AI4SOAR: A Security Intelligence Tool for Automated Incident Response

Manh-Dung Nguyen, Wissam Mallouli, Ana Rosa Cavalli, and Edgardo Montes de Oca
firstname.lastname@montimage.com
Montimage EURL
Paris, France

ABSTRACT

The cybersecurity landscape is fraught with challenges stemming from the increasing volume and complexity of security alerts. Traditional manual or semi-automated approaches to threat analysis and incident response often result in significant delays in identifying and mitigating security threats. In this paper, we address these challenges by proposing AI4SOAR, a security intelligence tool for automated incident response. AI4SOAR leverages similarity learning techniques and integrates seamlessly with the open-source SOAR platform Shuffle. We conduct a comprehensive survey of existing open-source SOAR platforms, highlighting their strengths and weaknesses. Additionally, we present a similarity-based learning approach to quickly identify suitable playbooks for incoming alerts. We implement AI4SOAR and demonstrate its application through a use case for automated incident response against SSH brute-force attacks.

KEYWORDS

SOAR, Automated Incident Response, Playbook Execution, Similarity Learning

ACM Reference Format:
Manh-Dung Nguyen, Wissam Mallouli, Ana Rosa Cavalli, and Edgardo Montes de Oca. 2024. AI4SOAR: A Security Intelligence Tool for Automated Incident Response. In *The 19th International Conference on Availability, Reliability and Security (ARES 2024), July 30-August 2, 2024, Vienna, Austria.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3664476.3670450

1 INTRODUCTION

Existing security solutions are designed to monitor an organization's IT infrastructure and network activities, generating security alerts and taking necessary actions upon detecting potential security threats. However, these cybersecurity systems often produce a high volume of alerts, typically managed and processed by security analysts using primarily manual or semi-automated procedures. Although most of data breach incidents take minutes to execute, companies often take weeks or even months to identify these attacks. For instance, when receiving alerts from Intrusion Detection Systems (IDS) about malicious behaviours, a security expert may switch to a defensive terminal to gather pertinent information by searching for network resources and confirming the threat. Subsequently, upon confirmation, the security expert instructs a firewall to isolate or block traffic from the affected area and updates threat intelligence with pertinent information. BakerHostetler's annual report showed that in 2023 security experts took an average of 63 days to discover an incident and an additional 20 days to implement corrective actions [16]. Therefore, a SOAR solution is specifically designed to address challenges associated with manual threat analysis and response delays to security incidents, providing enhanced security for an organization's ICT infrastructure. SOAR solutions are capable of automatically identifying suspicious activities within an organization's environment and proactively taking measures to mitigate potential cyberattacks.

Challenges. There are three primary challenges associated with SOAR systems [21]. Firstly, when integrating new security tools (modules), security analysts are required to either create new playbooks or update and maintain existing playbooks for known alerts [22]. This demands significant expertise in handling security incidents [18], along with in-depth knowledge of the integrated security tools and their functionalities [22, 23]. Secondly, once playbooks are defined, they become hard coded for a fixed set of alerts, resulting in a relatively static and rigid structure [6]. While this may be acceptable for investigative playbooks that don't require frequent changes [18], it becomes problematic for response playbooks that may need adjustments to address emerging threats and previously unseen alerts. Consequently, the response playbooks in the SOAR system may become ineffective against novel alerts, necessitating the rapid creation of new playbooks. Thirdly, in many SOAR systems, rule-based methods are employed to map playbooks to alerts [20]. The drawback of rule-based systems is that playbook relevance depends on rules rather than the context of the alert, making it challenging to validate, update, and ensure the completeness and correctness of the rules. Faced with these challenges and the time-consuming nature of playbook creation and error resolution, security analysts require assistance in efficiently and swiftly creating, updating, and maintaining playbooks and associated rules.

Solutions. Overcoming the limitations of existing SOAR solutions, such as limited customization and challenges in handling dynamic and complex threats, requires leveraging advanced techniques like Artificial Intelligence (AI) to create more effective and select the most suitable SOAR playbooks. We propose AI4SOAR, a security intelligence tool built on top of the open-source SOAR platform Shuffle for automated incident response, enabling security analysts to create custom playbooks for cyber threats like SSH brute force attacks via an intuitive interface. It utilizes similarity

