

MSc Cyber Security and Digital Forensics

Module: Research Methods and Professional Development

Module Code: ACFA7004

Assessment Component 1: Research Proposal

Submission Deadline: 13th August 2025, 14:00 (2PM)

Lecturer: Dr Suresh Shanmugasundaram

Title of Proposal: Development of an AI-Assisted SOC Alert Prioritization System Using Large Language Models

Student Name: Mahedi Hasan Rabbi

Student Number: 2517228



Prifysgol Cymru

Y Drindod Dewi Sant

University of Wales

Trinity Saint David

Table of Contents

Abstract.....	3
1. Introduction	3
1.1 Context and Background	3
1.2 Research Problem	3
1.3 Research Aim and Objectives.....	4
1.4 Significance of the Research	4
2. Literature Review	5
2.1 Introduction to the Literature Review.....	5
2.2 Alert Fatigue and the Prioritization Problem	5
2.3 Machine Learning Approaches to Alert Classification.....	5
2.4 Large Language Models in Cybersecurity	6
2.5 Explainability and Human-AI Teaming in Security Operations.....	6
2.6 Key Gaps and Positioning of This Study	6
3. Research Design and Methodology.....	7
3.1 Research Design	7
3.2 Research Methods and Materials.....	7
3.2.1 System Architecture and Development.....	7
3.2.2 Quantitative Evaluation — Experimental Method	8
3.2.3 Qualitative Evaluation — Analyst Usability Study	8
3.2.4 Data Sources and Collection	8
3.2.5 Ethical Considerations.....	9
3.2.6 Software and Hardware	9
4. Expected Outcome.....	9
5. Timeline	10
6. References.....	11
7. Appendices	12

Abstract

Security Operations Centres (SOCs) are currently dealing with an extremely critical situation where the sheer volume of alerts generated exceeds what analysts can manually process, leading to failed detections, burnout and poor security posture. In order to address this challenge, this project aims to create an AI-enabled SOC alert prioritization tool using Large Language Models (LLMs), specifically fine-tuned versions of GPT-4 and open source LLMs like Mistral-7B to provide intelligent triaging and ranking of security incidents. The tool will be designed to work seamlessly with industry standard SIEM products, consume both structured and unstructured security incident data and output prioritized and contextually rich incident recommendations. The DSR methodology will be used to benchmark the tool in terms of triage performance against existing SIEM based solutions on metrics like precision, recall, Mean Time to Triage (MTTT) and reduction in analyst cognitive load.

1. Introduction

1.1 Context and Background

There has been a paradigm shift within the global cybersecurity ecosystem over the last decade. This is because the threat actors have significantly escalated the intensity, frequency, and sophistication levels of their attacks on the organisational infrastructure.^[1] As a result, Security Operation Centers (SOC) have emerged as the main component of enterprise cyber defense operations, providing constant monitoring and incident detection and response operations in highly diverse IT infrastructures.

The SOC operation relies heavily on SIEM software such as Splunk, Microsoft Sentinel, and IBM QRadar, that collects and correlates the data in order to generate security alerts. This methodology has led to what practitioners call 'alert fatigue' - a condition under which analysts receive too many notifications that make it impossible for them to properly evaluate each one of them.^[2] According to Gartner, a standard enterprise SIEM generates up to 10,000 to 150,000 alerts per day, of which only less than 5% are considered to be actual threats and require any analyst's attention.^[3]

Derner and Batistič (2023) examined the security risks of ChatGPT, including malicious content generation, information disclosure and the potential bypass of safety safeguards.^[4] LLMs like GPT-4 by OpenAI, LLaMA-3 from Meta and Mistral-7B from Mistral AI have shown impressive proficiency in semantic reasoning, context generation and natural language explanation, all of which directly cater to alert triage.

