OST: An AI enabled One-Stop Station Platform for Cyber Security Incident Reporting

Alwaleed Al Rashdi, Shancang Li

[a] *Cardiff University, Cardiff, CF24 4AG, UK*

Abstract

This paper proposes a "One Stop Station" (OST) with an AI-powered "CyberBot" to streamline incident reporting and management. The OST tackles inefficiencies by offering a unified platform that streamlines threat reporting, delivers real-time threat intelligence, and enhances user interaction with an intuitive interface. Leveraging AI like GPT-3.5 and Rasa, the OST automates responses, generates detailed reports, and integrates with existing tools like VirusTotal. This demonstrably improves response speed and accuracy. Testing results show a potential 600% reduction in reporting time. The OST empowers both cyber security professionals and less technical users, reducing workload and enhancing overall incident management. This project highlights the potential of AI in cyber security and positions the OST as a pioneer. It reinforces discussions on leveraging AI to fortify digital defences and integrate AI into daily life.

Keywords: Cyber security, Incident Reporting, Generative AI

1. Introduction

1.1. Background

Cybercrime has become a pervasive global threat, affecting over one in five people worldwide [1]. As cyber-attacks grow increasingly sophisticated, traditional security measures are proving inadequate. The OST was developed to address this critical challenge by offering a proactive and intelligent solution. Recognizing the urgent need to safeguard sensitive data, the OST's integrated approach streamlines incident reporting and accelerates threat hunting, enabling organizations to minimize the impact of cyber-attacks.

The proposed OST acts as a comprehensive system that provides users with a single point for all cyber security-related matters, in addition to having access to the tools required to investigate cyber security incidents, the platform is equipped with a password cracking software and a word list of 14,341,564 unique elements, which should be capable of tackling even the hardest passwords. Tying all this together is the CyberBot, which is a virtual assistant that can access all the tools using a single text field, and is integrated to a database to store all the case information.

The main purpose of the OST is to allow the first responders to quickly create an incident report, and log information for future reference. The development of the OST is particularly relevant in sectors possessing dedicated Security Operations Centers (SOCs). By streamlining incident reporting, the OST can enhance operational efficiency and enable SOC teams to focus on higher-level threat analysis and response. This research aims to pioneer a new era of AI specialization, where chatbots can be tailored to provide direct benefits to organizations.

The main contributions of this work can be summarized as:

- A centralized incident management platform (OST) has been developed that automates incident report generation. Leveraging advanced threat intelligence, the OST provides tailored cyber recommendations based on user-supplied incident details.
- Standardization of incident reports is achieved through a unified template, ensuring consistency across all records.
- By integrating AI capabilities, the OST offers a comprehensive suite of tools for enhanced incident management. The OST chatbot can access real-time data and conduct intelligent analysis to support incident response efforts.
- Adherence to National Cyber Security Centre (NCSC) guidelines is a core feature of the OST, with additional provisions for risk mitigation. Sensitive information is securely stored in a local SQL database fortified with robust security measures.

2. Related Works

Unlike most existing cyber security platforms that predominantly focus on specific facets of the security life-cycle, this work aims to develop an AI-

empowered comprehensive one-stop solution, which integrates multiple tools excelling in threat detection, incident response, or vulnerability management but often lack a holistic approach.

Security Operations Centres (SOCs) have emerged as centralized hubs [2], they often rely on multiple disparate systems for effective operation. Some platforms have begun integrating multiple security functions, such as SIEM (Security Information and Event Management) and SOAR (Security Orchestration, Automation, and Response) solutions [3, 4]. These platforms offer centralized logging, analysis, and response capabilities.

Since the project deals with GPT3.5, custom instructions enable training the model with data and setting limitations on specific functions that can make the model more secure, for example ChatGPT is usually accused of spurring responses. created by its "hallucinations", meaning that it is just generated and not true. Fine-tuning the model and hardcoding prompts such as tell the model to "Answer the question as truthfully as possible, and if you're unsure of the answer, say "Sorry, I don't know"." [5] eliminates the risk of hallucinations by 100% , which avoids the detrimental risk that can be caused in cyber security, especially if it mislabels a malicious link to be safe.