ARES 2024, July 30-August 2, 2024, Vienna, Austria

ACM ISBN 979-8-4007-1718-5/24/07
https://doi.org/10.1145/3664476.3670450

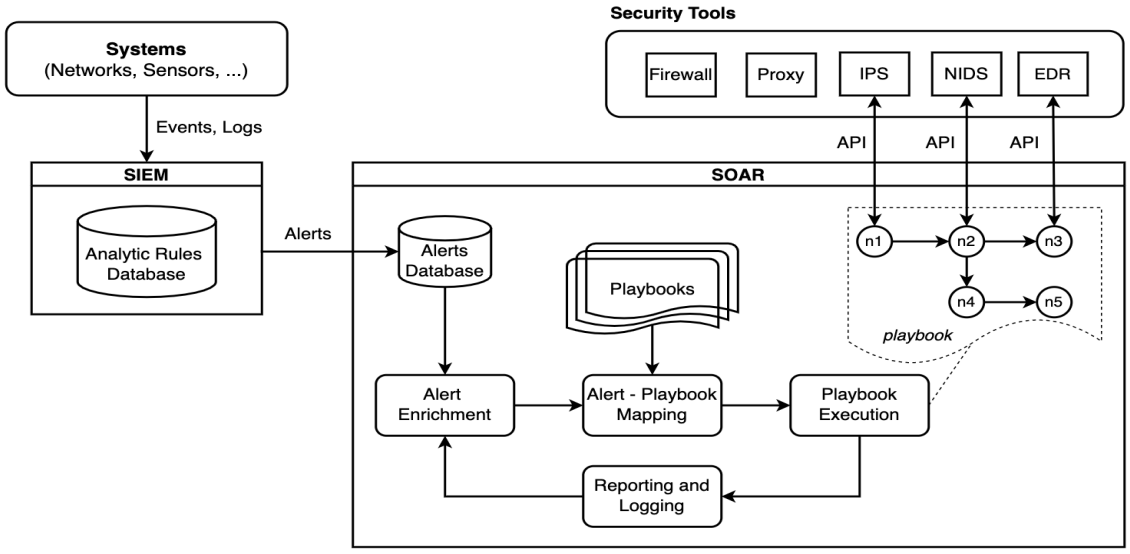

Figure 1: Overview architecture of a SOAR platform

learning algorithms to calculate similarity scores between new and historical alerts, enabling quick identification of suitable playbooks for incoming alerts. AI4SOAR simplifies the process of creating playbooks by allowing users to define and customize responses for common threats, such as SSH brute force attacks. With the ability to execute playbooks manually or automatically via APIs, organizations can respond to incidents promptly and effectively.

Contributions. Our contributions are as follows:

- We present a comprehensive survey of existing open-source SOAR platforms, discussing their strengths and weaknesses in detail.
- We propose a similarity-based learning approach that calculates similarity scores between new and historical alerts, enabling quick identification of suitable playbooks for incoming alerts.
- We implement the tool AI4SOAR[1] and demonstrate its application via a use case for automated incident response against SSH brute-force attacks.

The rest of the paper is organized as follows. Section 2 provides a background of a SOAR platform and discusses our study on existing open-source SOAR platforms for automated incident response. In Section 3, we present in detail the open-source SOAR platform Shuffle for creating and executing playbooks with a concrete example. Section 4 introduces AI4SOAR, a security intelligence tool for automated incident response based on similarity learning techniques and leveraging the open-source Shuffle platform. We then discuss the implementation and evaluate our tool in a concrete use case for automated incident response against SSH brute-force attacks in Section 5. Finally, Section 6 concludes the paper and offers insights into potential future directions for research and development.

[1] https://github.com/montimage/ai4soar

2 BACKGROUND & RELATED WORK

This section offers an overview of a SOAR platform and its role as a recommender system for automated incident response. Furthermore, we present a survey of existing open-source SOAR platforms, discussing their strengths and weaknesses in detail.

2.1 SOAR platform

Figure 1 shows an overview architecture of a SOAR platform. SOAR platforms frequently collaborate with Security Incident and Event Management (SIEM) components, which are responsible for identifying malicious or suspicious activities within raw logs or events from the system, including diverse sensors or network devices. The SIEM is configured to generate alerts in a format compatible with the SOAR platform, encompassing information such as alert type, affected systems, and event descriptions. Subsequently, upon detecting specified signatures or anomalies, these alerts are transmitted in real-time from the SIEM to the designated endpoint or API of the SOAR platform. The SOAR platform then performs data mapping and normalization, ensuring consistency and compatibility with its internal data model. This step is crucial for extracting relevant information from SIEM alerts. Next, the SOAR platform enriches SIEM alerts by retrieving contextual data from external sources like threat intelligence feeds. Using its alert-playbook mapping engine, the SOAR platform determines the appropriate (predefined or partial) playbook or workflows to execute based on alert characteristics, tailoring the response to the specific incident. Automated playbook execution follows, involving both investigative actions (e.g., querying additional data sources, user/asset information or verifying URL reputation) and response actions (e.g., isolating systems, notifying stakeholders, blocking malicious IP addresses in the firewall or terminating potential malicious processes on an endpoint). Throughout the process, the SOAR platform generates detailed reports and logs, providing valuable insights for post-incident

Table 1: Strengths and weaknesses of existing open-source SOAR platforms

Platform	Strengths	Weaknesses
Walkoff	Easy-to-use with a drag-and-drop workflow editor. Flexibility, modular, easy integration with other tools and visual analytics.	Development was dropped, the GitHub repository has been archived by the owner.
Shuffle	Simple to use and install, available on Docker. Highly documented. Integration with SIEM, threat intelligence feeds and other security tools.	Still under development, with ongoing enhancements and bug fixing.
TheHive	STIX2 standard. Numerous analysers. Comprehensive and efficient database. Few documentations. Difficult deployment process.	High hardware requirements. Difficult deployment process.
Patrowl	Multidisciplinary engines. Results normalization. Scan automation. Pro-edition and SaaS availability. API availability.	Engines configuration might be tedious. Limited data.
Alertflex	High integration. Alerts filtering, prioritization, and visualization. Detection threats, misconfigurations, vulnerabilities.	Few documentations. Difficult deployment process.

analysis, compliance reporting, and continuous improvement of response workflows, such as enriching alerts in the database or making existing playbooks more comprehensive and complete.

