UDC: 534.843: 004.9 DOI: https://doi.org/10.15407/jai2021.01.022

CONSTRUCTION OF A MULTI-AGENT ATTACK DETECTION SYSTEM BASED ON ARTIFICIAL INTELLIGENCE MODELS

O. Belej[1], Spas N.[2], Artyshchuk I.[3], Fedastsou M.[4]
[1,2,3,4] Lviv Polytechnic National University, Ukraine
5 Mytropolyt Andrei str., Building 4, Room 324, Lviv, 79015

[1]http://orcid.org/0000-0003-4150-7425
[2]http://orcid.org/0000-0002-6908-2900
[3]http://orcid.org/0000-0001-7287-8451
[4]http://orcid.org/0000-0003-0927-6890

Abstract. Statistics of recent years on attacking actions on information systems show both the growth of known attackers and the growth of new models and directions of attacks. In this regard, the task of collecting information about events occurring in the information system and related to the main objects of the information system, and conducting their effective analysis is relevant. The main requirements for the tools of analysis are: speed and ability to adapt to new circumstances - adaptability. Means that meet these requirements are artificial intelligence systems. In particular, there are a number of research that use neural networks as a means of analysis. There are different types of neural networks, which differ depending on the tasks to be solved and are more suitable for different input data.

The proposed multi-agent attack detection system collects and analyzes the collected information about the events of the information system using two types of neural networks. A multilayer perceptron is used to analyze various logs of information system objects. The Jordan network is used to analyze directly collected information about the events of information system objects. The use of a multi-agent attack detection system can increase the security of the information system. Features of modern attacks are considered. The urgency of the task of detecting attacks is substantiated. The peculiarities of the attack process were considered. The actions of attackers of different types at different stages of the attack are analyzed. It was shown which methods of detecting attacks should be used at different stages of the attack by an attacker. A model of a multi-agent attack detection system is proposed. An interpretation of the results of the analysis of information system events by the method of detecting attacks was proposed, as well as an algorithm for joint decision-making by agents based on several sources of information about their status. A model of an attack detection system that takes into account these features is proposed. This attack detection system collects information at several levels of the information system and uses it to analyze the artificial intelligence system.

Keywords: attack, attack detection system, neural network, intelligent agent, multi-agent system, joint decision making.

ПОБУДОВА МУЛЬТИАГЕНТНОЇ СИСТЕМИ ВИЯВЛЕННЯ АТАК НА ОСНОВІ МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ

О.І. Белей[1], Н.Я. Спас[2], І.В. Артищук[3], М.В. Федосцов[4]
[1,2,3,4] Національний університет «Львівська політехніка», Україна
вул. Митрополита Андрея 5, 4 навч. корпус, 324 кімнат, Львів, 79015

[1]http://orcid.org/0000-0003-4150-7425
[2]http://orcid.org/0000-0002-6908-2900
[3]http://orcid.org/0000-0001-7287-8451
[4]http://orcid.org/0000-0003-0927-6890

Анотація. Статистика останніх років щодо атакуючих дій на інформаційні системи показує як зростання відомих атакуючих, так і зростання нових зразків і напрямків реалізації атак. У зв'язку з цим актуальною є задача збору відомостей про події, що відбуваються в інформаційній системі і відносяться до основних об'єктів інформаційної системи, і проведення їх ефективного аналізу. Основними вимогами до засобів аналізу є: швидкість і можливість пристосування до нових обставин - адаптивність. Засобами, що задовольняють цим вимогам, є системи штучного інтелекту. Зокрема існує ряд досліджень, в яких застосовуються нейронні мережі

як засіб аналізу. Виділяють різні типи нейронних мереж, що розрізняються в залежності від розв'язуваних завдань і більш підходящі для різних вхідних даних.

