# Development of a Network Attack Detection System Based on Hybrid Neuro-Fuzzy Algorithms

Olexander Belej[1] [0000-0003-4150-7425], Liubov Halkiv[2] [0000-0001-5166-8674]

[1] Lviv Polytechnic National University, 5 Mytropolyt Andrei str., Building 4, Room 324, Lviv 79013, Ukraine

Oleksandr.I.Belei@lpnu.ua

[2] Lviv Polytechnic National University, 12, Stepan Bandera str, 79013, Lviv, Ukraine

Lubov.I.Halkiv@lpnu.ua

**Abstract.** The imperfection of existing intrusion detection methods, as well as the changing nature of malicious actions by the attacker, make computer systems unsafe, therefore it is important to identify new types of attacks and respond to them on time. The developed hybrid scheme for detecting and classifying network attacks based on a combination of adaptive classifiers. The study proposed a generalized scheme for combining classifiers to detect network attacks. Based on it, a software tool has been developed that allows you to analyze network traffic for the presence of abnormal network activity. To reduce the number of features used, it is proposed to use the method of principal components. The main feature of the proposed approach is a multilevel analysis of network traffic, as well as the use of various adaptive modules in the attack detection process. Computational experiments were carried out on two open data sets using various methods of combining classifiers. The developed modules can be used to process data received from sensors of the information and security events management system.

**Keywords**: network attack, intrusion detection system, algorithm, signal, traffic, protocol, transmission.

## 1 Introduction

The detection of network attacks is one of the urgent problems of information security due to the rapid development of computer technology and the imperfection of existing methods for detecting attacks. To correct the situation in this area, the current efforts of researchers are aimed at finding and applying new, including hybrid and adaptive, detection schemes.

An intrusion detection system (IDS) is an important component of computer network protection. Its main task is to monitor the network or system for malicious activity. Although the problem of detecting network attacks is quite old, it is still relevant.

A few years ago, network intrusions were rare because they required extensive knowledge of operating systems, network protocols. Today, any user can carry out malicious actions on the network by downloading one of the exploit programs available on the Internet.

One of the most popular areas of development of computer security systems today is the technology of artificial immune systems [1]. It is assumed that the use of methods and techniques for protecting the body from germs and viruses will overcome the problems of classical means of ensuring information security.

In this paper, an attempt is made to create an intrusion detection system based on the immune model. Based on the architecture of the IDS developed by Kim and Bentley, which is called the immune model of intrusion detection [2]. Among others, it most closely mimics the behavior of a biological immune system. However, according to the authors of this article, some properties of immunity are absent in the Kim and Bentley model, which negatively affects its effectiveness.

The detection of network attacks is one of the urgent problems of information security due to the rapid development of computer technology and the imperfection of existing methods for detecting attacks. To correct the situation in this area, the current efforts of researchers are aimed at finding and applying new, including hybrid and adaptive, detection schemes.

**Formal problem statement**

In this study, an attempt is made to identify network attacks based on the analysis of connections characterizing the activity of remote hosts. In this case, monitoring network activity usually consists of the following steps. At the first stage, the filtering of incoming and outgoing packets collected with the help of sensor agents installed by controlled network nodes is performed. At this stage, packet aggregation also takes place to form signs of an established session concerning a unique pair of sockets. At the next stage, the data collected from the first level is stored in a repository for their subsequent processing and interpretation. At the last stage, the administrator is notified of the detected changes in the security policy and offers possible solutions to prevent the further development of the invasion.

Network IDSs are based on the analysis of network traffic that passes through the sensors and, in the event of anomalies, is sent to the analyzers for further decision making. In large information systems (IS), the problem of a large amount of network traffic arises, since standard approaches to data processing are no longer effective. To increase the efficiency of identifying situations associated with a possible intrusion, recently, modern technologies of data mining have been widely used. One of the most effective approaches to classifying a large amount of data is the use of neural networks. This method allows you to detect not only already known network attacks, but also to identify new ones.

