Bulletin of L.N. Gumilyov Eurasian National University. Mathematics, computer science, mechanics series, 2024, Vol. 148, №3, P. 6-17.
http://bulmathmc.enu.kz, E-mail: vest_math@enu.kz

IRSTI: 20.53.17

MACHINE LEARNING ALGORITHMS IN SIEM SYSTEMS FOR ENHANCED DETECTION AND MANAGEMENT OF SECURITY EVENTS[1]

A. Nurusheva[1], A. Abdiraman[2], D. Satybaldina[3], N. Goranin[4],
[1,3] *L.N. Gumilyov Eurasian National University, Satpayev str., 2, Astana, Kazakhstan*
[2] *Astana IT University, Mangilik yel str., 55/11, Astana, Kazakhstan*
[4] *Vilnius Gediminas Technical University (VilniusTech), Sauletekio al. 11, 10223, Vilnius, Lithuania*
(E-mail: [1] asselnurusheva7@gmail.com, [2] a.s.abdiraman@gmail.com, [3] dinasaty@gmail.com,
[4] nikolaj.goranin@vgtu.lt)

Abstract. As cyber threats become increasingly sophisticated, traditional Security Information and Event Management (SIEM) systems face challenges in effectively identifying and responding to these dangers. This research presents the development of a SIEM system integrated with machine learning (ML) to enhance threat detection, anomaly identification, and automated incident response. The integration of ML allows the SIEM system to go beyond conventional rule-based approaches, enabling the detection of previously unknown threats by learning from historical data. The system employs advanced algorithms to analyze large-scale log data and network traffic, providing real-time insights and reducing false positives. Key features of this SIEM include anomaly detection, predictive analytics, and adaptive thresholds, which allow it to adjust dynamically based on contextual data. By adapting to new and evolving cyber threats, the system provides a more resilient and proactive defense against potential attacks. The results indicate that integrating machine learning into SIEM systems can offer organizations a more effective, scalable, and adaptive security solution, ensuring the protection of critical infrastructure and data in a rapidly changing digital landscape.

Keywords: cyber threats, machine learning, SIEM, information security management, incident response, critical infrastructure.

DOI: https://doi.org/10.32523/bulmathenu.2024/3.1

2000 Mathematics Subject Classification: 68M25; 68M10; 68T01

1. Introduction

The integration of Security Information and Event Management (SIEM) systems with machine learning components can significantly enhance the effectiveness of data security. As digitalization progresses in modern society, there has been a corresponding rise in cyber threats. The development of an event management and data security system with machine learning elements is highly relevant

[1] This research is funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP19175746)

and increasingly essential, as advancements in artificial intelligence offer new ways to improve both security and defense mechanisms.

Incorporating machine learning into SIEM systems not only advances academic discourse but also provides practical solutions that enhance the sustainability and responsiveness of data security measures in an ever-evolving technological landscape. The sharp increase in modern cyber threats, including malware, ransomware, and advanced persistent threats, underscores the urgent need for improved systems. Machine learning integration allows SIEM frameworks to more effectively prevent unauthorized access, quickly identify potential security breaches, and respond proactively to cyber threats.

Conventional SIEM systems primarily rely on static rules and signature-based detection mechanisms to identify threats. However, as cyber threats become increasingly sophisticated, such systems struggle to detect new and evolving attacks. Traditional methods are limited in their ability to analyze vast amounts of data and often fail to recognize emerging, unknown threats. This highlights the necessity for advanced analytical approaches capable of identifying novel and subtle cyber threats in a timely manner.

SIEM systems are critical components within Security Operations Centers (SOCs), providing essential functions for the collection, normalization, and analysis of security events gathered from various sources. These systems play a pivotal role in addressing the growing complexity of cyber threats by enabling the efficient detection and management of security incidents through comprehensive data processing and event correlation [1].

In turn machine learning (ML) offers powerful tools for analyzing large datasets and uncovering hidden patterns and anomalies. When integrated into SIEM systems, ML significantly enhances the system's ability to anomaly detection, incident prediction, automation of analysis, adaptability.

The objective of this research is to investigate and address emerging challenges in the domain of event management and data security by developing an advanced Security Information and Event Management (SIEM) framework integrated with machine learning components. The aim is to advance the state-of-the-art in event management and data security by providing a comprehensive and innovative SIEM solution that leverages machine learning to tackle the evolving and increasingly complex cyber threat landscape. A key goal of this research is to enhance the efficiency and accuracy of incident detection, response, and resolution processes. By employing machine learning algorithms, the proposed system aspires to create a self-learning framework capable of identifying previously unknown threats and responding to them in real time.

ML models can learn the typical behavior of systems and identify deviations from the norm, which may indicate the presence of security threats [2]. This allows for more dynamic and effective detection of attacks that would otherwise go unnoticed by rule-based systems. By leveraging historical data and patterns, ML can predict potential security incidents, enabling preemptive measures to be taken before an attack occurs. This predictive capability greatly improves the proactive nature of SIEM systems [3]. ML enables the automation of data analysis processes, reducing the reliance on human intervention and significantly speeding up the detection and response to threats. This leads to a more efficient handling of large volumes of security-related data. Also, one of the key strengths of ML is its ability to adapt to new and evolving threats. Unlike static systems, machine learning models can evolve in response to changing conditions, making them more effective in detecting and mitigating novel threats over time.

Incorporating machine learning into SIEM systems thus enhances their overall efficiency, making them more capable of addressing the complexities of modern cybersecurity challenges [4].

This research focuses on the application of machine learning components to enhance the effectiveness of event management and data security systems. Traditional SIEM frameworks rely on correlation rules to identify events associated with security threats. By incorporating machine learning, the proposed system enables SIEM frameworks to autonomously learn from security event data, allowing for the detection of previously unknown and emerging threats [5, 6].

2. Methods

The research area for this project focuses on the development and evaluation of a machine learning-based alert scanning system within a SIEM framework. The primary objective is to enhance detection and incident resolution capabilities through the application of machine learning techniques. The research methodology involves several key steps: first, conducting a comprehensive literature review to understand the current implementation of SIEM systems, particularly those incorporating machine learning; second, developing and deploying an automated alert scanning system; and third, testing the system using real-world cyber-attack scenarios to analyze its performance and effectiveness in detecting and responding to threats. The results from these tests will be used to refine and optimize the system's capabilities.