Запропонована багатоагентна система виявлення атак здійснює збір і аналіз зібраних відомостей про події інформаційної системи за допомогою двох типів нейронних мереж. Для аналізу різних журналів об'єктів інформаційної системи застосовується багатошаровий персептрон. Для аналізу безпосередньо зібраних відомостей про події об'єктів інформаційної системи застосовується мережу Джордана. Застосування багатоагентної системи виявлення атак дозволяє підвищити захищеність інформаційної системи. Розглянуто особливості сучасних атак. Обґрунтовано актуальність завдання виявлення атак. Були розглянуті особливості процесу проведення атаки. Проаналізовано дії зловмисників різних типів на різних етапах атаки. Було показано, які методи виявлення атак необхідно застосовувати на різних етапах реалізації атак зловмисником. Запропоновано модель багатоагентної системи виявлення атак. Була запропонована інтерпретація результатів аналізу подій інформаційної системи методом виявлення атак, а також представлений алгоритм прийняття спільного рішення агентами на підставі кількох джерел відомостей про їх стан. Запропоновано модель системи виявлення атак, що враховує ці особливості. Дана система виявлення атак реалізує збір інформації на декількох рівнях інформаційної системи і використовує для аналізу системи штучного інтелекту.

Ключові слова: атака, система виявлення атак, нейронна мережа, інтелектуальний агент, багатоагентна система, прийняття спільного рішення.

Introduction

Currently, the main features of attacks are that. There is a constant increase in the complexity of attacking influences, their technological level has grown significantly even compared to last year. Often, attacks have a multi-step algorithm of actions and a distributed nature. The most attacks are initially carried out through the browser - using many vulnerabilities both in the browsers themselves and in third-party applications interacting with them. This leads to the fact that often the same malware can be spread using a dozen different vulnerabilities, which leads to a proportional increase in the number of types of attacks.

To take into account these features, modern intrusion detection systems (IDS) must perform distributed collection and analysis of information, as well as its intellectual analysis. In addition, to detect new attacking influences, the number of which, according to statistics, is large.

As the analysis of modern free IDS shows, none of them fully meets the requirements, but the trend shows that most of them perform analysis at several levels of the information system or have the ability to expand. In addition, attempts are being made to realize the ability of the IDS to adapt to new types of attacking influences.

Problem statement

Based on the analysis of various methods used to analyze data in identifying attacking influences, neural networks and genetic algorithms are the most preferable for solving the problem. Within the framework of this research, neural networks are used.

Since the collection of data must be carried out at several levels and the information system, the following sources of information were selected, the analysis of which will make it possible to identify attacking influences:

- data about network packets;
- router log information;
- information from the operating system security log;
- information from the operating system registry;
- information about the processes of the operating system.

The use of adaptive data analysis methods, like neural networks, has a large number of false positives. To reduce this indicator, within the framework of the research, intelligent agents are used that interact with each other to find a joint solution within the framework of the state of the entire information system, and not a separate event.

Each intrusion detection agent is described by a state (*P, B, S, G, I*), where:

- *P* - sensation. Represents information about the environment collected by an agent, that is, a set of inputs from an agent.
- *B* - beliefs. A set of beliefs, that is, information and knowledge about the environment of the agent. An agent's beliefs are a neural network. At the first stage, agents collect information about the functioning of the information system, and on their basis, a training sample for the neural network is created.
- *S* - situation. The specific state of the environment, that is, the specific values of the input data and the result of their classification by the neural network.
- *G* - goals. Defined as the desired state of the environment.
- *I* - intentions. Many possible action plans for the agent.

Accordingly, intrusion detection agents have the following basic functions:

- generation and revision of beliefs.

This function is responsible for collecting information for training and, if necessary, retraining the neural network and the training itself:

- assessment of the situation. Obtaining the results of evaluating the collected information about the information system by a neural network;

- target activation. Depending on the value of the neural network output, the agent chooses a set of elementary actions that must be performed in a given situation;

- appointment. The agent determines the final plan of action, determining the sequence of elementary actions;

- performance. Implementation of the selected elementary actions by the agent.

The interaction of agents with each other allows making joint decisions, reducing the errors of a single neural network.

At present, experimental researches are being carried out on a software package developed for this architecture.