In a SOAR platform, a playbook (or workflow) is a sequence of activities that security analysts manually define. These playbooks align with incident response plan (IRP) policy documents, offering a step-by-step guide for security analysts to investigate and respond to alerts. Playbooks are categorized into three types based on the level of automation. First, manual playbooks involve a series of tasks executed manually by security analysts, requiring human intervention at various stages. Second, semi-automated playbooks represent a hybrid approach, integrating both automated and manual subtasks to provide a more flexible response strategy. Third, fully automated playbooks are entirely automated, relying on predefined responses and automated actions to handle security incidents without direct human involvement. Within the broader classification, two primary types of playbooks, investigative and response playbooks, further refine the orchestration process. Investigative playbooks focus on guiding the steps involved in collecting additional contextual information about a security alert. On the other hand, response playbooks encode actions to mitigate known or expected security events, incidents, or threats, preventing harm to the entire network.

As illustrated in Figure 1, a playbook can be visually represented as a directed graph, wherein the nodes represent the specific configuration of security tools, and the edges are the direction of logic flow and associated conditions. These modules serve as connectors, linking security tools to the SOAR platform through application programming interfaces (APIs), and are responsible for executing actions to address a given security alert. For example, the Network Intrusion Detection System (NIDS) tool seamlessly connects the organization's network sensors to the SOAR platform, using APIs for efficient communication. In this orchestrated environment, it can automate responses to detected security threats by analysing network traffic patterns, identifying intrusions, and triggering actions like isolating affected segments, notifying security teams, or initiating predefined investigative workflows.

2.2 Recommender systems

The taxonomy of recommender systems in a simple 4-phase workflow of incident handling [19] consists of Triage, Analysis and Response, Intelligence and Prevention, and Management. SOAR platforms offer a versatile and comprehensive approach to incident management across all tiers of the cybersecurity lifecycle. In the initial phase of Triage, SOAR aids in automating alert analysis, prioritizing incidents based on severity, and recommending appropriate playbooks for further investigation. Moving to the Analysis and Response tier, SOAR excels in automating detailed analysis, correlating data from diverse sources, and suggesting effective response actions, drawing insights from past incidents to optimize strategies. In the Intelligence and Prevention tier, SOAR integrates threat intelligence, analyses patterns, and recommends proactive measures, contributing to attack prediction and intelligence-based decision-making. Finally, in the Management tier, SOAR plays a pivotal role in strategic decision-making by assisting in defence planning, prioritizing investments, and continually enhancing the organization's overall security posture. Through automation, orchestration, and intelligent response capabilities, SOAR adapts to the specific needs of each tier, streamlining incident management processes and fortifying cybersecurity resilience.

2.3 Comparison of open-source SOAR platforms

In the realm of SOAR platforms, several solutions [2, 3], whether commercial or open source, offer distinct features. Concerning commercial solutions for SOAR, there is a diverse range of options, but the main challenge lies in their often-high costs. For example, commercial SOAR software like Siemplify (now Chronicle SOAR [4], part of Google Cloud) and Cortex XSOAR [5] by PaloAlto is priced in the thousands of dollars, depending on factors such as the organization's size, deployment requirements, and additional features. While some of these solutions offer free versions, often referred to as Community Editions, they come with various limitations. Therefore, our focus is primarily on open-source solutions, particularly those with AI features or integration-friendly capabilities to facilitate the seamless incorporation of the framework with our AI developments.

Table 1 provides an overview of the strengths and weaknesses of existing open-source SOAR platforms that we have reviewed and tested. WALKOFF [15] is an automation framework that, through integration with various tools, enables users to define sequences of actions, providing an automated solution of repetitive tasks. Shuffle

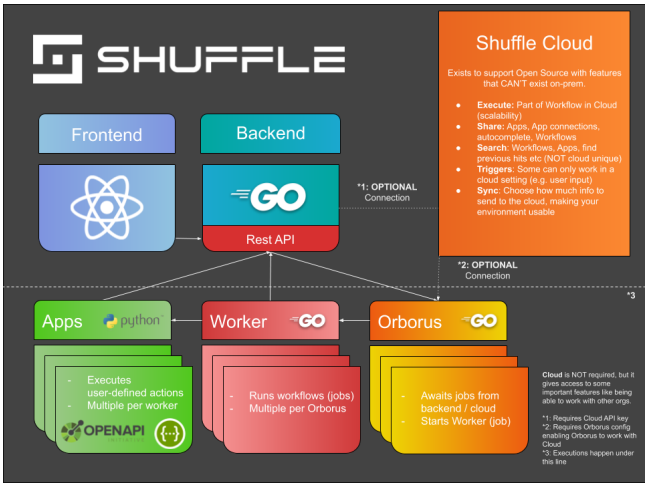

Figure 2: Shuffle architecture [8]

