Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

A cyber threat intelligence model using MISP and machine learning in a SOC environment

Asia Othman Aljahdali *

Cybersecurity Department, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia

ARTICLE INFO

Article history:
Received 5 April 2025
Received in revised form
11 September 2025
Accepted 7 October 2025

Keywords:
Cyber threat intelligence
Security operations center
Machine learning
Fraud detection
Mobile transactions

ABSTRACT

Information and communication technology (ICT) has become a major global driver, but it also exposes organizations to frequent cyber threats, making asset protection increasingly difficult. Cyber threat intelligence (CTI) is essential for improving cybersecurity, especially when integrated into a security operations center (SOC) for real-time threat monitoring and analysis. This study proposes a real-time CTI framework within a SOC environment, hosted on Linode, which integrates the Malware Information Sharing Platform (MISP) and a Security Information and Event Management (SIEM) system to collect indicators of compromise (IoCs). The framework uses machine learning to detect fraud in mobile money transactions such as cash-in, cash-out, debit, payment, and transfer. Fraudulent activity often involves the use of stolen identity information for unauthorized transactions. The system generates detailed alert reports and provides predictive insights into potential threats, helping organizations strengthen user trust and protect their reputation. Experimental results using financial datasets show high performance: logistic regression achieved 98.83% accuracy, while the random forest model reached a test accuracy of 95.86% and cross-validation accuracy of 95.76%. The F1-score was 0.9586, and the ROC-AUC score was 0.9923, indicating strong classification capability.

1. Introduction

Nowadays, there is a significant increase in sophisticated cyberattacks, including advanced persistent threats and zero-day attacks. Such attacks can easily bypass established defenses like firewalls and intrusion detection systems (IDS), compromise crucial infrastructure, and result in catastrophic failures (Krishnapriya and Singh, 2024). It has become harder and harder to keep up with these growing cyber threats; therefore, prompt dissemination of pertinent information about these risks is crucial for effective protection and mitigation. Any information that could assist a company in identifying, assessing, monitoring, and responding to cyber threats is referred to as cyber threat intelligence (CTI). In the era of big data, it's critical to keep in mind that the term "intelligence" often refers to information that has been gathered, examined, analyzed, and transformed into a set of actions that can be taken or that has become actionable (Koloveas et al., 2021). A CTI framework is the best answer for dealing with threats to information systems and information security because of its comprehensiveness. It is based on organized, evidence-based threat information and is seen as relevant and useful knowledge. Threat information is, therefore, a crucial component of informing decision-makers about the present state of their organization's security and outlining essential security actions (Möller, 2020). These systems can consume huge amounts of data, offer advanced protection capabilities, and respond to occurrences in real time. These platforms should feature automated information transformation and intelligence creation processes to offer a more effective, proactive, and timely defense strategy.

Ransomware continues to be one of the most common and persistent malware varieties. Organizations must invest in cyber threat intelligence to monitor corporate networks to (1) detect how and when a breach occurs, (2) be able to identify compromised systems, and (3) be able to determine how adversaries modified their systems (Preuveneers et al., 2020). The content of CTI formats comprises observable artifacts, indicators of

* Corresponding Author.
Email Address: aoaljahdali@uj.edu.sa
https://doi.org/10.21833/ijaas.2025.11.001
Corresponding author's ORCID profile:
https://orcid.org/0000-0002-9013-9465

compromise (IoCs), or tactics, techniques, and procedures (TTPs), starting with the underlying threat information. CTI artifacts include malware hashes and malicious IP addresses. Recent research suggests that companies may use mining and analytical approaches to retrieve artifacts from unstructured data. This data analysis then facilitates several crucial tasks, including information exchange (and receiving) and incident response, thanks to CTI frameworks, standards, and other formats (Schlette et al., 2021). By offering services, including controlling vulnerabilities, threat intelligence, digital investigation, and data collection and analysis, the security operation center (SOC) plays a critical role in most enterprises. Most of the information used in the real-time environment is gathered from open-source and non-secure sources. The data collected is then sent to the SOC, which evaluates it using threat intelligence before reporting it to end users to determine whether there are any risks or threats (Varatharaj et al., 2021). A security analyst's ability to detect threats depends on their operational and strategic understanding of today's security environment and relevant intelligence, which allows them to better understand the data coming from the security information and event management (SIEM) systems in the SOC. The analyst is specifically helped by background knowledge of the system's context (e.g., which software programs are installed on their system? What is the typical behavior of the system? What vulnerabilities might there be?) and the outside world (e.g., intelligence on new attacks that have been launched or are being discussed as potential threats) (Mittal et al., 2019). While SIEM solutions offer control over networks and systems, SOC staff must study the operational environment and the current threat landscape to get an advantage over cyber threat actors and foresee possible threats and dangers. A SOC's success depends on various factors, including its access to useful threat intelligence.

Machine learning algorithms are one of the most advanced tools for detecting cybercrime. Machine learning techniques can be used to overcome the restrictions and limits present in traditional detection approaches because they can learn from data and experience without being explicitly coded. ML methods are also used on the attacking side to get beyond the defense wall. On the other hand, ML approaches are used to develop quick and effective defense strategies (Shaukat et al., 2020).

Three important categories of ML can be distinguished: semi-supervised, unsupervised, and supervised learning. In supervised learning, we already know the labels and target classes for the data. Unsupervised learning is based on finding patterns in the data with no prior knowledge of the target classes. Semi-supervised learning refers to blending supervised and unsupervised learning (Angra and Ahuja, 2017).

Most organizations nowadays are primarily concerned with firewalls, intrusion prevention systems (IPS), and SIEMs. Organizations can learn about their opponents' next actions with a CTI system. As a result, the business may actively defend itself against potential cyberattacks. Given the significant benefits of CTI platforms, our objective for this paper is to create a CTI system with the help of existing CTI platforms within the industry today. CTI platforms are useful aggregators of intelligence sourced from various data feeds and play an essential role in helping SOC operators enhance their results. Machine learning (ML) algorithms will be used in training the model to detect anomalies.

