Pakistan Journal of Life and Social Sciences
www.pjlss.edu.pk

https://doi.org/10.57239/PJLSS-2026-24.1.0035

RESEARCH ARTICLE

An Intelligent Information Security System for Detecting and Preventing Network Attacks

Elena Revyakina*[1], Andrey Gazizov[2]

[1,2]Don State Technical University, Rostov-on-Don, Russia

ARTICLE INFO

Received: Apr 20, 2026

Accepted: May 18, 2026

Keywords

Intelligent information
Security System
Network attack detection
Neural networks
Network traffic analysis
Machine learning
Firewall

***Corresponding Author:**
revyelena@yandex.ru

ABSTRACT

This article considers the problem of ensuring information security of critical information infrastructure facilities in the face of the increasing number and complexity of cyber-attacks. The development of an intelligent information security system designed to detect and prevent network attacks based on the analysis of quantitative and temporal characteristics of network traffic is presented. The system implements a comprehensive approach combining active network packet capture at the network card level, calculation of key network connection metrics, and their classification using a recurrent neural network. A comparative study of MLP, CNN, and RNN architectures is conducted, demonstrating the advantage of the recurrent model in terms of accuracy and processing speed. The developed system integrates with the iptables firewall, providing automatic blocking of malicious IP addresses. Laboratory testing results confirmed the system's ability to effectively recognize various types of network attacks, including port scanning, dos attacks, and brute force, while maintaining high performance.

INTRODUCTION

The modern development of information technology is accompanied by a steady increase in the complexity and interconnectedness of critical infrastructure, leading to an increase in information security threats. With the digitalization of the economy and public administration, ensuring the security of such infrastructure is becoming a strategically important task. The rapid development of artificial intelligence methods makes it possible to create intelligent systems capable of not only identifying threats in real time but also making prompt decisions to minimize the consequences of incidents.

The relevance of developing an intelligent information security system is driven by several factors. First, the increasing complexity and diversity of information security threats in a tense geopolitical environment necessitates a transition from traditional security measures to more flexible and adaptive systems leveraging machine learning and big data analytics. Second, critical information infrastructure plays a key role in the functioning of modern society, and their compromise can lead to serious economic and social consequences (Slepovichev et al., 2009). The goal of this work is to develop an intelligent information security system that ensures the protection of critical information infrastructure from information security threats.

The theoretical significance of the study lies in its examination of artificial intelligence methods for recognizing malicious network activity, as well as in identifying network connection metrics relevant for neural network training and establishing a relationship between a specific malicious network event and specific network characteristics. The practical significance of the work lies in the potential for enhancing the security of information infrastructure objects and creating products based on the developed software for direct implementation in industry.

Theoretical Aspects of Ensuring the Protection of Critical Information Infrastructure

The primary objective of any data security solution is to minimize the possibility of a successful cyberattack. Intelligent information security systems are tasked with systematically identifying and analyzing incidents and predicting potential threats. Incident detection systems must ensure continuous monitoring of ongoing events, preventing downtime (Tsvetkova, Aidinyan, 2014).

The key task of a specific intelligent information security system is to collect data streams from information sources on the workstation. The obtained data is then prepared for subsequent analysis, filtering out noise and correcting incomplete data. These actions help minimize false positives (positive) triggering of the intelligent system. The task of recognizing anomalies and incident markers falls to the intelligent module, which must have a high-quality training base. The applied methods of machine learning and predictive analytics cope with the task of detecting attacks that adapt to a specific victim, showing a more accurate recognition result compared to standard signature-based methods. This is facilitated by neural networks and clustering, capable of detecting changes that are at first glance imperceptible even to a specialist, significantly increasing accuracy (Gladkikh, 2020). The adaptability of the system is based on maintaining adequate operability, without a critical deterioration in response accuracy with an increase in incoming traffic or increased consumption of system resources. When designing such systems, it is necessary to initially account for the possibility of increasing incoming traffic, setting throughput values in the range of 10 Gb / s - 100 Gb / s without packet loss, with acceptable delays (Shamsutdinov et al., 2024). Such indicators will allow solving problems in the event of increased peak loads without loss of performance.