These advancements, coupled with cybersecurity-specific LLM fine-tuning (SecurityLLM, CyberSecEval benchmarks), the application of reasoning layers of LLM in products like Microsoft Security Copilot and the recently published AI Risk Management guidelines by NIST suggest that LLM-supported SOC operations is an upcoming operational paradigm.^[5]

1.2 Research Problem

Despite the promising capabilities of LLMs in a security environment, their implementation for SOC alerts prioritization is in its infancy and underexplored.

The current methods for prioritizing SIEM alerts employ static rule-based systems and scoring systems (e.g., Common Vulnerability Scoring System or CVSS scoring), which do not incorporate context at all – they cannot consider asset importance, attack chain phase,

threat actor Tactics, Techniques, and Procedures (TTPs) or changes in the threat landscape due to constant manual adjustment.[\[6\]](#)

There are several shortcomings that describe the current state of practice. Firstly, the current approaches to alert prioritization do not understand the semantics of the information they process structured information and pattern-matching without any interpretation of the context of the combination of alerts.[\[7\]](#) Secondly, despite the fact that machine learning techniques are used for anomaly detection in order to decrease false positives, such approaches require large amounts of labeled data, are not resistant to new attack vectors (zero-day attacks) and have uninterpretable prioritization output.[\[8\]](#)

Thirdly, and most importantly, there is no systematic evaluation and application of LLMs for alert triage in SOC workflows. The current literature is mostly limited to concept demonstration.[\[9\]](#) This gap, between the theoretical ability of LLMs and their practical applications in cybersecurity, is what this research seeks to address.

1.3 Research Aim and Objectives

Aim: Designing, developing and evaluating an AI-assisted system for SOC alerts prioritization based on fine-tuned Large Language Models to enhance the accuracy, speed and explainability of security alerts prioritization process as compared to the traditional SIEM-based systems.

Objectives:

O1: Conducting a systematic literature review of existing approaches to SOC alert prioritization and current use cases of LLMs in cybersecurity finished by September 2026.

O2: Designing a modular alert prioritization system with LLMs integration compatible with at least one of the major SIEM systems (Microsoft Sentinel) and defining its components such as data ingestion, prompt engineering and output formatting finished by October 2026.

O3: Developing and fine-tuning the LLMs pipeline using the labeled dataset of cybersecurity alerts and developing a system that will be able to generate prioritized alerts with rationale finished by December 2026.

O4: Evaluation of the developed system as compared to a baseline SIEM system using a controlled experiment with measuring such metrics as precision, recall, F1-score, MTTT and explainability of the results perceived by analysts — finished by January 2027.

O5: Critical evaluation of the ethical, operational and deployment aspects of LLMs integration into SOC environment finished by February 2027.

1.4 Significance of the Research

Importance of the research is multi-fold. From an analytical point of view, development of a reliable and accurate LLM-based alert prioritization system will offer SOC teams an advanced decision-making tool which will explain, using natural language, why particular alert needs to be analyzed, decrease cognitive load on analysts, and take into consideration context instead of using static rules.[\[10\]](#)

From a workforce point of view, according to ISC², there is a shortage of 3.4 million cybersecurity professionals globally.[\[11\]](#) Assistance provided by AI can help small teams cover all necessary alerts. From a social perspective, more effective functioning of SOC results in less harm from breaches, hence helping to achieve a greater goal building a

resilient digital infrastructure that is an essential part of UK Cyber Strategy 2022 and EU NIS2 Directive.[\[12\]](#)

2. Literature Review

2.1 Introduction to the Literature Review

This literature review is aimed at investigating two intersecting areas SOC alert management and usage of LLMs in the cybersecurity field. The literature review incorporates peer-reviewed articles published between 2019 and 2026 found in IEEE Xplore, ACM Digital Library, and Springer and some industry reports by Gartner, MITRE, and NIST. Scope of the literature review includes (i) alert fatigue and priorities challenges; (ii) machine learning approaches to security alert classification; (iii) LLMs in cybersecurity advantages and disadvantages; and (iv) explainability and human-machine cooperation in SOC.

