Method for detecting unauthorized influences in the process of network interaction based on intelligent traffic analysis*

Pavlo Skladannyi[1,2,*,†], Yuliia Kostiuk[1,†], Bohdan Bebeshko[1,†], Hennadii Hulak[1,†], and Volodymyr Astapenia[1,†]

[1] *Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudriavska str., 04053 Kyiv, Ukraine*
[2] *Institute of Mathematical Machines and Systems Problems of the National Academy of Sciences of Ukraine, 42 Ac. Glushkov ave., 03680 Kyiv, Ukraine*

Abstract

Modern network infrastructure is characterized by high heterogeneity, scalability, and complexity, which complicate ensuring stable operation in the face of dynamic threats. In particular, unauthorized influences pose a special danger – hidden or undeclared actions that traditional signature analysis methods cannot detect. This article proposes a method for detecting unauthorized influences in network interactions based on intelligent traffic analysis. The problem lies in the insufficient effectiveness of traditional methods in detecting new or hidden attacks. The study aims to develop a hybrid approach that combines traffic clustering, analytical indices, and eXplainable Artificial Intelligence (XAI). Self-Organizing Maps (SOMs) are used for traffic cluster analysis, and a criticality index and derivative are used to assess the impact of features. SHAP and LIME methods are used to interpret the results. Experiments are conducted on real-world network datasets. Improvements in accuracy metrics (F1, TPR, MCC) are obtained compared to baseline models. The scientific novelty lies in the combination of SOM clustering with index-based impact assessment and explainable traffic analysis. The method is suitable for implementation in intrusion detection systems and for improving the resilience of corporate networks to atypical threats.

Keywords

explainable artificial intelligence, XAI, self-organizing map. SOM, network traffic anomaly detection, cyber influence modeling, intelligent intrusion detection, critical feature analysis, cyber threats, unsupervised learning

1. Introduction

The rapid development of digital infrastructure, the deployment of heterogeneous network architectures, and the widespread introduction of services based on cloud, telemetry, and multimedia technologies have significantly complicated information security management in the context of dynamic interaction between systems [1–3]. The problem of ensuring the stable functioning of network components in conditions of growing data volumes, traffic heterogeneity, and evolving cyber threats is becoming particularly relevant.

Classic attack detection systems, based mainly on signature analysis, demonstrate low efficiency against new or modified attack types that lack prior signatures in knowledge bases [4]. In this context, unauthorized or undeclared attacks pose a particular threat—actions that are carried out without being explicitly recorded in the system and may go unnoticed due to the absence of patterns, signs, or signatures [5, 6]. Such impacts can violate the fundamental properties of information (integrity, availability, confidentiality) and lead to hidden degradation of the information and communication infrastructure's functioning.

* *CSDP'2026: Cyber Security and Data Protection, May 7, 2026, Lviv, Ukraine*
* Corresponding author.
† These authors contributed equally.
p.skladannyi@kubg.edu.ua (P. Skladannyi); y.kostiuk@kubg.edu.ua (Y. Kostiuk); b.bebeshko@kubg.edu.ua (B. Bebeshko); h.hulak@kubg.edu.ua (H. Hulak); v.astapenia@kubg.edu.ua (V. Astapenia)
0000-0002-7775-6039 (P. Skladannyi); 0000-0001-5423-0985 (Y. Kostiuk); 0000-0001-6599-0808 (B. Bebeshko); 0000-0001-9131-9233 (H. Hulak); 0000-0003-0124-216X (V. Astapenia)

The problem is that most existing methods of protection or threat detection do not account for the real-time behavioral dynamics of traffic flows, do not adapt to new abnormal influences, and do not perform a comprehensive analysis of the traffic generation environment [1, 7]. Therefore, there is a need to develop a method capable of detecting hidden, unauthorized influences through intelligent analysis of network interactions, using complex anomaly models, behavioral profiles, and mechanisms to explain the decisions made.

The problem is that most existing methods of protection or threat detection do not account for the real-time behavioral dynamics of traffic flows, do not adapt to new abnormal influences, and do not perform a comprehensive analysis of the traffic generation environment [1, 7]. Therefore, there is a need to develop a method capable of detecting hidden, unauthorized influences through intelligent analysis of network interactions, using complex anomaly models, behavioral profiles, and mechanisms to explain the decisions made.

The purpose of this study is to develop a method for detecting unauthorized influences in network interactions through intelligent analysis of network traffic using Kohonen-type neural networks (SOM) and interpretive models, such as SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME). SHAP is an additive method for explaining model decisions based on Shapley theory, which allows a quantitative assessment of each feature's contribution to the classifier's final decision, ensuring both global and local interpretability.

LIME is a local explanation method that approximates a complex model in a small neighborhood of a single example using a simple linear model, allowing for the interpretation of decisions even in so-called "black boxes" such as deep neural networks. The proposed approach aims to enhance the accuracy of anomaly detection in uncontrolled network traffic, eliminating the need for labeled data.

The scientific novelty of the research lies in the development of a comprehensive approach to detecting unauthorized influences during network interactions by combining Kohonen's neural network (SOM) with XAI methods, specifically SHAP and LIME. The proposed model not only allows for the detection of abnormal network flows but also enables the interpretation of the reasons for classifying traffic as potentially dangerous. Another novelty is the formation of a generalized feature vector that includes transport-level characteristics, time parameters, load parameters, and traffic cluster structure features, thereby improving segmentation accuracy. In addition, the study introduces a criterion index $K(x)$ that aggregates the influence of individual features on the classification result, accounting for the model's sensitivity to input changes, formalized by the sensitivity derivative δ and weight coefficients γ. This approach ensures not only accurate traffic classification but also transparency in decision-making, which is crucial for the practical implementation of network infrastructure protection systems.

The practical significance of this work lies in its potential application to new-generation cyber defense systems, particularly SIEM platforms, industrial control systems (ICS/SCADA), the Internet of Things (IoT), and critical infrastructure [8, 9]. The research results can be utilized to develop adaptive mechanisms for early threat detection, mitigate the risks of exploiting unknown vulnerabilities, and enhance the resilience of network services against complex, targeted attacks.

In the context of this work, unauthorized influences are understood to mean not only classic attacks (e.g., DDoS, port scanning, exploits), but also complex hidden actions in network interactions that change the behavioral characteristics of traffic without explicitly violating security policies. Such influences may include abnormal interaction patterns, slow or conditionally safe scans, injection-based modifications to request contexts, or targeted traffic modeling to avoid detection. Detecting such impacts requires a deeper level of analysis than is possible with traditional signature- or threshold-based methods, which is why SOM- and XAI-based intelligent traffic analysis is used in this work [1–3]. Thus, developing a new method capable of identifying hidden, unauthorized influences in network interactions with high accuracy is a pressing task for modern cybersecurity and determines the scientific and applied value of this research.

The article presents a formalized model for detecting unauthorized influences in network interactions that combines statistical, signature-based, and neural network traffic analysis. A

hybrid SOM+XAI method with an index-based assessment of criticality, $K(x)$, is proposed, which simultaneously provides clustering of anomalous flows and explainability of decisions via SHAP/LIME. A prototype system for detecting unauthorized influences in real time has been developed and implemented, and comparative experiments on the CICIDS-2017 and UNSW-NB15 datasets have confirmed its superiority over basic IDS solutions. Additionally, ablation analysis of SOM and XAI component parameters was performed, which allowed us to justify the optimal model settings for practical application.

2. Literature overview

The relevance of detecting unauthorized influences in network interactions stems from the spread of hidden threats that are not detected by traditional signature-based protection systems. In response, current research focuses on intelligent analysis of network traffic, building behavioral profiles, and detecting anomalies without the need for labeled data.

I. Ullah and Q. Mahmoud [1] demonstrated the effectiveness of neural network models based on One-Class Neural Networks and Autoencoders for detecting rare anomalies in IoT networks. The proposed solution enables the detection of deviations from normal traffic without relying on signatures, which is crucial for identifying unauthorized influences that operate covertly.

Despite the widespread use of machine learning methods to detect traffic anomalies, many of these methods have significant limitations. For example, the One-Class SVM method exhibits limited scalability in high-dimensional spaces and is sensitive to kernel parameter selection, which complicates its application in dynamic network environments. In addition, achieving satisfactory performance often requires careful tuning, which is challenging to automate.

The Isolation Forest algorithm, which is based on isolating random samples, is fast and effective for anomaly detection tasks. However, it does not provide interpretability and has reduced accuracy in complex multidimensional feature spaces where anomalies are not clearly separated. Similar shortcomings are observed in autoencoder-based methods: without clear features or labeled data, they are prone to overgeneralization.

Traditional signature-based systems, such as Snort or Zeek, demonstrate high accuracy only in detecting known attacks but are ineffective when new attack types or modified behavior patterns emerge. They cannot adapt to new threats in real time without constantly updating their signature database. Thus, there is a need for a hybrid approach that combines the efficiency of unsupervised clustering with the ability to explain the results obtained, which determines the feasibility of using SOM and Explainable AI.

The work of W. Chua et al. [5] analyzes the effectiveness of the Isolation Forest algorithm for detecting anomalies in web traffic. The method has demonstrated a high ability to detect isolated deviations, which makes it promising for detecting unknown influences without prior patterns. S. Kerimkhulle et al. [8] emphasized the use of fuzzy logic to assess risks in the Internet of Things, particularly for identifying potential threats that lack a clear structure. This approach is relevant for modeling behavioral signs of unauthorized influences in environments with incomplete or uncertain information. J. Feng et al. [7] proposed a Tensor RNN with differential privacy to protect IoT system traffic. Despite its focus on privacy, the architecture demonstrates the ability to analyze temporal patterns and detect hidden activity, a key sign of unauthorized influences.

I. Hamid and M. M. H. Rahman [2] emphasize the feasibility of using XGBoost and deep learning in cyber risk management systems. In particular, they highlight the importance of combining classification models to detect complex multi-vector threats. The work of R. C. Poonia et al. [3] presents the concept of real-time risk management in cyber-physical systems. It considers the use of distributed sensor systems and dynamic analysis, which meets the need for rapid response to abnormal influences in the data transmission infrastructure. K. Al-Dosari and N. Fetais [10] conducted a meta-analysis of information security systems for small and medium-sized businesses, concluding that there are insufficient means of identifying subtle influences that do not fall under the classic categories of attacks. This highlights the importance of implementing

behavioral models with flexible detection thresholds. M. Mylrea et al. [4] proposed a framework for detecting threats from internal attackers, focused on complex hidden influences in cyber-physical systems. The systematization of signs of covert activity and the development of an analytical web tool demonstrate the effectiveness of a multi-level approach to detecting influences that may remain undetected by classical systems.