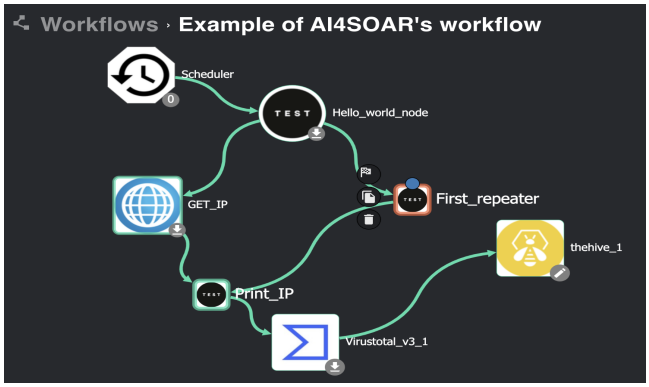

Figure 3: Example of AI4SOAR's workflow

[8], another open-source SOAR platform sharing many similarities with WALKOFF, distinguishes itself through its comprehensive documentation [10], high customizability, and user-friendly interface. With numerous integrations and Docker availability, it offers a straightforward solution for users. However, being still under development, users may encounter occasional instability. TheHive [14], previously open source in version 4 and below, now provides a commercialize version 5 under the name of StrangeBee [13]. It excels in maintaining a comprehensive and efficient database, adhering to the STIX2 standard, and supporting automated incident response. Nevertheless, its challenging deployment process and high hardware requirements can present obstacles. Patrowl [7], with its multidisciplinary engines, results normalization, and availability of a Pro edition, provides robust capabilities, but users may find configuring engines somewhat tedious, and the tool has limited data compared to others. Finally, Alertflex [1], developed since 2016, emphasizes high integration with tools like MITRE and MISP, but its challenging deployment and limited documentation might impact user adoption.

3 SHUFFLE ARCHITECTURE

The following section introduces a more in-depth exploration of Shuffle's architecture and its principal modules. Additionally, we provide a demonstration of Shuffle's workflow for the use case of automated analysis of IP addresses to handle security incidents within this SOAR platform.

3.1 Architecture

Shuffle [8] is one of the pioneering open-source SOAR platforms. Despite now providing various subscription plans, it continues to offer an open-source version. Shuffle actively develops workflows across various use case categories, which are organized into eight distinct groups: (1) Communication, (2) Case Management, (3) SIEM, (4) Assets, (5) IAM, (6) Intelligence, (7) Network, (8) Eradication Cyber Incident Detection, Prevention, Remediation, Case Management, Communication, etc. Each use case involves specific SOC tools tailored to scenarios. Furthermore, Shuffle provides an intuitive graphical user interface where users can create organizations, manage users belonging to organizations, and, most importantly, create playbooks. Shuffle integrates 135 security applications, which users can utilize to create new playbooks across various categories, such as Collect, Enrich, Detect, Respond, and Verify, with a variety of public playbooks provided by the community available for adaptation and use. Moreover, users can also incorporate their own applications from OpenAPI or Swagger.

Figure 2 illustrates Shuffle's workflow design environment, with the pointing arrow indicating the direction of input data in Shuffle, accessible by all nodes within the workflow. The Shuffle platform consists of two principal tools: Apps and Workflows, as depicted in Figure 2. Firstly, Apps represent plug-and-play functionalities that facilitate integration with other applications, predominantly relying on the OpenAPI, a Web API standard. Currently, Shuffle supports apps for a range of security tools, including The Hive, Cortex, VirusTotal, MISP, Elastic Search, etc. These tools effectively manage cybersecurity concerns, addressing aspects like threat and vulnerability management, authority management, security incident response, and automation of security operations. Secondly, Workflows, serves as the central hub where various elements come together. By combining Apps, Triggers, and Variables, workflows essentially operate as playbooks, responding to potential threats within the system and, in some cases, even proactively preventing potential security risks. When seamlessly integrated with TheHive version 4, known for its comprehensive database, adherence to the STIX2 standard, and support for automated incident response, the combined solution enhances incident management capabilities. Additionally, the integration with Cortex further amplifies the strength by leveraging its analytical and response capabilities.

3.2 Example of playbook

This section discusses in detail how to create a playbook in Shuffle to automate the process of obtaining our local IP, scanning it on VirusTotal, and generating alerts on TheHive. We create several nodes in the workflow as follows:

- A *Hello_world* node, which is our start node, meaning the first action to be executed.