Beyond detection, such systems are tasked with adequately responding to incidents. An intelligent information security system must have mechanisms for tight integration with existing host firewalls (or possess firewall capabilities itself), with the ability to insert a new rule based on the detected incident. Based on the response, a report on the incident is automatically generated and subsequently notified to the responsible person within the organization. The information entity is obligated to notify the regulatory authority of the incident; the preparation and automatic transmission of this information can also be delegated to the intelligent information security system for more rapid notification. The actions of the intelligent system should be recorded in a detailed log for subsequent analysis in the event of an emergency.

Analyzing forums and websites dedicated to information security, one can often find articles devoted to the complexities and weaknesses of information security systems. Websites such as securitymedia.org and securitylab.ru contain articles with varying opinions on the integration and operation of information security systems (Fisun, 2020). Issues highlighted include high sensitivity to false positives, which is a consequence of poorly thought-out software architecture and the use of inappropriate datasets. Many such solutions utilize outdated methods of detecting malicious activity, such as signature data, which can effectively only detect attack profiles known to their signatures.

This leads to a further drawback: the inaccuracy of recognizing previously unknown attack types, which may differ slightly from known ones but still have an unknown signature for databases. Such solutions are mostly commercial products, with high implementation and maintenance costs. The complexity of such systems adds to the need for additional training for information security specialists, which increases the financial requirements for building reliable protection based on such systems.

Analysis of Artificial Intelligence Methods for Recognizing Network Attacks

Artificial intelligence and artificial neural networks are at the core of the intellectualization of existing information security technologies. An artificial neuron, through mathematical modeling, mimics the operation of a human neuron, with inputs (simulating dendrites), weights that determine the significance of input data, an adder, an activation function, and an output (simulating an axon). Artificial neurons, interconnected in a network, form an artificial neural network capable of solving various types of problems, including regression, multi-class classification, and recognition.

Various neural network architectures are used for network attack detection tasks. Multilayer The multilayer perceptron (MLP) is the basic architecture of a multilayer fully connected network, whose advantages include versatility and ease of implementation. However, MLP exhibits quadratic growth of parameters with increasing input features and is incapable of processing temporal and spatial relationships. Convolutional neural networks (CNNs) specialize in data with spatial properties and use specialized filter layers to extract local patterns (Vasiliev, Shamsutdinov, 2016). Recurrent neural networks (RNNs) are tailored to solving problems with sequential data containing time series by creating cyclic connections between layers of neurons, which allows for the preservation of information from previous inputs and introduces the concept of contextual memory.

Intelligent information security systems are technically complex software, hardware, and hardware-software systems. Their architecture is based on data-collecting sensors. When monitoring network anomalies, sensors are installed on critical network nodes using network TAP devices or port mirroring. The collected data is analyzed by an analytics engine that utilizes two primary methods: signature analysis, based on comparing data with patterns of known attacks, and behavioral analysis, based on intelligent statistical and algorithmic analysis systems using machine learning.

For IPS-class systems, a response module that automates the blocking of potential attacks is critical. Deep integration with firewalls allows for the creation of new rules to prevent repeat attacks (Kotenko et al., 2021). Behavioral analysis in such systems is based on the creation of patterns of host, user, or software behavior using statistical methods, which are then used to construct a normal distribution and a model of behavior under typical conditions.

Network traffic is a complex system of interactions between devices, where each parameter and their combination can convey information about potential danger. Key internet connection metrics that correlate with legitimacy and malicious activity include timestamps, source and destination IP addresses, ports, transport layer protocol, network flow duration, number of packets in both directions, volume of data transferred, mean and standard deviation of interpacket intervals, number of TCP flags of various types, average packet length, and average activity and pause times between packets (Turdieva, 2022).