Recent studies have focused on detecting anomalies in network traffic using machine learning methods, including One-Class Neural Network (OC-NN) [1], Isolation Forest [5], Autoencoder [1], Support Vector Machine (SVM) [2], and SOM [3]. OC-NN performs well with a small amount of labeled data but is difficult to train. Isolation Forest is effective for processing large samples, but does not provide high accuracy for complex patterns. Autoencoder and SVM work well in high-signal-to-noise conditions but have limited interpretability. SOM is distinguished by its ability to visualize data structure and cluster unlabeled traffic, but it requires tuning of the map size and learning rate. Table 1 provides a brief comparative analysis of these methods.

Table1
Comparison of modern methods for detecting anomalies in traffic

Method	Advantages	Disadvantages
OC-NN	Works without attacks; adaptability	High complexity of configuration
Isolation Forest	Simplicity, efficiency	Low interpretability
Autoencoder	Good reconstruction of norms	Vulnerability to overfitting
SVM	High accuracy with clear separation	Poor scalability
SOM	Unlabeled clustering, visualization	Dependence on training parameters

In these works, SOM is primarily used as a clustering tool, with little in-depth interpretation of the results [1, 5, 8]. At the same time, modern cybersecurity requirements call for explanations of detected anomalies, which are not practically implemented in the approaches mentioned [11]. Thus, the gap lies in the absence of models that combine SOM clustering with XAI interpretation based on SHAP or LIME, which is the subject of this study.

Thus, the literature analysis shows the active development of intelligent methods for detecting anomalies and cyber risks [3, 10, 12–15]. However, most approaches focus on general anomalies and do not account for the specifics of unauthorized (undeclared) influences that operate covertly, without signatures or structured patterns [4, 9, 16, 17]. The method proposed in this article addresses this scientific gap by combining behavioral modeling, traffic generation, and environmental control, along with XAI analysis, to detect hidden influences in network interactions accurately.

3. Research method

The study uses an integrated approach that combines machine learning, behavioral traffic analysis, and explainable analytics (XAI) methods [2, 11, 18, 19]. Network interaction is formalized as a directed graph, where nodes are network objects, and links are data exchange sessions [20]. The normal activity profile was constructed using One-Class Neural Network (OC-NN), Autoencoder, and Isolation Forest models, which enable the detection of abnormal deviations without the need for labeled anomalies [1, 5, 12]. To improve classification accuracy, the XGBoost (Extreme Gradient Boosting) ensemble algorithm was used, a powerful gradient-boosted decision-tree (tree) ensemble method optimized for speed and performance [15, 19, 21, 22]. XGBoost is widely used in classification and regression tasks due to its high accuracy, resistance to overfitting, and ability to handle large datasets.

During the experiments, SOM maps were tested with different grid sizes (10×10, 20×20, 30×30). The best results in terms of F1-score and processing time were achieved with a size of 20×20, which

was chosen as the main option [3, 7, 23]. The initial value of the learning rate η(t) was set at 0.1 with an exponential decay to 0.01. The open dataset CICIDS-2017 (Canadian Institute for Cybersecurity Intrusion Detection System) was used to train and test the model, which contains real examples of normal and abnormal traffic, including DoS, PortScan, and BruteForce attacks [1, 13, 14]. The data was split 70% for training and 30% for testing.

SHAP and LIME are used to explain the decisions of intelligent models, allowing the weight of features to be interpreted when making decisions about traffic anomalies. WMI (Windows Management Instrumentation) data is used to monitor host behavior in the system. WMI is an interface for obtaining telemetry information about processes, network connections, services, and events in the Windows environment [9]. Additionally, a mechanism for controlling the integrity of the traffic generation environment based on this WMI data has been implemented, allowing the scope of collected information to be expanded without the need to install third-party agents [24]. This approach increases the transparency of observation and reduces the impact on the target system.

All experimental studies were conducted in a standardized computing environment to ensure the reproducibility of the results obtained. The software configuration included Python 3.11 and modern libraries for machine learning and intelligent traffic analysis: TensorFlow 2.15, Keras 2.15, Scikit-learn 1.5, NumPy 1.26, Pandas 2.2, SciPy 1.12, Matplotlib 3.8, SOMPY 1.2, SHAP 0.45, and LIME 0.2.0. The REST interface for prototype integration was built with Flask 3.0, and containerization was performed with Docker 24, ensuring independence from system libraries and stability of the execution environment. To ensure reproducibility in TensorFlow, NumPy, and SOMPY, the initial states of random generators (random seed = 42) were fixed, and each training procedure was repeated 10 times. The deviation of the F1-score metric between repeated runs did not exceed ±0.7%, and the stability of the SOM cluster boundaries remained within ±1.9% of the Euclidean distance. The average training time for the SOM model (20×20 neurons) was 14.2 seconds, and the inference time for a single feature vector was approximately 2.1 ms, confirming the method's suitability for real-time use in network environments with a load of up to 500 packets/s. The calculations were performed on a workstation equipped with an AMD Ryzen 7 5700X processor, 32 GB of RAM, and an NVIDIA RTX 3060 (12 GB) GPU, which enabled the processing of large-scale traffic samples and the training of models in near real-time.

The experimental part of the study was based on the use of representative sets of network traffic containing realistic profiles of normal and abnormal activity. The study used two widely recognized datasets: CICIDS-2017 and UNSW-NB15, which enable objective comparison with modern threat detection methods. The initial processing involved removing missing values, normalizing numerical features using the MinMax transformation, encoding categorical fields with One-Hot Encoding, and filtering correlated features with a correlation coefficient of $|r| > 0.92$. To avoid the dominance of certain types of traffic, the sample was balanced using a combination of Random Undersampling and SMOTE. The proposed methodology involved dividing the data into training (70%), validation (15%), and testing (15%) sets using stratified sampling. The quality assessment criteria were Precision, Recall, F1-score, and ROC-AUC, which allowed us to form a comprehensive picture of the effectiveness of the developed method in various network activity scenarios.

The prototype was implemented using the aforementioned software environment. At the practical level, Scapy libraries were utilized to analyze and generate network packets, while Flask was employed to implement a REST API and integrate with the administrator interface [25]. Validation was performed on the CICIDS-2017 and UNSW-NB15 datasets (a dataset of network traffic with attacks), using the metrics Precision (accuracy of classification of positive events), Recall (completeness of detection), F1-score (harmonic mean between accuracy and completeness), ROC-AUC (area under the ROC curve), and MCC (Mathews correlation coefficient for assessing classification balance) [6, 16, 17, 26–29]. The proposed method allows the detection of unauthorized influences in network interactions with high accuracy, ensuring flexibility, scalability, and explainability of results.

4. Main material

An analysis of the current state of research in the field of ensuring the reliability of computer networks and traffic processing indicates a rapid increase in the number of complex, multi-vector, and unauthorized influences that remain undetectable by classical signature-based methods [1, 2, 5]. In this regard, it is crucial to develop methods for detecting hidden, undeclared influences in network interaction processes that are implemented without prior signs in threat detection system knowledge bases [3, 12]. In such conditions, traditional tools focused on fixing known vulnerabilities and attacks demonstrate limited effectiveness, and reactive approaches are unable to adapt to dynamic changes in network topology and attacker behavior.

The scientific literature reveals a clear trend toward implementing proactive, intelligent solutions that combine machine learning, behavioral traffic analysis, and explainable analytics methods [11, 18]. The object of study in such systems is network traffic, which exhibits self-similarity, is prone to bursts of activity and uneven load distribution, and also exhibits complex dynamics that depend on the architectural and technical parameters of the environment [12–14]. Studies have shown that detecting unauthorized influences is only possible if a multi-level profile of expected system behavior is constructed, in which even minor deviations can indicate deliberate anomalous activity.

Based on a generalization of modern methods, the most relevant approaches have been identified: statistical analysis [6], contextual-semantic monitoring [10], expert models [9], environment integrity control mechanisms, neural network algorithms (in particular, One-Class Neural Network, Autoencoder, SOM) [1, 5, 23], and XAI tools SHAP and LIME [6]. The combined use of these approaches allows not only the detection of unknown attacks but also the interpretation of the reasons for their occurrence, reducing the level of false-positive decisions [19, 21]. Research on the telemetry of server nodes, the system environment, and traffic generation dynamics has shown that changes in environmental parameters that are undetected by classic IDS systems play a crucial role in creating conditions for hidden influences.

To identify such influences, it is proposed to decompose the network interaction process by identifying critical objects of analysis: the traffic generation environment, flow characteristics, interface nodes, and interrelationships among them [20, 28]. For each object, a set of behavioral and structural features is defined that must be considered when forming an adaptive security profile [29, 30]. The importance of parameters is assessed using expert ranking, followed by the calculation of the concordance coefficient to evaluate statistical significance [15, 17, 22, 24]. The results enable the selection of informative features that are most sensitive to influences not captured by formalized models. Further grouping of these features into functional clusters allows the formation of structurally consistent security profiles for different types of network segments. This provides the basis for building a dynamic response model that can adapt to environmental changes and identify hidden, unauthorized influences on time.

The structure of data flows and the distribution of system components across network infrastructure nodes are illustrated in Figure 1. The diagram shows the interaction among the main modules of the model, including traffic generation, monitoring, profiling, impact analysis, XAI interpretation, and decision-making within the security loop.

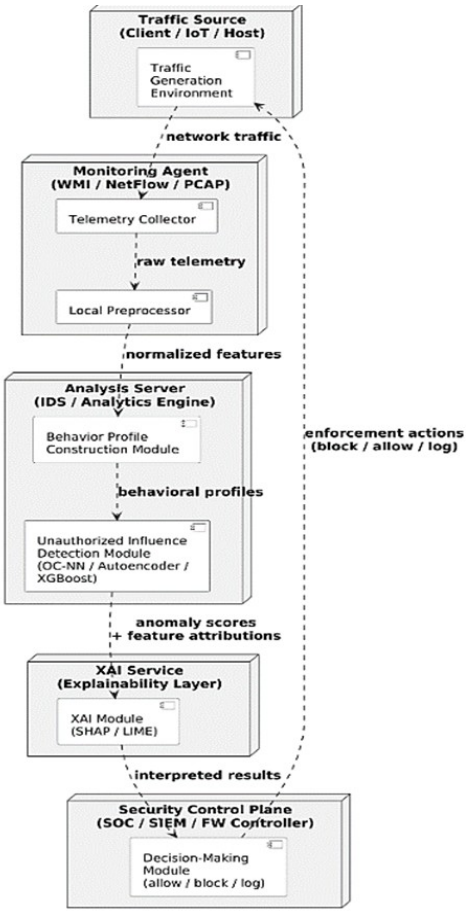

Figure 1: Diagram of the deployment of components of a system for detecting unauthorized influences based on intelligent traffic analysis

The algorithmic sequence of the system's operation is illustrated in Figure 2. The diagram shows the system's data processing logic, from network and WMI monitoring to classification, explanation of results, and decision-making regarding the presence of an impact.