Analysis of the last publications

The modern stage of development of information system (IS) is based on the achievements of telecommunication technologies used for distributed information processing [1, 2].

This has led to the fact that most attacks are distributed in nature. Over the course of 2020, the complexity of the attacking effects has been constantly increasing. Their technological level has grown significantly even compared to last year. The attacks often had a multi-step algorithm of actions and a distributed nature. According to data for the first quarter of 2020, 28% more attacks were carried out than in the previous quarter[3]. Attack spread rates also increased by 61% [4]. All this confirms the relevance of research in the field of attack detection.

In work [5], a description of the methodology is presented and the temporal behavior of employing each SEDAR strategy is mathematically described, both in the absence and presence of faults. The paper [6] proposes a framework for analyzing the collected logs in order to provide the defenders with relevant insights on the attacks that have been conducted.

Typically, attacks include the following steps [7]: collecting data about the object of the attack; implementation of unauthorized access (NSD) to the required IS node; attacking effect; spreading the attack to other IS nodes. Previously, the architecture of a typical information system for the problem of detecting attacks was proposed [8]. Let's denote the set of router events as R, the set of network packets - N, the set of host events (both for a normal workstation and the server) - W.

In the first step, the attacker collects information about the IP. For example, an external attacker can perform a port scan from an external network with respect to the IS. Since the IS is separated from the external network by a router, at the first step, an external attacker generates a subset of events $R_1 \subset R$.

An internal attacker located in a different segment of the IS relative to the target host, at this stage, collects information about the configuration of the IS. To do this, he can, for example, form a query to the DNS server. By doing this, an internal attacker generates subsets of events $R_2 \subset R$ on the internal IP routers and segment $N_1 \subset N$ packets that do not contain the target host.

An internal attacker on the same IP segment as the target host is collecting information about the target host itself. Its actions generate subsets of the packets of the segment $N_2 \subset N$ containing the target host and the events of the target host $W_1 \subset W$.

To analyze information about the events generated by the attacker at this stage, it is effective to use methods that identify certain signs of malicious behavior, i.e. methods of detecting abuse. This is due to the fact that all the operations of the intruder, with the help of which he obtains the information he needs, in most cases does not cause any deviation from the normal behavior of the IS.

At the second stage, an external attacker must get the unauthorized service to the internal IS network. At the same time, it generates a subset of events $R_3 \subset R$ the router that separates the IS from the external network. After successfully overcoming this step, the behavior of an external attacker becomes similar to the actions of an internal attacker located in a different IP segment relative to the target host, and he performs the first and second steps similar to this type of attacker.

At the second step, an internal attacker located in a different IP segment relative to the target host of the attack needs to carry out unauthorized access to the network segment containing the target host of the attack $R_4 \subset R$. Its actions generate subsets of events on internal IP routers and segment packets $N_3 \subset N$ that do not contain the target host. After successful completion of this step, the behavior of an internal attacker located in a different IP segment relative to the target host becomes similar to the actions of an internal attacker located in the same IP segment as the target host, and he performs the first and second

steps similarly this type of attacker.

At this step, an internal attacker who is in the same segment $N_4 \subset N$ with the target host tries to get the tamper to the target host. Such actions generate subsets of packets $W_2 \subset W$ of the segment containing the target host and events of the target host.

The study [9] proposes a new vegetation index, Normalized Projected Red & SWIR (NPRS), for detection of spruce bark beetle attacks. To detect DDoS attacks on SDN we propose DDoS detection using Machine Learning with Ensemble Algorithm. At the experimental stage, authors [10] used InSDN as a dataset.

Authors [11] propose a pilot spoofing attack detector followed by a beamforming scheme for secure data transmission. Another authors [12] provide a polynomial-time algorithm to compute a correlated equilibrium for the multistage attack case.

Detection of an attack at the stage of obtaining an NSD is possible both using methods of detecting abuse, and using methods that react to deviations from normal behavior, anomaly detection methods [13]. This is due to the fact that, on the one hand, any intrusion is characterized by certain characteristic features of an attack, and on the other hand, the same intrusion can also be described as some deviation from the normal behavior of the information system (IS). Therefore, the most effective is the combined use of these methods of detecting attacks [14].