The main task of processing network data lies with the intrusion detection subsystem, which is responsible for analyzing network traffic. The effectiveness of IDS directly depends on the chosen method of constructing an algorithm for detecting attacks and processing a large amount of network data. The aim of this work is to develop an intrusion detection algorithm by processing and filtering a large amount of network traffic, based on an artificial neural network.

## 2 Literature review

Currently, many works investigate the applicability of the methods in question to detect attacks. We will present some of these works in more detail. As a training set for artificial neural networks, various data can serve. So, in [3, 4], to solve this problem, it is proposed to use a multilayer neural network trained on data from a variety of Snort signatures. In works [5, 6] many KDD Cup records are used as training data.

Modeled connections are another source of data. An example of a work in which the author used a three-layer direct distribution neural network as a binary classifier of network connections is an article [7]. These works are united by the general idea of using one classifier aimed at identifying illegitimate traffic.

Other authors [8, 9] propose a model of the immune system with the life cycle of T-lymphocytes, which took into account the process of their maturation in the thymus, as well as activation, the formation of immune memory cells through a stimulation signal and the death of immune cells. The lightweight immune system (LISYS) developed is designed to detect intrusions in a distributed environment. Although this system has several advantages, including relatively low computational overhead, reliability, and scalability, the authors note some of the disadvantages. The inability to detect network attacks using the UDP protocol and the ability to trick the system by conducting distributed attacks over time such as slow-scanning [10]. It provides an overview of a two-level intrusion detection system, which combines the advantages of Kohonen's immune systems and networks. At the first stage, the signs of network connections are filtered using immune detectors trained by the negative selection method, thereby eliminating those samples that correspond to normal connections. In the second stage, abnormal instances are processed by self-organizing cards and grouped into separate clusters with similar features. It also presented a two-level model in which immune systems were used to detect anomalies, and neural networks with the principal component method were used to detect abnormalities [11].

The application of neuro-fuzzy systems to intrusion detection problems was discussed in [12]. Their authors used the parameters of the KDD Cup 99 and DARPA 1998 datasets as input for fuzzy classifiers based on the fuzzy inference.

There are also various hybrid approaches, for example, a combination of the apparatus of immune systems and neural networks [13]. Multilayer neural networks that are generated using the clonal selection method are selected as immune detectors. The experiments were carried out on the KDD Cup 99 dataset; they proved the high ability of the detectors to adapt to new types of attacks.

Another hybrid solution to intrusion detection is to combine several neural networks into a single classifier. So, work [14] is devoted to the application of the IDS method and radial basis networks for the classification of records from the NSL-KDD data set. The final classifier is presented as a composition of classifiers sequentially built on different samples and simple voting procedures. As a result, the classification level was increased by about 1.6% compared with the classifiers taken separately. Other authors [15] propose submitting output values from several neural networks trained by various algorithms to the input of the weighted voting and majority voting. On a test sample of 6890 samples, a classification accuracy of over 99% was

achieved.

Article [16] describes a two-level scheme for detecting and classifying attacks. Several adaptive neuron-fuzzy modules are combined. Each of them is designed to detect only one class of compounds, processes the parameters of the KDD Cup 99 records. The final classification is performed by the fuzzy decision module, which implements the neuro-fuzzy inference system with two membership functions. This module is responsible for determining how abnormal the record being processed is. Its class corresponds to the class of the fuzzy module of the first level with the highest output value.

Previous work [17, 18, 19] has focused more on the transmission of chaotic signals using dynamic models. The offered algorithm of transfer images can be used for the decision of control tasks for integrity data and transfer of the information in modern telecommunication and information systems [17]. In the article [18] considered an algorithm of constructing an electronic-digital message signature based on encryption using elliptic curves. The article [19] is devoted to the calculation of the characteristics of dynamic chaos based on the traffic of the corporate computer network.

This article proposes an approach that develops the considered work. It is based on the implementation of multi-level and adaptive detection of network attacks through a combination of neural networks. Adaptive detectors perform parallel processing of input features of connections. The detectors are trained on various data samples and are designed to classify a single compound.