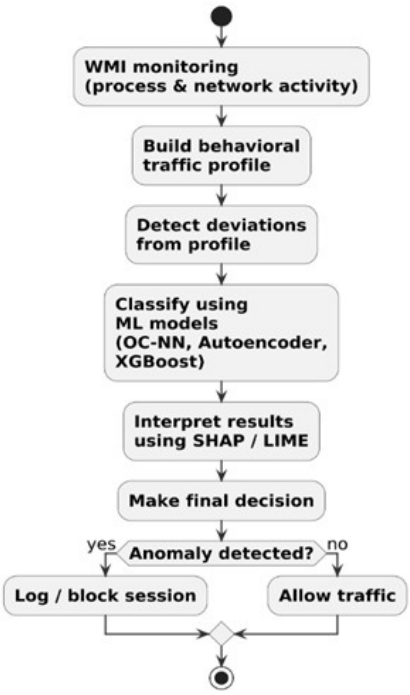

Figure 2: Block diagram of the algorithm for analyzing unauthorized influences in the network based on intelligent traffic analysis

The block diagram (Figure 3) shows the sequence of operations of the traffic generation environment control module. The system performs step-by-step monitoring of WMI parameters, generates a matrix of characteristics, ranks indicators by importance, builds a dynamic profile of regular activity, and transmits the results to the primary anomaly analysis module.

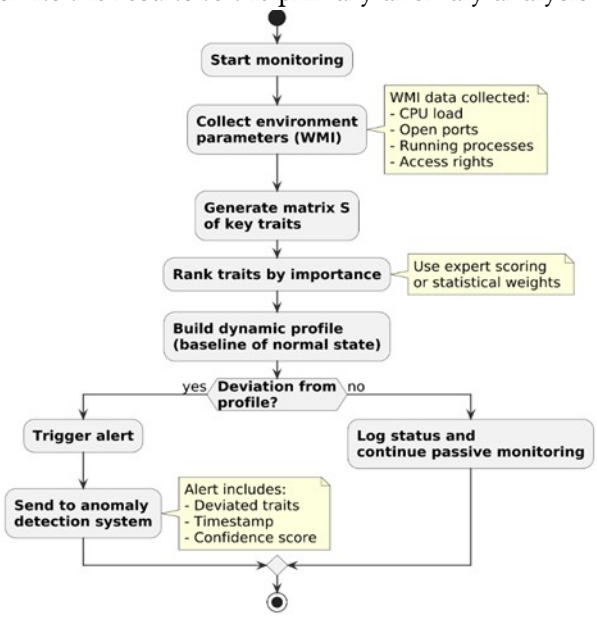

Figure 3: Block diagram of the traffic generation environment control algorithm for detecting unauthorized influences

The information model of intelligent network traffic analysis for detecting unauthorized influences can be formalized as a tuple [8, 20]

$$M = \langle O, Q, V, \Psi, E, K \rangle \tag{1}$$

where O is a set of classes of analysis objects (network nodes, traffic generation nodes, protocol structures), Q is a set of "reference" or normative characteristics of an object (reference profile) stored in the behavior template database, V is current telemetric observations of an object (a specific node/virtual machine/module), $\Psi(Q_i, V_i)$ a correspondence function for assessing the deviation of parameter V_i from standart Q_i, E expert weighting coefficients of importance for characteristics (determined based on statistics or security policy priorities), K resulting assessments of the state of an object or its part (a vector of evaluations from 0 to 1). This model provides a consistent representation of an object's behavioral characteristics and lays the groundwork for further detection of deviations at the level of individual parameters and their combinations. The integration of normative profiles, telemetry, and expert weights enables the formation of a formalized risk vector, which is subsequently used to identify unauthorized influences and inform decisions within the traffic analysis system.

The process of identifying influences involves several consecutive stages. First, each parameter V_i is compared with the corresponding standard Q_i, where a match in values is interpreted as no influence, and a discrepancy is interpreted as a potential deviation. For each network node, a binary vector $\vec{q} \in \{0,1\}^2$, is formed, reflecting the presence or absence of anomalies for each of the parameters [3]. Next, the integral index of influence criticality is calculated using the formula [2, 10]

$$omega = \sum_{i=1}^{n} e_i \cdot psi(Q_i, V_i) \tag{2}$$

where e_i are importance coefficients (e.g., access level, service criticality, leakage risk). The resulting ω value allows you to quantitatively assess the degree of deviation of the current state from the reference profile and determine the risk posed by the detected impact. Based on this

index, the system classifies the event as normal, suspicious, or critical, which ensures further automated response and prioritization of incidents in the traffic analysis process.

For each object of analysis, a normalized result value is determined, indicating the level of risk impact

$$K_i = \max\left(1 - \frac{\min \vec{q}}{\max \vec{q}}\right). \tag{3}$$

This normalized value reflects the intensity of the deviation of the object's parameters from the normative profile and enables the unification of risk assessment across different types of nodes and flows. An increased K_i value signals an increase in the criticality of the impact and serves as the basis for activating the appropriate response mechanisms in the intelligent traffic analysis system.

The next stage involves utilizing intelligent traffic analysis that takes into account the behavioral characteristics of traffic flows. Based on the TCP/IP stack at the OSI levels (from transport to application), key features are selected, such as session duration, data transfer volume, byte distribution, connection frequency, and the number of SYN/ACK requests, among others [5, 15]. The system combines signature filtering methods with deep learning algorithms, specifically Autoencoder, One-Class Neural Network (OC-NN), and XGBoost, which enables the detection of both known and unknown types of anomalies [1, 12, 19]. To ensure transparency in decision-making, XAI modules (SHAP and LIME) are integrated into the system, enabling the interpretation of the weight of each feature and the identification of the dominant factors behind unauthorized influence.

This combination of statistical, signature, and neural network methods forms a multi-level detection system that can adapt to changes in traffic intensity and structure in real-time. The use of hybrid models increases resistance to false positives, as the analysis results undergo multi-channel verification through various analytical mechanisms. As a result, the system cannot only detect deviations but also explain their nature, ensuring a high level of interpretability and suitability for rapid response.

The process of determining the criticality of an impact is implemented through continuous monitoring of environmental parameters [9], intelligent classification of the type of impact [21, 23], and the formation of a risk function taking into account weighting coefficients [24], as shown in Figure 4. The diagram shows the key stages of the analysis, which end with the normalization of the criticality index and the decision on the threat level [14, 31].

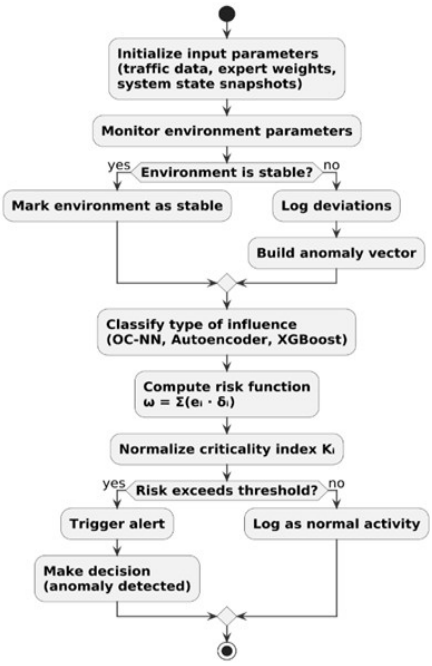

Figure 4: Activity diagram of the process of determining the criticality of unauthorized influence

As part of the proposed method of intelligent analysis of network traffic to detect unauthorized influences, an updated signature model is used, which implements context-sensitive comparison of behavioral characteristics with the basic system profile [2, 3, 5]. A signature is considered a multidimensional sequence of factors relevant to the normal functioning of network protocols in real-time.

Each signature includes both static parameters that determine the protocol's structural properties and dynamic indicators that reflect changes in network activity over time. Thanks to this, the model can identify deviations associated with both packet-format violations and abnormal behavior in transport and application services, which is not possible with traditional signature-based approaches.

At the first stage, a traffic flow is formed

$$\mathrm{P}=\langle \mathrm{H}_1, \mathrm{H}_{2...} \mathrm{H}_n, \rangle \quad (4)$$

where H_i is a vector of features that characterizes traffic behavior in a specific time interval [9]. The number of such vectors corresponds to the number of segments of the analyzed traffic obtained from network telemetry or monitoring systems (WMI, NetFlow, PacketCapture). Each vector H_i reflects the local state of network activity and serves as a basic unit for further clustering and anomaly detection. Such step-by-step segmentation allows us to account for the temporal dynamics of traffic. It ensures the correct formation of feature sequences for further analysis using SOM and XAI methods.

Next, key parameters are extracted from the H_i stream that reflects critical traffic characteristics —delays, packet size, protocol type, non-standard ports, frequency deviation, etc. These parameters form a set of signature features S that is updated through dynamic learning

$$S=\langle S_1, S_{2...} S_m, \rangle \quad (5)$$

Further analysis of this set allows us to identify the most informative features that have the most significant impact on changes in traffic behavior profiles. Thanks to constant updating and adaptation of the S set, the system can adapt to new network activity scenarios and more effectively detect previously unknown or modified types of influence.

Contextual comparison is performed using a modified mechanism that constructs a correspondence function between the input stream and the reference signature as a binary vector. If $\gamma_i=0$, this indicates that the i^{th} feature corresponds to the reference value; $\gamma_i=1$ indicates a detected deviation. In the case of multiple deviations, a structured vector of contextual violations is formed, which reflects the type, strength, and location of the anomaly within the stream. The resulting binary profile is transferred to the integrated risk assessment module, where it is aggregated with feature weight coefficients to determine the impact's criticality level.

To localize the source of unauthorized influence, an intelligent model is used to identify the dominant anomaly feature. If we denote it as δ, then the dominant signature takes the form

$$\delta=S_1^{\uparrow}, S_2^{\uparrow}, \ldots S_k^{\uparrow} \quad (6)$$

where $S_i^{\uparrow}$ are features with the highest deviation from the baseline profile (determined via SHAP or LIME for a specific flow). This signature is then used as a marker of the attack source, as it allows us to determine which traffic parameter was changed first and with what intensity. Including δ in the correlation analysis process increases the accuracy of identifying cause-and-effect relationships in the behavior of network agents and reduces the time it takes to locate an incident.