2.2 Alert Fatigue and the Prioritization Problem

Tariq et al. (2025) reviewed the problem of alert fatigue in Security Operations Centres and identified alert volume and workload as important challenges affecting SOC analysts.[\[2\]](#)

Traditional SIEM priority settings usually involve CVSS scores and vendor priority rules. At the same time, Dissanayake et al. (2022) show that CVSS-based prioritization demonstrates a systematic bias towards vulnerability severity compared to exploitability and organizational specifics, which results in high severity of unexploitable vulnerabilities and leaves actively exploited medium severity threats out of focus.[\[6\]](#) The fact is supported by 2024 Verizon DBIR report, according to which 60% of exploited vulnerabilities have a CVSS score lower than 7.0[\[13\]](#)

In addition, Pendlebury et al. (2019), in their pioneering research on machine learning-based network intrusion detection, highlight the concept drift as a major problem because of the model losing its predictive capacity quickly due to changing network behaviors.[\[7\]](#)

2.3 Machine Learning Approaches to Alert Classification

Many research papers have examined security event classification using ML-based methods. Supervised classifiers, including Random Forest, Gradient Boosted Trees (XGBoost), and deep learning neural networks (LSTM, CNN) were used to classify intrusions from the datasets like NSL-KDD, CICIDS-2017, and UNSW-NB15.[\[8\]](#) They demonstrate high accuracy on benchmark datasets, however, they suffer from several problems in practical application – distribution shift of the training data, class imbalance, and the black-box nature of the prediction, which is a problem of interpretability and limits analysts' trust.

One of the limitations common for all machine learning methods is the use of structured and numerical features. It limits the ability of models to utilize the contextual information from unstructured data like alert descriptions, notes from analysts, threat intelligence feeds, and external databases with vulnerabilities.[\[14\]](#) This is exactly what LLMs can help to resolve.

2.4 Large Language Models in Cybersecurity

LLM application in cybersecurity is a young but fast-evolving research area. The transformer architecture (Vaswani et al., 2017) allowed developing the pre-trained models capable of encoding the semantic meaning of the text and doing zero-shot and few-shot task transfer without using labeled domain data.[\[15\]](#)

Derner and Batistič (2023) performed a survey of GPT-4 capability to analyze the vulnerability description and showed that it was able to map CVE descriptions to MITRE ATT&CK techniques with 71% accuracy.[\[4\]](#) Microsoft's Security Copilot, based on GPT-4 and cybersecurity-specific fine-tuning, demonstrated time reduction in triaging complex events but emphasized the hallucination danger as one of the critical safety issues.[\[5\]](#)

SecureBERT (Aghaei et al., 2022) demonstrated the substantial improvement in NER, entity classification, and threat reports summarization after domain-specific fine-tuning.[\[16\]](#) Mistral-7B and LLaMA-3 become computational efficient alternatives for on-premise deployment.

2.5 Explainability and Human-AI Teaming in Security Operations

Explainability is key to analyst trust in recommendations made by AI systems.[\[10\]](#) Rastogi et al. (2025) found that SOC analysts value relevant, evidence-backed and context-rich explanations, highlighting the importance of explainability for trust and practical use of AI in security operations.[\[17\]](#)