In the past few years, several OST platforms have been developed to provide a centralized hub for managing and coordinating incident response activities, but most of these works focus on tracking and managing incidents (e.g., Rapid 7 InsightDR). Palo Alto Networks developed Cortex XSOAR to help organizations automate and streamline incident response processes. IBM developed security QRadar to focus on security information and event management (SIEM) for incident response and threat detection, while Splunk also specializes in incident response and threat detection.

In recent, deep learning, specific GPT models are increasingly used to help users to generate incident management insights. Fine-tuning the ChatGPT model was used in biomedicine and health research [6], showing the increasing accuracy of these models after they have been fed information in a specific domain. This is the main reason that a custom GPT-3.5 model was used in this paper to summarize the incident report and generate recommendations.

Some companies have attempted to reduce human intervention in Security Operations Centres (SOC) usually by implementing a chatbot , but significant technical gaps remain, particularly in the integration with existing systems and scalability of the system.

For example **Onyoursix** claims to have a fully customizable bot that

is active "24/7, 365", but the lack of access, detailed technical documentation raises questions about the product's maturity and readiness for official launch, especially at an enterprise level.

Similarly, **Cisco's Securex platform**, integrated Extended Detection and Response (XDR) with chatbot functionality to enhance SOC operations.However, despite Cisco's reputation and extensive resources (€ 182.61 Billion market cap) [7] Cisco has announced the platforms decommission as of July 2024.

The examples above underscore the broader technical gaps within current chatbot solutions in cyber security. While implementing AI into cyber security seems very promising, the issues with integration and scalability cause current solutions to fail.

The extent to which a cyber threat can be neutralized depends mainly on the incident response team and how well they capture the full scope of the attack [8]. During the research, it became clear that there is no full cyber security incident report platform that uses a chatbot. However, by combining the knowledge of past researchers, this papers focuses on creating a revolutionary step in the cyber industry and emphasizes the seamless integration with diverse cyber security ecosystems.

3. Methodology

3.1. Problem

SOC teams face significant challenges due to understaffing, with over 70 percent of cyber security professionals reporting that their organizations are understaffed [9], meaning that some cases are either not properly documented or ignored using the discretion of the first responders, harming the companies infrastructure and putting data at risk, the OST supplements human work with automation thus reducing the effects of understaffing, ensuring the report is as detailed as possible, and includes a well written summary.

Additionally, a discussion on a Reddit forum about the use of AI in cyber security highlighted that threat intelligence report standardization remains underdeveloped [10], especially as countries like Saudi Arabia are investing $ 40 billion into artificial intelligence [11]. This government support highlights the necessity to implement solutions that leverage computational power and AI. This aligns with the goal of the OST, which is why AI was chosen as the main focus of this project

3.2. Solution

Developing a cyber platform that -theoretically- simplifies hours of investigation into a few minutes requires multiple steps. The foundation of the chatbot application is built on a custom flask server that integrates with existing systems through APIs, such as SIEM (Security Information and Event Management) systems. The system boasts a modular design that allows for simple implementation of systems like VirusTotal for threat scanning, SQL databases for incident logging, email servers for automated reporting, and includes proper API documentation and webhook configurations to enable real-time data transmission. Currently, the data is being manually inputted from the SIEM system into the OST. The key functionalities provided by the OST include:

- **AI empowered Incident Management**. Having all the tools required in one place and having a chatbot that can access real-time data and perform intelligence gathering.
- **Advanced Threat Analysis**. Integration with VirusTotal and other tools for scanning and analysing malicious files and URLs.
- **Automatic creation of incident reports**. The OST will also generate cyber recommendations for the incident, using the threat intelligence it has been fed and the case information inputted by the user.
- **Standardization of incident reports**. Each report will be the exact same since a single template will be used, this will ensure that there will be consistency between the reports.
- **Compliance and Risk Management**. The incident report will be compliant with the National Cyber Security Centre guidelines, and will also have an extra section relating to risk mitigation.
- **Centralized storage location**. A local SQL database will be used in the development on the OST, the database will be developed to a high standard of safety to secure the confidential information being transmitted.
- **Scalability and Customization**. The project will use Rasa for the chatbot making the information gathering scalable by simply adding more intents, and the website is built using a Flask template meaning

that new tools can be added by creating a div container and inserting the code into it.