## 3 Hybrid signature model and adaptive approach

The proposed hybridization scheme for signature and adaptive approaches is presented in Fig. 1. To analyze packets and generate session parameters, the IDS system was used additional scripts were added for processing network events. The system has a built-in script interpreter that provides processing of various network events.

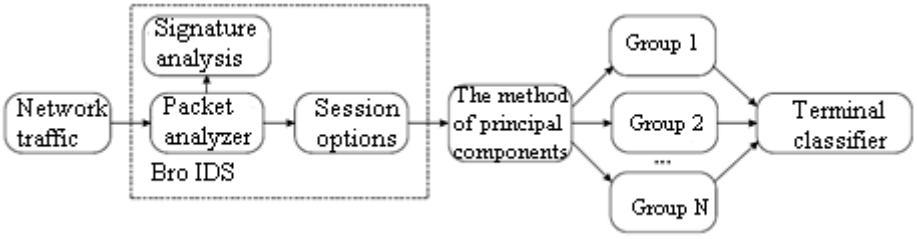


**Fig. 1.** Hybrid signature model and adaptive approach.

When creating a record about each formed connection, the parameters necessary for the classification of compounds are selected and calculated from the data field and header of each packet. Statistics are also collected on currently open connections. Optionally, to reduce the dimension, the input vector is compressed by the principal component method. Each group of detectors trained to recognize one particular type

of connection processes the last set of features. The final classification result is issued by the terminal classifier.

Within this scheme, several attack detection techniques can be used. The first of them implies that each group consists of several classifiers of the same type, which are trained on different samples from the training set. This allows you to create detectors that differ from each other in adjusted weights, and thereby increase the quantitative indicators of attack detection. The second technique involves the use of heterogeneous classifiers within each group. Here, instances of each of the three models are used to recognize a specific type of attack. In both cases, the output values from all classifiers can be considered as an input vector for the terminal classifier.

Let's take a closer look at the attack detection process. It can be divided into three stages.

At the first stage, the IP-defragmentation of raw packets and the assembly of TCP session segments are performed. Each connection (session) is a sequence of TCP packets for a certain time interval, during which data is transferred between a pair of remote hosts <addr_src, port_src> and <addr_dst, port_dst> using a specific protocol. In the process of analyzing network traffic, packets that initiate the start of a session (SYN) and serve as a sign of its termination (FIN, RST, timeout) are monitored. Each session can be characterized as a set of parameters, the elements of which are conditionally divided into two groups: attributes obtained from the packet headers, and statistical data. Also, this stage is responsible for the initial detection of network anomalies by verifying that the contents of individual packets match the specified regular expressions in the signature set.

The second step involves applying the principal component method and converting the output parameters of each session into a compressed set of attributes. Each element of the new vector is a linear combination of the elements of the old vector with coefficients, which are elements of the eigenvectors of the covariance matrix of the source data.

The third stage is the application of several groups of adaptive classifiers. Each group consists of detectors responsible for recognizing one particular type of connection. To increase the classification rate, each detector processes an incoming set of parameters in parallel with the others.

A terminal classifier can be represented in several ways. The simplest of them is the majority voting procedure. In this case, the connection class is the one for which a larger number of detectors within a certain group voted. Another method for implementing the terminal classifier is a weighted vote and its modification. In this case, each classifier will be assigned an appropriate coefficient assigned to it in the learning process. This will allow more reasonable use of one or another classifier, depending on its indicators of the correctness of recognition of attacks calculated on the samples of the training set. The next approach is to choose the best classifier for each particular record and ignore the rest. The main difficulty here is the construction for each classifier of a given area of competence in the space of attributes. It is also proposed to use output values from first-level classifiers as a training set for a terminal classifier. This technique, known as a multi-tiered generalization, can be built based on data mining methods.

To achieve the goal, it is proposed to use three artificial neural networks (ANNs) with various topologies: a multilayer perceptron (MLP), a network of radial basis functions (RBF) and a self-organizing Kohonen map (SOM). The same traffic is sent to the input of each ANN for the analysis of IP packet headers together with an inaccurate search for intrusion signatures directly in the applied contents of IP packets. To train an ANN, it is necessary to use both packets containing known signatures and normal IP datagrams [8]. In the process of detection, each ANN determines how much the network packet submitted to the input corresponds to a normal or abnormal situation.