The XAI literature makes a crucial distinction between post-hoc explanation methodologies (such as LIME and SHAP) which attempt to retroactively approximate the decision making process of a model, and fundamentally interpretable models which generate explanations as part of their decision making process. It is important to recognize LLMs for what they are: novelty architecture which generates text as its fundamental output, rather than a post-hoc explanation of a mathematical formula. The difference is critical to understanding why an LLM-native explanation is an improvement over traditional post-hoc approximations. In order to understand an approximation of an explanation (such as the visualisations used in SHAP or LIME), the analyst must first understand the algorithm used to approximate the explanation. The analyst must interpret the output of the explanation algorithm in the context of the problem being solved, which requires domain knowledge. An LLM explanation, by contrast, takes the form of natural language which can be understood without additional mental overhead. An analyst who wishes to understand why an LLM-based explanation recommends a particular course of action can simply read the rationale in natural language (‘...because the lateral movement matches TA0008 and the targeted host is a Tier-1 asset which has not been patched...’) without first having to understand the inner workings of the explanation algorithm. This is critical to analyst adoption, as shown by the findings of Vance et al. (2022). They found that social trust (the perception of trustworthiness of an AI by the analyst) was a more important factor to overall analyst acceptance of AI than objective measures such as accuracy.[\[10\]](#)

2.6 Key Gaps and Positioning of This Study

The literature review revealed three research gaps that this paper aimed to address. First, previous studies did not produce an end-to-end evaluated system that utilizes large language models (LLMs) to consume and prioritize alerts from a security operations center (SOC) pipeline. Second, the performance of prioritization using such models has not been compared against operational security information and event management (SIEM) solutions using standard benchmarks. Third, the design considerations for implementing such solutions using open-source and on-premise LLMs were under-

researched. Notably, the current study addressed each of the research gaps it identified. It demonstrated the feasibility of developing an end-to-end LLM-assisted triage system that includes a prioritization module while comparing the performance of such a model against standard metrics. Another methodological strength of this study was the use of operational data or its analogs in assessing the effectiveness of the alternative triage option. Indeed, most of the literature employs synthetic or academic intrusion data that does not represent the properties of alerts in operational SCSs. The current study used the CICIDS-2018 benchmark intrusion detection dataset alongside synthetic data generated using the Atomic Red Team framework that covers over 100 MITRE ATT&CK techniques. Therefore, the findings were generalizable to an operational SOC, which is critical in understanding the effectiveness of the proposed options in practice. Thus, the findings of this study demonstrated that there is a need for an integrated triage system, which the paper attempted to address.

3. Research Design and Methodology

3.1 Research Design

This study will employ a Design Science Research paradigm (DSR) proposed by Hevner et al. (2004) and used extensively in the field of information systems and cybersecurity engineering research.^[18] The reason for selecting this research methodology is that the study will produce an artifact in the form of an AI-assisted SOC prioritization system. According to Hevner et al., the DSR paradigm is appropriate for this kind of study because it assumes that design science research results should be rigorous, i.e., grounded in existing knowledge, and relevant, i.e., meet the research domain's objectives.

DSR will be implemented using the following four stages:

1. Problem Identification and Motivation, 2. Design and Development, 3. Demonstration, 4. Evaluation and Communication.^[19] The research will apply a positivist approach to quantitative evaluation and an interpretivist approach to qualitative evaluation, which is a form of a mixed-methods research design. [5] The design will also follow the Knowledge Contribution Framework put forward by Gregor and Hevner.

3.2 Research Methods and Materials

3.2.1 System Architecture and Development

The LLM Alert Prioritization System (LAPS) will be developed as a Python-based pipeline comprising four modules:

Module	Function	Key Technology
1 – Alert Ingestion	Consume alerts from Microsoft Sentinel via REST API; normalise to a common JSON schema	Python 3.12, Azure SDK
2 – Context Enrichment	Enrich alerts with CVE data, MITRE ATT&CK TTP mapping, and asset criticality from CMDB	NVD API, ATT&CK Navigator
3 – LLM Inference Engine	Pass enriched alert bundles to LLM with structured prompt; receive ranked prioritization + rationale	GPT-4-turbo / Mistral-7B

4 – Output Interface	Return structured JSON to Sentinel with priority score, confidence, and natural language rationale	FastAPI, Sentinel Connector
----------------------	--	-----------------------------

Table 1: LAPS Module Architecture

Two language models will be developed and compared: (a) GPT-4-turbo (via Azure OpenAI API, cloud-hosted) and (b) Mistral-7B-Instruct fine-tuned on MITRE ATT&CK corpus (on-premise for data sovereignty). The prompt engineering will leverage chain-of-thought reasoning, stipulating the role (e.g., SOC analyst), alert context, asset criticality, recent threat intelligence, and output schema (JSON with priority_rank, confidence_score, triage_rationale fields).