Analyzing these parameters allows us to identify characteristic signs of various types of attacks. Short streams lasting less than one second may indicate port scanning or DoS attacks. Long streams exceeding ten minutes are often associated with communication with the C&C server or a data leak. Asymmetry in the number of packets in the forward and reverse directions may indicate DoS attacks or data leaks. High SYN flag values are often associated with SYN flood attacks, and abnormal activity using the PSH flag indicates forced payload sending during vulnerability exploitation.

Development of an Information Security System Based on Intelligent Recognition of Network Attacks

The software being developed is designed to operate both on a server and on an automated workstation (AWS), actively capturing data about ongoing internet connections (Internet sniffer). Program launch and access to executable files should be restricted to root users (users with administrator rights). Since the intelligent system is expected to operate on servers, possibly also on an AWS, a graphical interface is not essential. An interface would consume more system resources and offer no significant advantage over terminal-based program launch and management. Since encryption of traffic via various protocols is standard practice in network data transmission, a method for detecting malicious activity independent of the presence or absence of encryption should be employed to facilitate the task (Guts, Enns, 2017). Therefore, qualitative and quantitative data will be used, providing a reliable method for recognizing a network attack. To prevent the operating system from influencing the received data, direct interaction with the server or AWS network card is necessary.

Based on the received data and previously obtained indicators, the intelligent module must analyze and conclude whether the current connection contains malicious traffic. The intelligent module

uses one of the artificial neural network models. The specific architecture must be selected experimentally, by determining accuracy, performance, and training time.

Since network traffic is a complex structure with a large number of variables associated with quantitative and qualitative metrics, creating a custom training dataset at this stage of program development is impractical. Therefore, one of the readily available open-source training datasets specialized for network attacks should be used. When training an artificial neural network, the selected dataset should be relevant to current challenges, with relevant types of network attacks and metrics. It is possible to use pre-prepared data, normalized and standardized. Otherwise, to improve accuracy and reduce false positives, the data should be prepared before training, removing all duplicates and anomalous values (Lapshakova et al., 2022). Based on the neural network model that demonstrated the best accuracy performance and the selected training dataset, an artificial neural network model should be trained and saved.

For each new network connection, an intelligent assessment of its maliciousness is made, and if a threat is detected, a response mechanism should be triggered. In this case, interaction with the operating system's default firewall should occur. This is iptables, which is included in Linux distributions. It is important to consider the firewall's operation, as it operates at a higher level of abstraction than the network card. The program should determine whether the incoming IP address has already been blocked. This avoids excessive load on the intelligent module and prevents duplicate rule chains in iptables. The firewall rule issues a command to terminate the connection and rejects any new connection attempt from this IP address. The responsible party should also be notified via email of the incident, along with information about the IP address and the time of the incident.

The program's architecture represents interconnected elements with a server, and network packet data is received from the network adapter. These packets are collected for each specific connection, and quantitative and qualitative parameters are calculated based on them, allowing for the identification of malicious activity. These parameters are analyzed in real time by an artificial neural network, previously trained and saved in a file with the ".h5" extension. Pre-prepared data in a CSV file was used for training. When a deviation in network connection parameters is detected, a response is sent by sending a command to the firewall. The firewall, in turn, communicates with the network adapter, disconnecting and blocking further connections to the malicious source. To prevent overload, the intelligent component retrieves information from the firewall about previously blocked IP addresses of sources and generates a report.

The Algorithm of the Developed Intelligent System and its Training

Linux operating systems, which is driven by modern requirements and the need to use certified software in critical information infrastructure facilities. The AltLinux distribution, which is targeted at government and corporate information infrastructures and provides long-term support with updates that fix vulnerabilities and critical errors, was chosen as the testing platform.

An intelligent information security system that works with network traffic detects and prevents network attacks by performing a series of sequential actions. The program's operations are divided into three sequential global actions: network connection monitoring, intelligent network connection analysis, and incident response based on intelligent analysis.