The OST leverages several AI models to enhance its capabilities:

1) **GPT-3.5 (Generative AI)**: GPT3.5 is a Generative AI model developed by OpenAI, that excels in processing and generating human-like text. In the OST platform, ChatGPT is used for generating SQL queries and detailed summaries of incidents, while also answering questions and providing recommendations. However, due to the studies highlighting the risks associated with the availability of AI services like ChatGPT [12], the development of the OST will limit the use of third-party services as much as possible, and it will also offer a redundancy mechanism to maintain the chatbot's operational period.

Another major fear for companies of their staff using services like ChatGPT with business data was not only caused by the security issue of sharing confidential information with a third-party company, but also due to an understanding that unauthorized access to users account would also cause the data to be jeopardized, which influenced the OST to use a secure API call and encrypting the data sent to and from the server. All data passed to ChatGPT is encrypted on the backend of the flask server using the post method (methods=['POST']) and access by need-to-know basis only.

2) **RasaNLU Model**: Rasa was selected to be the interface for the chatbot in conjunction with GPT3.5 to supplement the shortcomings of using ChatGPT highlighted above. Additionally, using Rasa will allow for more control over the training data which can eliminate the "hallucination" effect that is common in ChatGPT, and flexibility in fine-tuning with APIs rather than datasets, Rasa can also be run locally which can be seen as a method of reducing security threats by keeping all data offline until it is necessary. Above all Rasa stands out due to its ability to integrate seamlessly with spaCy for the entity recognition, which demonstrated its ability to extract information from the users input with a high level of accuracy and speed in parsing. Rasa was selected for the core of the NLU because it is specifically designed for creating chatbots, offering advanced features for conversation management, it also has great integration abilities, making it a valuable tool for the chat interface, it is used to develop the core functionality of the chatbot, enabling it to understand and process natural language, manage dialogues, and preform custom actions based on user input.

3) **Hugging Face**, Hugging Face was proposed to be the main source for machine learning algorithms along with datasets, but due to the recent discovery of a vulnerability in the platform [13], this work will use the GPT-3.5 model and pass custom prompts to achieve the report generation functionality.

3.3. Key techniques

The OST integrates all the tools mentioned below in symphony to form a comprehensive, centralized platform. Having a large number of tools working independently yet being able to communicate with each other means that the data transfer model must be meticulously designed, this ensures efficient and secure data flow and processing.

After the user logs in and begins a chat with the CyberBot, the Flask server will display a text field in which the user will type their prompt. The prompt is then sent to the Rasa server via a secure AJAX request. The Rasa chatbot will then process the input, and run a function depending on the user's intent. For instance, if the user wanted to create an incident report, the Rasa server will send a response to the user through the AJAX server and enabling the user to reply to the answer which will trigger the Rasa server to start the information gathering process, subsequently storing this data in an SQL database.

Following the information collection and storage, data is securely read by Python code that filters the data; the information undergoes data cleansing to remove irrelevant information, and prepares it for analysis, the data fields like IP address and domain name are passed through a Python function that takes the information as an input and returns the sanity of the input using an API call, the response is then saved into the SQL database for the next step which is report generation.

After the database is fully populated with as much case information as possible and the sanity checks for all data fields have been completed, another Python function then packages the details for the case and concatenates it with an engineered prompt to send to the ChatGPT model via an API call. The code then waits for a response from the model stating that the report generation has been completed; after receiving the final report in text format, the data is then parsed by another Python function which formats the data and saves it in a PDF format on the user's machine using an OS command. This data transfer model is designed with scalability in mind. Using SQL allows larger volumes of data to be transferred without compromising on

processing efficiency. A simplified model of the data flow between the various services of OST can be seen in Figure 1.