If $\delta=\varnothing$, it is assumed that there is no anomaly. If $|\delta|>0$, its criticality and type of impact are determined—network scan, DoS, data leak, or intrusion attempt [1, 10, 18, 19, 29]. Thus, the proposed methodology enables adaptive expansion of the signature template through the built-in interpretation of machine learning models, allowing for the detection of atypical influences not described by traditional signatures and the localization of the source of influence, taking into account the context of the current interaction.

The diagram in Figure 5 illustrates the complete cycle of network traffic processing, from data acquisition to decision-making regarding the presence of unauthorized influence. In the initial stages, features are extracted and compared with signatures, after which classification is performed using modern models (OC-NN, Autoencoder, XGBoost). The results are processed through the XAI level (SHAP, LIME) [11], ensuring the interpretability of the conclusions. At the final stage, the system's final decision is formed.

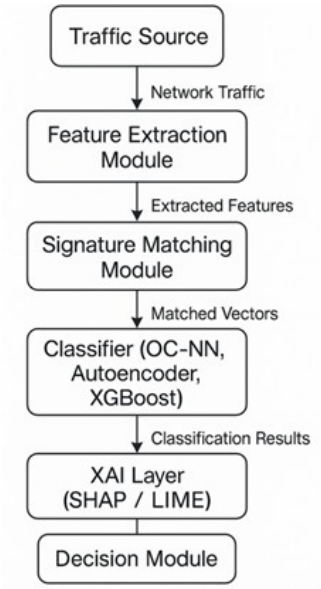

Figure 5: Process of detecting unauthorized influences in traffic based on signature analysis and XAI

Kohonen neural networks (SOM) are used to analyze behavioral patterns, allowing traffic to be grouped without prior labeling. SOM effectively clusters traffic based on similarity of characteristics, allowing abnormal segments to be identified. Input vector:

$$X = \langle src\_ip, dst\_ip, src\_port, dst\_port, proto, flags, payload\_size, timestamp \rangle \quad (7)$$

After projecting vectors X into the feature space, SOM forms a topological map on which vectors with similar behavior are placed in compact clusters. At the same time, anomalous segments are removed from the normative groups. This self-organizing structure enables the identification of previously unknown types of influence without the need for supervised learning. The resulting clusters serve as the basis for further interpretation of decisions using XAI mechanisms.

The adaptation of neuron weights is carried out according to

$$W_{ij}^{(t+1)} = W_{ij}^{(t)} + \eta(t) \cdot (X_j) - W_{ij}^{(t)} \quad (8)$$

where $\eta(t)$ is a learning coefficient (within the range [0.1–0.01]), W_{ij} is the weight of the j^{th} feature of the i^{th} neuron. A gradual decrease in the learning coefficient $\eta(t)$ ensures model convergence and stabilization of the topological structure of the feature map. As a result, neurons gradually form stable cluster centroids that reflect typical traffic behavior patterns. Ultimately, SOM demonstrates the ability to highly sensitively distinguish between normal and abnormal segments even in a changing network environment.

The implementation utilizes a two-dimensional Kohonen map with a size of 15×15 neurons, which provides sufficient resolution for grouping network traffic behavior patterns [12, 14]. The initial weight initialization was performed randomly from a uniform distribution within the feature range. A linearly decreasing function was used for the learning rate $\eta(t)$ linearly decreasing function applied [31]. Linear color coding in RGB space was used to visualize SOM maps [23]. This approach enables rapid initial learning, followed by a gradual decline in adaptability, which contributes to stable map formation. The colors reflect the distance of the features to the cluster

centers—the stronger the deviation, the brighter the color [19, 30]. This enables real-time detection of suspicious areas (anomalies), which are visually highlighted on the feature map [11, 15]. Advantage: no need to dive deep into tables—critical points are immediately visible, especially when integrated with alert system notifications.

The graph (Figure 6) shows a SOM where the color of each neuron reflects its deviation from the average behavior profile. Brighter RGB palette values correspond to a higher level of abnormality, allowing you to quickly identify potentially problematic areas of network traffic.

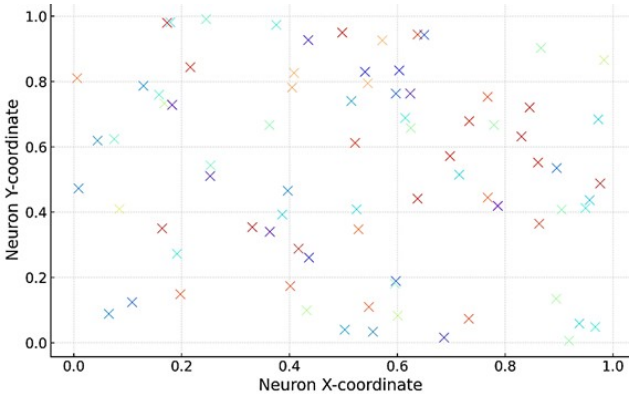

Figure 6: Visualization of a SOM map with RGB encoding of anomalies

To improve the accuracy of the analysis, a combination of volume and frequency analysis modes is used, taking into account the following parameters

$$Tr = \langle \frac{S}{D}, Tp, L, Tt \rangle \tag{9}$$

where S is a source, D is a destination IP address, Tp is a protocol type and port, L is a packet length, Tt is a timestamp [1, 5]. The following methods are used: volume mode monitors traffic by size and frequency, and frequency mode analyzes request repetition within a time window [2, 5]. For example, the frequency metric of requests from an IP address

$$F_{ip} = \frac{N_{ip}}{T} \tag{10}$$

where N_{ip} is a number of requests, T is a time [3, 5]. This allows you to identify slow attacks, periodic scans, abnormal ping pulsations, and other types of network influences that disrupt normal interaction. The combination of these modes provides a multidimensional overview of traffic behavior, allowing both sudden spikes in activity and hidden low-intensity impacts to be detected. Additionally, the dynamics of parameter changes within a sliding time window are evaluated, thereby increasing the system's sensitivity to long-term anomalies. The integration of the F_{ip} frequency metric with the S signature features forms a context-oriented flow profile suitable for further clustering analysis. As a result, the system can correctly distinguish legitimate load fluctuations from signs of potential cyber influence.

The graph in Figure 7 shows typical request frequency dynamics for a single IP address, where peak values may indicate scanning or DoS activity. The interval 30–40 (highlighted in red) simulates a DoS-like attack, characterized by a sharp increase in request frequency. The 70–72 range (marked in orange) reflects short-term scanning activity. Such an analysis of the F_{ip} the frequency metric enables analysts or monitoring systems to quickly detect abnormal patterns in network traffic.

$$\Delta_t = \frac{1}{n} \sum_{n-1}^{i=1} |T_{(i+1)} - T_i|. \tag{11}$$

High Δ_t values indicate instability in the time intervals between packets, characteristic of network noise, periodic scans, or attempts to bypass filters. Conversely, consistently low values may indicate automated traffic generation or the operation of botnet components with clearly fixed timing. Analysis of pulsations in combination with other parameters enables more accurate differentiation between legitimate and anomalous flows.

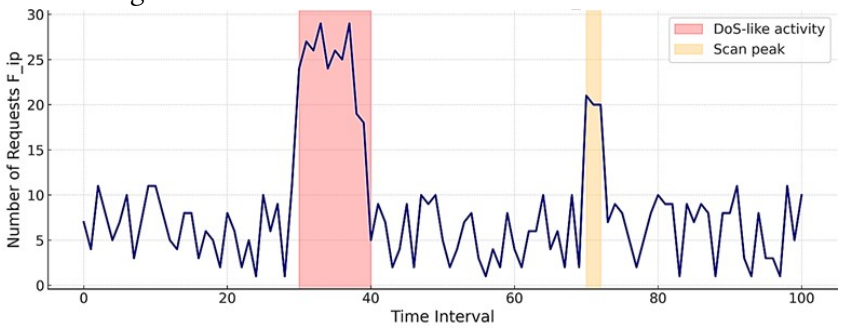

Figure 7: Example of how the request frequency metric works F_{ip} Pulsation (delay variation)

To implement the method of intelligent analysis of network traffic, the neural architecture of the SOM is used [1, 5, 7]. The input object is a vector of traffic characteristics

$$\vec{x}_t = \langle f_1, f_2, \dots f_n \rangle \tag{12}$$

where f_i are flow characteristics, which may include source and destination IP addresses, ports, packet length, reception time, protocol type, request frequency, delay, etc. [2, 5]. Each vector $\vec{x}_t$ is formed for a fixed point in time or a network monitoring segment. Before being fed into the SOM, the vectors undergo normalization and feature scaling, which minimizes the impact of imbalances between values of different natures. Such pre-processing ensures the correct formation of the map topology and increases the model's sensitivity to hidden anomalous patterns in traffic.

At the initial stage, the weights of each neuron in the map are initialized randomly

$$W_i^{(0)} = \text{random}\left([0,1]^2\right) \tag{13}$$

which allows the neural network to adapt to the distribution of the input data. Initial stochastic initialization offers a range of initial neuron representations, which contribute to improved coverage of the feature space during the early stages of training. Subsequently, during self-organization, the weights gradually shift toward the most typical traffic structures, forming a stable topological map of normal behavior.

During training, the nearest neuron to each input vector is selected

$$BMU = \arg\min_i \|\vec{x}_t - W_i\| \tag{14}$$

i.e., the neuron whose weight vector W_i has the smallest Euclidean distance to the current flow. Once the Best Matching Unit (BMU) is determined, a weight update mechanism is activated in this neuron and its neighborhood, ensuring gradual adaptation of the map topology to the input data structure. This approach allows SOM to form compact clusters of normal behavior and effectively separate anomalous traffic segments from regular patterns.

This neuron (and its neighbors) are updated according to the rule

$$W_i(t+1) = W_i(t) + \eta(t) \cdot h_{bi}(t) \cdot (\vec{x}_t - W_i(t)) \tag{15}$$

where $\eta(t)$ is a learning speed function that decreases over time

$$\eta(t) = \eta_0 \cdot \exp\left(-\frac{t}{\lambda}\right), \tag{16}$$

where $h_{bi}(t)$ is an influence function that models the degree of learning for neurons distant from the BMU

$$h_{bi}(t)=\exp\left(-\frac{\|r_b-r_i\|^2}{2\cdot r(t)^2}\right) \tag{17}$$

where r_b and r_i are positions according to BMU and the i^{th} neuron on the topological map, and $r(t)$ is the neighborhood radius, which also decreases exponentially over time. This gradual decrease in $\eta(t)$ and $r(t)$ ensures a smooth transition from global learning, when the map adapts to the overall data structure, to local refinement of clusters in the final iterations. In the early stages, the large value of $r(t)$ allows the BMU to update the wide neighborhood, forming the macrostructure of the topology. As the neighborhood radius decreases, learning focuses on the nearest neurons, which helps to detect subtle differences in traffic behavior more accurately. As a result, the SOM forms stable clusters of normal patterns and highlights anomalies as segments that do not fit into the established topology.