3.2.2 Quantitative Evaluation — Experimental Method

The system will be evaluated using the CICIDS-2018 intrusion detection benchmark dataset and a synthetic SOC alert dataset generated using Atomic Red Team, annotated by certified SOC analyst subject matter experts.[\[20\]](#)

The experiment will have three conditions: (i) SIEM-baseline prioritization (CVSS-rule based), (ii) LAPS-GPT4, and (iii) LAPS-Mistral. As outcome variables, we will report Precision, Recall, and F1-score (for critical alert detection), Mean Time to Triage (MTTT), False Positive Rate (FPR), and Prioritization Concordance Rate (PCR), defined as the percentage of top-10 alerts that concurred with the expert analyst ranking.

3.2.3 Qualitative Evaluation — Analyst Usability Study

An analyst usability study will be carried out with a convenience sample of cybersecurity practitioners (target n=15) to evaluate the utility of findings in practice. Participants will be asked to respond to 20 scenarios presented in SIEM-baseline and LAPS (in randomized counterbalanced order) and rate them in terms of confidence in triage, explainability, and time-on-task.[\[17\]](#)

The items will be developed based on the NASA Task Load Index (NASA-TLX) for cognitive workload estimation and Technology Acceptance Model (TAM) for assessing acceptance and adoption. Item-scale transformation will rely on the use of the Wilcoxon signed-rank test to identify within-group differences.

3.2.4 Data Sources and Collection

Dataset	Source	Justification	Access
CICIDS-2018	Canadian Institute for Cybersecurity	Realistic labelled network intrusion data; widely validated in IDS research	Open access
Atomic Red Team Alerts	Red Canary / MITRE	Realistic SIEM alert simulation covering 100+ ATT&CK techniques	Open source
CVE/NVD	NIST National Vulnerability Database	Vulnerability context enrichment for alert correlation	Public API

MITRE ATT&CK	MITRE Corporation	TTP mapping for alert context; used in prompt grounding	Open access
--------------	-------------------	---	-------------

Table 2: Research Data Sources

3.2.5 Ethical Considerations

Primary ethical concerns come from the proposed analyst usability study. Informed consent will be obtained from all participants using a UWTSD Ethics approved consent form. Participants will be issued anonymous numeric IDs; no personal data will be stored. All data will be encrypted and stored in the UWTSD Office 365 only (ISO 27001 certified).

No live data containing PII or sensitive organisational information will be passed to external API endpoints for processing. All processing will be done either using synthetic or public data sets. The on-premise Mistral-7B model is used for the project to satisfy restricted data sovereignty requirements in a regulated SOC environment. An ethics form will be submitted separately according to the ACFA7004 requirements.

3.2.6 Software and Hardware

Category	Tools / Resources
Programming Language	Python 3.12 (primary); PowerShell (Sentinel integration)
LLM Frameworks	OpenAI Python SDK (GPT-4); HuggingFace Transformers + llama.cpp (Mistral-7B)
SIEM Platform	Microsoft Sentinel (Azure trial subscription)
Data Processing	Pandas, NumPy, Scikit-learn, NLTK
Qualitative Analysis	NVivo 14 (analyst usability study thematic coding)
API Integration	FastAPI (LAPS API gateway); Azure SDK for Python
Development Environment	VS Code, Git/GitHub, Docker (containerised deployment)
Hardware	Primary: UWTSD HPC cluster (GPU: NVIDIA A100 40GB for Mistral fine-tuning); Secondary: Personal workstation (CPU inference)