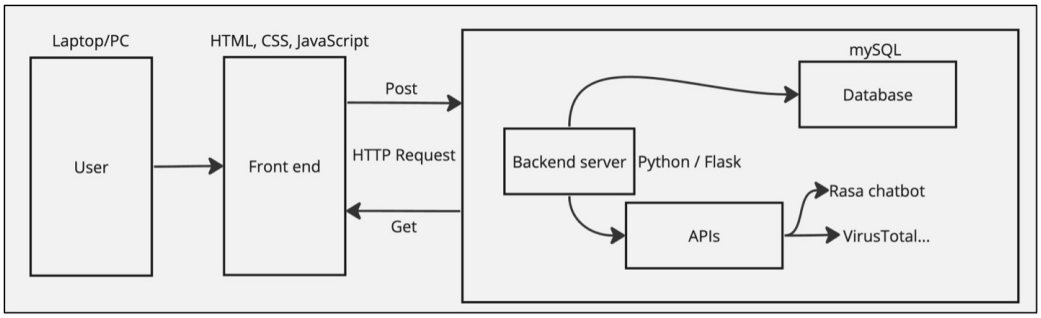

Figure 1: OST workflow

The OST ensures no information is missed by storing all data that was mentioned in the existing frameworks, including National Institute of Standards and Technology, NASA, and SANS Framework, etc., covering the basic information, affected clients, assessed secrecy, report to the stakeholders, description of the event, time, incident category, extent of the consequences, and the mitigation strategy.

The above frameworks were chosen due to the high success rate and standardisation in many countries [14], the report also covers all stages from containment, eradication, to recovery, and ensures that the issue is reflected on, to avoid the likelihood of it happening again. Using pre-existing frameworks to generate the report maintains the uniform information that creates a standardised report.

The final step after gathering the information is to analyse the data, the level of advancement the cyber security industry has achieved is unmatchable in a short time frame, for that reason the OST incorporates existing systems such as VirusTotal and Hashcat, these system's ability to enhance the chatbot's threat detection was a key consideration driven by the literature reviewed, due to their high success rate and wide ability to customize them. One of the biggest issues encountered was the availability of thousands of intelligence gathering tools, but the OST selected the best performing options and implement them so the results can be trusted, forming the research question:

"Can a chatbot help cyber security teams create incident reports faster and with more accuracy?"

3.4. Integrating cyber security tools and types of cyber incidents

To ensure effective functionality for the OST, research was conducted to find the top tools required by members of the SOC team, which will then be incorporated into the platform. Table. 1 summarises the tools and type of cyber incidents that OST can provide.

Table 1: Caption

Required Cyber Security Tools	**Types of Cyber Incidents**
• VirusTotal IP address check • VirusTotal Website Check • Email scanner • Threat map • Access to GPT for general queries • File password cracker: (hash extracted with john the reaper, password cracking using Hashcat)	• Phishing Attacks • Malware Infections • Unsafe website (user opened a website that is not safe) • Denial of Service (DoS) / Distributed Denial of Service (DDoS) Attacks • Data Breaches • Social Engineering Attacks • Man-in-the-Middle (MitM) Attacks • Unsafe file name (like having a file called password.txt) • Brute Force Attacks • Zero-Day Exploits

3.5. Website design

The OST provides a website portal with several webpages, each dedicated to a specific function of the system. This structure allows for a clear development plan. The first step is to log into/register to the website, after that the

user will be presented with all the tools required to create an incident report. The hierarchical structure created for the site map ensures that the user does not find it difficult to navigate through the site, as the tools are positioned in a logical way [15]. The arrow in the sitemap simply demonstrates the way that a user can flow through the interface and are bidirectional