The data utilized in this study was gathered from multiple sources to support the development and evaluation of the SIEM framework. First, correlation rules were employed, which enable automatic actions to be triggered based on real-time events received by the ObjectServer, thereby reducing the number of events that operators need to troubleshoot in the Event Viewer. Second, Sigma rules, an open-source format designed for easy description of significant log events, were incorporated. This format is highly adaptable and can be applied to various types of log files, facilitating clear and simple rule creation. Lastly, data from the Incident Response Platform (IRP) system was collected post-implementation, including detection cases and response times, to further analyze and enhance the system's operational performance [7].

The data analysis process began with the preprocessing stage, which involved cleaning and standardizing the collected log records to facilitate easier investigation and decision-making. This step ensured that the data was normalized, transforming various log formats into a common structure that could be efficiently processed by the SIEM framework. The system was designed to generate alerts when potential threats were detected, allowing for automated responses or recommendations for manual interventions. The results produced by the framework guided the engineers in refining the system, which included updating signatures, adjusting rules, training additional machine learning models, and modifying other aspects of the system's configuration to enhance its performance.

The tools and techniques employed in this study were critical to the successful development and evaluation of the SIEM framework. Data checking involved identifying and referencing the appropriate documentation for the programs used, ensuring the accurate downloading and configuration of correlation rules and their integration into the SIEM system. Network monitoring tools, such as the ELK stack and Splunk, were employed to track network activity and identify trends that could signal potential journal alerts. These tools were also used for network activity monitoring and pattern analysis to detect possible security anomalies. Version control systems, like the ELK stack, were utilized to manage different versions of the SIEM system, ensuring that port conflicts and other potential issues were addressed [8].

Additionally, a test environment was established by acquiring the necessary system images to ensure proper operation of the framework. Finally, testing and assessment tools were applied to evaluate the effectiveness of the SIEM system when integrated with machine learning. This evaluation included the use of software testing and performance assessment methodologies to measure the system's overall viability.

The development process begins with a foundational understanding of machine learning algorithms, encompassing various types of attacks, their behaviors, signatures, and the data embedded in log files. This knowledge is then integrated into the architecture of the scanning system, enhancing its capabilities for more effective alarm detection and incident resolution.

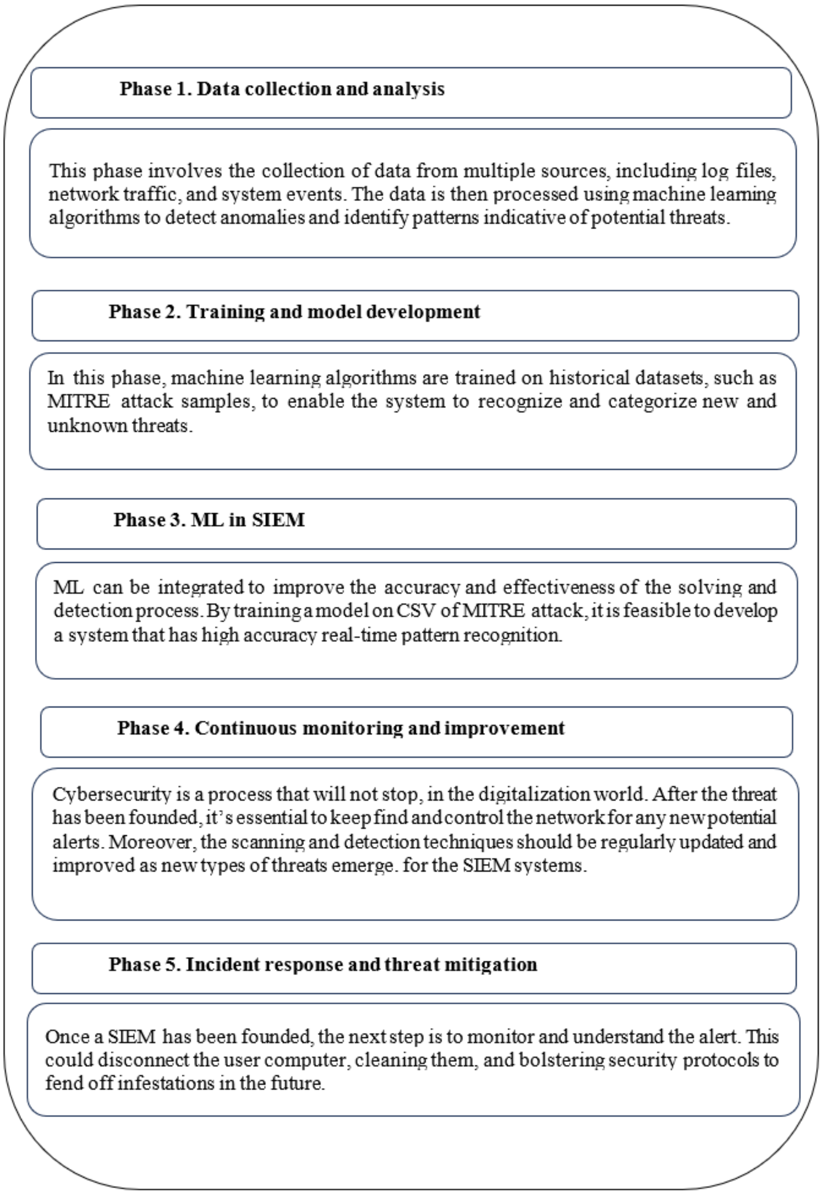
The next step involves training the machine learning algorithms on datasets derived from MITRE attack samples, enabling the system to analyze log files and detect anomalies. In this context, artificial intelligence (AI) functions as the decision-making component, allowing the system to predict and categorize potential threats with improved speed and accuracy [9].

A key feature of this approach is real-time scanning, which supports continuous monitoring and immediate response upon the detection of an alert. This real-time capability is crucial in reducing the workload on Level 1 (L1) engineers by automating much of the initial threat detection process.

Finally, the system generates response scenarios, providing actionable recommendations to users and alerting human supervisors for further intervention when necessary. The method is designed to offer clear, phased responses to identified threats, ensuring prompt and appropriate actions to mitigate risks. Reducing information security risks is most important for critical information and communication infrastructure systems [10, 11].

Figure 1 outlines the various phases and their corresponding descriptions in the development and implementation of the SIEM system integrated with machine learning. Each phase represents a critical step in the system's operation, from data analysis to threat detection and response.