At the third stage, all types of attackers have similar behavior, their task is to elevate privileges on the host in order to gain access to the target of the attack. These actions generate a subset of events $W_3 \subset W$ on the target host.

At the fourth stage, the attacker performs actions on the host, which will allow him, if necessary, to continue the attack on the resources of other IS nodes. These actions generate a subset of events $W_4 \subset W$ on the target host.

The method of the intrusion detection system is proposed to increase the reliability of intrusion detection using a heuristic model of the intrusion detection system and increase the battery life of the system [15].

Mutli Agent Systems (MAS) are very suitable for intrusion detection systems as they meet the characteristics required by the networks and Big Data issues [16]. A distributed event-triggered control law is developed with scheduling of controller updating times determined in the presence of DoS attacks [17].

The research [18] involves the design of a novel intrusion detection system and the implementation and evaluation of its analysis model. The developed technique of hierarchical hybridization of binary classifiers makes it possible to build multilevel schemes with the arbitrary embedding of classifiers in each other and their "passive" connection in the process of analysis of the input vector [19].

The paper [20] proposes an approach for IoT service composition using a multi-agent system where several agents are engaged to satisfy the user's request. Authors [21] have created an environment that simulates a smart home with a range of devices and sensors. The agent manages the built-in device.

The paper [22] discusses the importance of CI, the application of artificial immune systems in the protection of CI information systems and proposes a new model for smart grid protection using biologically inspired concepts. The research [23] introduces a combined multi-agent and multilayered game formulation where it incorporates a trust model to assess each node/object, which is participating in IoT communications from a security perspective.

Effective detection of attacks at the stages of attack and attack development is possible using anomaly detection methods, since the actions of intruders at these stages can vary greatly depending on the objectives of the attack and therefore cannot be unambiguously determined by a fixed set of attack signs.

The aim of research

In the information system, every second there is a huge number of events that affect the state of the information system and determine the transitions of the information system from one state to another. From the point of view of the security of information stored, processed and transmitted within the information system, three states can be distinguished: normal, abnormal, and dangerous.

In a normal state, the information system operates in a day-to-day mode, in accordance with its tasks and in accordance with the documents regulating its operation, and confidentiality, availability and integrity are ensured for the protected information. Let us refer to normal events as such events that transfer the information system to a normal state.

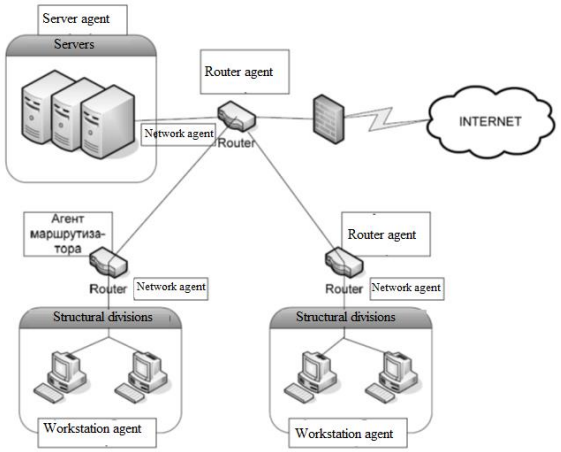
In an abnormal state, the functioning of an information system differs from functioning in a normal state; however, there are no unambiguous signs of the manifestation of attacking influences, confidentiality, availability and integrity are also ensured for the protected information. Let us refer

to anomalous events as those events that transfer the information system to an abnormal state.

The main material

Since it is necessary to use both the abuse detection method and the anomaly detection method, the attack detection method was chosen - neural networks, which, depending on training, can detect both abuse and anomalies. For this, the training sample for the neural network is formed so that it contains samples of both the normal behavior of the IS and the malicious actions of intruders.