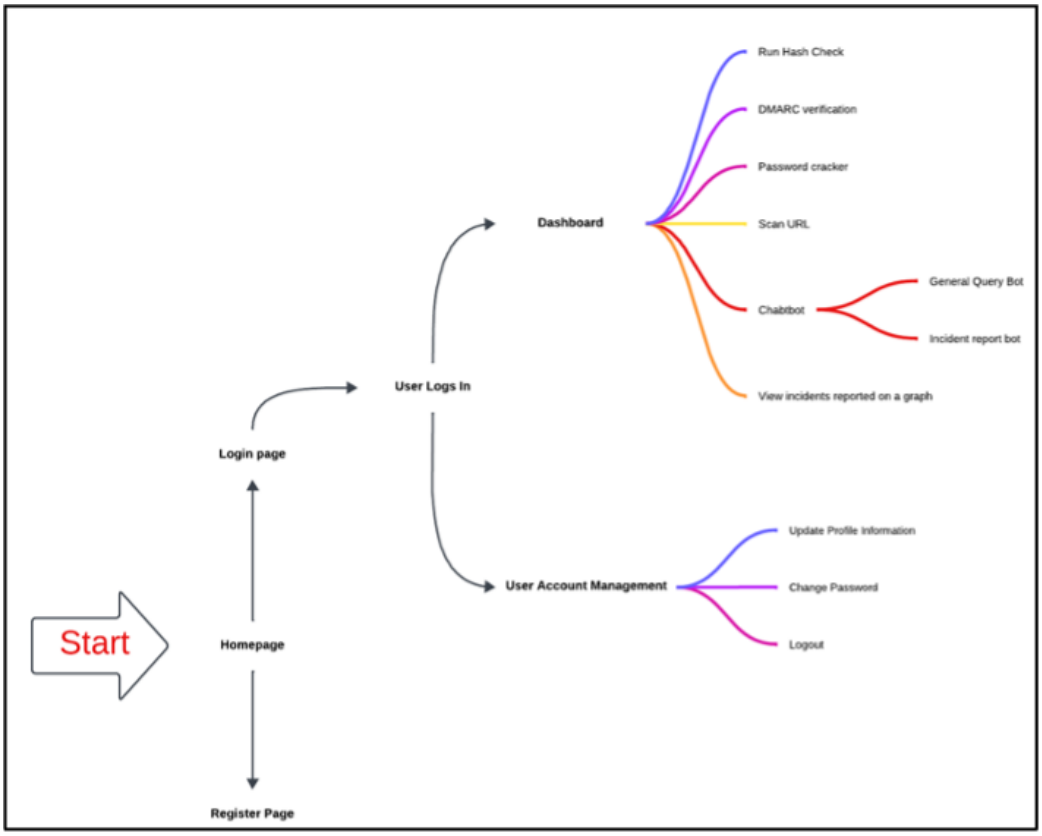

Figure 2: Website Flowchart

4. Validation

4.1. Implementation

Creating the brain of the chatbot begins with a deep understanding of user interactions and ends with a functional model capable of engaging in meaningful dialogues, the development stages aim to solve the inefficiencies of manual information gathering and report writing. A simplified diagram

addressing the chatbot architecture can be seen in the figure below, showing the intents that the user can request, and how the bot responds to them. Where the squares are actions completed by the chatbot and the diamond-shapes represent a decision to be made by the user, the oval is the termination step.

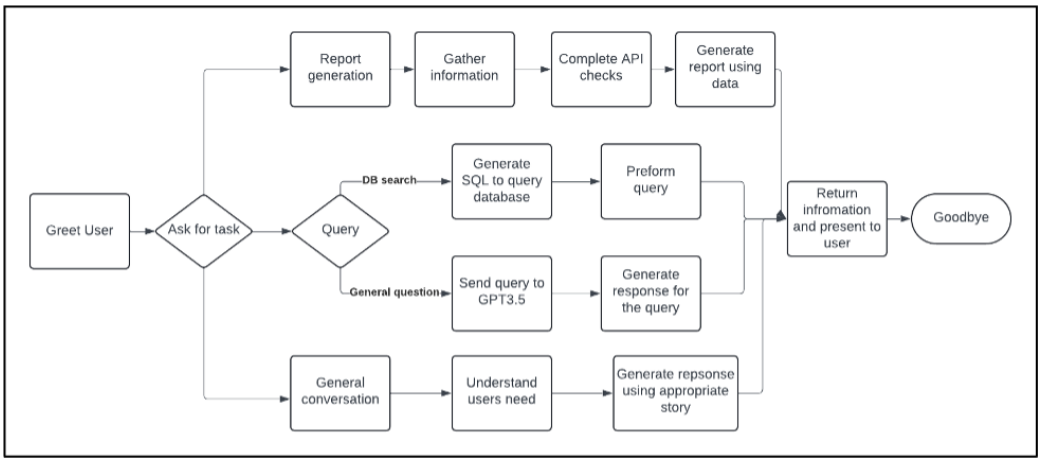

Figure 3: Rasa Flowchart

Creating the above design in the Rasa model through intents and entities, the chatbot can now understand the user's needs and manage user dialogue. The last step is to combine all the information created and involves feeding the bot all the defined entities, stories, and domain data. To test the model an interactive test using extra dialogue is employed, this allows the machine learning algorithm to make the chatbot predictions more accurate by correcting any misclassified information where necessary.