This research proposes a real-time cyber threat intelligence framework in a SOC environment. The proposed framework is built on Linode and integrates the Malware Information Sharing Platform (MISP) with security information and event management (SIEM) to collect indicators of compromise (IoC) feeds. Additionally, machine learning is utilized to train models. Finally, the proposed system sends a detailed report on the alert. Companies can receive predictive output about impending threats by deploying the CTI System. The system enhances user trust, which protects an organization's reputation.

The main contribution of this study lies in developing a cyber threat intelligence model by integrating the CTI platform with the SIEM solution. Online CTI resources cover all recent major attack vectors, such as malicious URLs, phishing URLs, spam, bot IPs, social media, and websites. Additionally, the ML algorithm is used to detect IoC. This paper is organized as follows: Sections 2 and 3 discuss the background and a wide range of related work for a better understanding of existing cyber threat intelligence systems and SOC real-time monitoring. Section 4 proposes a threat intelligence model. Section 4 discusses the implementation of the proposed system model. Finally, Section 6 concludes the research work presented in this paper and identifies directions for future work.

2. Background

Threat intelligence is described by the National Institute of Standards and Technology (NIST) as "threat information that has been aggregated, transformed, analyzed, interpreted, or supplemented to provide the appropriate context for decision-making processes." A key feature to emphasize is that CTI information typically goes through aggregate, convert, analyze, interpret, and enrich phases to be rendered functional and avoid remaining merely information (Aljuhami and Bamasoud, 2021). The second crucial factor to emphasize is decision-making. Intelligence gathering doesn't always result in a conclusion, but it frequently leads to one about how to respond to an attack attempt, safeguard someone's assets, or reduce an incident. The essential criteria for any cyber threat protection and warning system should include CTI modeling and infrastructure node threat type identification. Many research papers have been published on this subject in recent years because of

the interest that it has received from the academic and business sectors in the disciplines of cybersecurity and data mining (Gao et al., 2020).

The intelligence lifecycle includes four stages: direction, collection, processing, and dissemination. Direction is the initial phase that determines the organization's intelligence requirements by concentrating on existing threats and prioritizing the resources to protect them. The collection phase involves gathering all the necessary information that meets the intelligence requirements from the relevant sources. In the processing phase, the collected data is synthesized and analyzed using structured analytical techniques, then turned into a report for relevant stakeholders. The last phase is dissemination, in which an intelligence report is distributed to pertinent parties so they can use it to carry out decisions based on the information gathered during the earlier procedure (Ainslie et al., 2023).

Threat intelligence platforms can be shared or exchanged. The cooperative sharing of CTI among organizations is a process that benefits all involved organizations. Because of the various ways threats might influence the infrastructure-building components of an organization (for instance, phishing and spear-phishing assaults, eavesdropping attacks, man-in-the-middle attacks, etc.) (Bandara et al., 2021), CTI sharing is difficult in practice. Although an exchange platform for CTI could potentially increase societal security, potential participants are frequently hesitant to share their CTI and instead choose to consume it exclusively, at least in voluntary-based systems. Such conduct negates the concept of knowledge sharing. Governments, on the other hand, are pressuring businesses and operators to report cyberattacks and their results; failing to do so might result in consequences for the operators. Participants are typically discouraged from sharing information willingly by obligations and penalties and will only disclose and report what is legally necessary (Riesco et al., 2020).

CTI can potentially develop into an unmanageable alarm stream (Aljuhami and Bamasoud, 2021). Current SOC operations have a history of considerable information overload, false positives, and false negatives. As a result of these issues, security analysts may experience alert fatigue, which can lead to severe burnout and mental health issues (Samtani et al., 2020). In response to these problems, AI and machine learning have begun to show promise in filtering out noise, reducing alert fatigue, and improving outcomes (Samtani et al., 2020). The effective sharing of CTI, which allows the development of multi-layer automated tools with advanced and effective defensive capabilities, is the foundation of cyber-threat detection and prevention. These tools consistently analyze large amounts of homogenous information relating to attackers' tactics, techniques, and procedures (TTPs), IoCs, etc. (Ramsdale et al., 2020). The US National Vulnerabilities Database (NVD) and the Common Vulnerability Scoring System (CVSS) are both used in the community effort to provide public databases of software vulnerabilities. The NVD uses CVSS to keep track of, score, record, and communicate information regarding vulnerabilities found and reported by businesses and individuals. For practitioners, using a scoring system approved by the community of cybersecurity experts is useful since the numerical index gives an idea of the severity and the effect of vulnerability on the infrastructure. We should also mention that the MITRE Corporation, in cooperation with NIST and the NVD, maintains the Common Vulnerabilities and Exposures (CVE) database (Czekster et al., 2022). MITRE ATT&CK is an attack pattern knowledge base for cyber actors, methods, and tactics. It is an open-source repository that the scientific community and enterprises may use to exploit knowledge. It also provides threat feeds in Structured Threat Information Exchange (STIX) format for use in business security equipment. This framework offers guidance on how to lessen attacks. With the assistance of the scientific community, it updates its database constantly (Conti et al., 2018). Organizations work together to establish defensive actions against sophisticated attack vectors to improve the detection and prevention of cyber risks by exchanging knowledge about dangers like IoCs (Gao et al., 2020).

Forensic data files describe IoCs related to the investigation of malicious activity on a network or device. Examples include an attacker's IP address, a phishing campaign's domain name, or the hash of a malware file. Enabling technologies are used to gather, investigate, analyze, and perhaps exchange digital evidence from various digital data sources to prevent the same security issue from occurring elsewhere.

Several tools have been developed in the last few years to gather, store, and interchange the many CTI formats. Honorable mentions include ThreatConnect, IBM's X-Force, Facebook's ThreatExchange, Collaborative Research into Threats (CRITS), the MISP, CTX/Soltra Edge, and Collective Intelligence Frameworks (CIF). Further tools are available that concentrate largely on parsing and analyzing CTI. For instance, ActorTrackr is an application that links and maintains data about APT actors and information about the actual exploits those attackers have used. It does not automate the data input process; instead, it depends on users and a few repositories. The analysis tool Autometer only supports atomic signs, such as IP addresses, hashes, and URLs.