At the output of each ANN, the results are presented as a multidimensional vector, each coordinate of which corresponds to one of the types of attacks. If the ANN detects an attack, then the corresponding coordinate takes a single value, otherwise, it takes zero.

The use of three ANNs with different topologies necessitates the conversion of network traffic to a single format so that each of them can equally analyze the data supplied to it at the input. The most suitable format is NLS-KDD, which is a standard dataset for IDS. The NLS-KDD dataset contains raw network packets that are independent of the operating systems used. In total, this data set includes 4898431 samples. NLS-KDD includes 37 different network attacks that can be divided into four large groups: denial of service, privilege escalation, remote access, and network scanning. All types of network attacks are presented in Table 1.

**Table 1.** Types of network attacks from the NLS-KDD dataset

| Group of attacks | Attacks |
|---|---|
| Denial of service | Back, Land, Neptune, Pod, Smurf, Teardrop, Mailbomb, Processtable, Udpstorm, Apache2, Worm |
| Network scan | Satan, IPsweep, Nmap, Portsweep, Mscan |
| Getting Remote Access | Guess_password, Ftp_write, Imap, Phf, Multihop, WarezmasterXlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named |
| Privilege escalation | Buffer_overflow, Loadmodule Rootkit, Perl, Sqlattack, Xterm, Ps |

The NLS-KDD format contains 41 fields. Bringing IP packets to NLS_KDD consists of extracting some features from network traffic. Such signs include connection duration, connection protocol, connection type, service and destination port, successful connection flag, number of failed authorization attempts, number of connection requests with administrator rights. The process of converting network packets to the NLS-KDD format is shown in Fig. 2.

Upon completion of converting network data to a single NLS-KDD format, they are fed to the input of the ANN. After analyzing the next portion of network traffic, the results of the ANN are recorded in multidimensional vectors. The resulting vector is found as the sum of the output vectors of the three ANNs. If the coordinate is zero or one in the resulting vector, then there is no attack. A unit means that a false positive occurred on one of the ANNs. If the value of the vector coordinate is 2 or 3, then an attack of the corresponding type has occurred. In Fig. 3, the process of obtaining the resulting vector is simplified.

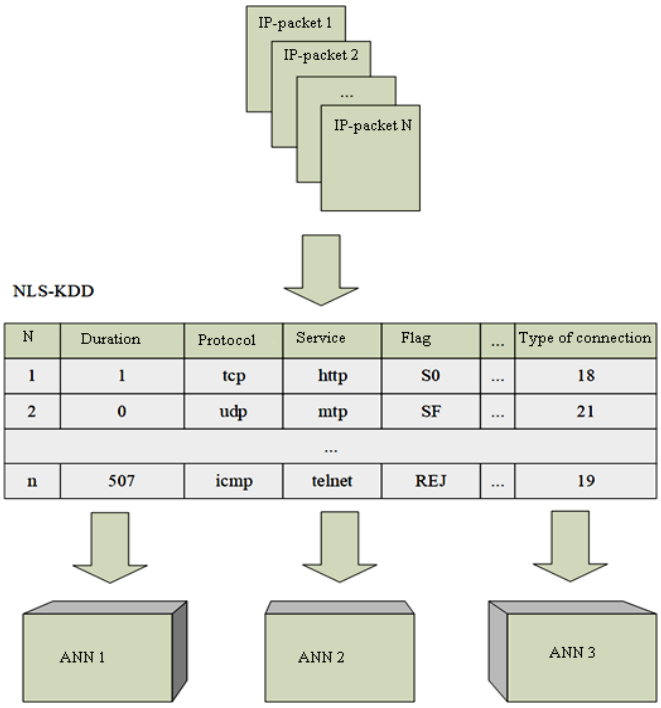


**Fig. 2.** The process of converting network packets to the NLS-KDD format for further submission to the input of the ANN of various topologies.