Custom actions allow the chatbot to surpass the basic conversational capabilities, and give it the ability to fetch real-time data from external APIs and interact with databases, which is required for cases where the chatbot is not general, rather aims to solve a specific task. The tasks which rely on third parties are:

1. Information extraction: To collect the case details by asking the user questions incrementally, each time the user is asked a question the chatbot back-end opens a listening server and waits for the user's response, for example the listener action is used to log the name of the person creating the report.

2. Database management: After collecting all the information and temporarily holding it, the next task is to store the data in a more permanent way, this is done by creating an integration between the chatbot and the SQL database. Another intent of the chatbot is to be able to retrieve information from the database, this is very complex as the user will be inputting data in plain English, but database retrieval uses Structured Query Language (SQL).
3. GPT AI Model Integration: Since the CyberBot leverages some of its capabilities to extract entities, answer questions, generate reports and SQL queries using the GPT-3.5 turbo model, to use OpenAI's model an API must be called.
4. Security and Analysis Features: One of the threat intelligence providers in the OST is VirusTotal, their services are being utilized to scan IP addresses, websites, domains, and file hashes.

4.2. Experiments

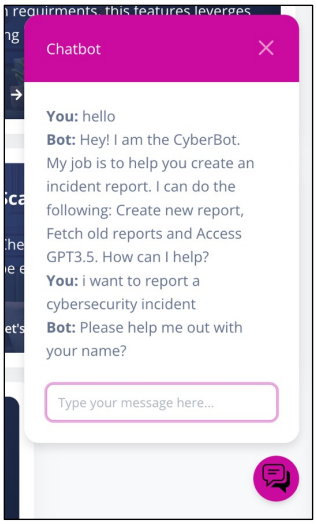

Figure 4: Operating the chatbot from the OST website

Overall, the speed of writing a traditional case report with a summary and information gathering took me 24 minutes and 30 seconds. However, when using the OST with the same information it took 4 minutes, this is much faster at about **600% time reduction**, (when manually timed by

myself) highlighting the time saved by using an AI chatbot to complete all the checks and using the CyberBot to write the summary of the report, in terms of mistakes both the manual and the OST generated report had the exact same details and information intelligence, the major difference was that in the manual report had some typos, while **the OST report had 100% grammatically correct English**. In conclusion, the system works as intended and I am confident that the project is a success. However, testing with real participants will be required to gain further confidence, but it is not possible due to the difficulty to deploying the chatbot along within the timeframe for this project and might have shown less promising results. However, current testing passed with outstanding results.