Google's APT Groups and Operations and Malware Search Engine not only retrieves relevant links but also retrieves threat information matching supplied keywords from certain defined sources. Cacador, Forager, and Jager are tools that collect indications from textual reports about incidents; they do not support the most crucial high-level TTPs that we want to extract from text (Ghazi et al., 2018). However, the security measures now in place are not enough to stop the continuously changing attacks

that attackers can launch because of their growing expertise (Varatharaj et al., 2021).

Machine learning (ML) and data mining techniques are increasingly being used because of their effectiveness in malware analysis (both static and dynamic analysis) and network anomaly detection. In addition to the techniques that cyber defenders use to stop or detect cyber-attacks, other mechanisms could fool the attackers, such as the use of honeypots. Overall, a composite of these techniques would be needed to give security analysts and practitioners the most recent information (Conti et al., 2018). Popular deep learning methods are increasingly being used in machine learning applications. Convolutional neural networks (CNNs) are used to identify hacker forum postings that may contain helpful CTI, and their performance is comparable to that of more traditional techniques like support vector machines (SVMs) (Varatharaj et al., 2021). Barik et al. (2022) proposed a framework that examines network packets in real time using deep learning techniques, including the CNN model, which provides the highest accuracy of 98.64% with a precision of 98%. The study demonstrates the effectiveness of deep learning techniques in detecting cybersecurity threats. Barik et al. (2024) developed a framework to detect cyberattacks on weighted conditional stepwise adversarial networks by employing the particle swarm optimization algorithm and support vector classifier. The suggested framework achieved an accuracy of 99.36% in normal traffic and 98.55% in malicious traffic.

3. Literature review

Varatharaj et al. (2021) developed a CTI model to help SOCs monitor and assess an organization's security in real time. Their CTI system uses a multi-layer strategy. Each layer's output delivers findings that are sent to the following layer. Three layers comprise the system, which are constructed above the Security Onion. Data from offline and internet sources is input into Layer 1. After receiving the input from Layer 1, Layer 2 implements two components: filtering and data reduction. Layer 3 then provides a thorough report. The first layer's input is made up of financial datasets, which are used to detect financial fraud. Machine learning is used to train models.

Karatisoglou (2022) presented the BRIDGE CTI framework's core components. The author suggested the use of BRIDGE, a tool that takes advantage of the STIX standard, employs blockchain technology, and automatically transforms the necessary intelligence into the form researchers and other professionals require. BRIDGE streamlines the flow of intelligence between CTI and cybersecurity professionals. The CTI team supplies the system's input, and the system then generates CTI reports that are stored in the database. For the system pipeline's automation to run correctly, the data inside the block must adhere to a single CTI standard. As a result, the STIX 2.0 language standard was chosen as the report's format. Threat intelligence providers and consumers are both permitted to communicate with the blockchain. A new block is added to the blockchain every time the CTI team needs to store a threat intelligence report containing the report's contents.

Papanikolaou et al. (2023) proposed the CTIMP system. This system focuses on providing a variety of intelligent mechanisms for monitoring data integrity, reporting new threats, detecting and recording security incidents, and responding instantly to automated processes. It primarily focuses on meeting the critical information infrastructure needs of different organizations. The system first focuses on the timely identification of events using automated, thorough log analysis. The system administrator's visualization console shows any warnings or events. This interface minimizes the probability of reaching the wrong conclusions by providing rapid and accurate simultaneous analysis of a very large number of security occurrences in the supervised business network.

The foundation of CTIMP's predictability updates and upgrades is the collection of cyber-threat data from reputable open-access sources, such as pre-built IOCs created by security professionals, etc., that are filtered and correlated with the express goal of supporting infrastructure. Table 1 analyzes the current research work on CTI systems and their underlying components.

As indicated in Table 1, different techniques have been used to develop models that help the SOC team improve their work and the correlations used in the SIEM based on IoC from different resources. Varatharaj et al. (2021) used ML to detect fraud. On the other hand, Karatisoglou (2022) found that the blockchain is more accurate for storing STIX reports to deliver the IoC to SIEM.

Papanikolaou et al. (2023) suggested an architecture enabling expanded collaboration, the intelligent choice of sophisticated coping mechanisms, and automatic self-healing procedures for dealing with threats to provide administrators with the situational awareness they need.

Dekker and Alevizos (2024) proposed threat intelligence based on security assessment and a decision-making strategy. The study utilizes causal graphs to reduce uncertainty and considers tactics, techniques, and procedures to improve the predictability of adversary behavior. The study incorporates uncertainty into assessment analysis and in evaluating the effectiveness of cybersecurity control, which would help chief information security officers (CISOs) to protect resources from cyber risks.

Karatisoglou's (2022) model manually adds reports to the blockchain and suffers from the limited resources of CTI reports. Whereas the proposed model of Papanikolaou et al. (2023) can be strengthened by employing more sophisticated anomaly detection methods. For intelligent processes, finding the best ways to represent various forms of structured or unstructured data to give self-

healing rules, the system's structure should be studied to determine how it may be used with data transformation methods.

A critical analysis identified an unfilled gap, which is the use of online CTI resources that cover all recent major attack vectors like malicious URLs, phishing URLs, spam, bot IPs, social media, and websites, in addition to machine learning algorithms. This could be achieved by integrating the CTI platform with a SIEM solution along with ML algorithms to find IoC, which will be discussed in this paper.