A model of a multi-agent intrusion detection system (IDS) is proposed that carries out a distributed collection and analysis of the listed information about the state of the IS (fig. 1).



Structural divisions

Fig. 1. A model of multi-agent intrusion detection systems

The model includes: router agents that analyze information about router events reflected in the router A_R ; network agents analyzing information about packets transmitted over the network A_N ; server agents analyzing information about events occurring on IS servers A_S ; workstation agents analyzing information about events occurring on IS workstations A_W .

A multi-agent IDS can be represented as a graph *G = (A, B)*, where A is a set of agents, B is a set of graph edges; an edge exists if and only if a pair of agents can communicate with each other, bypassing other agents.

A router agent and a network agent, or a network agent and a workstation agent, are connected by edges.

The knowledge base of all agents is a neural network. In this case, the neural networks of all agents are trained on training samples containing both samples of normal behavior and samples of the attacker's actions on the IS, which allows agents to classify the analyzed event as a normal event or as an attack. In practice, the output of a neural network is a continuous signal at a given interval [*a, b*], where *a* is the lower bound of the interval, when received at the output, the event is interpreted as an attack; *b* - the upper bound of the interval, upon receipt of which at the output, the event is interpreted as normal.

When obtaining the value of *c* at the output of the neural network $p = \frac{c-a}{b-a}$, the value can be interpreted as the probability with which the neural network classifies the analyzed event as normal. The smaller the value of *p*, the closer *c* is to the lower bound of the interval a, and the more likely the event can be classified as an attack.

Since the output of the neural network is interpreted as a probability, the initial interval [a, b] is divided into 5 subintervals for convenience $(a_1, b_1), (a_2, b_2), (a_3, b_3), (a_4, b_4), (a_5, b_5)$, where $a_1 = a, a_i = b_{i+1}$ for $i = 2,...,5, b_5 = b$

Depending on the interval in which the next output of the neural network fell, ie. with what probability the event was classified as normal, it belongs to one of the hazard classes, and IDS agents can perform various actions according to the settings of multi-agent IDS: recording the event in the IDS log; informing the administrator about the event; process blocking; disconnection; interrupting the process.

In addition, if the output of the neural network did not fall into the extreme left interval, ie. was not assigned to the highest class of danger, nor in the extreme right interval, i.e. has not been classified as a lower hazard, the IDS agents make a joint decision on how to classify the event to reduce the likelihood of a single agent making a decision error by comparing data from multiple sources.

To make a joint decision, each agent forms his own preferences, ie. indicates with what priority it assigns the event to each of the 5 classes. In this case, preferences of the form are formed $O_i \succ O_j \succ O_k \succ O_l \succ O_m$ where *i* determines the interval in which the output of the neural network fell. The next class in the agent preferences is defined as the next closest subinterval to the value of the output of the neural network c, using the following algorithm:

$$\begin{aligned} &\text{If } (c > a_i + \frac{b_i + a_i}{2}), \\ &\text{then } j = i+1; a_j = a_i; b_j = b_{i+1}, \\ &\text{else } j = i = 1; a_j = a_{i-1}; b_j = b_i \end{aligned} \quad (1)$$

When one of the extreme subintervals is added to the preference, the other subintervals are

added to the preference in the order of removal from the current preference. After forming the preferences of each agent, the agents are grouped together to make a joint decision. According to the actions of the attackers discussed earlier, the group of agents for an external attacker looks like this $A = \{A_R^1, A_N^1, A_R^2, A_N^1, A_W^1\}$. Moreover, in the graph *G*, there must be a path that includes all agents of the group $\exists G_1 = (A_R^1, A_N^1, A_R^2, A_N^1, A_W^1)$.

The general decision is by vote. The winner of the vote, ie the jointly accepted class to which the event will be assigned will be the Condor winner level that satisfies the condition accordingly $\forall o' \in O, \#(o \succ o') \geq \#(o' \succ o)$. Due to the peculiarity of the formation of agents' preferences, the so-called Condorcet paradox is excluded, in which the winner cannot be identified.