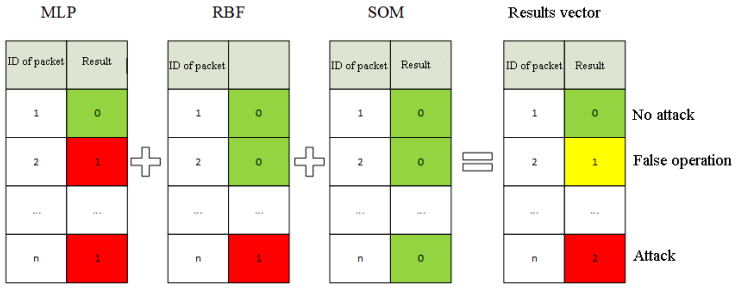


**Fig. 3.** Simplified representation of the addition of resulting vectors.

Thus, the algorithm of IDS (Fig. 4), which processes and filters a large amount of network traffic based on the ANN, will consist of the following: receiving suspicious network traffic from the sensor subsystem; converting network packets to the NLS-KDD format; analysis of a portion of n network packets using three neural networks with different topologies of MLP, SVM, and SOM; analysis results are entered into multi-dimensional vectors; the resulting vector is obtained by adding the three output vectors MLP, SVM, and SOV. After that, it is transferred to the data presentation subsystem.

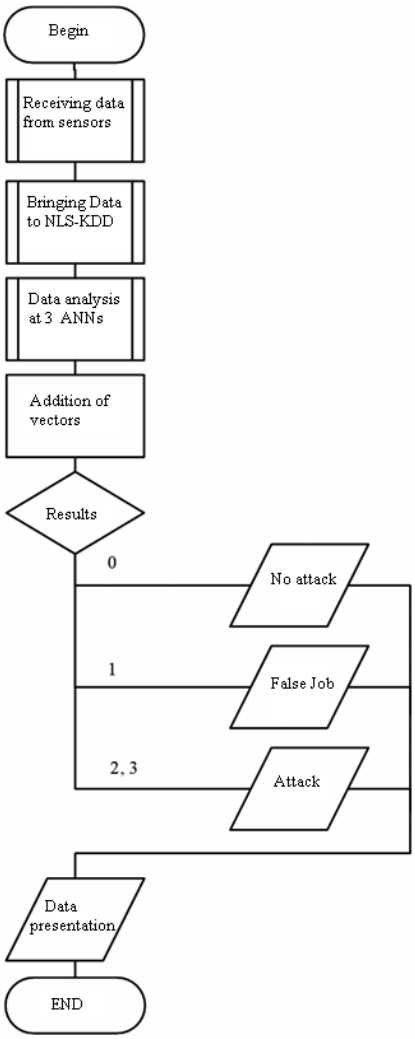


**Fig. 4.** Algorithm for detecting network attacks using three ANNs with different topologies.

The capabilities of the proposed algorithm can be expanded both by increasing the number of ANNs used and by using other types of ANNs depending on the task.

## 4 Experiments and results

To test the described models, two open data sets were selected: KDD Cup 99 and NSL-KDD. For these sets, each record represents an image of a real network session, described as a set of 41 parameters, and is marked as an attack or normal connection. The developed intrusion detection system monitors the level of data obtained from packet headers and statistical information generated by the sliding window method.

Therefore, 29 parameters were selected that satisfy this requirement. Also, for the classification of connections, 5 types of DDoS attacks and 4 types of attacks such as port scans were selected. The data received from the Bro system is processed by modules that implement a mechanism for classifying compounds using the proposed detectors. The following numerical indicators were selected to test the effectiveness of the functioning of the models:

1. Percentage of samples of normal compounds recognized as attacks (false positive):

$$FP = \frac{n_{FP}}{n_{FP} + n_{TN}} \cdot 100\% . \quad (1)$$

2. The percentage of correctly recognized samples of abnormal compounds (true positive):

$$TP = \frac{n_{TP}}{n_{TP} + n_{FN}} \cdot 100\% . \quad (2)$$

3. The percentage of network connection patterns whose class was correct classification:

$$CC = \frac{n_{CC}}{n} \cdot 100\% . \quad (3)$$

The threshold value for immune detectors was chosen experimentally in such a way that

$$TP - F + CC \to \max \quad (4)$$

was fulfilled on the training data.