4. Proposed system

The proposed architecture's subsystems and mechanisms are described in detail below. Fig. 1 shows an overview of the system. Online CTI Platform. Almost 6,000 businesses use the open-source CTI-sharing platform known as MISP. As a CTI-sharing platform with cutting-edge technology, it strives to serve a wide group of security information professionals with varying demands and goals (Stojkovski et al., 2021). The proposed system employs Sysmon (System Monitor) event logs, which are valuable for detecting suspicious activity on Windows-based systems (Chen et al., 2020). Sysmon log records a wide range of events that occur on a computer. These contain information such as unique process identification, child processes, command-line prompts, network information such as IP addresses and ports, and more, depending on the event. A popular ML method for classification and other learning tasks is logistic regression, which is used in this study. A logistic regression approach is used to solve classification issues. It is a predictive analysis that depicts data and shows how variables interact. Logistic regression is applied to an input variable X, and the outcome variable Y is either 1 (yes) or 0 (no). One of the most crucial aspects of this study is the financial dataset, as it aids in creating an accurate machine-learning model. The financial dataset was obtained via Kaggle. This dataset was extremely useful in detecting fraud (Varatharaj et al., 2021). The accuracy rate of a machine-learning model will rise as more datasets are acquired. The financial dataset we obtained from Kaggle is made up of 31 characteristics, 28 of which have been made anonymous and are denoted by the labels V1 through V28. The final three elements are timing, transaction amount, and whether the transaction was fraudulent. The dataset has no missing values. Additionally, Jupiter Notebook is used, which is an original web application for producing and sharing computational documents. It provides a straightforward, simplified, document-centric interface.

Table 1: CTI systems analysis

Study/model	Main components	Availability	Strengths	Limitations
Varatharaj et al. (2021) / Oracle VM + Security Onion	Financial datasets, ELK stack, Python, ML classifiers (LR, LDA, KNN, CART, SVM, RF)	All tools freely available	Wide range of ML algorithms; achieved multi-level accuracy	Needs more comprehensive datasets
Karatisoglou (2022) / BRIDGE	CTI reports, STIX 2.0, blockchain, SIEM, Sigma rules	–	High accuracy; effective for SOC teams	Manual addition of reports; limited CTI resources
Papanikolaou et al. (2023) / CTIMP	OSSEC, decoders, MISP, dependency mapper, SIEM, self-healing rules	–	Real-time monitoring; strong active protection	Needs more advanced anomaly detection; requires further study on data transformation

OpenSearch is an open-source search and analytics suite used in a wide range of applications, such as real-time monitoring, log analytics, and website searches. It is also a highly scalable system for delivering quick access and response to massive amounts of data. It includes an integrated visualization tool, OpenSearch Dashboards, that allows users to easily examine their data. OpenSearch is based on the Apache Lucene search library and provides various search and analytics features such as k-nearest neighbors (KNN) search, SQL, anomaly detection, machine learning commons, trace analytics, full-text search, and more. The proposed model uses Wazuh as an SIEM solution. Wazuh is a free, open-source platform for threat detection and security monitoring in accordance with preset security standards.

It may be used to gather and analyze data in real time and monitor endpoints like PCs, laptops, servers, or network equipment like firewalls and routers. Wazuh offers the following features: security analytics, vulnerability detector, file integrity monitoring, log data analysis, and intrusion detection. The datasets used in this study mostly consist of financial data. The data is transferred to the SOC, where every activity is monitored. The major functions of the SOC are analysis, detection, and response to cybersecurity issues. The SOC offers services including data collection and analysis, threat intelligence, digital forensics, and vulnerability management.

The proposed system makes use of CTI. The system consists of Wazuh integrated with the MISP CTI feed using an API as a CTI online resource. This phase entry explains how to automate Wazuh to use the MISP API (Fig. 2). Analysts do not have to manually correlate a Wazuh alert with MISP to detect potential IoCs, which is a huge advantage of this automation. For instance, Wazuh will strip out the domain from a DNS query made by an endpoint and pass it to MISP to see if MISP has this information in its threat feeds. If the value is present

in MISP, MISP will reply with the event ID and further information about the IoC. When MISP responds positively, a high-severity Wazuh alert is generated, and an IoC-detected message is displayed. Instead of manually looking for IoCs, the SOC team is immediately alerted to expanded data. A report will be sent via email in real time and will contain the alert information.

5. Experimental setup and results

For the suggested system's development, all instances are hosted on Linode, which is a cloud virtualization platform. Three Linux Ubuntu 22.04 LTS machines with x86_64 architecture are used, including hosting Wazuh Manager and MISP Server. Additionally, a Windows 10 Pro machine with x86_64 architecture is used as an agent in the SIEM to test the operation.

The client side carries out the Sysmon installation and configuration. Sysmon may be installed and configured in a variety of ways. Sysmon is installed in this project with the option to include hashing algorithms. Fig. 3 depicts the setup configuration. Because the standard Sysmon setup does not permit deactivating events, further configuration is necessary to activate or disable events. As a result, Sysmon is configured with advanced filter settings in the config.xml file using XML.

