI(192.168.1.101) action=Edit cfgtid=12714067 cfgpath=firewall.policy cfgobj=7 cfgattr=uuid[c2b1795e-c488-51ec-ee70-f00a4eaee6a9]srcaddr[all IPSec_RICH_172.24.216.50] msg=Edit firewall.policy 7
14:58:39	Sensitive data search	No logs/alerts are assumed to be sent to the SIEM after the firewall configuration change
15:26:52	Data exfiltration	No logs/alerts are assumed to be sent to the SIEM after the firewall configuration change

mechanisms are typically not publicly documented, which limits reproducibility and prevents controlled comparisons across studies.

To address this limitation, we selected the open-source version of Prelude SIEM, known as Prelude OSS, as the evaluation target. Prelude has long been used in both research and practice, and its open architecture enables detailed inspection of alert-processing behavior and configuration settings. Prelude OSS is explicitly described as suitable for research use in small-scale environments and is freely available[16]. These properties make it appropriate for reproducible and controlled usability evaluations. Examples of the Prelude OSS interface are shown in Figures 2 and 3.

B. Evaluation Environment

Each evaluator accessed the server running Prelude OSS through a web browser and performed the usability evaluation while role-playing as an SOC analyst who monitors security events and makes escalation decisions. The evaluation setup consisted of two 31.5-inch 4K monitors. The SIEM interface was displayed on the right monitor, while supporting docu-

Time	Action	Log/Alert
01:34:49	SSH login to web server	Accepted password for adm from 172.18.10.1 port 22 ssh2
01:35:30	SSH login to desktop PC	Accepted password for adm from 192.168.3.3 port 22 ssh2
01:39:43	Sensitive data search	No logs/alerts are assumed to be sent to the SIEM after the firewall configuration change
02:01:53	Data exfiltration	No logs/alerts are assumed to be sent to the SIEM after the firewall configuration change

TABLE III
CHRONOLOGY OF THE ATTACK SCENARIO RELATED TO AN INSIDER THREAT

Fig. 2. List view of alerts in Prelude OSS.

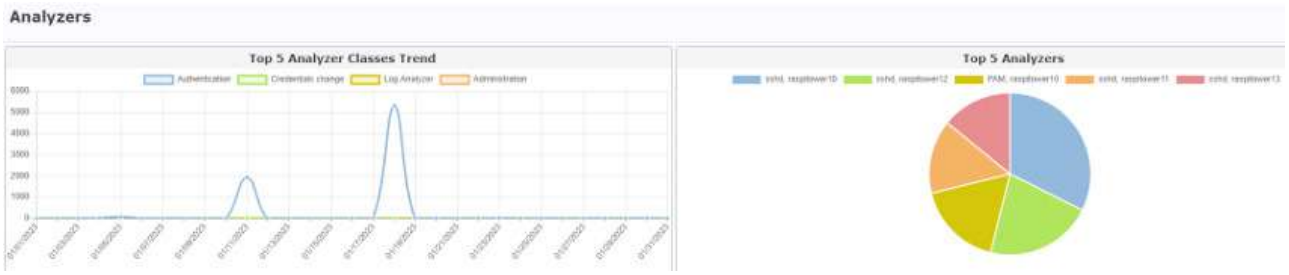

Fig. 3. Graphical representation of information in Prelude OSS.

Fig. 4. Photograph of the evaluation environment. The right monitor displays the SIEM interface, while the left monitor displays supporting materials and scenario information.

ments, including evaluation guidelines and scenario descriptions, were displayed on the left. Evaluators were free to search for additional information during the evaluation.

Printed documentation containing the evaluation task description and attack scenarios was provided, and handwritten notes were encouraged using paper notebooks and pens when needed (Figure 4).

C. Implementation of the Heuristic Walkthrough

Before conducting the evaluation, we prepared a guideline document to ensure consistent interpretation of the evaluation procedure among all evaluators.

During the cognitive walkthrough phase, evaluators completed the following task sequence:

1) Access the SIEM.
2) Examine the management interface and identify alerts that appear to require attention, making escalation decisions as necessary.
3) For each decision, record the following information:
   - Estimated duration of the attack
   - Number of attack instances
   - Whether the attack succeeded or failed and the rationale for that judgment
   - Locations and reasons for any escalation decisions

No predefined escalation guidelines or operational manuals were provided, in order to observe how evaluators interact with the SIEM using only the information available through the interface.