The first step involves monitoring network activity and tracking basic IP packet header parameters, timing, and quantitative metrics. The operating system and system firewall do not affect packet reception, as the program operates at the network card level. Upon receiving a packet, the system tracks its TCP flags and other metrics. Thanks to multithreading, the program can process incoming packets in multiple threads, "remembering" active connections (Revyakina, Gazizov, 2025). This "remembering" capability enables tracking metrics such as the time between packets, the total weight of all packets, and so on. Throughout the entire time the intelligent system is running, packets are collected for each specific connection. The extracted and calculated data is then passed on to the intelligent network connection analysis system, after undergoing preprocessing for more accurate prediction and the adoption of an appropriate format.

The first received data from a packet is immediately fed to a pre-trained artificial neural network for intelligent analysis. Training occurs by providing a labeled dataset (supervised learning), which contains network target metrics and a class label characterizing anomalous or legitimate traffic. After training, the intelligent module is capable of astatically analyzing and classifying network traffic based on previously defined parameters. Incoming connection parameters are analyzed, and the final class label for that connection is predicted (Cherckesova et al., 2024a). If an anomaly is detected in the connection, a response process is initiated; if the route is legitimate, the response stage is skipped.

If a network anomaly is detected in a network connection, an incident response process is initiated. This response is triggered by the firewall, which adds a corresponding rule to terminate the connection. This rule also prevents new connections and discards all incoming packets from the blocked IP address. Only system administrators have access to the firewall, and deleting a rule from it also requires administrative rights (Cherckesova et al., 2024c). The program is an automated add-on to firewalls, capable of automatically taking actions aimed at preventing a network attack.

The second stage of the algorithm uses a trained artificial neural network capable of recognizing network anomalies. The training process differs from conventional programming or any other type of algorithm. Training is the automatic adaptation of all neuron connections (their weights) to optimally solve a given problem. A trained neural network, using training data, finds complex relationships, gaining the ability to solve the problem on new, initial input data (provided the data relates to the same subject area).

A supervised learning method was used to train artificial neural network models. In this type of training, the dataset is divided into two components: a set of input features and the target response values (class, number). When received by the neural network, the input data is processed using the initial weight values (at the first iteration, the weight values are randomly selected). The output value of the neural network is compared with the target value for this input parameter vector (Cherckesova et al., 2024b). Two methods are used to calculate weight value adjustments during training: backpropagation and forwardpropagation. Backpropagation involves a more complex process involving calculating the error gradient and propagating it in the opposite direction, from the network output to its output, followed by weight adjustments.

Using the forward error propagation method, input data passes through all layers of the intelligent system, producing the final result at the network's output. Error is defined as the difference between the expected value and the actual output of the neural network. The training algorithm for an artificial neural network is shown in Figure 1.

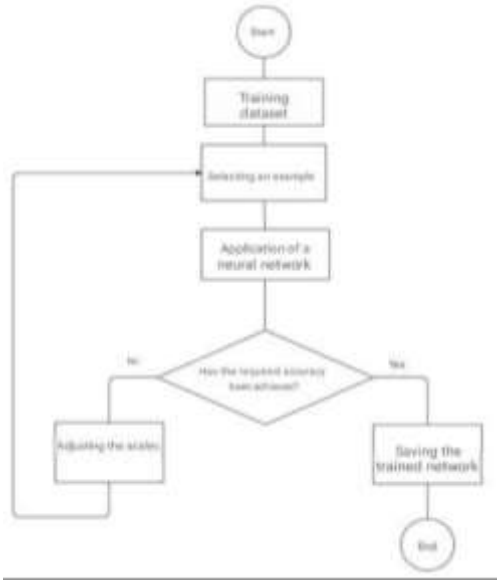

Figure 1: Artificial neural network training algorithm

The software is implemented using the object-oriented Python programming language, which boasts a rich ecosystem of specialized libraries for information security and network analysis. The Scapy library provides low-level access to network traffic, enabling the interception, modification, and analysis of network packets at Layer 2 of the OSI network model. Keras and TensorFlow are used to create and train neural networks, providing access to a variety of architectures and enabling computational optimization. Scikit-learn is used for data preprocessing, normalization, standardization, and evaluation of training results. Pandas provides convenient work with tabular data, and NumPy enables mathematical calculations and multidimensional matrices.