Each phase represents a critical step in the system's operation, from data analysis to threat detection and response. These phases are integral to ensuring the system's ability to predict, detect, and respond to cyber threats effectively. Above is a detailed summary of the phases and their descriptions. Each phase in this table contributes to the overall robustness and efficiency of the SIEM system, ensuring that it can handle complex and evolving cybersecurity challenges.



Phase 1. Data collection and analysis

This phase involves the collection of data from multiple sources, including log files, network traffic, and system events. The data is then processed using machine learning algorithms to detect anomalies and identify patterns indicative of potential threats.

Phase 2. Training and model development

In this phase, machine learning algorithms are trained on historical datasets, such as MITRE attack samples, to enable the system to recognize and categorize new and unknown threats.

Phase 3. ML in SIEM

ML can be integrated to improve the accuracy and effectiveness of the solving and detection process. By training a model on CSV of MITRE attack, it is feasible to develop a system that has high accuracy real-time pattern recognition.

Phase 4. Continuous monitoring and improvement

Cybersecurity is a process that will not stop, in the digitalization world. After the threat has been founded, it's essential to keep find and control the network for any new potential alerts. Moreover, the scanning and detection techniques should be regularly updated and improved as new types of threats emerge. for the SIEM systems.

Phase 5. Incident response and threat mitigation

Once a SIEM has been founded, the next step is to monitor and understand the alert. This could disconnect the user computer, cleaning them, and bolstering security protocols to fend off infestations in the future.

Figure 1 – Phases and their corresponding descriptions of the SIEM system integrated with ML

3. Results and discussion

Figure 2 below presents the workflow of a Security Information and Event Management (SIEM) system integrated with machine learning (ML) to enhance threat detection and response.

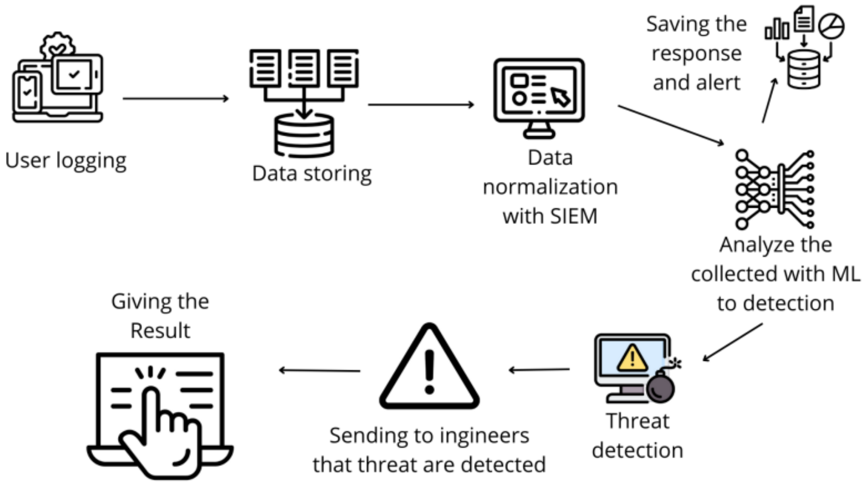

User logging
Data storing
Data normalization with SIEM
Saving the response and alert
Analyze the collected with ML to detection
Threat detection
Sending to ingineers that threat are detected
Giving the Result

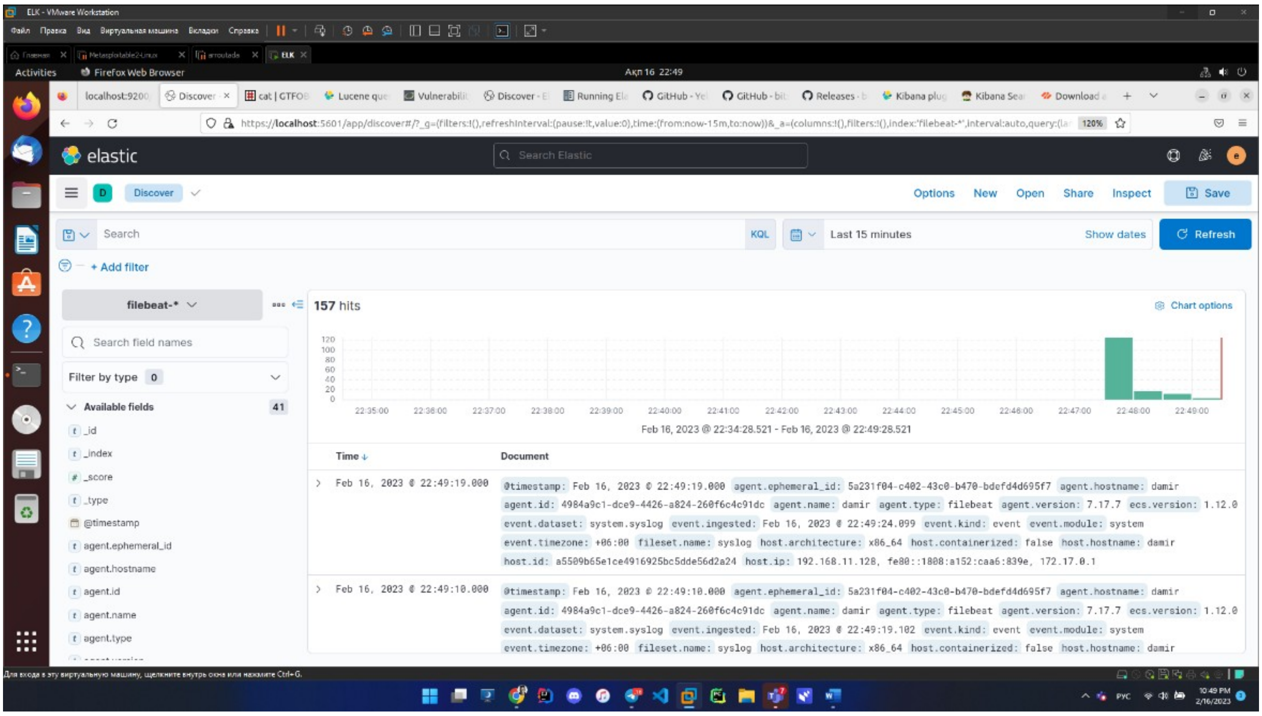
Figure 2 – Workflow of SIEM System with Machine Learning for Threat Detection