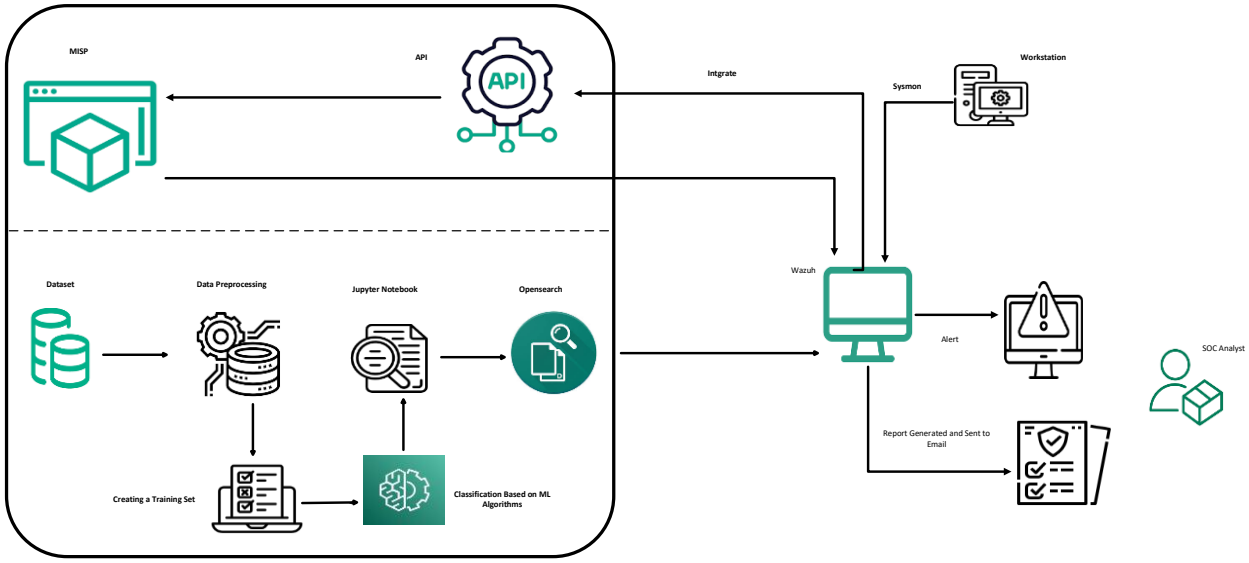

Fig. 1: Proposed model

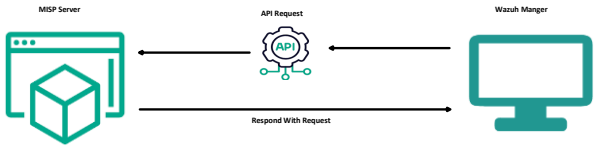

Fig. 2: MISP API workflow

```
C:\Users\[REDACTED]\Sysmon\Sysmon.exe -c

System Monitor v14.14 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2023 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard
Sysinternals - www.sysinternals.com

Current configuration:
 - Service name:         Sysmon
 - Driver name:          SysmonDrv
 - Config file:          C:\Users\[REDACTED]\Sysmon\\
 - Config hash:          SHA256:[REDACTED]

 - HashingAlgorithms:    SHA1, MD5, SHA256, IMPHASH
 - Network connection:   enabled
 - Archive Directory:    \Sysmon\\
 - Image loading:        enabled
 - CRL checking:         disabled
 - DNS lookup:           disabled
```

Fig. 3: Sysmon's basic configuration

Because of the large number of events captured, event logs are quite noisy. Not all occurrences are necessarily of interest to this research. The critical duty is to monitor the activity of malicious programs that gain access to the network and attempt to remain there. In such cases, it is vital to monitor the activity of the processes and the parent processes that produce them. The XML configuration file is modified to filter events generated by the Sysmon log. The filter tags are listed in the configuration file's event filtering node. Many events, such as Software Protection Service, Windows Update events, and other typical events, are removed from the event log in this project to filter the host system's output and limit the number of logs to gather. A simple configuration will be added to the Wazuh agent file (ossec.conf) to ship the event logs from the client to the Wazuh Manager server.

5.1. Wazuh and MISP integration

This integration will function with a custom Python script to tell the Wazuh Manager how to make the API call to MISP. A few actions must be taken for the request to be successful. This script performs the API request to MISP. Certain settings are added to the ossec.conf file as an integration block to run this custom misp.py script every time the Wazuh Manager needs it. Whenever an event is detected in the manager, it is decoded first. Predecoding is a basic procedure that extracts only static information from well-known fields of an event. Decoding is used to extract non-static data,

making it easier to develop rules for it. At least one rule in the hierarchy must be associated with the event's decoder to trigger an alert. If the rules are met, an alert is generated. Finally, one must establish a rule so that Wazuh generates an alert if MISP returns a positive hit. Whenever the endpoint issues a DNS request for the domain, resulting in Sysmon Event 22, the manager will then send an API call to MISP, and MISP will answer with a positive match, resulting in an alert.

5.2. Machine learning training

Logistic regression can be used in fraud detection to build a model that accurately classifies transactions as fraudulent or non-fraudulent based on their characteristics. The model is trained on a labeled dataset of historical transactions, where each transaction is labeled as either fraudulent or non-fraudulent. The model can then predict the likelihood that a new transaction is fraudulent based on its characteristics and trigger an alert if the probability of fraud exceeds a certain threshold.

Overall, logistic regression is a useful tool in fraud detection because it is a relatively simple and interpretable algorithm that can effectively capture complex relationships between transaction characteristics and fraud risk.

During the pre-processing stage, 99.6% of transactions in the dataset were non-fraudulent, while 0.4% were fraudulent. Fig. 4 shows that most fraudulent transactions occur in the categories of Cash Out and Transfer, which is quite logical, and the mean number of fraudulent transactions is much lower than the mean number of regular transactions. We noticed that in threat plots, the data is skewed. The random undersampling technique is used to train the dataset with a balanced class distribution to achieve high performance and fit a robust decision boundary. This strategy is especially useful for classification algorithms working with unbalanced datasets since it can help reduce bias towards the majority class and enhance model performance overall. Undersampling ensures that the algorithm obtains an equal representation of both classes by limiting the number of instances in the majority class, enabling it to learn from both and generate more accurate predictions. All fraudulent transactions were tallied, and then, at random, we chose the same number of non-fraudulent transactions and joined them to generate the balance training sample. Fig. 5 shows that the skewness is 0.18. The dataset was split 70/30 for training the model and testing the performance of the algorithms. From the model evaluation (or confusion matrix) presented in Fig. 6, we evaluate the following:

$$Accuracy = (TP + TN)/Total$$
$$Precision = TP/(TP + FP)$$
$$Recall = TP/(TP + FN)$$

The study is particularly concerned with the recall number to identify fraudulent transactions. Due to data imbalances, many observations can be forecast as false negatives, which means that we anticipate a regular transaction, but it is fraudulent. This will be captured by recall. Attempting to improve recall usually results in a reduction in precision. However, in this situation, predicting that a transaction is fraudulent and it turns out not to be fraudulent is not a major issue in comparison to the reverse. As a result, many evaluations will be dependent on recall values. As part of the process of validating a model's accuracy and effectiveness, it is important to identify which features have the greatest impact on its performance. By analyzing feature importance, we can ensure that the model is not assigning undue importance to irrelevant or insignificant variables.

Feature importance is determined by evaluating each feature's contribution to improving the model's performance and can be visualized through a scoring system. This score represents the frequency with which the feature has been shown to enhance the model's predictive capabilities. As a result of this analysis, we can gain insight into the most critical variables and optimize our model accordingly.