The CSE-CIC-IDS2018 dataset, created by the Canadian Cybersecurity Institute in collaboration with the Communications Security Agency of Canada, was selected for training the intelligent module. This dataset was implemented on the Amazon cloud computing platform. Web Services and takes into account the limitations of the lab environment, expanding the list of attack machines to over 50 devices, and the victim network comprising over 400 workstations and 30 servers, divided by a realistic network topology comprising five functional units. The simulation ran for ten days, during which seven attack categories were implemented, including brute-force, web-based attacks and DoS.

Preparing the data for training included removing duplicates, combining ten types of malicious activity into a single class of “malicious traffic” (the process of combining classes is shown in Figure 2), and reducing the number of features from 78 to the 14 most significant indicators corresponding to the previously considered metrics (the reduction in the number of columns with features is shown in Figure 3).

```
import pandas as pd
import numpy as np
ids_dataset = pd.read_csv("ids2018_sample.csv")

# Замена бесконечности на NaN
ids_dataset.replace([np.inf, -np.inf], np.nan, inplace=True)
# Удаление пропущенных значений
ids_dataset.dropna(inplace=True)
# Удаление дубликатов
ids_dataset.drop_duplicates(inplace = True)
# Сокращаем количество классов до двух
ids_dataset['Label'] = ids_dataset['Label'].replace(1, 0)
for i in range(1,12):
    ids_dataset['Label'] = ids_dataset['Label'].replace(1, 1)
```

Figure 2: Data preparation and merging of malicious activity classes

After preprocessing, the total number of rows in the dataset was approximately 511,000, equally divided between the two classes. The data were standardized to zero mean and unit variance.

The optimal neural network model was selected by testing three architectures: a multilayer perceptron (MLP), a convolutional neural network (CNN), and a recurrent neural network (RNN). For each model, various hyperparameters were considered: the number of layers, the number of neurons in the hidden layers, and the size of the training set. batch, number of epochs. The Dropout method was used between layers to reduce overfitting. The hidden layer activation function was ReLU , and the output layer activation function was Sigmoid (binary classification). Adam was chosen as the optimization algorithm, and the loss function was binary cross-entropy.

```
def filter_columns_in_csv(output_path, columns_to_keep):
    try:
        # Чтение CSV файла
        df = pd.read_csv("C:\ids2018_sampled.csv")
        # Проверка, что все указанные столбцы присутствуют в файле
        missing_cols = [col for col in columns_to_keep if col not in df.columns]
        if missing_cols:
            raise ValueError(f"Следующие столбцы отсутствуют в файле: {missing_cols}")
        # Фильтрация DataFrame по нужным столбцам
        filtered_df = df[columns_to_keep]
        # Сохранение в новый CSV-файл
        filtered_df.to_csv(output_path, index=False)
    except Exception as e:
        print(f"Ошибка: {e}")
if __name__ == "__main__":
    output_file = "D:/smaller_data_normal_col.csv"
    columns = ["Flow Duration","Tot Fwd Pkts","Tot Bwd Pkts","Flow Byts/s",
"Flow Pkts/s","Flow IAT Mean","Flow IAT Std","SYN Flag Cnt","RST Flag Cnt",
"PSH Flag Cnt","ACK Flag Cnt","Pkt Size Avg","Active Mean","Idle Mean","Label"]
    filter_columns_in_csv(output_file, columns)
```

Figure 3: Reducing the number of columns with features

Training results showed that the multilayer perceptron model was inferior to the convolutional and recurrent models in terms of accuracy and loss. The convolutional and recurrent neural networks demonstrated comparable results, but the recurrent network provided higher processing speed, which is critical for real-time systems. The best results were achieved using an RNN model with two recurrent layers (64 and 32 neurons, respectively) and one hidden layer of a fully connected network with 32 neurons, trained for 20 epochs with a batch size of 128. The finished software runs on the AltLinux operating system; all necessary files are contained in a single directory.