The graph in Figure 8 shows the placement of neurons in the SOM (marked in gray) and input traffic vectors (blue). For each input vector, the best-matching neuron (BMU) is identified and marked in green. Neurons associated with input streams that have a distance to the BMU exceeding a specified threshold are classified as anomalous and are highlighted in red with a border. This visualization allows you to quickly locate suspicious network segments and improve the interpretation of abnormal traffic without manually analyzing tables or metrics.

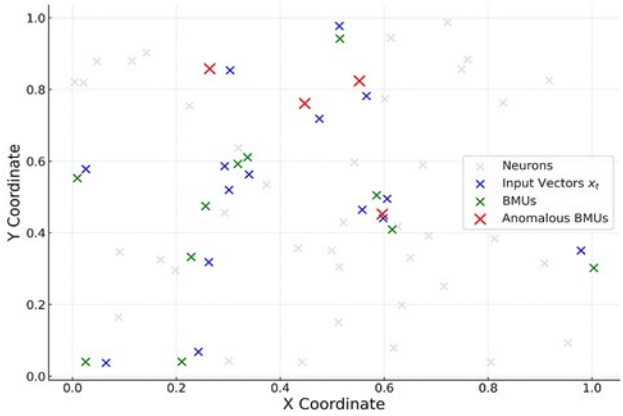

Figure 8: Map of anomalies with BMU markings

The neighborhood radius $r(t)$ in a SOM determines which neurons (except for BMU) will also be updated during training. This radius decreases over time, allowing the system to transition gradually from global to local training. An exponential decrease in the neighborhood radius is typically used [1, 7]

$$r(t)=r_0\cdot\exp\left(-\frac{t}{\frac{T}{\log(r_0)}}\right) \tag{18}$$

where r_0 is an initial neighborhood radius (e.g., half the map size), t current training iteration number, $\lambda=\frac{T}{\log(r_0)}$ is a time constant that determines the rate of radius decrease, T is the total number of iterations. Thanks to this dynamic, SOM first forms the map's global structural organization, aligning neurons with the main traffic patterns. In later iterations, learning focuses on local details, which increases the model's ability to distinguish between network influences that are similar in appearance but different in behavior.

The graph in Figure 9 shows the exponential decrease in the learning rate $\eta(t)$ and the neighborhood radius r(t) during the SOM operation. The blue line corresponds to the decrease in

$\eta(t)$, while the orange line corresponds to the change in $r(t)$. Both functions decrease over time, ensuring the network's transition from a global organization to the local refinement of data structures.

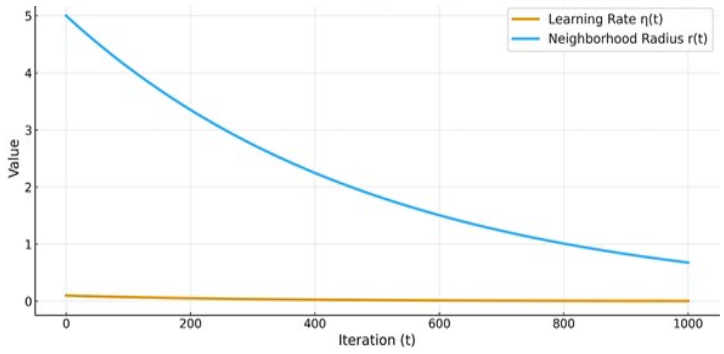

Figure 9: Dynamics of the learning function $\eta(t)$ and the neighborhood radius $r(t)$

After completing training, SOM determines the distance to the nearest neuron for each new traffic item [1, 7, 15]

$$d_t = \|\vec{x}_t - W_{\text{BMU}}\|. \tag{19}$$

The obtained value d_t serves as the basis for classifying traffic as normal or abnormal based on the threshold value determined during model calibration. The more d_t exceeds the set threshold, the higher the probability of hidden or undeclared influence in network interaction. To increase noise resistance, the threshold θ is adaptively adjusted based on statistics of recent observations, allowing short-term traffic fluctuations to be filtered out. In addition, the value of d_t can be integrated into multi-level risk assessment mechanisms, where it acts as a key indicator of abnormal flow behavior. If this distance exceeds a predefined threshold θ, the incoming flow is marked as an anomaly

$$A(x_t) = \begin{cases} 1, \text{if } d_t > \theta, \\ 0, \text{then}. \end{cases} \tag{20}$$

This approach provides a quick binary assessment of traffic status without the need for additional tagging or complex computational procedures. In addition, adaptive threshold selection θ allows dynamic changes in the network environment and increases the system's resistance to new types of influence.

XAI methods are used to interpret the detected anomaly, specifically SHAP or LIME [2, 5, 11]. These approaches allow for calculating the importance of each feature in decision-making

$$\delta_i = \left| \frac{\partial y}{\partial f_i} \right| \tag{21}$$

where y is a model output, f_i is a specific traffic feature. The obtained δ_i values allow us to determine which traffic parameters had the greatest impact on classifying the segment as anomalous, significantly increasing the transparency of the model's operation. This allows not only to record the fact of a violation, but also to establish its nature—for example, atypical request frequency, port change, abnormal packet length, or non-standard behavioral signature.

The graph (Figure 10) illustrates the significance of individual traffic features in the SOM classifier's decision regarding a flow's anomaly, as determined by the SHAP explanatory model. The axis shows the values of δ_i, which evaluate the contribution of each feature to the final output of the model payload_size, timestamp, and dst_ip, have the most significant impact, indicating their critical role in detecting potentially malicious or atypical flows. This visual XAI approach not only detects anomalies but also explains why the system detected them, increasing trust in the neural network model.

The set of the most important features with the highest deviations forms the dominant anomaly signature $\delta = S_1^{\uparrow}, S_2^{\uparrow}, \ldots, S_k^{\uparrow}$. This signature reflects the key factors that led to the deviation from the reference profile and is used for further analysis of the attack's nature and impact. It allows not only to localize the source of the anomaly, but also to assess its potential danger, taking into account the weight coefficients of the characteristics.

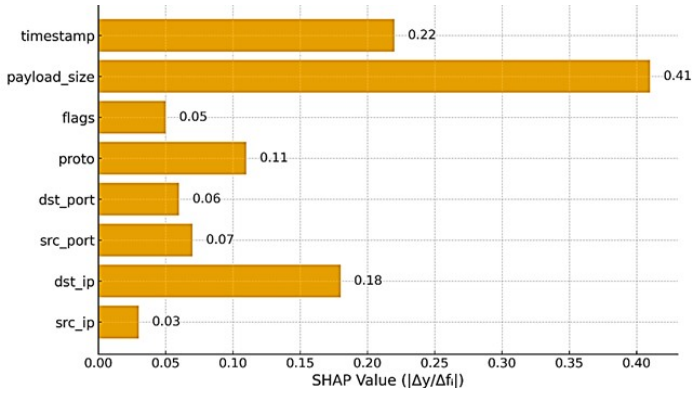

Figure 10: SHAP/LIME explanations for the SOM classifier

To detect anomalies in visual space, linear color coding in RGB format is used, which depends on the distance of each neuron to the nearest input vectors

$$RGB_i = \text{map}\|\vec{x}_t - W_i\|; \min, \max. \tag{22}$$

The result is a topological map of color intensity, where cool tones correspond to areas with normal traffic and warm tones correspond to areas with increased deviation. This visualization enables you to quickly identify clusters of anomalies and track their spatial distribution within the SOM model, which is particularly useful for rapid incident analysis.

Neurons with high deviations are highlighted in bright colors, enabling critical areas to be quickly identified without the need for manual analysis of tables. Neuron clusters are formed according to the principle of minimum internal distance

$$C_k = \{ i \mid \|W_i - \mu_k\| < \epsilon \} \tag{23}$$

where μ_k is a cluster center, and ϵ is an acceptable error. Clusters with increased neuron density and large deviations from local "anomalous areas" that reflect typical patterns of malicious activity. This enables not only the recording of the impact's occurrence but also the determination of its nature and potential source, thereby increasing the accuracy of subsequent incident correlation.

To finally determine the risk level, a criticality index is formed based on a combination of a binary vector of compliance and the weight of characteristics

$$K(x_k) = \sum_{i=1}^{n} y_i \cdot \delta_i \tag{24}$$

where $\gamma_i \in \{0,1\}$ the presence of anomalous deviation δ_i is important. The obtained index value enables us to quantitatively assess the danger of the flow and prioritize further actions for the response system. High values of $K(x_k)$ automatically signal critical impacts that require immediate correlation with event logs and triggering of protection mechanisms.

The histogram (Figure 11) shows the value of the criticality index $K(x_k)$ for each of the eight clusters of the SOM. The bars display the total weight of the most deviant features characteristic of the corresponding cluster. Higher values of $K(x_k)$indicate an increased level of anomaly or risk associated with traffic falling into a particular group. Such visualization enables you to quickly identify the most critical network segments that require attention or response, and can be integrated into a monitoring system to support informed decision-making.

Within the proposed approach, the primary focus is on the formalized detection of unauthorized influences during network interaction through multi-level intelligent analysis of network traffic [1, 2, 5, 6, 15]. The methodology is based on comparing network activity parameters with patterns of regular functioning, combining statistical, signature, and neural network processing.

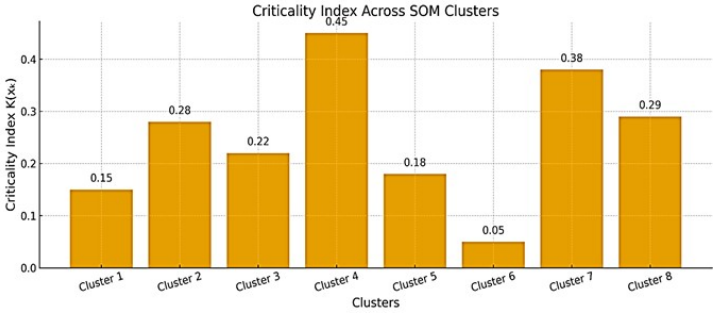

Figure 11: Histogram of critical clusters by index $K(x_k)$

At the first stage of analysis, a vector of observation features $\widetilde{X} \sim T_g^{(\beta)}$, is extracted from the event stream T_g a sequence of changes in the state of the network adapter (for example, the arrival or departure of a packet), which characterizes the current network activity. For each object $\widetilde{X}_i$ at time t_i comparison is made with the corresponding Template of Normal Network Functioning (TNNF). The boundaries of the permissible interval are defined as the lower limit $\widetilde{X}_i > \lambda_{\min}$ and the upper limit $\widetilde{X}_i < \lambda_{\max} = k \cdot \lambda_{ave}$, where $k = 0{,}8$. If the value $\widetilde{X}_i$ exceeds the set limits, the system records a potential deviation and transfers it to the in-depth analysis module to clarify the nature of the impact. Additionally, a binary anomaly indicator is formed that accounts for both the strength of the deviation and the context of previous events in the stream. This enables us to accurately distinguish between short-term fluctuations and persistent anomalies associated with hidden network attacks.