The experimental results for each approach with the indicated values of these parameters, which were used as the terminal classifier by the majority voting procedure, are presented in Table 2.

**Table 2.** Indicators FP, TP, CC

| An approach | KDD Cup 99 | | | NSL-KDD | | |
|---|---|---|---|---|---|---|
| | FP | TP | CC | FP | TP | CC |
| Neural networks | 3,56 | 98,95 | 73,56 | 7,08 | 98,24 | 56,86 |
| Immune detectors | 2,26 | 91,16 | 94,82 | 12,51 | 93,06 | 65,65 |
| Neuro-fuzzy classifiers | 11,65 | 98,29 | 88,89 | 16,94 | 96,37 | 83,06 |
| Combination of approaches | 1,31 | 99,98 | 77,04 | 5,11 | 99,85 | 57,96 |

As can be seen from the results obtained, neural network and neuron-fuzzy classi-

fiers, in comparison with immune detectors, have a greater number of correctly recognized compounds when analyzing network parameters. So these classifiers are a more preferable tool for detecting network attacks. At the same time, immune detectors due to the dynamic configuration of the balance can modify their structure in response to detected attacks. Therefore, in the process of analyzing network traffic, the performance of recognition of network attacks by immune detectors increases over time. This evolutionary mechanism, as well as an updated set of training data, contributes to the recognition of previously unknown types of attacks.

Among the proposed three models, neural networks showed the best degree of recognition of network connection patterns. Neuron-fuzzy classifiers are characterized by a long learning process. This is explained by the complexity of the calculations associated with the multi-level structure of the network, and the setting of the parameters of membership functions in fuzzy rules. Nevertheless, the attack detection rate for these classifiers is high and comparable to the indicator of neural networks.

To evaluate the effectiveness of the proposed intrusion detection approaches, a series of experiments was conducted. The KDD Sup 99 database was used to train and test neural network models. This is one of those few intrusion detection bases that has attracted the attention of researchers due to its well-designed structure and accessibility. To study the characteristics of the proposed systems, we set three main indicators: the share of detected, the share of recognized attacks for each class and the number of false positives of the system. The share of detected attacks is defined as the number of attack images of a particular class detected by the system divided by the total number of records of attacks of this class in the database. Similarly, the share of recognized is determined. False positives indicate the total number of images of normal network operations that are erroneously classified as attacks.

The results of testing the recognition of an attack class for the operation of an intrusion detection system for sensor wireless networks are shown in Table 3.

**Table 3.** Model 1 test results for intrusion detection

| Class | Total | Discovered | Recognized |
|---|---|---|---|
| DoS | 391458 | 391441 (99.99%) | 370741 (94.71%) |
| U2R | 52 | 48 (92.31%) | 42 (80.77%) |
| R2L | 1126 | 1113 (98.85%) | 658 (58.44%) |
| Probe | 4107 | 4094 (99.68%) | 4081 (99.37%) |
| Normal condition | | | |
| Normal | 97277 | --- | 50831 (52.25%) |

Thus, the best result was achieved for DoS and Probe attacks. U2R and R2L are determined slightly worse, respectively, 80.77% and 58.44%. Also, there is a percentage of false positives for the system. A summary of each of the options for building an attack detection system is shown in Table 4.

Thus, model 3 is characterized by high accuracy (93.21%) and the least number of false positives. Using model 1, 86.3% of the input images were recognized, and model 2, 92.97%. Models 2 and 3 can be successfully used to work with large sets of complex data structures.