The final program contains two Python files: a saved file with the trained neural network model and a file with input data normalization parameters. The main file, ISZI_start.py, is the primary one and contains the functionality for intelligent analysis and incident response with alert sending. After launch, the DataCollector class from collector.py is called, which implements the network sniffer logic and calculates the target quantitative and qualitative internet connection metrics.

When receiving a packet, the system monitors TCP flags and other metrics, using multithreading to process incoming packets in multiple threads while maintaining information about active connections. This allows for tracking metrics such as the time between packets and the total weight of all packets throughout the entire operation. Extracted and calculated data is fed into a pre-trained recurrent neural network for intelligent analysis, which analyzes and classifies network traffic in a non-static mode based on previously defined parameters.

If a connection anomaly is detected, a response process is initiated through the iptables firewall. The system adds a rule to terminate the connection and discards all incoming packets from the blocked IP address. Before blocking, the IP address is checked for inclusion in existing iptables rules to prevent excessive load on the intelligent module. Incident detection is accompanied by an alert to the responsible party.

The functionality of the intelligent information security system was tested in laboratory conditions using the AltLinux operating system. To verify the correct recognition of legitimate activity, visits to the websites ya.ru and wikipedia.org were performed. The intelligent part of the system correctly recognized these connections as legitimate and did not take any blocking actions, as evidenced by the absence of new rules in the firewall (Figure 4).

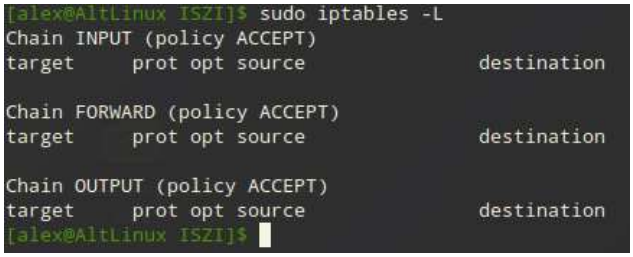

Figure 4: Rules for the interaxial screen for a legitimate connection

To test the detection of a malicious connection, the attacker's behavior was simulated by scanning the system's open ports using the Nmap network scanner from the Kali distribution. Linux. The developed intelligent information security system successfully identified the malicious connection and immediately responded to the incident. The IP address of the machine running Kali Linux was blocked by the firewall, as evidenced by the appearance of a corresponding rule in iptables (Figure 5).

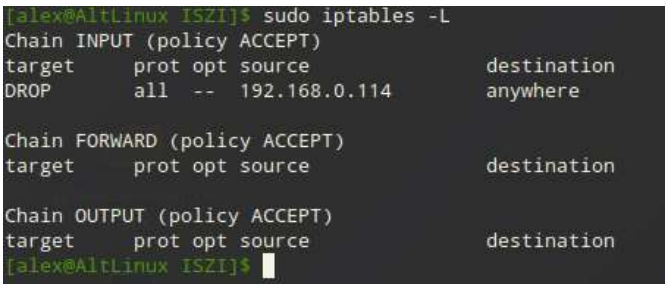

Figure 5: Firewall rule when blocking an IP address

Port scanning was ineffective, confirming the effectiveness of network attack prevention. Lab tests demonstrated that the developed intelligent information security system correctly and effectively distinguishes between legitimate and malicious traffic.

CONCLUSION

During the course of this work, an intelligent information security system was developed that can effectively detect network attacks based on the timing and quantitative characteristics of network traffic. During the development process, programming tools were used to create a network sniffer and define and train an artificial neural network. A study of the modern information infrastructure sector and an analysis of past information security incidents identified the main threats and vulnerabilities typical for this area. An analysis of modern information security approaches and technologies demonstrated the need to transition from signature-based attack detection methods to behavioral methods based on machine learning.

A study of artificial intelligence methods for anomaly and threat detection enabled the identification of the most suitable neural network architectures for network traffic classification. The architecture of intelligent information security systems is examined using examples of intrusion detection and prevention systems. Key requirements for the system under development are outlined, including operation on domestic operating systems, active traffic capture at the network interface card level, the use of quantitative and temporal metrics independent of encryption, automatic response through the firewall, and notification of responsible personnel.