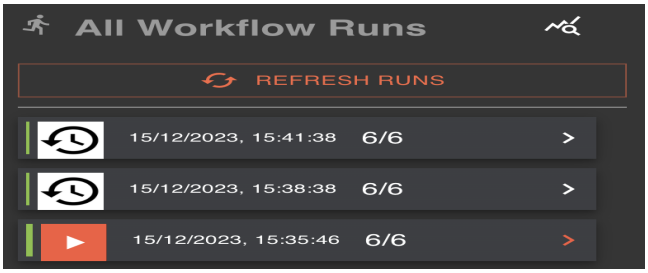

Figure 4: All workflow executions occur every 3 minutes

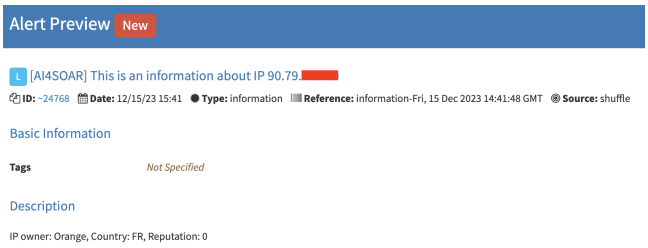

Figure 5: Alert preview showing some information of our local IP extracted from Virustotal's reports

- A *First_repeater* node that repeats data from the *Hello_world* node to show how data is passing between nodes.
- A *GET_IP* node making a HTTP GET request from the website[2] to obtain our local IP address in json format.
- A *Print_IP* node, which reads the json data from the *GET_IP* node and prints the IP.
- A *VirusTotal* node, that searches for information of our IP.
- A *TheHive* node for creating alerts.
- A *Scheduler* node that schedules the workflow to execute at specified intervals, here we set it to run every 180 seconds.

In the workflow, we use TheHive, an incident response platform, to generate an alert concerning our local IP address information. As illustrated in Figure 4, the workflow operates every 3 minutes and TheHive generates alerts on schedule. Furthermore, each alert description provides details about our local IP, including the owner of the Autonomous System to which the IP belongs, the country of location, and the IP's reputation score derived from the VirusTotal community's votes, as illustrated in Figure 5. Concretely, our local IP holds a reputation score of 0, indicating its harmlessness. In real-world scenarios where we encounter malicious IP addresses with negative reputation scores, we can undertake various mitigation actions to effectively respond to and mitigate the potential threat. These actions may involve redirecting traffic from the identified malicious IP to a honeypot environment or automatically blocking the malicious IP on our honeypots to prevent further interaction.

4 AI4SOAR FRAMEWORK

The following section presents the AI4SOAR architecture and its principal components, which leverage AI algorithms to calculate similarity scores between new and historical alerts, enabling quick identification of suitable playbooks for incoming alerts.

4.1 Architecture

Figure 6 illustrates an overview of AI4SOAR architecture, designed to enhance incident management and response through intelligent automation. The *Playbook Consumer* module is responsible for ingesting (partial) playbooks from external security tools. Alternatively, it can also receive events or logs from the SIEM tools for further analysis. The *Intelligent Playbook Orchestration* module serves as the core module, employing AI algorithms for refining the completeness of the received partial playbooks and creating a more robust foundation for incident response. For example, this module can obtain the current attack/incident description and dynamically create, reconfigure, or add branches to existing playbooks, tailoring responses to be executed for specific attacks. After optimization, the *Orchestration Engine* module is triggered, operating within Docker containers and utilizing the open-source SOAR platform Shuffle as the orchestrator. This engine can integrate with TheHive and Cortex, executing the playbooks and interacting with other security tools through APIs. Finally, the *Analysis and Reporting* module plays a crucial role in collecting results and logs generated by the Orchestration Engine module. Operating in a closed-loop fashion, this information is fed back to the Intelligent Playbook Orchestration to facilitate the refinement of AI algorithms for adaptive responses to emerging threats. Incident information and playbook effectiveness are shared with security threat intelligence tools to foster collaborative threat intelligence, enhancing the framework's effectiveness against similar attacks. For instance, AI4SOAR can collaborate with a cyber threat intelligence tool by receiving processed logs for improving AI-driven playbooks and then forwarding malicious data collected from playbook executions to trigger honeypots and strengthen its capabilities in identifying and deceiving malicious actors.

4.2 Intelligent Playbook Orchestration module

4.2.1 Predefined playbooks. We compile and generate a dataset of playbooks designed for incident response, following the MITRE technique. The objective is to ensure a comprehensive collection of predefined sequences of actions tailored to address various alert scenarios against common attacks in different use cases. These playbooks are influenced by generic playbook datasets gathered from various public sources, such as Shuffle [11] and Splunk [12]. This curated set of playbooks serves as a valuable resource for security teams, offering structured guidance and predefined responses to common security incidents. Initially, historical alerts are associated with predefined playbooks, via *playbook_id*, providing contextual information for incident response.