Another dataset is used to train two different models, K-Nearest Neighbors (KNN) and Random Forest (RF) algorithms. Both models were assessed using several metrics, including accuracy, F1-score, ROC-AUC, and confusion matrix analysis. A stratified train-test split was used, allocating 80% of the dataset for training and 20% for testing. Cross-validation was also applied with 5 folds to validate model generalizability. The dataset is available at https://www.kaggle.com/datasets/ismetsemedov/transactions. The original dataset has delivered more than 1.1M synthetic transactions. A balanced subset of the original data was sampled to overcome the issue of class imbalance and to avoid biased model training. This subset consists of 100,000 fraudulent purchases (is_fraud = 1) and 100,000 non-fraud transactions (is_fraud = 0). The dataset consists of the following types of features: identifiers, merchant information, geolocation and currency, device and access information, transaction context, velocity features, and label. In the data preprocessing stage, column filtering is used, and records containing NAs were excluded. Label encoding is used to convert categorical columns into numeric strings. Feature scaling is used, where StandardScaler is used for standardizing the distribution of numerical features. We randomly shuffled the combined balanced dataset containing 200,000 transactions after preprocessing, dividing it into 80% training and 20% testing sets for model development.

The KNN model achieved a test accuracy of 89.20% and a cross-validation accuracy of 89.05%. The F1-score was 0.8936, and the ROC-AUC score reached 0.9430, indicating strong discriminatory power, and the model is highly capable of distinguishing between the two classes. For both classes, precision is around 0.89–0.90, meaning that when the model predicts a label, it's correct about 89–90% of the time. Based on the recall

results, the model is slightly better at identifying True cases (90%) compared to False cases (88%). The results of F1-Score show a balanced score across classes at 0.89, indicating a strong harmonic mean of precision and recall. The classification report is summarized in Table 2.

Table 2: KNN classification result

Class	Precision	Recall	F1-score	Support
False	0.90	0.88	0.89	19,904
True	0.89	0.90	0.89	20,096
Accuracy			0.892	40,000
Macro avg	0.89	0.89	0.89	40,000
Weighted avg	0.89	0.89	0.89	40,000

The confusion matrix in Table 3 shows that 2,350 false positives and 1,968 false negatives were recorded, revealing a moderate rate of misclassification.

Table 3: KNN confusion matrix

	Predicted false	Predicted true
Actual false	17,554	2,350
Actual true	1,968	18,128

The random forest model achieved a test accuracy of 95.86% and a cross-validation accuracy of 95.76%. The F1-score was 0.9586, and the ROC-AUC score was 0.9923, indicating excellent performance in distinguishing between the classes. These numbers reflect exceptional model performance, especially in terms of discriminative ability, as evidenced by the near-perfect ROC-AUC score. Table 4 summarizes the classification report.

Table 4: Random forest classification report

Class	Precision	Recall	F1-score	Support
False	0.95	0.96	0.96	19,904
True	0.96	0.95	0.96	20,096
Accuracy			0.9586	40,000
Macro avg	0.96	0.96	0.96	40,000
Weighted avg	0.96	0.96	0.96	40,000

The confusion matrix in Table 5 demonstrates a reduced number of misclassifications, with 722 false positives and 935 false negatives, highlighting the robustness of the Random Forest classifier.

Table 5: Random forest confusion matrix

	Predicted false	Predicted true
Actual false	19,182	722
Actual true	935	19,161

Comparing random forest results with KNN, Random Forest clearly outperforms KNN. It has higher accuracy and F1-score, better ROC-AUC, and Lower misclassification rates.

To connect the Python code algorithm to the SIEM, OpenSearch was installed and configured with Wazuh, a search and analytics plugin, to allow easy access to a Python Jupyter Notebook. Wazuh provides additional functionalities such as security features, monitoring, and analytics for OpenSearch, enhancing its capabilities for handling large datasets and performing advanced searches.

5.3. The report

To send the report via email, we configure the email address and password on the Wazuh Server, as shown in Fig. 7. Next, we add the email configuration in /var/ossec/etc/ossec.conf allowing Wazuh to send the alert report to the email in real time, as shown in Fig. 8. Finally, Wazuh's manager sends a notification containing information about the alert.

The suggested model uses MISP and ML to create CTI, identify and address cyber threats, and gather data and IoC. The financial dataset was obtained via Kaggle. Table 6 shows the results compared to Varatharaj et al. (2021); the logistic regression accuracy of the proposed system is 98.83%, which is higher than Varatharaj et al. (2021).

On the other hand, Karatisoglou's (2022) model took 0.01 sec to fetch data, which is faster than the proposed model by 0.05 sec; this is due to the integration of the logistic regression algorithm in the proposed system. Karatisoglou's (2022) model used blockchain technology and does not employ machine learning algorithms.

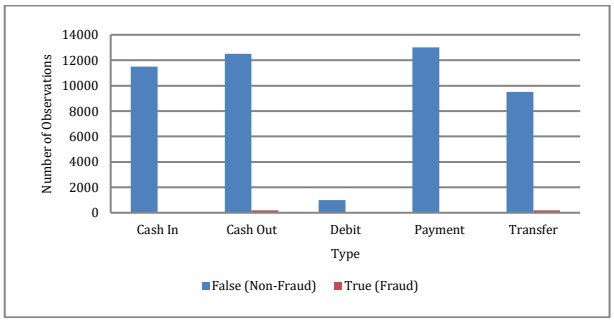

Fig. 4: Number of transactions per type and amount

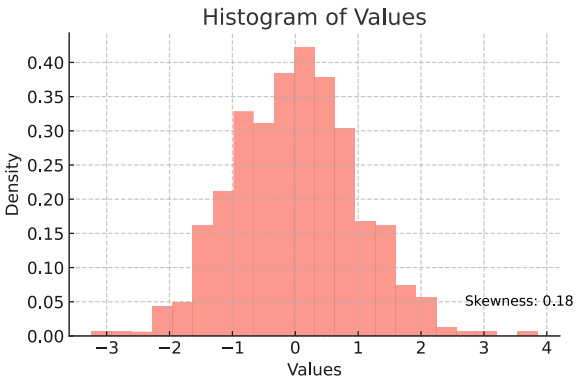

Fig. 5: Data sample skewness

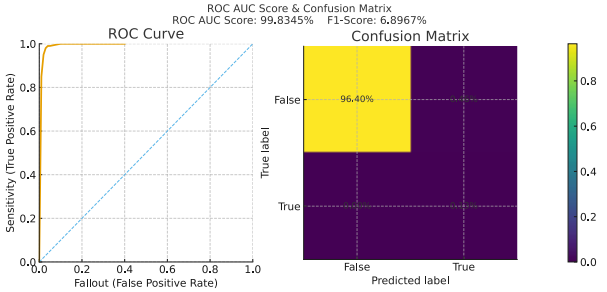

Fig. 6: Confusion matrix

```
[root@wazuh-server wazuh-user]# echo [smtp.gmail.com]:587 [REDACTED]@[REDACTED].com:[REDACTED] > /etc/postfix/sasl_passwd

[root@wazuh-server wazuh-user]# postmap /etc/postfix/sasl_passwd

[root@wazuh-server wazuh-user]# chmod 400 /etc/postfix/sasl_passwd
```