The proposed multi-agent IDS allows to increase the accuracy of the classification of IP events, ie. reduce errors of the first and second kind.

In a dangerous state, the normal functioning of the information system is disrupted, leading to a violation of the confidentiality, availability or integrity of the information stored, processed or transmitted in the information system. We classify as dangerous events such events that translate the information system into a dangerous state. Such a division can be represented as a finite state machine (fig. 2).

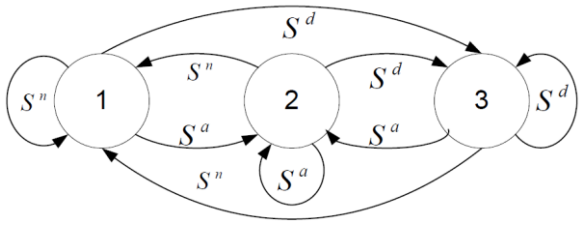


Fig. 2. The finite state machine for the transition of the multi-agent attack detection system from one state

Here, “1” corresponds to the normal state of the information system, “2” to the abnormal state of the information system, and “3” to the dangerous state of the information system.

Given the large number of events occurring in the information system, it is necessary to carry out automatic collection and automated analysis of information describing the occurring events.

There are the following main sources of event information that should be analyzed to determine the state in which the information system is: information about network packets; router logs; operating system logs; application logs; database logs; process information.

These sources differ primarily in the nature of the dependence of the events that occur in them. Event sources, which are logs, reflect a selection of information system events that are most important to the developer of the relevant information system object. Therefore, although some events belonging to these sources arise in groups that can be considered as a chain of interrelated events, in general, these sources do not maintain the correlation of the original flow of events, which makes it possible to define them as a set of independent events.

On the other hand, if we consider information about network packets or processes running in the system, they are direct information about the objects of the information system, which makes it possible to talk about their interdependence.

When analyzing information about events from any of the above sources, it is necessary to solve the classification problem by relating the current event to one of the three classes of events. In addition, important indicators for analysis tools are the speed of analysis and the simplicity of the process of adapting tools to new types of events, ie learning tools for analysis.

One of the possible directions of development of means of analysis of events of the information system corresponding to these tendencies is use of neural networks. The use of neural networks is due to their following properties: detection of hidden patterns, classification according to the identified patterns, high resistance to noise of the processed data and the absence of the need for strict formalization of the tasks.

Neural networks are by nature a parallel means of information processing. This "innate" parallel processing allows you to effectively use all the resources of modern hardware solutions, which increases the speed of analysis.

Two types of neural networks are required to conduct a comprehensive analysis of information about information system events related to all of these sources. Both must carry out the classification. One should be suitable for analyzing independent events, while the other should be suitable for analyzing interrelated events.

Two types of neural networks that meet these requirements were selected: a multilayer perceptron and a Jordan network.

The multilayer perceptron (fig. 3) is characterized by the direct propagation of signals from the inputs of the neural network to its outputs, which allows the analysis of independent events. Since the output result of the neural network

depends only on the values of the input vector transmitted to the neural network at the moment.

The figure denotes the components of the input vector, the weights of the synaptic connections between the input and hidden layers, the outputs of the neurons of the hidden layer, the weights of the synaptic connections between the hidden and output layers, and the outputs of the neural network.

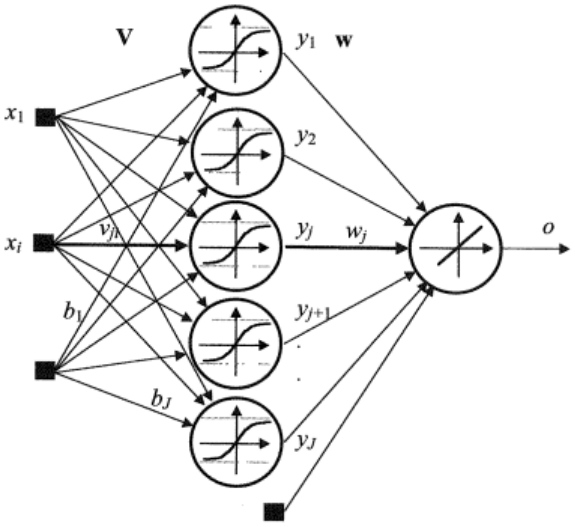


Fig. 3. Multilayer perceptron with one hidden layer in the multi-agent attack detection system