**Table 4.** Test results of proposed intrusion detection models in intrusion detection systems for sensor wireless networks

| Model | Detected attacks | Recognized Attacks | False positives | Total share recognized, % |
|---|---|---|---|---|
| Model 1 | 396696 (99.98%) | 375522 (94,65%) | 46446 (47.75%) | 86.30% |
| Model 2 | 395949 (99.80%) | 375391 (94.61%) | 13398 (13.77%) | 92.97% |
| Model 3 | 396549 (99.95%) | 375730 (94.70%) | 12549 (12.90%) | 93.21% |

Also, for each type of attack, the detection rates of various detectors were calculated. So, Table 5 shows indicators of the effectiveness of recognition of attacks on a set of KDD Cup 99 combined classifiers. In the presented table, the left column is the type of connection, the top row is the type of classifiers, their intersection is the percentage of correctly recognized attacks.

**Table 5.** Attack recognition performance indicators with combined classifiers for sensor wireless networks, %

| | back | neptune | pod | smurf | teardrop | ipsweep | nmap | ports-p | satan |
|---|---|---|---|---|---|---|---|---|---|
| back | 100 | 0 | 0 | 0 | 0 | 0 | 0 | 0,46 | 3,37 |
| neptune | 0 | 99,98 | 0 | 0 | 0 | 0 | 0,56 | 99,99 | 99,93 |
| pod | 0 | 0 | 60,98 | 0 | 34,09 | 1,89 | 1,89 | 100 | 37,88 |
| smurf | 0 | 0 | 0 | 99,92 | 0 | 0 | 0,05 | 0 | 0,09 |
| teardrop | 0 | 0 | 56,63 | 0 | 100 | 0,2 | 0 | 100 | 100 |
| ipsweep | 0 | 0 | 0,4 | 0 | 0 | 100 | 91,95 | 1,69 | 0,4 |
| nmap | 0 | 0 | 0,43 | 0 | 0 | 44,16 | 100 | 44,59 | 44,59 |
| ports-p | 0,1 | 3,09 | 0,1 | 0 | 0 | 67,6 | 0,39 | 100 | 89,68 |
| satan | 0 | 88,64 | 0 | 0 | 0 | 9,28 | 0,25 | 88,26 | 99,87 |
| normal | 0,06 | 0,03 | 0,05 | 0,03 | 0 | 0,1 | 0,44 | 0,25 | 0,56 |

By combining individual classifiers, it was possible to increase the detection rates by 3.85 and 3.96% compared with the average value of this value for individual classifiers for KDD Cup 99 and NSL-KDD, respectively. At the same time, the indicator of correct classification remained above this indicator for neural networks.

The constructed intrusion detection system can be used in real computer networks after preliminary training on the traffic characteristic of this network. In particular, at present, the study is aimed at conducting experiments related to the newly generated data. For this purpose, a software stand was created to simulate the computer network of a large enterprise used for experiments with the developed SIEM system, taking into account various vulnerabilities in software and hardware.

## 5 Conclusion

An implementation of a hybrid intrusion detection system, designed to analyze traffic in computer networks using the TCP/IP protocol stack, is presented. One of their advantages is the flexible extensibility of functionality by writing additional scripts for processing network events. This allows you to embed the necessary scripts designed to generate session parameters. To analyze such parameters, three modules are proposed that are based on various adaptive models: neural networks, immune detectors, and neuron-fuzzy classifiers. The experiments were conducted on two data sets: KDD Cup NSL-KDD. The results showed that, in comparison with the existing detection approaches evaluated on these sets, the proposed integration scheme allows a compromise between the recognition of unknown types of threats and false positives.

As we noted above, there are a lot of ways to detect an intrusion into the system and no fewer ways to hide it. We examined only one of them and proposed their mechanism for detecting network attacks. Today, hackers are becoming smarter, so you need to do a dump of RAM and its analysis. After that, we do a selective analysis of the files on the hard drive. And only at the end - a full analysis of the hard disk, if necessary.

Further research will focus on finding and applying other hybrid approaches to detecting attacks, creating experimental data sets and conducting experiments.