4.2.2 Alerts Preprocessing. The alerts preprocessing phase is crucial for managing and analyzing alert data efficiently in AI4SOAR. It involves two key steps: alerts encoding and alerts embedding. While alerts encoding transforms raw data into a machine-learning-friendly format, alerts embedding captures essential data representations in a lower-dimensional space, aiding in more effective processing and analysis.

[2] https://api.ipify.org/?format=json

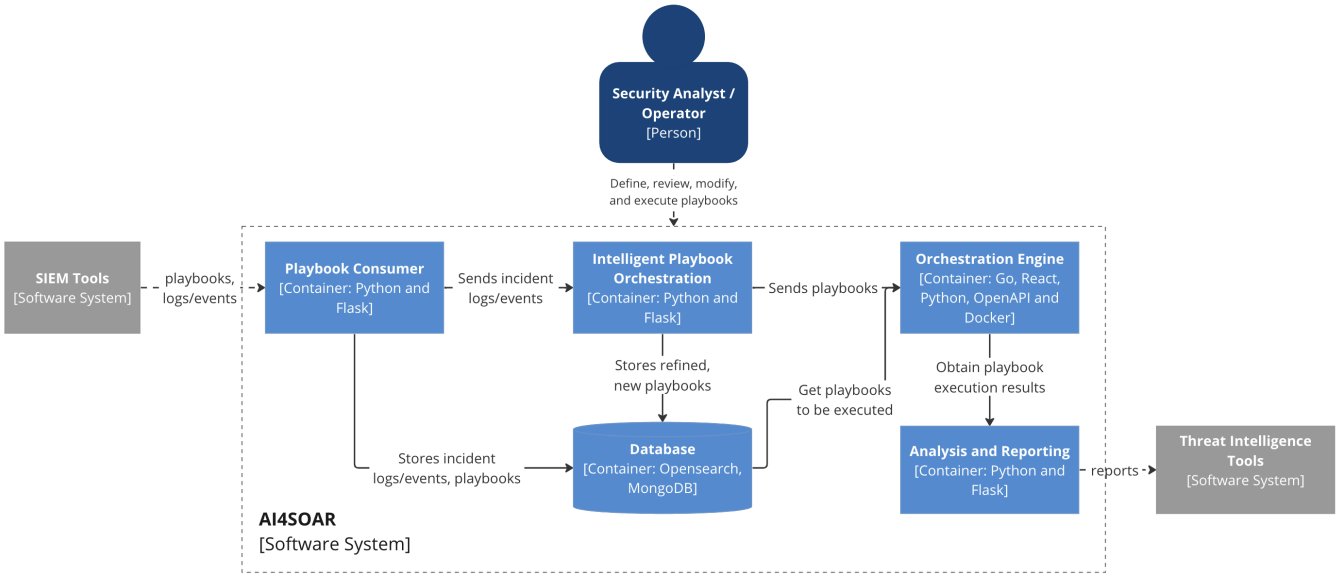

Figure 6: Overview of AI4SOAR's architecture

Alerts encoding begins by converting alerts, both historical and incoming, into a standardized format suitable for machine learning tasks. This involves decoding alerts in JSON format and extracting relevant fields like *technique*, *srcip*, *srcport*, *dstuser*, and *hostname*. For the *technique* feature, it checks for the presence of MITRE techniques and assigns binary values based on their occurrence. These one-hot encoded vectors, representing alerts with sparse key information, are then concatenated into numpy arrays for processing efficiency.

Next, these vectors are encoded into lower-dimensional embedding vectors using an autoencoder model, minimizing the difference between input and reconstructed output vectors. The resulting embedding vectors serve as dense representations of alerts, capturing important features and ensuring versatility across various alerts. The autoencoder model, defined using TensorFlow's Keras API, comprises an input layer, a dense hidden layer with a rectified linear unit (ReLU) activation function, and an output layer with a sigmoid activation function. It is trained on the one-hot encoded alert vectors, learning to reconstruct input vectors while compressing them into a lower-dimensional space over 1000 epochs. The encoded alerts facilitates further analysis with a more concise and meaningful alert representation.

4.2.3 Similarity learning. Suppose that the encoded alerts can be represented by n metrics (i.e., n attributes). This set of n attributes can be depicted by a vector in an n-dimensional space. Calculating the similarity and dissimilarity between two encoded alerts becomes the task of measuring the orientation (the angle) and magnitude (the length) of their respective vectors. Figure 7 illustrates an example in a 3-dimensional space.

The AI4SOAR tool harnesses similarity learning techniques to effectively map incoming alerts to predefined playbooks. To compute the similarity between incoming alerts and historical ones,

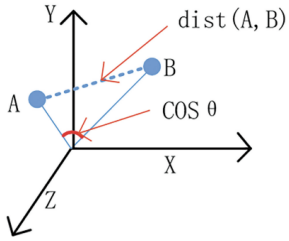

Figure 7: Similarity calculation in a 3-dimensional space

the tool employs various similarity and distance measures, such as cosine similarity, Euclidean distance, and Manhattan distance [17]. These measures facilitate the calculation of similarity scores ranging from 0 to 1, where higher scores denote greater similarity between alerts (e.g., if the similarity score equals 0.98, there is a 98% probability that two compared alerts are considered equivalent). By examining the orientation and magnitude of vectors representing different states in a multidimensional space, AI4SOAR discerns the similarity or dissimilarity between them. During the training phase, the tool selects appropriate measures to maximize similarity scores for known alerts and minimize them for dissimilar ones, thereby enhancing the accuracy of alert identification and playbook selection. Overall, the integration of similarity learning techniques enhances the tool's capability to identify the most suitable playbook for effectively managing incoming alerts.