Fig. 7: Email configuration command line on Wazuh server

```
<ossec_config>
<global>
<jsonout_output>yes</jsonout_output>
<alerts_log>yes</alerts_log>
<logall>no</logall>
<logall_json>no</logall_json>
<email_notification>yes</email_notification>
<smtp_server>localhost</smtp_server>
<email_from>[REDACTED]@[REDACTED].com</email_from>
<email_to>[REDACTED]@[REDACTED].ae</email_to>
<email_maxperhour>120</email_maxperhour>
<email_log_source>alerts.log</email_log_source>
<agents_disconnection_time>10m</agents_disconnection_time>
<agents_disconnection_alert_time>0</agents_disconnection_alert_time>
</global>

<alerts>
<log_alert_level>3</log_alert_level>
```

Fig. 8: Ossec.conf file configuration to send the report via email

Table 6: Comparison of proposed system with previous work

Study/model	Key methods	Previous accuracy / results	Proposed system results	Notes
Varatharaj et al. (2021)	Shallow and deep learning (LR, LDA, KNN, CART, SVM, RF, Isolation Forest, LOF)	LR: 97.0%, LDA: 96.6%, KNN: 95.0%, CART: 90.0%, SVM: 97.0%, RF: 96.0%, Isolation Forest: 99.8%, LOF: 99.7%	Logistic regression: 98.83%; Random forest: 95.86% (test), 95.76% (CV), F1 = 0.9586, ROC-AUC = 0.9923	Dashboard: Kibana (previous); Wazuh (proposed)
Karatisoglou (2022) / BRIDGE	Blockchain-based CTI, STIX 2.0, SIEM	Query fetch: 0.01 sec	Query fetch: 0.06 sec	Proposed model slower due to ML integration; BRIDGE does not use ML

6. Conclusion and future work

Cybersecurity is extremely significant in the field of information and communication technology. Cyber threat intelligence is becoming increasingly prevalent. Various risks have emerged since the introduction of information technology. These threats are extremely well-established in the digital world. According to the findings of cybersecurity experts and researchers, cyber threats are rapidly increasing. CTI is necessary for a SOC environment because of these increasing threats. This research developed a real-time cyber threat intelligence framework. The proposed framework is built on Linode and integrates the MISP with SIEM to collect indicators of compromise (IoC) feeds. Additionally, the ML model was trained using a logistic regression algorithm to incorporate components such as filtering and cutting the data. For the evaluation of performance, we performed a performance comparison using a financial dataset. The experimental results show that the suggested framework outperformed the current existing model in terms of accurate classifications. In the future, the researchers intend to collect additional datasets and expand the system to include more features that can be integrated with CTI, such as The Hive and ChatGPT, ensuring that the built-in CTI system will be more valuable now and in the future.

The scalability and performance of the proposed model in larger SOC environments depend on several factors. Large SOCs process a significantly higher volume of logs, alerts, and threat indicators (IoCs). Therefore, scaling MISP through its API and distributed synchronization features to federate with multiple MISP instances can support the scalability of the proposed model. However, correlating internal alerts with MISP IoCs is challenging, and as data volume increases, computational demands also grow. To address this, the use of Elasticsearch can improve scalability by enabling fast searches and correlations, while storing frequently matched indicators can reduce repetitive correlation costs. Finally, integration with Security Orchestration, Automation, and Response (SOAR) platforms for automated responses further enhances the scalability of the proposed model.

List of abbreviations

CIF	Collective intelligence frameworks
CRITS	Collaborative research into threats
CTI	Cyber threat intelligence
CVE	Common vulnerabilities and exposures
CVSS	Common vulnerability scoring system
IDS	Intrusion detection systems
IPS	Intrusion prevention systems
KNN	K-nearest neighbors
MISP	Malware information sharing platform
ML	Machine learning
NIST	National institute of standards and technologies
NVD	National vulnerabilities database
RF	Random forest
SIEM	Security information and event management
SOAR	Security orchestration, automation, and

	response
SOC	Security operations center
STIX	Structured threat information exchange
VM	Virtual machine
ELK	Elasticsearch–Logstash–Kibana
LR	Logistic regression
LDA	Linear discriminant analysis
CART	Classification and regression tree
SVM	Support vector machine
OSSEC	Open source security (host-based intrusion detection system)

Data availability

The datasets analyzed during the current study are available in Kaggle https://www.kaggle.com/datasets/sriharshaeedala/financial-fraud-detection-dataset/data and https://www.kaggle.com/datasets/ismetsemedov/transactions.