## References


1. Jing, Y., Feng, W.: Simulation Modeling of Network Intrusion Detection Based on Artificial Immune System. In: Life System Modeling and Intelligent Computing. Communications in Computer and Information Science, v. 97. Springer, Berlin, Heidelberg, pp. 1-4 (2010)
2. Kim, J., Bentley, P.: An Artificial Immune Model for Network Intrusion Detection. Computer Applications in Engineering Education, pp. 1-4 (2001).
3. Kumar, V.: Signature-Based Intrusion Detection System Using SNORT. International Journal of Computer Applications & Information Technology, pp. 1-7 (2012).
4. Guan, X., Li, Y.: Notice of Retraction: An New Intrusion Prevention Attack System Model Based on Immune Principle. 2nd International Conference on E-business and Information System Security, Wuhan, pp. 1-4 (2010).
5. Tavallaee, M., Bagheri, E., Lu W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, pp. 1-6 (2009).
6. Siddique, K., Akhtar, Z., Aslam Khan F., Kim, Y.: KDD Cup 99 Data Sets: A Perspective on the Role of Data Sets in Network Intrusion Detection Research. In Computer, vol. 52, no. 2, pp. 41-51 (2019).
7. Lei, J.Z., Ghorbani, A.A.: Improved Competitive Learning Neural Networks for Network Intrusion and Fraud Detection. Neurocomputing, vol. 75, Iss. 1, pp. 135–145 (2012).
8. Meng, Q.: An immune-neuroendocrine-inspired inspired artificial homeostatic security-coordination model for E-Government system. 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), Dengleng, pp. 6960-6963 (2011).

9. Giatzitzoglou, D.G., Sotiropoulos D.N., Tsihrintzis, G.A.: AIRS-x: An eXtension to the Original Artificial Immune Recognition Learning Algorithm. 2019 International Conference on Computer, Information and Telecommunication Systems (CITS), Beijing, China, pp. 1-5 (2019).
10. Yang, H., Yiwen, L., Tao, L., Ting, Z.: A model of Collaborative Artificial Immune System. 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, pp. 101-104 (2010).
11. Hamid, M.B.A., Rahman, T.K.A.: Short Term Load Forecasting Using an Artificial Neural Network Trained by Artificial Immune System Learning Algorithm. 12th International Conference on Computer Modelling and Simulation, Cambridge, pp. 408-413 (2010).
12. Orang, Z. A.: Using Adaptive NeuroFuzzy Inference System in Alert Management of Intrusion Detection Systems. Intern. Journal of Computer Network and Information Security, vol. 4, N 11, p. 32–38 (2012).
13. Komar, M., Golovko, V., Sachenko, A., Bezobrazov, S.: Development of Neural Network Immune Detectors for Computer Attacks Recognition and Classification. IEEE 7th Intern. Conf. on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), vol. 2, pp. 665–668 (2013).
14. Govindarajan, M., Chandrasekaran, R.M.: Intrusion Detection Using an Ensemble of Classification Methods. Proc. of the World Congress on Engineering and Computer Science, vol. 1, pp. 459–464 (2012).
15. Almeida, L.A. Álvarez, Santos, J. Carlos Martinez: Evaluating Features Selection on NSL-KDD Data-Set to Train a Support Vector Machine-Based Intrusion Detection System. IEEE Colombian Conference on Applications in Computational Intelligence (ColCACI), Barranquilla, Colombia, pp. 1-5 (2019).
16. Syarif, I., Zaluska, E., Prugel-Bennett, A., Wills, G.: Application of Bagging, Boosting and Stacking to In-trusion Detection. Machine Learning and Data Mining in Pattern Recognition, vol. 7376, pp. 593-602 (2012).
17. Belej, O., Lohutova T., Banaś, M.: Algorithm for Image Transfer Using Dynamic Chaos. 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), Ukraine, 2019, pp. 1-5 (2019).
18. Belej O.: The Cryptography of Elliptical Curves Application for Formation of the Electronic Digital Signature. Advances in Intelligent Systems and Computing, vol 938, Springer, Cham, pp. 43-57(2020).
19. Belej O.: The Controlling of Transmission of Chaotic Signals in Communication Systems Based on Dynamic Models. CEUR Workshop Proceedings, Vol. 2353, pp. 664-673 (2019).