The developed intelligent information security system has been tested for its ability to recognize and prevent network attacks. The CSE-CIC-IDS2018 training dataset was selected and prepared, and a recurrent neural network was trained, demonstrating the best results in terms of accuracy and processing speed. Laboratory testing confirmed the system's effectiveness in recognizing both legitimate traffic and malicious connections, including port scanning.

Further development of the software product may include the creation of its own training data set in the operational environment to improve recognition accuracy, the development of an intelligent host activity recognition module for comprehensive information system protection, as well as integration with SIEM systems and the expansion of notification channels to improve the convenience of interaction with information security specialists.

REFERENCES

Cherckesova L, Revyakina E, Buryakova O, Gazizov A. Creation of an encryption algorithm resistant to attacks through side channels of leakage. E3S Web Conf 2024a,583:06011. http://dx.doi.org/10.1051/e3sconf/202458306011

Cherckesova L, Revyakina E, Roshchina E, Porksheyan V. The development of countermeasures against session hijacking. E3S Web Conf 2024b,531:03019. https://doi.org/10.1051/e3sconf/202453103019

Cherckesova L, Revyakina E, Safaryan O, Porksheyan V, Kazaryan M. Analysis of the possibilities of carrying out attacks on the functions of transferring control to operating system console using active intelligence methods. Int Res J Multidisc Scope 2024c,5:516-34. http://dx.doi.org/10.47857/irjms.2024.v05i02.0558

Fisun VV. Intellectual information security management system of critical infrastructure objects. Sci Prosp 2020,11:181-6.

Gladkikh AM. Osnovnyye metody analiza setevogo trafika [Basic methods of network traffic analysis]. Iss Sci Educ 2020,19:23-8.

Guts AK, Enns EP. Program for simulation of computer network and network attacks. Math Struct Model 2017,3:139-49. https://doi.org/10.24147/2222-8772.2017.3.139-149

Kotenko IV, Saenko IB, Branitskiy AA, Parashchuk IB, Gaifulina DA. Intelligent system of analytical processing of digital network content for protection against inappropriate information. Inform Automat 2021,20:755-92. https://doi.org/10.15622/ia.20.4.1

Lapshakova AV, Milyutina AM, Juraeva DKh, Khalyavin NI. Network attacks based on machine learning. In: Nauka i Obrazovaniye v Epokhu Peremen: Perspektivy Razvitiya, Novyye

Paradigmy [Science and Education in the Era of Change: Development Prospects, New Paradigm]. Kaluga: Manuskript; 2022. p. 44-7.
Revyakina E, Gazizov A. Development of methods and tools for implementing and detecting network steganography. Pak J Life Soc Sci 2025,23:238-48. https://doi.org/10.57239/PJLSS-2025-23.2.00161
Shamsutdinov RR, Vasiliev VI, Vulfin AM. Intelligent system for monitoring information security of the industrial internet of things using artificial immune systems mechanisms. Syst Eng Inform Technol 2024,6:14-31. https://doi.org/10.54708/2658-5014-SIIT-2024-no4-p14
Slepovichev II, Irmatov PV, Komarova MS, Bezhin AA. DDoS attack detection using fuzzy neural network. Izvestiya Saratov Univ Math Mech Inform 2009,9:84-9. https://doi.org/10.18500/1816-9791-2009-9-3-84-89
Tsvetkova OL, Aidinyan AR. Intelligent system evaluation information security of the enterprise from internal threats. Herald Comput Inform Technol 2014,8:48-53. https://doi.org/10.14489/vkit.2014.08.pp.048-053
Turdieva GS. Network attacks and their use of protection. Universum Tech Sci 2022,2-1:60-2.
Vasiliev VI, Shamsutdinov RR. Intelligent system of information security incident analysis (based on the methodology of siem-systems using immunocomputing mechanisms). Model Optimiz Inform Technol 2019,7:536-47. https://doi.org/10.26102/2310-6018/2019.24.1.011