To support the walkthrough, evaluators referred to four guiding questions commonly used in cognitive walkthrough methodologies:

1) Does the user understand what needs to be done?

2) Will the user recognize how to proceed by exploring the interface?
3) Can the user associate the goal with the correct operation?
4) Based on system feedback, can the user determine whether actions are proceeding correctly?

Throughout the walkthrough, screenshots were taken to document interaction steps and were compiled as task logs.

D. Results of the Heuristic Evaluation

After completing the cognitive walkthrough, the evaluators assigned five-point ratings to all eleven usability criteria. These ratings served as the basis for assessing inter-rater reliability. To ensure that the criteria were interpreted consistently across evaluators, we conducted iterative rounds of discussion followed by re-rating. This process continued until Krippendorff's alpha coefficient (α) exceeded the commonly used threshold of 0.667 for acceptable agreement.

With a final α of 0.729, a sufficient level of inter-rater agreement was achieved in the rating-based assessment. Table IV summarizes the final ratings for all eleven criteria.

In the next section, we analyze the qualitative findings derived from the evaluators' free-form observations and group discussions. These qualitative insights represent the core outcomes of the expert review and complement the rating-based results shown above.

VIII. Discussion

A. Overview of Findings

The usability evaluation revealed a consistent set of strengths and limitations in the evaluated SIEM. While the system provided sufficiently rich information to support explanatory reasoning, persistent challenges were observed in alert volume management, handling of benign triggers, dependence on environmental knowledge, and interaction efficiency. These findings align with previously reported operational challenges in SOC environments, suggesting that the issues identified are not unique to a single tool but reflect broader structural characteristics of SOC platforms.

Although evaluators were able to complete investigative tasks, notable variability was observed in workflow efficiency and escalation behavior. This variability indicates that usability characteristics of SOC tools directly influence analysts' cognitive load and decision consistency.

B. Alert Volume, Context, and Analyst Burden

Alert volume emerged as a central usability challenge. While evaluators ultimately judged the volume as realistic, this assessment was strongly influenced by their prior knowledge of SOC operations. Previous studies report that SOC analysts routinely process thousands of alerts per day, many of which are benign triggers. From this perspective, the alert volume generated in our scenarios appeared reasonable.

However, evaluators noted that less experienced analysts may perceive the same volume as excessive and struggle to prioritize alerts effectively. This suggests that alert volume cannot be evaluated independently of analyst experience and contextual support. Without mechanisms for aggregation, differentiation, and contextualization, high alert volume amplifies cognitive burden, particularly for Tier 1 analysts.

The evaluation further showed that escalation decisions rely heavily on knowledge of the monitored environment, such as asset roles and service criticality. Because such contextual information was not integrated into the SIEM interface, analysts had to rely on external knowledge, increasing uncertainty and the likelihood of inconsistent escalation decisions.

C. Implications for the Design of SOC Tools

Across the themes, we identify several implications for future SOC tool design.

First, SIEM systems should incorporate alert aggregation and differentiation features that allow analysts to efficiently understand repetitive or closely related events. Without this capability, alert triage becomes unnecessarily time consuming.

Second, SOC tools would benefit from integrated contextualisation features that connect alert metadata with information about the monitored environment. Such integration could reduce reliance on external knowledge and improve consistency across analysts.

Third, help and reference documentation must extend beyond interface descriptions and include explanations of internal terminology and data structures to support accurate interpretation of alerts.

Fourth, interaction design should minimise duplicated operations and preserve interface states to support rapid investigative workflows.

Together, these implications highlight the importance of designing SOC tools that more effectively support cognitive processes, reduce uncertainty, and mitigate operational burden.

D. Implications for SOC Tool Usability Evaluation

Finally, the findings validate the proposed heuristic walkthrough method and the eleven criteria developed for this study. The evaluation framework successfully revealed usability challenges that are consistent with those reported in prior SOC research, indicating its ability to surface both tool specific issues and broader patterns relevant to SOC operations. The structured use of criteria, combined with inter rater agreement analysis, provides a reproducible methodology that can be applied to other SOC tools in future work. Furthermore, the simulated environment used in this study enabled controlled evaluation while preserving realism, demonstrating its utility as a platform for systematic usability assessment.