The neurons of the layered layers of the multilayer perceptron, which perform the main processing of the input signal, have a sigmoid transfer function. For the given classification problem, the output of a neuron possessing a given transfer function has the following interpretation (fig. 4).

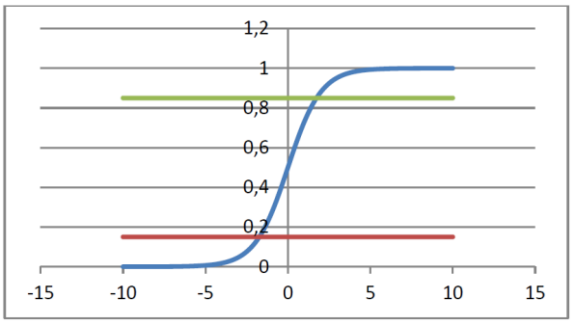


Fig. 4. Interpretation of the output of the sigmoid transfer function of the neuron in the multi-agent attack detection system

If the output of the sigmoid transfer function of the latent layer neuron is at the bottom of the graph, ie. less than some threshold value marked on the lower line graph, the event currently being analyzed by the neural network belongs to the event class.

If the output of the sigmoid transfer function of the neuron of the hidden layer is at the top of the graph, ie. more than some threshold value marked on the upper line graph, the event currently being analyzed by the neural network belongs to the event class.

If the output of the sigmoid transfer function of the latent layer neuron is on the slope of the sigmoid function, between the specified threshold values marked by lines in the graph, then the event currently being analyzed by the neural network belongs to the event class.

The Jordan neural network (fig. 5) is obtained on the basis of a multilayer perceptron by introducing feedback that transmits a signal from the neurons of the output layer to the corresponding neurons in the input layer.

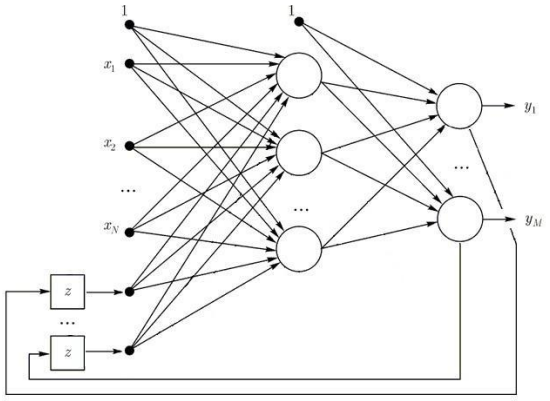


Fig. 5. Jordan's neural network in the multi-agent attack detection system

In the figure, the components of the input vector are denoted by, the outputs of the neural network are denoted by, the context neurons, the neurons of the input layer corresponding to the neurons of the output layer are denoted by.

In the Jordan network, signals are transmitted to context neurons with a delay of one clock cycle. Thus, the dependence of the neural network solution on the past states of the neural network, and hence the states of the information system, is obtained. This ability allows you to use a neural network to analyze related sets of events, where the analysis of events is independently incorrect.

Both types of neural networks are taught with a teacher. Therefore, for their training, the information security specialist must form a training sample containing samples of events of all three classes.

To implement the proposed principles, a multi-agent system for detecting attacks on the information system (fig. 6) has been developed, which collects information about events from

selected sources and analyzes this information using a neural network. The system presents neural networks of both types used for different sources.