According to figure 2 we can state, that the system begins by collecting user activities and logging data from various sources such as endpoints, servers, and network devices. These logs capture information about user actions and system events. The collected data is then securely stored in a centralized database for further processing. This ensures that all relevant information is accessible for analysis and threat detection. The stored data undergoes normalization, where it is standardized into a common format by the SIEM system. This step ensures that data from different sources can be analyzed uniformly and efficiently. The normalized data is analyzed using machine learning algorithms to detect anomalies or patterns indicative of potential security threats. The ML model continuously learns from historical data to improve its accuracy in identifying emerging threats. Based on the ML analysis, the system identifies potential threats and generates alerts when suspicious activity is detected. Once a threat is detected, the system sends alerts to engineers or security teams, notifying them of the potential threat for further investigation and response. The system provides recommendations and detailed information about the detected threat. The responses, alerts, and actions taken are saved for future reference and compliance reporting. This workflow ensures continuous monitoring, proactive threat detection, and timely response, improving overall cybersecurity through the integration of machine learning and SIEM technologies.

The ELK stack we used is a combination of four tools: Elasticsearch, Logstash, Kibana, and Filebeat. These tools work together to ensure the collection, processing, and analysis of large volumes of data. Filebeat is responsible for sending logs to Logstash, which processes and filters the data before indexing it in Elasticsearch. Kibana, in turn, provides real-time visualization and analysis of the data, enabling deeper understanding and monitoring of events occurring within the system.

The server was deployed with the installation of DVWA (Damn Vulnerable Web Application), a web application designed for penetration testing. This application allows users to enhance their skills in information security by offering a wide range of web vulnerabilities of varying complexity. It is equipped with a simple graphical interface, facilitating interaction and testing. Figure 3 presents the logs stored in Elasticsearch, which are visualized through the Kibana interface.

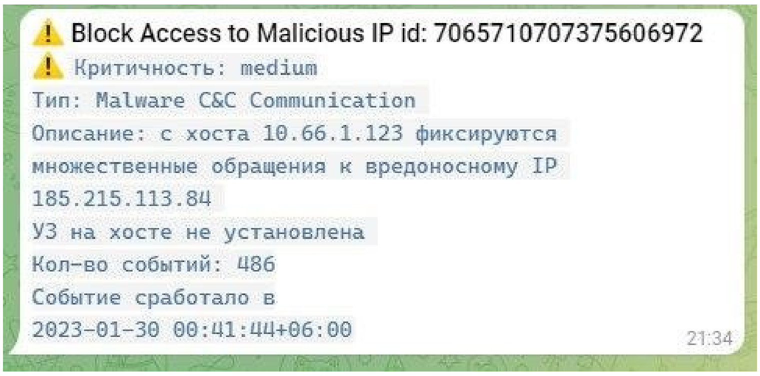
As part of this study, a ML algorithm in Python was developed to automatically send alerts to the Layer 1 of SOC via a Telegram bot. The algorithm ensures timely escalation of incidents, which allows for a prompt response to potential threats and minimizes risks to information security.



```
Time ↓                       Document
Feb 16, 2023 @ 22:49:19.000  @timestamp: Feb 16, 2023 @ 22:49:19.000 agent.ephemeral_id: 5a231f04-c402-43c0-b470-bdefd4d695f7 agent.hostname: damir
                             agent.id: 4984a9c1-dce9-4426-a824-260f6c4c91dc agent.name: damir agent.type: filebeat agent.version: 7.17.7 ecs.version: 1.12.0
                             event.dataset: system.syslog event.ingested: Feb 16, 2023 @ 22:49:24.099 event.kind: event event.module: system
                             event.timezone: +06:00 fileset.name: syslog host.architecture: x86_64 host.containerized: false host.hostname: damir
                             host.id: a5509b65e1ce4916925bc5dde56d2a24 host.ip: 192.168.11.128, fe80::1808:a152:caa6:839e, 172.17.0.1
Feb 16, 2023 @ 22:49:10.000  @timestamp: Feb 16, 2023 @ 22:49:10.000 agent.ephemeral_id: 5a231f04-c402-43c0-b470-bdefd4d695f7 agent.hostname: damir
                             agent.id: 4984a9c1-dce9-4426-a824-260f6c4c91dc agent.name: damir agent.type: filebeat agent.version: 7.17.7 ecs.version: 1.12.0
                             event.dataset: system.syslog event.ingested: Feb 16, 2023 @ 22:49:19.102 event.kind: event event.module: system
                             event.timezone: +06:00 fileset.name: syslog host.architecture: x86_64 host.containerized: false host.hostname: damir
```

Figure 3 – ELK system

Consequently, Figure 4 shows an alert sent via a Telegram bot containing key information about the nature of the detected attack.



⚠️ Block Access to Malicious IP id: 7065710707375606972
⚠️ Критичность: medium
Тип: Malware C&C Communication
Описание: с хоста 10.66.1.123 фиксируются множественные обращения к вредоносному IP 185.215.113.84
УЗ на хосте не установлена
Кол-во событий: 486
Событие сработало в 2023-01-30 00:41:44+06:00
21:34

Figure 4 – Telegram-bot for alerts from SIEM

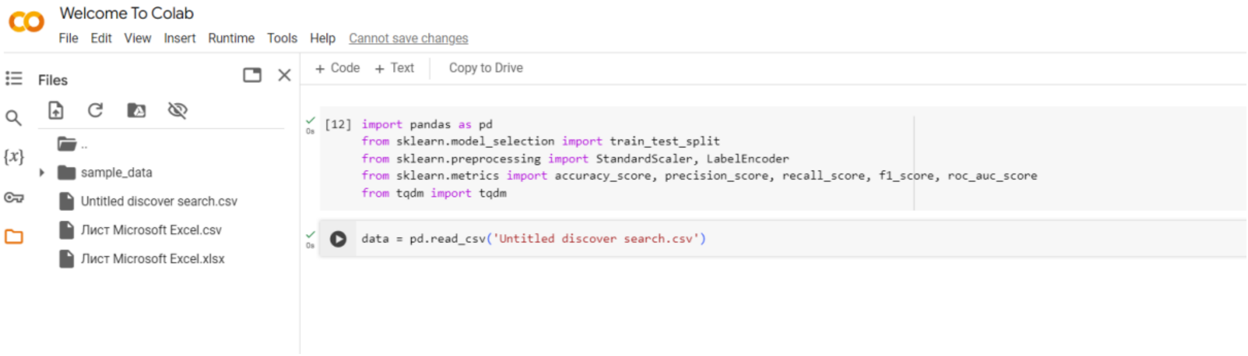
This alert in Figure above was generated based on data processed by the machine learning system and included several important parameters that facilitate accurate identification of the incident and its subsequent handling. Firstly, the message specifies the IP address from which suspicious activity was detected. This parameter is a critical indicator for identifying the source of a potential threat and allows analysts to quickly take further actions, such as blocking or analyzing network traffic originating from the given address.