These observations collectively address RQ1 by showing that SOC usability can be evaluated in a realistic yet fully controlled environment. They address RQ2 by demonstrating that the identified usability issues align with widely reported operational challenges, indicating that the method reveals tool-independent patterns of breakdown.

TABLE IV
FINALIZED FIVE-POINT RATINGS FOR THE ELEVEN USABILITY EVALUATION CRITERIA AFTER INTER-RATER AGREEMENT WAS ACHIEVED.

	Evaluation Item	**Author 1**	**Author 2**	**Author 3**
1	The number of displayed alerts is small.	2	2	2
2	The amount of displayed alerts is appropriate.	3	3	4
3	The quality of displayed alerts is good.	3	3	4
4	Appropriate measures are taken for benign triggers.	1	1	2
5	No prior knowledge of the monitored environment is required.	1	1	2
6	The displayed information is sufficiently comprehensive.	3	4	3
7	There are no factors that cause redundant or duplicate operations.	3	2	2
8	Help and reference documentation are available and sufficiently detailed.	1	1	2
9	When an alert indicating a potential threat is displayed, the entire process from detection to escalation can be completed using only the information shown on the screen.	2	2	3
10	It is possible to explain, with valid reasoning, why an alert should be addressed or can be safely ignored.	4	4	4
11	From an interface design perspective, it is possible to distinguish between alerts that require attention and those that can be ignored.	3	3	2

IX. LIMITATIONS

The proposed evaluation method and simulated environment offer a practical, reproducible basis for assessing SOC tool usability, but several limitations remain.

First, the empirical evaluation was conducted using only one SIEM system, Prelude OSS. Although the open architecture of Prelude OSS allows detailed examination of how alerts are processed and displayed, commercial SIEM products often include proprietary correlation mechanisms, organization-specific tuning, and workflow customizations. These differences may influence usability characteristics, so the findings presented in this paper should be interpreted as indicative rather than fully representative of the broader SIEM landscape.

Second, the simulated environment was developed using two attack scenarios that were designed to capture structural patterns frequently observed in incident reports. These scenarios were chosen because they reflect common sequences of compromise, such as initial access, reconnaissance, lateral movement, and data exfiltration. In real operational environments, incidents vary widely in scale, complexity, and duration. They may involve multi-stage intrusions, cloud-based compromise paths, or long-term adversarial presence. Increasing the diversity of scenarios would allow future evaluations to explore a broader range of usability challenges.

Third, the evaluation network represents a typical small-to-medium enterprise configuration. While this design aligns with many operational SOC environments, it does not reflect cloud-centric architectures or distributed organizations where multiple sites rely on remote connectivity. In such environments, additional log sources, distributed access points, and cloud service events would need to be incorporated. Extending the network model to cover these cases remains an important direction for future work.

Fourth, part of the benign log set was generated using a large language model. The generated logs were reviewed by experts and validated for consistency with realistic operational environments. Even so, LLM-generated logs may lack subtle statistical properties or vendor-specific artifacts that appear in authentic logs. Developing dedicated techniques for generating high-fidelity benign data for usability evaluation is a potential area for further research.

Fifth, the usability evaluation involved three evaluators with differing levels of SOC-related knowledge. This variation helped capture multiple perspectives, but the results may not fully reflect the practices of operational SOC teams, particularly Tier 1 analysts who perform alert triage in real environments. Broader participation from practitioners would strengthen the external validity of the evaluation.

Finally, the simulated environment delivers logs as static entries rather than dynamically generated traffic. As a result, interactions such as timing relationships, adaptive adversary behavior, or system state changes are not reproduced. Incorporating a more dynamic event generation mechanism could provide a closer approximation of real SOC operations and enable a more comprehensive analysis of tool usability.

X. CONCLUSION

This paper presented a structured methodology for evaluating the usability of SOC tools, combining an expert-based heuristic walkthrough with SOC-specific usability criteria derived from prior empirical studies. To support reproducible evaluations, we also introduced a simulated operational environment that generates realistic alert streams based on representative attack scenarios.

Applying the method to Prelude OSS demonstrated its ability to systematically reveal both strengths and recurring usability limitations common to many SOC tools, highlighting the need for usability improvements alongside ongoing efforts in detection accuracy and alert reduction.

Future work will extend the evaluation framework to a broader range of attack scenarios, network configurations, and SOC tools, as well as incorporate participants with real-world SOC experience to further strengthen its applicability and rigor.