4.3 Orchestration Engine module

The core of AI4SOAR is its Orchestration Engine, which is based on the open-source SOAR platform Shuffle. The Intelligent Playbook Orchestration module can utilize the RESTful APIs [9] provided by

the Orchestration Engine module (or the Shuffle platform) for various functions, including creating new workflows, refining existing ones, executing workflows, and fetching the results of workflow execution. For instance, sending a GET request to /workflows enables the retrieval of a list of predefined workflows. Subsequently, sending a POST request to /workflows/workflow_id/execute initiates the execution of a specified workflow, optionally with arguments, such as *"execution_argument"* and *"start"*, indicating the starting node. Finally, sending a POST request to /workflows/results allows the retrieval of the execution results, which are then processed and forwarded to other components for further analysis. All essential APIs of the Shuffle platform will be wrapped in AI4SOAR, making them easily accessible for automated incident response.

5 IMPLEMENTATION AND EVALUATION

5.1 Implementation

AI4SOAR is implemented using Docker Compose, seamlessly integrating essential security tools, such as Shuffle, TheHive, Cortex, and others. A web-based server built with Flask and Python serves as the central hub for managing SOAR functionalities. This server communicates with Shuffle via its REST APIs, ensuring smooth integration and workflow execution. Kafka facilitates data exchange in JSON format with external components, promoting efficient communication and data transfer. MongoDB acts as the primary database, storing crucial information such as alerts and playbooks for quick access. Additionally, AI4SOAR utilizes Python libraries, like NumPy, TensorFlow, and scikit-learn, for machine learning tasks, enhancing its capabilities for automated incident response and threat detection.


```
{
    ...
    "_index": "wazuh-alerts-4.x-2024.01.04",
    "_source": {
        "data": {
            "srcip": "95.214.27.52",
            "dstuser": "user",
            "srcport": "63228"
        },
        "rule": {
            "level": 5,
            "description": "sshd: authentication failed.",
            "mitre": {
                "technique": [
                    "Password Guessing",
                    "SSH"
                ],
                "tactic": [
                    "Credential Access",
                    "Lateral Movement"
                ]
            },
        },
        "id": "1704401393.810686",
        "full_log": "Jan 4 20:49:51 ai4soar sshd[94488]: Failed
        ↪  password for user from 10.0.2.2 port 63228 ssh2",
        "timestamp": "2024-01-04T20:49:53.154+0000"
    },
    "playbook_id": "cd5780a4-f624-400c-b734-1c1d98ff3820",
    ...
}
```


Listing 1: Example an alert generated by Wazuh in JSON and associated with a predefined playbook

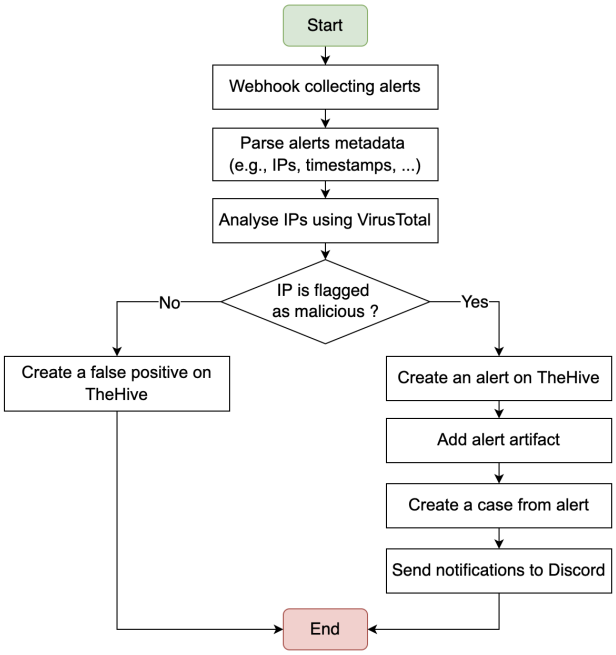

Figure 8: Playbook for SSH brute force attacks

5.2 Use Case: SSH brute force attacks

In this section, we examine a use case focused on automated incident response against SSH brute force attacks. Considering the Wazuh alert example in Listing 1, we find an SSH authentication failure, noting details like the timestamp and source IP address. This alert indicates a failed SSH authentication attempt, marked as "sshd: authentication failed" with severity level 5. Linked to tactics such as "Credential Access" and "Lateral Movement" in the MITRE framework, this incident involves techniques like "Password Guessing" and "SSH". Further inspection of the log entry provides crucial context for security analysis and response.