Second, the date and time of the incident are included in the notification, allowing precise temporal correlation of the event and determining exactly when the attack occurred. This information is crucial for log analysis and correlating with other security events within the system.

Additionally, the alert contains information about the type of attack, which is classified based on data analysis and the machine learning model. The attack type helps analysts understand the nature of the threat, whether it is an attempt at unauthorized access, a DDoS attack, or another form of cyberattack.

Thus, the alert sent via Telegram provides a detailed message that gives the second-line support all the necessary information for timely response. This automated approach significantly reduces the time spent on incident analysis and improves the overall efficiency of the security system.

After downloading the CSV file, the next step is to train the machine learning algorithm and verify its functionality as shown in Figure 5.


CO
Welcome To Colab
File Edit View Insert Runtime Tools Help Cannot save changes
Files
+ Code + Text Copy to Drive
..
sample_data
Untitled discover search.csv
Лист Microsoft Excel.csv
Лист Microsoft Excel.xlsx
{x}
[12]
0s
0s

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score
from tqdm import tqdm
```

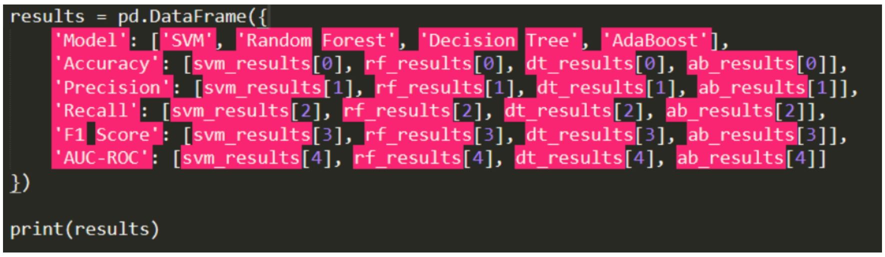
```
data = pd.read_csv('Untitled discover search.csv')
```

FIGURE 5 – Code review

Additionally, the SIEM collects logs and security events from various sources like firewalls, intrusion detection systems, etc. These logs are processed and cleaned, as SIEM data tends to be raw and unstructured. Preprocessing may involve transforming categorical data, normalizing values, and handling missing data. Features such as IP addresses, ports, timestamps, and other event attributes are extracted. This is a crucial step for any ML model to detect patterns or anomalies. Machine learning models can be trained to detect security events such as anomalies, intrusion attempts, or malicious activities. Examples of models used in cybersecurity are:

- Supervised Learning Models: Decision Trees, Random Forest, SVM, etc., are used for classifying known threats (requires labeled data).
- Unsupervised Learning Models: Anomaly detection models (e.g., Isolation Forest, DBSCAN) can be used to detect unknown threats or outliers.
- Deep Learning Models: Neural networks can be used for more complex intrusion detection.

The code shown in Figure 6 is creating a **pandas DataFrame** to store the evaluation results of several machine learning models and printing it.



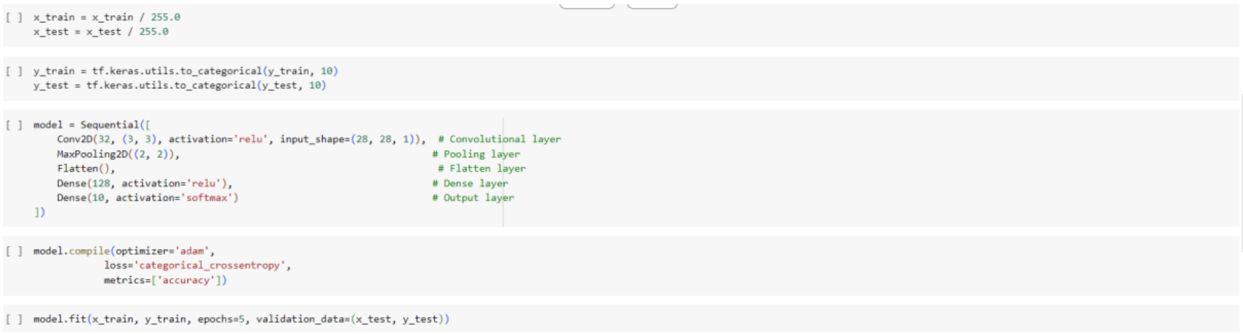
```
results = pd.DataFrame({
    'Model': ['SVM', 'Random Forest', 'Decision Tree', 'AdaBoost'],
    'Accuracy': [svm_results[0], rf_results[0], dt_results[0], ab_results[0]],
    'Precision': [svm_results[1], rf_results[1], dt_results[1], ab_results[1]],
    'Recall': [svm_results[2], rf_results[2], dt_results[2], ab_results[2]],
    'F1 Score': [svm_results[3], rf_results[3], dt_results[3], ab_results[3]],
    'AUC-ROC': [svm_results[4], rf_results[4], dt_results[4], ab_results[4]]
})

print(results)
```

FIGURE 6 – Creation of the algorithms

A DataFrame named results is created using the pd.DataFrame() function. This DataFrame contains various performance metrics for four machine learning models:

- 'SVM' (Support Vector Machine)
- 'Random Forest'
- 'Decision Tree'
- 'AdaBoost'

This code is designed to compare the performance of four different machine learning models (SVM, Random Forest, Decision Tree, and AdaBoost) using multiple evaluation metrics (Accuracy, Precision, Recall, F1 Score, AUC-ROC). The results are organized in a structured DataFrame for easy comparison and analysis.