REFERENCES

[1] B. A. Alahmadi, L. Axon, and I. Martinovic, "99% false positives: A qualitative study of SOC analysts' perspectives on security alarms," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA:

USENIX Association, Aug. 2022, pp. 2783–2800. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/alahmadi

[2] 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, Nov. 2024. [Online]. Available: https://doi.org/10.1145/3695462

[3] S. Tariq, M. Baruwal Chhetri, S. Nepal, and C. Paris, "Alert fatigue in security operations centres: Research challenges and opportunities," *ACM Comput. Surv.*, vol. 57, no. 9, Apr. 2025. [Online]. Available: https://doi.org/10.1145/3723158

[4] M. E. Aminanto, T. Ban, R. Isawa, T. Takahashi, and D. Inoue, "Threat alert prioritization using isolation forest and stacked auto encoder with day-forward-chaining analysis," *IEEE Access*, vol. 8, pp. 217 977–217 986, 2020.

[5] T. Ban, N. Samuel, T. Takahashi, and D. Inoue, "Combat security alert fatigue with ai-assisted techniques," in *Proceedings of the 14th Cyber Security Experimentation and Test Workshop*, ser. CSET '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 9–16. [Online]. Available: https://doi.org/10.1145/3474718.3474723

[6] C. Zhong, J. Yen, P. Liu, and R. F. Erbacher, "Learning from experts' experience: Toward automated cyber security data triage," *IEEE Systems Journal*, vol. 13, no. 1, pp. 603–614, 2019.

[7] M. Baruwal Chhetri, S. Tariq, R. Singh, F. Jalalvand, C. Paris, and S. Nepal, "Towards human-ai teaming to mitigate alert fatigue in security operations centres," *ACM Trans. Internet Technol.*, vol. 24, no. 3, Jul. 2024. [Online]. Available: https://doi.org/10.1145/3670009

[8] X. Wang, X. Yang, X. Liang, X. Zhang, W. Zhang, and X. Gong, "Combating alert fatigue with alertpro: Context-aware alert prioritization using reinforcement learning for multi-step attack detection," *Computers & Security*, vol. 137, p. 103583, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404823004935

[9] M. Vermeer, N. Kadenko, M. van Eeten, C. Gañán, and S. Parkin, "Alert alchemy: Soc workflows and decisions in the management of nids rules," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 2770–2784. [Online]. Available: https://doi.org/10.1145/3576915.3616581

[10] R. Stevens, D. Votipka, J. Dykstra, F. Tomlinson, E. Quartararo, C. Ahern, and M. L. Mazurek, "How ready is your ready? assessing the usability of incident response playbook frameworks," in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, ser. CHI '22. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: https://doi.org/10.1145/3491102.3517559

[11] K. Thimmaraju, S. I. Rispens, and G.-J. Ahn, "Human performance in security operations: A survey on burnout, well-being and flow state among practitioners," in *Proc. 2025 Workshop on Security Operations Center Operations and Construction (WOSOC 2025)*, 2025, pp. 2–4.

[12] J. Mink, H. Benkraouda, L. Yang, A. Ciptadi, A. Ahmadzadeh, D. Votipka, and G. Wang, "Everybody's got ml, tell me what else you have: Practitioners' perception of ml-based security tools and explanations," in *2023 IEEE Symposium on Security and Privacy (SP)*, 2023, pp. 2068–2085.

[13] J. Smith, L. N. Q. Do, and E. Murphy-Hill, "Why can't johnny fix vulnerabilities: A usability evaluation of static analysis tools for security," in *Sixteenth Symposium on Usable Privacy and Security (SOUPS 2020)*. USENIX Association, Aug. 2020, pp. 221–238. [Online]. Available: https://www.usenix.org/conference/soups2020/presentation/smith

[14] M. U. Aksu, E. Altuncu, and K. Bicakci, "A first look at the usability of openvas vulnerability scanner," in *Workshop on usable security (USEC)*, 2019.

[15] A. Kanaoka, "kanaoka-laboratory/soctoolusabilityevaluation: v1.0.0 - release," Dec. 2025. [Online]. Available: https://doi.org/10.5281/zenodo.17839819

[16] Overview - prelude siem. [Online]. Available: https://prelude-ids.org/prelude-2/