Further analysis is based on the calculation of sample statistics. In particular, for the set of elements X the sample mean is calculated

$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i \tag{25}$$

and the allowable range is set

$$\frac{1}{2} \cdot \overline{X} < X_i < 2 \cdot \overline{X}. \tag{26}$$

Such normalization enables the filtering out of random fluctuations and isolating only those values that significantly deviate from the flow's statistical stability. If an element goes beyond the permissible range, it is marked as a probable anomaly and is further analyzed in the context of the general traffic dynamics. Similarly, the limits of normal functioning are determined for the parameters of protocols, ports, and session durations. At the second stage, a frequency analysis of traffic types is performed. A set of packet types is isolated from each fragment of the flow

$$T_g^{(Tp)} = \{ p_1, p_2 \dots, p_n \} \tag{27}$$

where X_{pi} is the number of packets of type p_i, as well as the total number of captured packets

$$U = \sum_{i=1}^{n} X_{pi}, \tag{28}$$

The obtained data is used to determine the frequency distribution of packet types within the group, which allows the detection of atypical or dominant protocol events. Next, the relative frequencies of each packet type are calculated, based on which the system forms a profile of expected behavior for a specific traffic segment. The deviation of frequencies from the normative

values serves as an indicator of potential scans, activity spikes, or hidden attacks with protocol modification. The relative frequency of each type is calculated by

$$f(p_i)=\frac{X_{p_i}}{U}. \tag{29}$$

The resulting frequencies form a behavioral distribution of the traffic group, which is compared with a reference profile to detect anomalous shifts. A significant increase or decrease in $f(p_i)$ signals a possible change in the network operating mode or the manifestation of malicious activity, which requires additional correlation analysis. To fix anomalies, the flow variance is calculated

$$\sigma=\sqrt{E(X_{p_i}^2)-(E(X\,p_i))^2} \tag{30}$$

and the deviation from the TNNF is estimated by the generalized deviation function

$$\sigma_{\text{TNNF}}=1-\frac{\left|\widetilde{X_{\text{real}}}-X_{\text{template}}\right|}{X_{\text{template}}}. \tag{31}$$

The obtained value σ_{TNNF} allows us to quantitatively assess the degree of correspondence between real traffic and the expected pattern, and to detect even weakly expressed violations of network behavior. Low values of this indicator indicate significant anomalies and require in-depth analysis using correlation and clustering methods.

The formula calculates the normalized deviation of the current network indicator from the expected value of the regular network behavior pattern. A value close to 1, indicating high correspondence with the standard mode, indicates a high correspondence with the standard mode, while lower values signal potential anomalies [5, 15, 32]. This allows us to identify even slow or masked attacks that cause atypical changes in traffic intensity or structure.

Three approaches are used for integrated processing of the results: statistical, signature, and neural network. Each of them generates an intermediate estimate PR_i, which is weighted by the confidence coefficient CD_i, which reflects the accuracy of the model

$$KR_i=CD_i\cdot PR_i. \tag{32}$$

The obtained values KR_i are integrated into a standard indicator that combines the advantages of different methods and compensates for their individual limitations. This approach provides a more stable and reliable risk assessment, increasing the accuracy of detecting hidden network influences.

The final value for decision-making is formed as

$$R=\max(KR_1, KR_2, KR_3). \tag{33}$$

This approach ensures that the final decision is based on the most significant risk indicator among all three analysis subsystems. If the R value exceeds the regulatory threshold, the system classifies the impact as critical and initiates appropriate response mechanisms.

A special place in the methodology is occupied by neural network processing of traffic using SOM [7, 14, 17, 33]. Each neuron in the SOM map corresponds to a specific cluster of network behavior, and the distance between the traffic sample and the neuron's center determines its anomaly score. This distance is visualized using RGB coding: brighter colors indicate a greater degree of deviation, allowing you to quickly identify suspicious areas without manually reviewing feature tables.

In the proposed automated intrusion detection system, the information processing process is implemented as a modular cycle with intelligent traffic analysis mechanisms [25, 34]. The system's primary functionality is the real-time processing of data streams from the network environment. This process is divided into several logically interconnected stages.

The first stage involves collecting primary data from sensors, network agents, or SIEM components. The system intercepts key parameters: source and destination IP addresses, protocol

type, packet length, and arrival time. These data form structured transmission sessions that are sent to the pre-processing unit.

The second stage is data normalization and construction of feature vectors. All packet parameters are transformed into a vector of unified characteristics that can be fed to the input of the neural network. At this stage, noise is removed, missing values are filled, and the data is converted into a format compatible with SOM models.

The third stage is traffic clustering and anomaly detection using SOM [1, 15, 25]. The SOM builds a multidimensional topology of network traffic, where each cluster corresponds to a certain interaction profile. If the distance of the current vector to the nearest template exceeds a dynamic threshold, the system records an anomaly.

The next block is the decision block, which interprets anomalies based on the TNNF. If the detected deviation is significant, the system activates a response scenario: segment isolation, IP address blocking, sending a message to the administrator, or launching additional analysis.

The final stage involves updating the template database, enabling the system to adapt to legitimate changes in the network environment, prevent false positives, and build knowledge. The update can be automatic or controlled by the administrator. Thus, the information processing scheme in the system provides an adaptive, interpretable, and practical approach to detecting unauthorized influences in network traffic, using intelligent methods of analysis and visualization.

Figure 12 presents the stages of functioning of an intelligent network traffic analysis system for detecting unauthorized influences. The scheme includes data collection, normalization, clustering using the SOM model, anomaly detection, verification of compliance with the regular functioning template, and activation of response scenarios. If necessary, the system updates the template or initiates retraining to adapt to changes in the network environment.

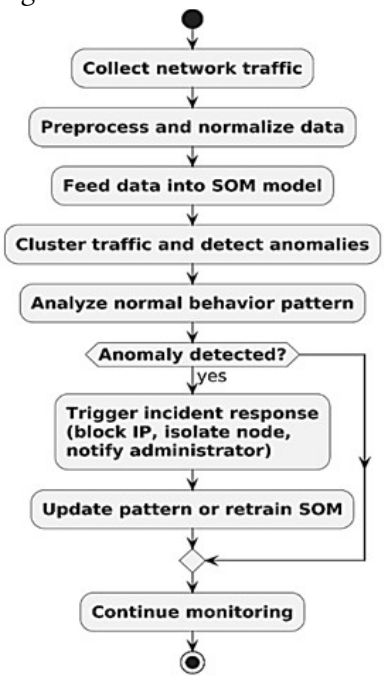

Figure 12: Information processing scheme in the tamper detection system

To assess the effectiveness of the proposed method for detecting unauthorized influences during network interaction, templates of regular operation and profiles of the traffic generation environment were developed. An “ideal” network, free from signs of malicious activity, was modeled as testing conditions. The test results showed that the reliability coefficient for the signature approach is 0.93, corresponding to a high score [1]. For statistical methods, the coefficient is 0.82 (average level) [5], and for neural network analysis based on self-organized Kohonen maps, it is 0.78 [15]. The analysis of individual and combined methods enabled us to provide an integrated

assessment of the system's effectiveness and justify the feasibility of their combined application for intelligent traffic analysis, identifying unauthorized influences in real-time.

The graph (Figure 13) illustrates a comparison of the effectiveness of three methods of intelligent network traffic analysis based on the reliability coefficient. The signature approach demonstrates the highest index (0.93), while the statistical (0.82) and the SOM-based neural network method (0.78) have lower values. The obtained results visually confirm the feasibility of combining methods to increase the reliability of detecting unauthorized influences.

Figure 13: Characteristics of the applied methods of intelligent traffic analysis

4. Practical implementation

The CICIDS-2017 dataset was used to train and test the model, which contains approximately 80,000 examples, comprising 55,000 normal traffic examples and 25,000 examples of various attack types. The data was split 70% for training and 30% for testing. The system was implemented using the software environment described above. As part of the practical implementation, the Scapy libraries were utilized to capture network packets, and Flask was employed to build a REST service and web interface, which simplifies integration with existing monitoring systems [11, 25, 32, 33]. The prototype was deployed in a Flask-based test environment, with further containerization using Docker to ensure scalability and integration with network monitoring [9, 32]. Data pre-processing included converting network sessions into feature vectors (according to formula (7)), normalizing values to the range [0, 1], and excluding records with missing or incorrect values.

The SOM model was trained on 80% of the regular traffic, while testing was performed on a mixed sample containing both normal and abnormal examples in a 1:1 ratio [7, 33]. The clustering results were interpreted using XAI methods (SHAP and LIME), which provided both global (feature importance) and local (analysis of specific decisions) explanations.

The size of the SOM map was chosen empirically as 20×20 neurons, which provided sufficient cluster density to detect small deviations in the traffic. The initial weights were randomly sampled from a uniform distribution. The learning coefficient $\eta(t)$ was varied linearly from 0.1 to 0.01 over 100 epochs, allowing the gradual reduction of the radius of influence and the stabilization of the map's topology [8, 34]. To increase the reliability of the results, the models were trained several times with different initial parameters, and the final clustering was performed by averaging the results.

To justify the choice of map size, a series of experiments was conducted with different SOM sizes (ranging from 10×10 to 30×30). The best results in terms of F1 and MCC metrics were observed at a size of 20×20 [7, 8, 33], which provided a compromise between resolution and processing speed. Increasing the size beyond 25×25 did not lead to a significant improvement in accuracy, but significantly increased the clustering time (up to 1.8 times). Reducing the map size resulted in the merging of anomalous clusters with normal ones, which in turn worsened recognition. The average processing time of one sample (feature vector) by the SOM map after training was ≈2.1 ms on a test machine (Intel i7-1165G7, 16 GB RAM), which allows using the method in real time in environments with a load of up to 450–500 samples/sec without loss of performance.

Within the practical implementation of the method for detecting unauthorized influences, an intelligent system was created that is capable of monitoring, processing, and analyzing network traffic in real-time to detect potential anomalies and deviations from the pattern of regular network functioning [12, 25]. To implement the system, the Python 3.10 environment was used, along with the Scapy libraries for packet capture, Pandas and NumPy for data pre-processing, TensorFlow and SOMPY for implementing the neural network architecture of Kohonen SOMs, as well as SHAP and LIME for explanatory interpretation of the results.