```
[ ] x_train = x_train / 255.0
    x_test = x_test / 255.0

[ ] y_train = tf.keras.utils.to_categorical(y_train, 10)
    y_test = tf.keras.utils.to_categorical(y_test, 10)

[ ] model = Sequential([
        Conv2D(32, (3, 3), activation='relu', input_shape=(28, 28, 1)),  # Convolutional layer
        MaxPooling2D((2, 2)),                                           # Pooling layer
        Flatten(),                                                       # Flatten layer
        Dense(128, activation='relu'),                                  # Dense layer
        Dense(10, activation='softmax')                                 # Output layer
    ])

[ ] model.compile(optimizer='adam',
                  loss='categorical_crossentropy',
                  metrics=['accuracy'])

[ ] model.fit(x_train, y_train, epochs=5, validation_data=(x_test, y_test))
```

Figure 7 – Model training process

As shown in Figure 7, each class could represent a different type of security event or attack. This convolutional layer extracts features from input data (such as event logs or patterns). In SIEM, this could be used to detect spatial or temporal patterns in log data. This is used for multi-class classification, where each class corresponds to a potential threat or security event. the trained model would be used to predict potential security incidents or classify various types of security events based on real-time data logs or historical data. The accuracy metric will show how well the model can detect or classify security-related anomalies in these logs.

The model trained is a Convolutional Neural Network (CNN). The model is a CNN designed for multi-class classification. This architecture is typically used for image data, but in the context of SIEM, it could be adapted to classify security events or detect anomalies from event logs by treating the logs in a similar structured manner. With the integration of a CNN model, this data can be analyzed more efficiently to detect anomalies, classify threats, and provide real-time alerts.

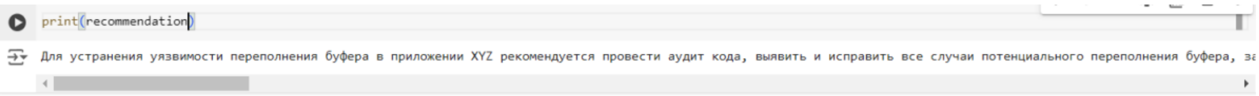
The application of CNN in SIEM systems for enhanced detection and management of security incidents is a promising approach, but it requires careful analysis of algorithm complexity and execution time. The main task of SIEM systems is to process massive data streams in real time to promptly detect threats and anomalies. CNN, as one of the powerful machine learning tools, can be applied to identify complex patterns in data, but its computational complexity is an important factor that affects practical applicability.

CNN are designed to extract local features, making them particularly suitable for analyzing time series and logs that flow into SIEM systems. One of the key factors affecting the execution time of CNN algorithms is their high computational cost, associated with the need to process large amounts of data and train the model on a large number of parameters. CNN execution time depends on the following aspects:

- **Data volume**: In SIEM systems, data streams can be enormous (e.g., network traffic, event logs, audit logs), which requires scalable solutions. The larger the volume of data, the longer the processing will take at the level of convolutions and fully connected layers.
- **Execution optimization**: To accelerate CNN execution, specialized hardware, such as Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs), is often used to enable parallel data processing. However, this requires additional resources and may increase the cost of implementing such solutions in SIEM systems.
- **Training and inference time**: In SIEM systems, both the speed of training the model and the real-time inference time is important. CNNs require considerable training time, but deployed models must quickly and efficiently process new data to detect anomalies. Optimization methods such as data preprocessing and dimensionality reduction can be applied to this end.

Despite the high complexity of Convolutional Neural Networks, they can become a key tool for improving the accuracy and efficiency of threat detection in SIEM systems. However, their successful implementation requires considering computational costs and execution time, as well as employing optimization technologies to ensure fast and effective data analysis in real time. When the CNN model detects a security incident, it triggers alerts within the SIEM system. The SIEM can:

- Generate Alerts: Immediate alerts are sent to security analysts, flagging the nature of the threat (e.g., malware, phishing attack, insider threat).
- Automated Responses: The SIEM can be configured to respond automatically to certain types of incidents as shown in Figure 8. For example, blocking a suspicious IP address or isolating a compromised device from the network.