Compliance with ethical standards

Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Ainslie S, Thompson D, Maynard S, and Ahmad A (2023). Cyber-threat intelligence for security decision-making: A review and research agenda for practice. Computers & Security, 132: 103352. https://doi.org/10.1016/j.cose.2023.103352

Aljuhami AM and Bamasoud DM (2021). Cyber threat intelligence in risk management. International Journal of Advanced Computer Science and Applications, 12(10): 156-164. https://doi.org/10.14569/IJACSA.2021.0121018

Angra S and Ahuja S (2017). Machine learning and its applications: A review. In the International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), IEEE, Chirala, India: 57-60. https://doi.org/10.1109/ICBDACI.2017.8070809

Bandara E, Liang X, Foytik P, and Shetty S (2021). Blockchain and self-sovereign identity empowered cyber threat information sharing platform. In the IEEE International Conference on Smart Computing (SMARTCOMP), IEEE, Irvine, USA: 258-263. https://doi.org/10.1109/SMARTCOMP52413.2021.00057

Barik K, Misra S, and Fernandez-Sanz L (2024). Adversarial attack detection framework based on optimized weighted conditional stepwise adversarial network. International Journal of Information Security, 23: 2353-2376. https://doi.org/10.1007/s10207-024-00844-w

Barik K, Misra S, Konar K, Fernandez-Sanz L, and Koyuncu M (2022). Cybersecurity deep: Approaches, attacks dataset, and comparative study. Applied Artificial Intelligence, 36(1): 2055399. https://doi.org/10.1080/08839514.2022.2055399

Chen CM, Syu GH, and Cai ZX (2020). Analyzing system log based on machine learning model. International Journal of Network Security, 22(6): 925-933.

Conti M, Dargahi T, and Dehghantanha A (2018). Cyber threat intelligence: Challenges and opportunities. In: Dehghantanha A, Conti M, and Dargahi T (Eds.), Cyber threat intelligence. Advances in information security, 70: 1-6. Springer, Cham, Switzerland. https://doi.org/10.1007/978-3-319-73951-9_1

Czekster RM, Metere R, and Morisset C (2022). cyberaCTIve: A STIX-based tool for cyber threat intelligence in complex models. Arxiv Preprint Arxiv:2204.03676. https://doi.org/10.48550/arXiv.2204.03676

Dekker M and Alevizos L (2024). A threat-intelligence driven methodology to incorporate uncertainty in cyber risk analysis and enhance decision-making. Security and Privacy, 7(1): e333. https://doi.org/10.1002/spy2.333

Gao Y, Li X, Peng H, Fang B, and Yu PS (2020). HinCTI: A cyber threat intelligence modeling and identification system based on heterogeneous information network. IEEE Transactions on Knowledge and Data Engineering, 34(2): 708-722. https://doi.org/10.1109/TKDE.2020.2987019

Ghazi Y, Anwar Z, Mumtaz R, Saleem S, and Tahir A (2018). A supervised machine learning based approach for automatically extracting high-level threat intelligence from unstructured sources. In the International Conference on Frontiers of Information Technology (FIT), IEEE, Islamabad, Pakistan: 129-134. https://doi.org/10.1109/FIT.2018.00030

Karatisoglou M (2022). CTI sharing optimizations and automating threat detection based on actionable intelligence. M.Sc. Thesis, University of Piraeus, University of Piraeus Institutional Repository, Piraeus, Greece.

Koloveas P, Chantzios T, Alevizopoulou S, Skiadopoulos S, and Tryfonopoulos C (2021). INTIME: A machine learning-based framework for gathering and leveraging web data to cyber-threat intelligence. Electronics, 10(7): 818. https://doi.org/10.3390/electronics10070818

Krishnapriya S and Singh S (2024). A comprehensive survey on advanced persistent threat (APT) detection techniques. Computers, Materials and Continua, 80(2): 2675-2719. https://doi.org/10.32604/cmc.2024.052447

Mittal S, Joshi A, and Finin T (2019). Cyber-all-intel: An AI for security related threat intelligence. Arxiv Preprint Arxiv:1905.02895. https://doi.org/10.48550/arXiv.1905.02895

Möller DPF (2020). Threat intelligence. In: Möller DPF (Ed.), Cybersecurity in digital transformation: Scope and applications: 29-45. Springer, Cham, Switzerland. https://doi.org/10.1007/978-3-030-60570-4_3

Papanikolaou A, Alevizopoulos A, Ilioudis C, Demertzis K, and Rantos K (2023). A cyber threat intelligence management platform for industrial environments. ArXiv Preprint ArXiv:2301.03445. https://doi.org/10.48550/arXiv.2301.03445

Preuveneers D, Joosen W, Bernal Bernabe J, and Skarmeta A (2020). Distributed security framework for reliable threat intelligence sharing. Security and Communication Networks, 2020: 8833765. https://doi.org/10.1155/2020/8833765

Ramsdale A, Shiaeles S, and Kolokotronis N (2020). A comparative analysis of cyber-threat intelligence sources, formats and languages. Electronics, 9(5): 824. https://doi.org/10.3390/electronics9050824

Riesco R, Larriva-Novo X, and Villagrá VA (2020). Cybersecurity threat intelligence knowledge exchange based on blockchain: Proposal of a new incentive model based on blockchain and Smart contracts to foster the cyber threat and risk intelligence exchange of information. Telecommunication Systems, 73: 259-288. https://doi.org/10.1007/s11235-019-00613-4

Samtani S, Kantarcioglu M, and Chen H (2020). Trailblazing the artificial intelligence for cybersecurity discipline: A multi-disciplinary research roadmap. ACM Transactions on Management Information Systems, 11(4): 17. https://doi.org/10.1145/3430360

Schlette D, Caselli M, and Pernul G (2021). A comparative study on cyber threat intelligence: The security incident response perspective. IEEE Communications Surveys & Tutorials, 23(4): 2525-2556. https://doi.org/10.1109/COMST.2021.3117338

Shaukat K, Luo S, Varadharajan V, Hameed IA, Chen S, Liu D, and Li J (2020). Performance comparison and current challenges of using machine learning techniques in cybersecurity. Energies, 13(10): 2509. https://doi.org/10.3390/en13102509

Stojkovski B, Lenzini G, Koenig V, and Rivas S (2021). What's in a cyber threat intelligence sharing platform? A mixed-methods user experience investigation of MISP. In the Proceedings of the 37th Annual Computer Security Applications Conference, ACM, New York, USA: 35-46. https://doi.org/10.1145/3485832.3488030

Varatharaj A, Rupasinghe PL, and Liyanapathirana C (2021). Development of cyber threat intelligence system in a SOC environment for real time environment. In the International Conference on Advanced Research in Computing (ICARC-2021), Sabaragamuwa University of Sri Lanka, Belihul Oya, Sri Lanka: 70-75.