At the initial stage, the system continuously captures network traffic using Scapy or tcpdump, after which the traffic is cleaned, normalized, and transformed into a vector form, which includes the following features: source and destination IP addresses, ports, protocol, packet size, capture time, number of connections, etc. [9, 25]. The obtained data is fed into the SOM network, which performs clustering and detects anomalous traffic segments based on a distance criterion relative to the template cluster of regular activity.

Each new traffic sample is compared with the TNNF, which was formed during training using reference normal data. In the event of a significant deviation, the system flags the segment as potentially malicious. To increase the transparency of the neural network model, XAI methods are employed. SHAP graphs illustrate which features (e.g., flow duration or packet size) most significantly influence the decision-making, while LIME provides local explanations for each individual prediction.

The processing results are transmitted to the administrator's web interface, implemented in Flask, where they are visualized as a SOM map with RGB coding of the degree of anomaly [32]. The administrator can instantly see suspicious nodes or segments that differ in color. If the set thresholds are exceeded, the system performs reactive actions: blocks the IP address, isolates the port, sends an alert to the SIEM system, and updates the operating template.

Testing took place in a simulated virtual local area network (LAN) consisting of two clients and a server, all connected to a standard gateway. Various types of attacks were generated in the network, including port scanning, Ping Flooding, SYN Flooding, and DNS Spoofing [13, 17]. As a result of practical application, high detection accuracy was achieved: signature analysis (93%), statistical (82%), and neural network (78%). A complex hybrid system, based on multiple analyses, demonstrated an efficiency of over 90% in detecting unauthorized influences without the need for labeled data.

The activity diagram (Figure 13) shows a generalized process for the system to detect unauthorized influences in network interactions. Traffic processing begins with the collection of network packets using the Scapy or Tcpdump tools, after which pre-processing and feature normalization are performed. Traffic is then clustered using SOM, and if anomalies are detected, an alert is generated, and SHAP/LIME explanatory analysis is launched. The results are displayed to the administrator as an interpretable visualization, enabling decision-making regarding the response.

Figure 14. Generalized activity diagram of the method of intelligent anomaly detection in network traffic.

Thus, the developed system demonstrates the ability to adaptively detect threats, quickly respond to changes in traffic behavior, and explain its actions to the operator, which is critically important in the context of modern cybersecurity.

An ablation study was conducted to determine the contribution of individual system components to the overall effectiveness. Removing the SHAP explanatory analysis module decreased the accuracy of decision interpretation and reduced the F1-score by 3.4%, confirming the significance of the XAI component in the final decision. Reducing the SOM map size from 20×20 to 10×10 resulted in a 18% decrease in cluster separability, as measured by the Euclidean metric, and a corresponding decrease in classification accuracy to 0.89. Similarly, disabling adaptive updates to the neighborhood radius and the learning rate led to instability in cluster boundaries and local overtraining.

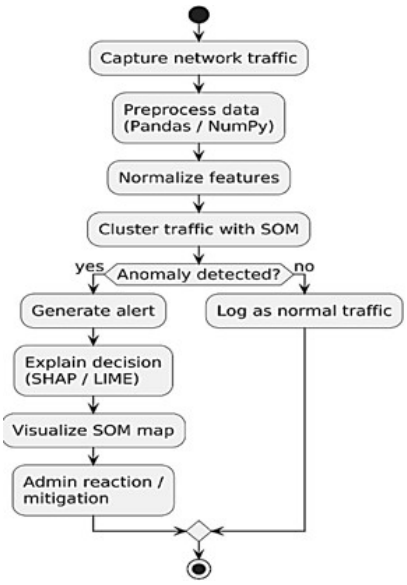

Figure 14: Generalized activity diagram of the method of intelligent anomaly detection in network traffic

The features related to TCP connection intensity, port entropy, and inbound/outbound traffic ratio demonstrated the most significant positive impact on the Detectability Index. The results of the ablation study confirm that the proposed combination of SOM, criticality index, and XAI modules provides a significantly better balance among accuracy, stability, and interpretability than any individual component.

5. Discussion

The proposed method for detecting unauthorized influences combines several approaches to network traffic analysis: signature, statistical, and neural network methods [1, 5, 12, 25, 32, 33]. Such a multi-aspect model significantly increases the reliability of anomaly detection even under low a priori awareness of the attack structure.

A comparison of the effectiveness of different methods revealed that signature-based approaches yield the highest accuracy for known attacks (CD = 0.93), but detect only those impacts that were previously defined [6, 17]. Statistical methods have lower accuracy (CD = 0.82), but demonstrate resilience to sudden changes in traffic characteristics [25, 33]. Neural networks, particularly SOMs, shown a CD of 0.78, yet provided flexible adaptation to new, previously unknown impacts. The advantage of the proposed system lies in the implementation of explainability procedures (Explainable AI), specifically the built-in SHAP and LIME methods for visualizing classification results [11]. This allows system administrators not only to receive an anomaly detection event, but also to understand its causes, which is critically important when making decisions about response [4, 30, 32, 34]. This is consistent with AI-based approaches to web application security, where intelligent models support attack detection, vulnerability analysis, and explainable risk-oriented decision-making [35]. Similar principles are also relevant to wireless network protection, where adaptive threat detection, intelligent traffic analysis, and automated response mechanisms are used to improve the resilience of communication infrastructures [36]. In addition, the SOM classifier's response time after training enables real-time clustering, as confirmed during testing (latency <3 ms/record), which is acceptable even under high load.

Another important aspect is the use of network operating patterns (NOMPs), which allows the formation of expected profiles of normal behavior [9, 25]. This simplified the formalization of deviation rules as mathematical models and intervals applied to traffic characteristics. Calculating the degree of deviation based on the discrepancy between real and template values improved alarm accuracy. Practical testing has demonstrated that combining the three approaches (signature,

statistical, and neural) increases the overall efficiency of the system by 12–17% compared to each method [13, 14, 17, 21]. Visualizing SOM results as RGB maps greatly simplifies the detection of anomaly clusters, especially for slow attacks or destructive pulsations.
However, it is worth noting the limitations of the method in particular, the dependence of accuracy on the quality of the formation of regular functioning patterns [9], the need for periodic retraining of the neural network when changing the topology or network policies [2, 33], as well as the potentially high load on the system when processing a large flow in real time.

The graph (Figure 15) illustrates the comparative effectiveness of four approaches for detecting unauthorized influences in network interactions. The highest accuracy rate is demonstrated by the combined method, which integrates signature, statistical, and neural network analysis, allowing for the combination of their strengths. Signature approaches remain the most effective among individual methods, while neural networks provide a better ability to adapt and detect new, previously unknown attacks.

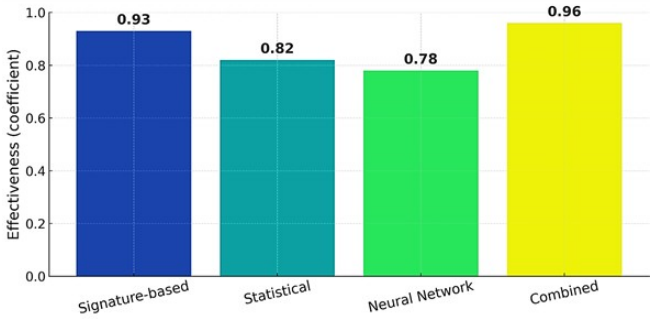

Figure 15: Overall effectiveness of traffic analysis methods

To assess the effectiveness of the proposed method, a comparison was made with other modern approaches to anomaly detection, including One-Class Neural Network (OC-NN), Isolation Forest, and the signature IDS Snort. Table 2 shows the results of the comparison by the metrics F1-score, TPR (True Positive Rate), and FPR (False Positive Rate).

Table 2
Comparison of the effectiveness of the proposed method with other intrusion detection systems

Method	F1-Score	TPR (Recall)	FPR	MCC
Proposed method (SOM+XAI)	0.92	0.95	0.06	0.88
Snort (signature)	0.67	0.72	0.11	0.59
OC-NN	0.84	0.86	0.07	0.76
Isolation Forest	0.78	0.81	0.09	0.69
SVM (One-Class)	0.79	0.80	0.08	0.71

Table 2 shows the comparative results of the proposed method and basic IDS approaches, including Snort, OC-NN (single-class neural network), Isolation Forest, and SVM. The evaluation was performed using the F1-measure, True Positive Rate (TPR), False Positive Rate (FPR), and Matthew's correlation coefficient (MCC) metrics [19, 33]. The data demonstrate higher accuracy and lower false-positive rates for the proposed method, indicating its effectiveness in detecting unauthorized influences in network interactions [8, 11, 25]. Despite its high accuracy, the method is sensitive to the choice of hyperparameters, in particular the size of the SOM map and the dynamics of $\eta(t)$, and requires careful pre-processing of incoming traffic [11, 12, 25, 34]. Future research plans to automate the configuration of these parameters and integrate the system with real traffic through a SIEM solution.

Despite the high efficiency of the proposed methodology, the study has several limitations. SOM performance depends significantly on the correct scaling of features and the choice of hyperparameters, which can affect clustering stability. The use of the CICIDS-2017 and UNSW-NB15 datasets partially limits the external validity of the results, as real enterprise traffic has different dynamics and event densities, which require additional testing on live streams. The explainability of SHAP/LIME-based decisions may decrease in high-dimensional feature spaces, requiring optimized sampling procedures. Additionally, the system requires periodic retraining when the network topology, access policies, or new threat types change.

In further research, it is advisable to consider the use of adaptive security profiles with dynamic parameter updates and integration with SIEM systems to automate responses and ensure continuous control over corporate network information security.

6. Conclusions

The article develops and substantiates a method for detecting unauthorized influences during network interaction using intelligent traffic analysis. The proposed model combines signature, statistical, and neural network approaches, enabling effective detection of both known and new attack types.

The proposed SOM+XAI model achieved an accuracy (F1-score) of 0.92 at TPR = 0.95 and FPR = 0.06, with an optimal map size of 20×20 and a processing time of approximately 0.12 seconds per session. This result demonstrates the effectiveness of the method for detecting unauthorized influences in real traffic. In the future, the system will be deployed as a Docker container with a RESTful interface built using Flask for seamless integration into the network infrastructure. The prospect of integrating the system into a SIEM solution (such as Splunk or Graylog) for real-time event detection is also being considered, as well as adapting the methodology to operate in an edge environment with limited resources.