```
print(recommendation)
```

```
Для устранения уязвимости переполнения буфера в приложении XYZ рекомендуется провести аудит кода, выявить и исправить все случаи потенциального переполнения буфера, за
```

Figure 8 – Results of integrating ML into SIEM

As shown above, system could automatically send recommendations in various cases, which minimize the human presence. By integrating a CNN model with a SIEM system, organizations can significantly enhance their threat detection capabilities. The ability of CNNs to recognize complex patterns and process large volumes of data makes them ideal for modern SIEM environments, where cybersecurity threats are growing increasingly sophisticated. This integration allows for more accurate classification of threats, real-time monitoring, and automated responses, helping organizations mitigate security risks more effectively.

For the setting up environment in this project we used Google Colab. This includes installing all the necessary libraries such as pandas, scikit-learn, and other essential packages for data manipulation and model evaluation. Integrate SIEM logs with this script, possibly via APIs or exporting data in CSV format.

We then upload our CSV file, ensuring the data is correctly imported. From there, we proceed to test the machine learning algorithm, splitting the data into training and test sets, and evaluating the model's performance by checking its accuracy and other relevant metrics. This process helps ensure that the code is functioning as expected and the algorithm is correctly identifying patterns in the data.

1. Train_test_split: used to split historical SIEM log data into training and test sets for model building and evaluation.
2. StandardScaler: Normalize numerical features in SIEM logs, such as the size of data transferred, or response time, to ensure ML algorithms perform better.
3. LabelEncoder: Convert categorical features like "event type" or "source type" into a numeric format for model training.
4. Metrics (accuracy_score, precision_score, etc.): These will help evaluate how well your ML model detects security incidents or anomalies from the logs. In an intrusion detection system, precision and recall are especially important for measuring false positives and true detection rates.
5. tqdm: This is useful for adding progress bars when processing large SIEM logs or during model training, which can take a long time.

Next Steps for Full Integration:

1. Data Ingestion: Integrate SIEM logs with this script, possibly via APIs or exporting data in CSV format.
2. Model Training: Choose or build a model that fits your use case (anomaly detection, classification, etc.).
3. Real-time Application: Once trained, the model could be integrated to work on real-time streams from SIEM, using Kafka, Logstash, or other log-streaming platforms.

We downloaded the CSV file, the next step is to train the machine algorithm and check whether it works correctly. We go to Google collab and start writing. First, we need to install all the necessary libraries and also upload our CSV file, and start testing and checking accuracy of the code.

This code is designed to compare the performance of four different machine learning models (SVM, Random Forest, Decision Tree, and AdaBoost) using multiple evaluation metrics (Accuracy, Precision, Recall, F1 Score, AUC-ROC). The results are organized in a structured DataFrame for easy comparison and analysis.

4. Conclusion

In conclusion, based on the information presented, an advanced solution is proposed for protecting against various attacks, such as phishing and cryptographic attacks, commonly faced by many organizations. To ensure the security of corporate data, it is essential to understand proper information storage methods, including the handling of passwords and files. Without such knowledge among employees, a company remains vulnerable to cyber threats, even when firewalls are in place. It is crucial to evaluate all possible protection strategies and implement necessary measures to secure sensitive data. Furthermore, in the event of an attack, rapid and accurate response is imperative, necessitating the development of an action plan to contain the threat and prevent its spread to other critical files.

The increasing sophistication and coordination of cyberattacks have led to the need for more advanced tools for protecting information systems, such as intrusion detection systems, antivirus software, and firewalls. Each of these security tools generates streams of events with varying levels of detail, and often, only by correlating events from multiple systems can an attack be accurately detected.

To further improve efficiency, the automation of incident response processes through the integration of SOAR systems and the development of custom scenarios for SIEM is recommended. Automation is becoming increasingly vital in the field of information security, playing a significant role across many areas. This approach enables continuous monitoring of threats specific to an organization.

The developed system has several key objectives, including data collection and normalization, data correlation, alert generation, visualization dashboard creation, data storage organization, search and analysis, and report generation. The system's advantages lie in its use of reliable open-source software, which eliminates the need for financial investment and increases the efficiency of SOC analysts.

Several significant milestones were achieved in the project, including the installation of ELK and TheHive on separate CentOS servers, the configuration of one CentOS server, and the deployment of DVWA on another.

Additionally, a Telegram bot was developed to display alerts, configured to send notifications for multiple brute-force attacks. The VirusTotal analyzer was integrated into TheHive and MISP databases to detect and immediately remove malicious files.

Overall, this system offers a comprehensive data processing solution, utilizing software and hardware tools capable of detecting and responding to critical incidents and events.

Acknowledgements

This research is funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP19175746).

References

1 Bhatt S. N., Manadhata P. K., Zomlot L. The operational role of security information and event management systems, IEEE Security Privacy Magazine. 2014. № 12. P. 35–41.

2 Thakur K., Kopecky S., Nuseir M., Ali L., Qiu M. An Analysis of Information Security Event Managers, IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud). 2016. Vol. 6. P. 210-215. doi: 10.1109/cscloud.2016.19.

3 Holm H. Signature based intrusion detection for zero-day attacks: (Not) a closed chapter?, 47th Hawaii International Conference on System Sciences. 2014. P. 4895-4904. doi: 10.1109/hicss.2014.600.

4 Di Sarno C., Garofalo A., Matteucci I., Vallini M. A novel security information and event management system for enhancing cyber security in a hydroelectric dam, International journal of critical infrastructure protection. 2016. Vol. 13. P. 39–51.doi: 10.1016/j.ijcip.2016.03.002.

5 Jordan M., Mitchell T. M. Machine learning: Trends, perspectives, and prospects, Science. 2015. Vol. 349. P. 255–260.

6 Zhou Zh. Machine learning. Springer nature. 2021. 459 p.
7 Naqa I. E., Murphy M. J. What is machine learning? Springer eBooks. 2015.
8 Alzubi J., Nayyar A., Kumar A. Machine learning from theory to algorithms: an overview, Journal of Physics. 2018. № 1142. P. 1-15.
9 Aljawarneh S., Aldwairi M., Yassein M. B. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, Journal of Computational Science. 2018. № 25. P. 152-160.
10 Aidynov T., Goranin N., Satybaldina D., Nurusheva A. A systematic literature review of current trends in electronic voting system protection using modern cryptography, Applied sciences. 2024. Vol. 14, № 7.1235