Table 3: Software and Hardware Requirements

4. Expected Outcome

Based on the literature explored and initial scoping, the following outcomes are expected:

System Performance: LAPS is expected to achieve higher precision/recall (F1-scores) of ≥ 0.85 on key alert types than the equivalent SIEM-baseline (< 0.7), thereby demonstrating improved triage performance, and MTTT is expected to be reduced by 30-50%, which aligns with human-AI teaming literature for similar tasks.[\[5\]](#)

Explainability Advantage: Higher usability ratings for the natural language rationales provided by the LLM than for SIEM-rule counterparts are expected for both explanation clarity and confidence in triage decisions, building on the work of Handa et al. (2023). [17]

Architecture Contribution: A documented open-source architecture published on GitHub for LAPS, building on the components used, for others to implement and improve upon.

Implications: The project would contribute to academic literature on LLM applications for cybersecurity operations, support security operations centers and MSSPs in adopting AI solutions for operational improvements, and advance policy discussions on responsible AI use in cybersecurity, in particular, for NIST and the EU AI Act.

Limitations: Both hallucination and computational costs associated with large language models present significant risks to the project's success, with the former threatening the very foundation of the LAPS concept. The analysis on public data may not generalize well to real-world SOC alerts, limiting external validity. The panel for the usability study (target n=15) may not be representative of a larger population at a statistically significant level. While not undermining the value of such a study, these limitations suggest caution in extrapolating results beyond the described context. Finally, the performance of the GPT-4-turbo model at the scale of real-world SOC traffic (50k-150k alerts per day) is unknown and may prove too expensive and slow to be practical, and thus the alternative Mistral-7B setup is expected to have an impact on either accuracy or speed (or both) that needs to be evaluated in the investigation.

5. Timeline

The following timeline presents major milestones across the 12-month project period (September 2026 – August 2027):

Phase	Activities	Milestone	Target Date
Phase 1 Foundation	SLR completion; framework finalisation; ethics submission; dataset acquisition	SLR Report + Ethics Approval	Oct 2026
Phase 2 Design	LAPS architecture design; SIEM integration specification; prompt engineering; tool setup	Architecture Design Document	Nov 2026
Phase 3 Development	Module 1–4 development; Mistral-7B fine-tuning; GPT-4 pipeline integration; unit testing	Working LAPS Prototype	Jan 2027
Phase 4 Evaluation	Controlled experiment (CICIDS-2018 + Atomic Red Team); analyst usability study; data collection	Evaluation Data Collected	Mar 2027
Phase 5 Analysis	Quantitative statistical analysis; qualitative	Analysis Complete	Apr 2027

	thematic analysis; findings synthesis		
Phase 6 Write-up	Full dissertation write-up; GitHub artifact publication; internal review; submission	Final Dissertation Submitted	Aug 2027

Table 4: Project Timeline and Milestones

The timeline incorporates buffer periods between phases to accommodate iterative refinement of the LLM pipeline and potential delays in analyst participant recruitment for the usability study.

6. References

All in-text citation superscripts link directly to the corresponding entry below.