An automated system has been implemented that can adaptively analyze traffic flows, form and update standard operation templates, and visualize results using SOM maps and characteristic graphs. To improve decision-making accuracy, a multiplicative model is employed to combine results, taking into account the reliability coefficient. Experimental studies in a virtual network infrastructure have demonstrated the high efficiency of the integrated approach (KD = 0.96), which exceeds the results of individual methods. Real-time visualization of results has ensured convenient interpretation of anomalies even without in-depth technical analysis.

The results confirm the feasibility of integrating intelligent algorithms into network impact detection systems to improve resilience and information security. A promising direction for further research is the automatic formation of response rules and the in-depth adaptation of models to dynamic threats in real-world environments. The limitations of the proposed methodology include the sensitivity of the SOM algorithm to the choice of initial weights and map hyperparameters, as well as the dependence of XAI explanation accuracy on the chosen approach (SHAP or LIME). Prospects for further research include integrating the system with a real SIEM environment, deploying it in a containerized environment, and studying its resistance to zero-day attacks.

In future research, we plan to integrate the proposed method into information security management systems (SIEM), which will enable automated real-time traffic analytics. We also plan to test the technique in a real corporate environment with dynamically changing traffic configurations and apply it in cloud architectures. An important prospect is the development of mechanisms for automatic adaptation to new types of influences without the need for complete retraining.

Declaration on Generative AI

While preparing this work, the authors used the AI programs Grammarly Pro to correct text grammar and Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] I. Ullah, Q. H. Mahmoud, Design and Development of RNN Anomaly Detection Model for IoT Networks, IEEE Access 10 (2022) 62722–62750. doi:10.1109/ACCESS.2022.3176317

[2] J. J. Sellers, R. J. Astore, et al, Understanding Space: An Introduction to Astronautics, 2nd Edition, McGraw-Hill Primis Custom Pub; 2nd edition, 2000.I. Hamid, M. M. H. Rahman, AI, Machine Learning and Deep Learning in Cyber Risk Management. Discov. Sustain. 6, 389 (2025). doi:10.1007/s43621-025-01012-3

[3] R. C. Poonia, et al., Real-Time Cyber-Physical Risk Management Leveraging Advanced Security Technologies, in: Yang, XS., Sherratt, S., Dey, N., Joshi, A. (eds) Proceedings of 9th Int. Congress on Information and Communication Technology. ICICT. Lecture Notes in Networks and Systems, vol 1011, 2024. Springer. doi:10.1007/978-981-97-4581-4_25

[4] M. Mylrea, et al., Insider Threat Cybersecurity Framework Webtool & Methodology: Defending against Complex Cyber-Physical Threats, in: 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 2018, 207–216. doi:10.1109/SPW.2018.00036

[5] W. Chua, et al., Web Traffic Anomaly Detection using Isolation Forest. Informatics, 11(4), 83. doi:10.3390/informatics11040083

[6] S. Tamy, et al., An Evaluation of Machine Learning Algorithms to Detect Attacks in Scada Network, in: 2019 7th Mediterranean Congress of Telecommunications (CMT), Fez, Morocco, 2019, 1–5. doi:10.1109/CMT.2019.8931327

[7] J. Feng, et al., Tensor Recurrent Neural Network with Differential Privacy, IEEE Trans. Comput. 73(3) (2024) 683–693. doi:10.1109/TC.2023.3236868

[8] S. Kerimkhulle, et al, Fuzzy Logic and Its Application in the Assessment of Information Security Risk of Industrial Internet of Things. Symmetry 15(10) (2023) 1958. doi:10.3390/sym15101958

[9] Y. Kostiuk, et al., Tools for Providing Information Security from Hidden Threats in Cloud Computing Infrastructure. Cybersecur. Edu. Sci. Technol. 4(28) (2025) 633–655. doi:10.28925/2663-4023.2025.28.857

[10] K. Al-Dosari, N. Fetais, Risk-Management Framework and Information-Security Systems for Small and Medium Enterprises (SMEs): A Meta-Analysis Approach. Electronics, 12(17) (2023) 3629. doi:10.3390/electronics12173629

[11] A. Yayla, L. Haghnegahdar, E. Dincelli, Explainable Artificial Intelligence for Smart Grid Intrusion Detection Systems, IT Prof. 24(5) (2022) 18–24. doi:10.1109/MITP.2022.3163731

[12] M. Nakip, E. Gelenbe, Online Self-Supervised Deep Learning for Intrusion Detection Systems, IEEE Trans. Inf. Forensics Secur. 19 (2024) 5668–5683. doi:10.1109/TIFS.2024.3402148

[13] A. Vedant, et al., Detecting Cyber Attacks in a Cyber-physical Power System: A Machine Learning Based Approach, in: 2022 Global Energy Conference (GEC), Batman, Turkey, 2022, 272–277, doi:10.1109/GEC55014.2022.9986990

[14] F. Alzhouri, et al., A Smart Network Intrusion Detection System for Cyber Security of Industrial IoT, in: 2023 4th Int. Conf. on Intelligent Data Science Technologies and Applications (IDSTA), Kuwai, Kuwait, 2023, 67–75. doi:10.1109/IDSTA58916.2023.10317861

[15] A. Dhakad, et al., Real Time Network Traffic Analysis using Artificial Intelligence, Machine Learning and Deep Learning: A Review of Methods, Tools and Applications, in: 2023 Int. Conf. on Self Sustainable Artificial Intelligence Systems (ICSSAS), Erode, India, 2023, 372–378. doi:10.1109/ICSSAS57918.2023.10331855

[16] R. Sinha, P. Agrawal, A. Rasool, Hyperparameter Tuned Cloud Based Cyber Physical Attack Detection using Stacking Ensemble Learning, in: 2024 IEEE Int. Students' Conf. on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2024, 1–6. doi:10.1109/SCEECS61402.2024.10482067

[17] H. S. Divyashree, et al., Enhanced DDoS Attack Detection in Cyber-Physical Systems Within Supply Chain 4.0 using Optimized TSO-DBN Deep Learning Technique, in: 2024 IEEE Int. Women in Engineering (WIE) Conf. on Electrical and Computer Engineering (WIECON-ECE), Chennai, India, 2024, 151–156. doi:10.1109/WIECON-ECE64149.2024.10914761

[18] T. Roberts, S. J. Tonna, Artificial Intelligence, Machine Learning, and Deep Learning Models for Risk Management, in: Risk Modeling: Practical Applications of Artificial Intelligence, Machine Learning, and Deep Learning , Wiley, 2022, 31–54. doi:10.1002/9781119824961.ch3

[19] S. Ankalaki, et al., Cyber Attack Prediction: From Traditional Machine Learning to Generative Artificial Intelligence, IEEE Access 13 (2025) 44662–44706. doi:10.1109/ACCESS.2025.3547433

[20] I. Opirskyy, et al., Detection of Network Attacks based on the RedHunt Virtual Machine, Adv. Cybersecur, 1st ed., 2025, 310–330.

[21] P. Skladannyi, et al., Development of Modular Neural Networks for Detecting Different Classes of Network Attacks. Cybersecur. Edu. Sci. Tech. 3(27) (2025) 534–548. doi:10.28925/2663-4023.2025.27.772

[22] V. P. M. R, V. S. Vardhan, S. S, V. k. K, M. M. R. G, AI-Driven Cyber Threat Detection and Log Analysis, in: 2025 Int. Conf. on Inventive Computation Technologies (ICICT), Kirtipur, Nepal, 2025, 676–681. doi:10.1109/ICICT64420.2025.11004938

[23] Y. Kostiuk, et al., Models and Technologies of Cognitive Agents for Decision-Making with Integration of Artificial Intelligence, in: Modern Data Science Technologies Doctoral Consortium (MoDaST 2025), vol. 4005, 2025, 82–96.

[24] O. Harasymchuk, et al., Intelligent Cyber Defence Systems: Detection of Ransomware and Protection of Wireless Networks based on Artificial Intelligence Technologies, Technology Center PC, 2025. doi:10.15587/978-617-8360-22-1

[25] O. Harasymchuk, et al., Modern Methods of Ensuring Information Protection in Cybersecurity Systems using Artificial Intelligence and Blockchain Technology, Technology Center PC, 2025. doi:10.15587/978-617-8360-12-2

[26] Y. Kostiuk, et al., Effectiveness of Information Security Control using Audit Logs, in: Cybersecurity Providing in Information and Telecommunication Systems (CPITS), vol. 3991, 2025, 524–538.

[27] E. Caville, W. W. Lo, S. Layeghy, M. Portmann, Anomal-E: A Self-Supervised Network Intrusion Detection System based on Graph Neural Networks, Knowl.-based Syst. 258 (2022) 110030.

[28] Y. Kostiuk, et al., Integrated Protection Strategies and Adaptive Resource Distribution for Secure Video Streaming over a Bluetooth Network, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3826 (2024) 129–138.

[29] V. S. S. R. Nallapareddy, Modified AI-based Learning Approach to Secure Data using Deep Learning Algorithms, in: 2024 Int. Conf. on Intelligent Computing and Emerging Communication Technologies (ICEC), Guntur, India, 2024, 1–7. doi:10.1109/ICEC59683.2024.10837269

[30] S. Vasylyshyn, et al., A Model of Decoy System based on Dynamic Attributes for Cybercrime Investigation, East.-Eur. J. Enterpr. Technol. 1(121) (2023) 6–20. doi:10.15587/1729-4061.2023.273363

[31] Y. Kostiuk, et al., Ensuring Cybersecurity and High-Speed Data Transmission in Wireless Networks, Inf. Secur. 30(3) (2024) 365–375.

[32] Y. Kostiuk, et al., Intelligent System for Simulation Modeling and Research of Information Objects, in: 1st Workshop Software Engineering and Semantic Technologies (SEST), vol. 4053 (2025) 237–251.

[33] P. Skladannyi, et al., Model and Methodology for the Formation of Adaptive Security Profiles for the Protection of Wireless Networks in the Face of Dynamic Cyber Threats, in: Workshop Cyber Security and Data Protection (CSDP), vol. 4042 (2025) 17–36.

[34] J. Hutchins, D. Ferrer, J. Fillers, C. Schuman, Black-Box Adversarial Attacks on Spiking Neural Network for Time Series Data, in: Int. Conf. on Neuromorphic Systems (ICONS), Arlington, VA, USA, 2024, 229–233. doi:10.1109/ICONS62911.2024.00040

[35] Partyka O., Partyka A., Imoize A.L. (2026) AI for Web Application Security. *Handbook of Cybersecurity Challenges and Solutions for Emerging Technologies*. DOI: doi:10.1201/9781003640790-14

[36] Opirskyy I., Partyka O., Partyka A., Imoize A.L. (2026) AI for Wireless Network Protection. *Handbook of Cybersecurity Challenges and Solutions for Emerging Technologies*. DOI: doi:10.1201/9781003640790-15