11 Abdiraman A., Goranin N., Balevicius S., Nurusheva A., Tumasonienė I. Application of multicriteria methods for improvement of information security metrics, Sustainability. 2023. Vol. 15, № 10.

Қауіпсіздік оқиғаларын тиімді анықтау мен басқарудағы SIEM жүйелеріндегі машиналық оқыту алгоритмдері

Ә. Нурушева [1] , Ә. Әбдіраман [2] , Д. Сатыбалдина [3] , Н. Горанин [4]
[1,3] Л.Н.Гумилев атындағы Еуразия ұлттық университеті, Сәтбаев көш., 2, Астана, Қазақстан
[2] Astana IT University, Мәңгілік ел көш., 55/11, Астана, Қазақстан
[4] Гедиминас атындағы Вильнюс техникалық университеті (VilniusTech), Sauletekio көш., 11, 10223, Вильнюс, Литва

Аңдатпа. Киберқауіптер күрделене түскен сайын, қауіпсіздік туралы ақпаратты және оқиғаларды басқарудың (SIEM) дәстүрлі жүйелері осы қауіптерді тиімді анықтауда және оларға жауап беруде қиындықтарға тап болады. Бұл зерттеу қауіпті анықтауды, аномалияларды анықтауды және автоматтандырылған оқиғаларға жауап беруді жақсарту үшін машиналық оқытумен (ML) біріктірілген SIEM жүйесінің дамуын ұсынады. ML интеграциясы SIEM жүйесіне әдеттегі ережелерге негізделген тәсілдерден шығуға және тарихи деректерден үйрену арқылы бұрын белгісіз қауіптерді анықтауға мүмкіндік береді. Жүйе журналдар мен желілік трафиктің ауқымды деректерін талдау, нақты уақыт режимінде ақпарат беру және жалған позитивтерді азайту үшін озық алгоритмдерді қолданады. Бұл SIEM негізгі мүмкіндіктеріне аномалияны анықтау, болжамды аналитика және контекстік деректер негізінде динамикалық реттеуге мүмкіндік беретін бейімделу шектері кіреді. Жаңа және дамып келе жатқан киберқауіптерге бейімделе отырып, жүйе ықтимал шабуылдарға қарсы икемді және белсенді қорғанысты қамтамасыз етеді. Нәтижелер машиналық оқытуды SIEM жүйелеріне интеграциялау ұйымдарға жылдам өзгеретін цифрлық ландшафтта маңызды инфрақұрылым мен деректерді қорғауды қамтамасыз ететін тиімдірек, масштабталатын және бейімделген қауіпсіздік шешімін ұсына алатынын көрсетеді.

Түйін сөздер: киберқауіптер, машиналық оқыту, SIEM, ақпараттық қауіпсіздікті басқару, инциденттерге жауап беру, маңызды инфрақұрылым.

Алгоритмы машинного обучения в системах SIEM для усовершенствованного обнаружения и управления событиями безопасности

А. Нурушева [1] , Ә. Әбдіраман [2] , Д. Сатыбалдина [3] , Н. Горанин [4]
[1,3] Евразийский национальный университет имени Л.Н.Гумилева, ул. Сатпаева, 2, Астана, Казахстан
[2] Astana IT University, ул. Мангилик ел, 55/11, Астана, Казахстан
[4] Вильнюсский Технический Университет им.Гедиминаса (VilniusTech), ул. Sauletekio 11, 10223, Вильнюс, Литва

Аннотация. По мере того, как киберугрозы становятся все более изощренными, традиционные системы управления информацией и событиями безопасности (SIEM) сталкиваются с трудностями в эффективном выявлении и реагировании на эти опасности. В этом исследовании представлена разработка системы SIEM, интегрированной с машинным обучением (ML) для улучшения обнаружения угроз, идентификации аномалий и автоматизированного реагирования на инциденты. Интеграция ML позволяет системе SIEM выйти за рамки традиционных подходов на основе правил, позволяя обнаруживать ранее неизвестные угрозы, обучаясь на исторических данных. Система использует передовые алгоритмы для анализа крупномасштабных данных журналов и сетевого трафика, предоставляя информацию в реальном времени и сокращая количество ложных срабатываний.

Ключевые особенности этой SIEM включают обнаружение аномалий, предиктивную аналитику и адаптивные пороговые значения, которые позволяют ей динамически подстраиваться на основе контекстных данных. Адаптируясь к новым и развивающимся киберугрозам, система обеспечивает более устойчивую и проактивную защиту от потенциальных атак. Результаты показывают, что интеграция машинного обучения в системы SIEM может предложить организациям более эффективное, масштабируемое и адаптивное решение по безопасности, обеспечивающее защиту критически важной инфраструктуры и данных в быстро меняющемся цифровом ландшафте.

Ключевые слова: киберугрозы, машинное обучение, SIEM, управление информационной безопасностью, реагирование на инциденты, критическая инфраструктура.

References

1 Bhatt S. N., Manadhata P. K., Zomlot L. The operational role of security information and event management systems, IEEE Security Privacy Magazine. 2014. № 12. P. 35–41.

2 Thakur K., Kopecky S., Nuseir M., Ali L., Qiu M. An Analysis of Information Security Event Managers, IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud). 2016. Vol. 6. P. 210-215. doi: 10.1109/cscloud.2016.19.

3 Holm H. Signature based intrusion detection for zero-day attacks: (Not) a closed chapter?, 47th Hawaii International Conference on System Sciences. 2014. P. 4895-4904. doi: 10.1109/hicss.2014.600.

4 Di Sarno C., Garofalo A., Matteucci I., Vallini M. A novel security information and event management system for enhancing cyber security in a hydroelectric dam, International journal of critical infrastructure protection. 2016. Vol. 13. P. 39–51.doi: 10.1016/j.ijcip.2016.03.002.

5 Jordan M., Mitchell T. M. Machine learning: Trends, perspectives, and prospects, Science. 2015. Vol. 349. P. 255–260.

6 Zhou Zh. Machine learning. Springer nature. 2021. 459 p.

7 Naqa I. E., Murphy M. J. What is machine learning? Springer eBooks. 2015.

8 Alzubi J., Nayyar A., Kumar A. Machine learning from theory to algorithms: an overview, Journal of Physics. 2018. № 1142. P. 1-15.

9 Aljawarneh S., Aldwairi M., Yassein M. B. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, Journal of Computational Science. 2018. № 25. P. 152-160.

10 Aidynov T., Goranin N., Satybaldina D., Nurusheva A. A systematic literature review of current trends in electronic voting system protection using modern cryptography, Applied sciences. 2024. Vol. 14, № 7.

11 Abdiraman A., Goranin N., Balevicius S., Nurusheva A., Tumasonienė I. Application of multicriteria methods for improvement of information security metrics, Sustainability. 2023. Vol. 15, № 10.

Information about authors:

Нурушева Әсел Мұратқызы – PhD, ақпараттық қауіпсіздік кафедрасының доцент м.а., Л.Н. Гумилев атындағы Еуразия ұлттық университеті, Сәтбаев к. 2, Астана, Қазақстан.

Әбдіраман Әлия Серғалиқызы - *байланыс үшін автор*, Интеллектуалды жүйелер мен киберқауіпсіздік департаментінің сеньор лекторы, Astana IT Univeristy, Мәңгілік ел көшесі, 55/11, Астана қ., Қазақстан.

Сатыбалдина Дина Жағыпарқызы - қауымдастырылған профессор, физика-математика ғылымдарының кандидаты, Л.Н. Гумилев атындағы Еуразия ұлттық университеті, Ақпараттық қауіпсіздік және криптология ғылыми-зерттеу институтының директоры, Сәтбаев к. 2, Астана қаласы, Қазақстан.

Горанин Николай - профессор, PhD, Ақпараттық жүйелер кафедрасының меңгерушісі, Гедиминас атындағы Вильнюс техникалық университеті (VilniusTech), Сәулетяке алл., 11, 10223, Вильнюс, Литва.

Nurusheva Assel Muratovna – PhD, acting associate professor at the Department of Information Security, L.N. Gumilyov Eurasian National University, Satpayev str., 2, Astana, Kazakhstan.

Abdiraman Aliya Sergalikyzy – *Corresponding author*, senior lecturer of the Department of intelligent systems and cybersecurity, Astana IT University, Mangilik yel Street, 55/11, Astana, Kazakhstan.

Satybaldina Dina Zhagyparovna – Associate Professor, Candidate of Physical and Mathematical Sciences, Director of the Research Institute of Information Security and Cryptology at the L.N. Gumilyov Eurasian National University, Satpayev 2, Astana, Kazakhstan.

Goranin Nikolaj – Professor, Associate Professor, Professor at the Department of Information Systems and the Head of the department, Vilnius Gediminas Technical University (VilniusTech), Sauletekio al. 11, 10223, Vilnius, Lithuania.

Received: 09.09.2024. Revised: 25.09.2024.
Approved: 29.09.2024. Available online: 30.09.2024.