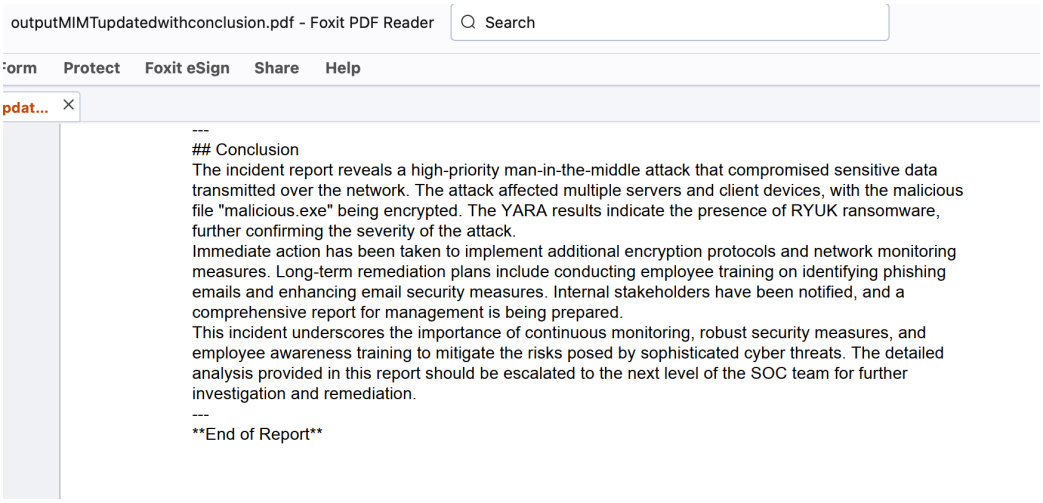

Figure 5: Demo report automatically generated by OST

5. Conclusion

In this work, the OST was developed to streamline cyber incident reporting by leveraging a chatbot-driven platform. We designed and implemented a solution to address the understaffing challenges faced by SOC teams. The OST's success demonstrates its potential to revolutionize the cyber security industry. By conducting thorough validation and user testing, the OST successfully consolidated essential tools into a single platform, enabling efficient incident reporting.

References

[1] Y. Liu, S. Li, Hybrid cyber threats detection using explainable ai in industrial iot, in: 2023 International Conference on Human-Centered Cognitive Systems (HCCS), 2023, pp. 1–6. doi:10.1109/HCCS59561.2023.10452621.

[2] A. A. Mughal, Building and securing the modern security operations center (soc), International Journal of Business Intelligence and Big Data Analytics 5 (1) (2022) 1–15.

[3] S. Li, S. Zhao, P. Yang, P. Andriotis, L. Xu, Q. Sun, Distributed consensus algorithm for events detection in cyber-physical systems, IEEE Internet of Things Journal 6 (2) (2019) 2299–2308.

[4] R. Brewer, Could soar save skills-short socs?, Computer Fraud & Security 2019 (10) (2019) 8–11.

[5] V. Dibia, Practical steps to reduce hallucination and improve performance of systems built with large language models, `https://newsletter.victordibia.com/p/practical-steps-to-reduce-hallucination`, accessed: 12 Jul. 2024 (2023).

[6] S. Tian, Q. Jin, L. Yeganova, P.-T. Lai, Q. Zhu, X. Chen, Y. Yang, Q. Chen, W. Kim, D. C. Comeau, R. Islamaj, A. Kapoor, X. Gao, Z. Lu, Opportunities and challenges for chatgpt and large language models in biomedicine and health, Briefings in Bioinformatics 25 (1) (2024) bbad493. doi:10.1093/bib/bbad493.
URL `https://doi.org/10.1093/bib/bbad493`

[7] C. market cap, Cisco (csco) - market capitalization, `https://companiesmarketcap.com/eur/cisco/marketcap/`, accessed: 24 August 2024 (2024).

[8] A. O'Neill, A. Ahmad, S. Maynard, Cybersecurity incident response in organisations: A meta-level framework for scenario-based training, in: Cybersecurity Incident Response in Organisations: A Meta-level Framework for Scenario-based Training, 2021, p. 11.

[9] (ISC)2, (ISC) 2 CYBERSECURITY WORKFORCE STUDY, (ISC)2, 2022.

URL https://www.isc2.org/-/media/Project/ISC2/Main/Media/documents/research/ISC2-Cybersecurity-Workforce-Study.pdf

[10] Reddit, Ai in cybersecurity, https://www.reddit.com/r/cybersecurity/comments/, accessed: 15 April 2024 (2024).

[11] Reuters, Saudi arabia plans 40 bln push into artificial intelligence, Reuters 1 (1), accessed: 15 April 2024 (2024).
URL https://www.reuters.com/world/middle-east/saudi-arabia-plans-40-bln-push-into-artificial-intelligence-nyt-reports-2024-03-19/

[12] T. Tran, Openai's chatgpt went completely off the rails for hours, The Daily BeastAccessed: 15 April 2024 (2024).
URL https://www.thedailybeast.com/openais-chatgpt-went-completely- off-the-rails-for-hours

[13] Vulnera, Hugging face vulnerability could lead to ai model supply chain attacks, accessed: 15 April 2024 (2024).
URL https://vulnera.com/newswire/hugging-face-vulnerability-could-lead-to-ai-model-supply-chain-attacks/

[14] K. Kashmer, A step-by-step guide to creating a cyber security incident report, https://techsevenpartners.com/a-step-by-step-guide-to-creating-a-cyber-security-incident-report/ (2022).

[15] Usability.gov, User-centered design basics — usability.gov, https://www.usability.gov/what-and-why/user-centered-design.html (2019).