The predefined playbook for SSH brute force attacks, seen in Figure 8, begins with a webhook collecting alerts for suspicious activities. These alerts are parsed to extract crucial metadata, including source and destination IP addresses, timestamps, etc. Next, the source IP is analyzed using the trusted threat intelligence platform VirusTotal. If not flagged as malicious, a false positive case is created. Otherwise, an alert is generated on TheHive, a collaborative incident response platform, with an associated alert artifact providing context. A case is then generated from the alert to initiate a structured response, followed by a notification sent to Discord, enabling swift action.

Upon receiving a new alert, AI4SOAR calculates its similarity score using methods like cosine similarity or Euclidean distance against historical alerts. It suggests the most suitable playbooks based on these scores, providing details like names, IDs, and descriptions. With AI4SOAR's playbook suggestions, analysts can

review and select the best one for execution, often with a 99% similarity score. After execution, a case linked to the SSH brute force attack alert appears in TheHive, along with the attacker's IP address. Additionally, a message is posted on Discord.

6 CONCLUSION

In conclusion, this paper presents AI4SOAR, a security intelligence tool designed to address the challenges associated with manual threat analysis and incident response delays. By leveraging similarity learning techniques and integrating with the open-source SOAR platform Shuffle, AI4SOAR offers organizations an efficient solution for quickly selecting suitable playbooks to be executed for automated incident response. The implementation and evaluation of AI4SOAR in a real-world use case demonstrate its practical utility and effectiveness in mitigating security threats. Moving forward, our focus will be on refining playbooks to address unforeseen threats, deploying and testing them across different use cases.

ACKNOWLEDGMENTS

This research is supported by the H2020 project AI4CYBER N° 101070450.

REFERENCES

[1] 2024. Alertflex. https://alertflex.org/
[2] 2024. Awesome Incident Response. https://github.com/meirwah/awesome-incident-response
[3] 2024. Awesome SOAR. https://github.com/correlatedsecurity/Awesome-SOAR
[4] 2024. Chronicle SOAR. https://chronicle.security/suite/soar/
[5] 2024. Cortex XSOAR. https://xsoar.pan.dev/
[6] 2024. How to be a SOAR winner: 8 successful strategies to unlocking more value from your security orchestration, automation and response (SOAR) solution. https://www.ibm.com/security/digital-assets/soar/how-to-be-a-soar-winner/
[7] 2024. Patrowl. https://www.patrowl.io/
[8] 2024. Shuffle: A general purpose security automation platform. https://github.com/Shuffle/Shuffle
[9] 2024. Shuffle APIs. https://shuffler.io/docs/API
[10] 2024. Shuffle documentation. https://shuffler.io/docs
[11] 2024. Shuffle workflows. https://github.com/Shuffle/workflows
[12] 2024. Splunk playbooks. https://research.splunk.com/playbooks/
[13] 2024. StrangeBee. https://www.strangebee.com/thehive
[14] 2024. TheHive. https://github.com/TheHive-Project/TheHive
[15] 2024. WALKOFF. https://github.com/nsacyber/WALKOFF
[16] BakerHosteller. 2023. 2023 Data Security Incident Response Report.
[17] Aurélien Bellet, Amaury Habrard, and Marc Sebban. 2022. *Metric learning*. Springer Nature.
[18] Robert A Bridges, Ashley E Rice, Sean Oesch, Jeffrey A Nichols, Cory Watson, Kevin Spakes, Savannah Norem, Mike Huettel, Brian Jewell, Brian Weber, et al. 2023. Testing SOAR tools in use. *Computers & Security* 129 (2023), 103201.
[19] Martin Husák and Milan Čermák. 2022. SoK: applications and challenges of using recommender systems in cybersecurity incident handling and response. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*. 1–10.
[20] Irina Kraeva and Gulnara Yakhyaeva. 2021. Application of the metric learning for security incident playbook recommendation. In *2021 IEEE 22nd International Conference of Young Professionals in Electron Devices and Materials (EDM)*. IEEE, 475–479.
[21] Ryuta Kremer, Prasanna N Wudali, Satoru Momiyama, Toshinori Araki, Jun Furukawa, Yuval Elovici, and Asaf Shabtai. 2023. IC-SECURE: Intelligent System for Assisting Security Experts in Generating Playbooks for Automated Incident Response. *arXiv preprint arXiv:2311.03825* (2023).
[22] Zarrin Tasnim Sworna, Muhammad Ali Babar, and Anjitha Sreekumar. 2023. IRP2API: Automated Mapping of Cyber Security Incident Response Plan to Security Tools' APIs. In *2023 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 546–557.
[23] Zarrin Tasnim Sworna, Chadni Islam, and Muhammad Ali Babar. 2023. Apiro: A framework for automated security tools api recommendation. *ACM Transactions on Software Engineering and Methodology* 32, 1 (2023), 1–42.