- [1] IBM Security, "X-Force Threat Intelligence Index 2024," IBM Corporation, 2024.
<https://www.ibm.com/reports/threat-intelligence>
- [2] S. Tariq, M. Baruwat Chhetri, S. Nepal, C. Paris, "Alert Fatigue in Security Operations Centres: Research Challenges and Opportunities," ACM Computing Surveys, vol. 57, no. 9, 2025. <https://dl.acm.org/doi/10.1145/3723158>
- [3] Gartner, "Market Guide for Security Information and Event Management," Gartner Research, 2024. <https://www.gartner.com/en/documents/market-guide-siem>
- [4] E. Derner and K. Batistič, "Beyond the Safeguards: Exploring the Security Risks of ChatGPT," arXiv preprint, 2023. <https://arxiv.org/abs/2305.08005>
- [5] Microsoft, "Microsoft Security Copilot: Early Access Program Evaluation," Microsoft Security Research, 2024. <https://www.microsoft.com/en-us/security/blog/2024/03/13/microsoft-copilot-for-security-is-generally-available>
- [6] N. Dissanayake, A. Jayatilaka, M. Zahedi and M. A. Babar, "Software Security Patch Management: A Systematic Literature Review of Challenges, Approaches, Tools and Practices," Information and Software Technology, vol. 144, 2022.
<https://doi.org/10.1016/j.infsof.2021.106771>
- [7] F. Pendlebury, F. Pierazzi, R. Jordaney, J. Kinder and L. Cavallaro, "TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time," Proc. USENIX Security Symposium, 2019.
<https://www.usenix.org/conference/usenixsecurity19/presentation/pendlebury>
- [8] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," Proc. ICISSP, 2018.
<https://doi.org/10.5220/0006639801080116>
- [9] N. Almodovar, J. Radigan, A. Al-Shaer and M. Gera, "LLM-Based Intelligent Agents in Network Security: A Survey," arXiv preprint, 2024. <https://arxiv.org/abs/2404.06255>
- [10] T. Vance, N. Howe and J. R. Corley, "Social Influence and the Use of AI in Cybersecurity Operations," Computers & Security, vol. 114, 2022.
<https://doi.org/10.1016/j.cose.2021.102587>
- [11] (ISC)², "Cybersecurity Workforce Study 2024," (ISC)² Research, 2024.
<https://www.isc2.org/research/workforce-study>
- [12] HM Government, "National Cyber Strategy 2022," Cabinet Office, 2022.
<https://www.gov.uk/government/publications/national-cyber-strategy-2022>

- [13] Verizon, "2024 Data Breach Investigations Report," Verizon Business, 2024.
<https://www.verizon.com/business/resources/reports/dbir/>
- [14] J. Jiang, X. Chen, K. K. R. Choo and T. Baker, "Towards a Deep Learning-Assisted Threat Intelligence Model in Cloud Environments," IEEE Transactions on Cloud Computing, vol. 11, no. 2, 2023. <https://doi.org/10.1109/TCC.2022.3160184>
- [15] A. Vaswani et al., "Attention Is All You Need," Advances in Neural Information Processing Systems (NeurIPS), 2017. <https://arxiv.org/abs/1706.03762>
- [16] E. Aghaei, X. Niu, W. Shadid and E. Al-Shaer, "SecureBERT: A Domain-Specific Language Model for Cybersecurity," Proc. SecureComm, 2022. https://doi.org/10.1007/978-3-031-25538-0_4
- [17] N. Rastogi, S. Pant, D. Dhanuka, A. Saxena and P. Mairal, "Too Much to Trust? Measuring the Security and Cognitive Impacts of Explainability in AI-Driven SOC's," arXiv preprint / ACM CCS 2025, 2025. <https://arxiv.org/abs/2503.02065>
- [18] A. R. Hevner, S. T. March, J. Park and S. Ram, "Design Science in Information Systems Research," MIS Quarterly, vol. 28, no. 1, pp. 75–105, 2004.
<https://doi.org/10.2307/25148625>
- [19] S. Gregor and A. R. Hevner, "Positioning and Presenting Design Science Research for Maximum Impact," MIS Quarterly, vol. 37, no. 2, pp. 337–355, 2013.
<https://doi.org/10.25300/MISQ/2013/37.2.01>
- [20] Red Canary, "Atomic Red Team: Open Source Adversary Emulation Library," Red Canary, 2024. <https://github.com/redcanaryco/atomic-red-team>

7. Appendices

Appendix A: Ethics Form (submitted separately via Moodle submission link per ACFA7004 requirements).

Appendix B: LAPS Prompt Engineering Template (to be completed during Phase 2 – Design).