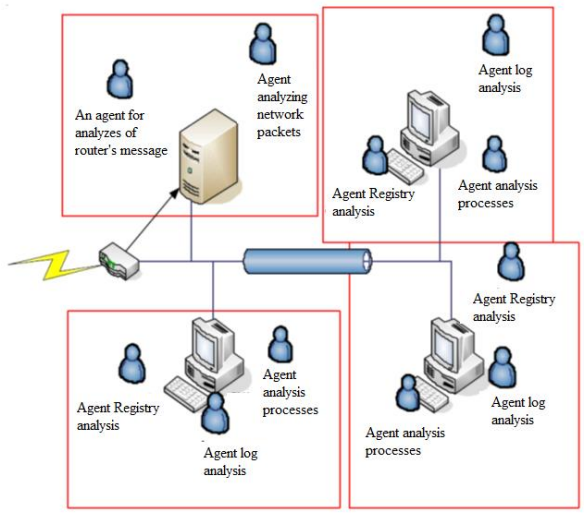


Fig. 6. The architecture of multi-agent attack detection system

Router agents and some workstation and server agents that work with event logs analyze the collected information using a multilayer perceptron. Network agents and some workstation and server agents that directly collect information about the state of information system objects use Jordan's neural network for analysis.

Conclusions

Experimental researches were performed for the developed multi-agent attack detection system using neural networks of both types. According to the results of experiments, the number of misses of attacks decreased by an average of 1.1 times, the number of false positives decreased by an average of 3.8 times. This allows us to conclude that the security of the information system increases when using the developed multi-agent attack detection system.

References

1. Shendryk, Vira & Boiko, Andrii. (2015). Stages of Information System Development in the Process Approach. Procedia Computer Science. 77. Doi: 10.1016/j.procs.2015.12.365.
2. Ferrer, I. & Ríos, José & Ciurana, Joaquim. (2009). An approach to integrate manufacturing process information in part design phases. Journal of Materials Processing Technology. 209. 2085-2091.
3. Doi: 10.1016/j.jmatprotec.2008.05.009.
4. Christopher Bailey, Rogério de Lemos, Malicious changeload for the resilience evaluation of self-adaptive authorisation infrastructures, Future Generation Computer Systems, Volume 113, 2020, Pages 113-131, doi: 10.1016/j.future.2020.06.045.
5. Julian Jang-Jaccard, Surya Nepal, A survey of emerging threats in cybersecurity, Journal of Computer and System Sciences, Volume 80, Issue 5, 2014, Pages 973-993, doi: 10.1016/j.jcss.2014.02.005.
6. Diego Montezanti, Enzo Rucci, Armando De Giusti, Marcelo Naiouf, Dolores Rexachs, Emilio Luque, Soft errors detection and automatic recovery based on replication combined with different levels of checkpointing, Future Generation Computer Systems, Volume 113, 2020, Pages 240-254, doi: 10.1016/j.future.2020.07.003.
7. Antonella Guzzo, Michele Ianni, Andrea Pugliese, Domenico Saccà, Modeling and efficiently detecting security-critical sequences of actions, Future Generation Computer Systems, Volume 113, 2020, Pages 196-206, doi: 10.1016/j.future.2020.06.054.
8. Saied A., Overill R. E., Radzik T. (2016) *Detection of known and unknown DDoS attacks using Artificial Neural Networks.* Neurocomputing. Vol. 172. 385–393. doi: 10.1016/j.neucom.2015.04.101
9. Al-Ayyoub, Mahmoud & Jararweh, Yaser & Daraghmeh, Mustafa & Qutaibah, Althebyan. (2015). Multi-Agent Based Dynamic Resource Provisioning and Monitoring for Cloud Computing Systems Infrastructure. Cluster Computing. 18. Doi: 10.1007/s10586-015-0449-5.
10. L. Huo, E. Lindberg and H. Persson, "Normalized Projected Red & SWIR (NPRS): A New Vegetation Index for Forest Health Estimation and Its Application on Spruce Bark Beetle Attack Detection," IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, 2020, pp. 4618-4621, doi: 10.1109/IGARSS39084.2020.9323611.
11. D. Firdaus, R. Munadi and Y. Purwanto, "DDoS Attack Detection in Software Defined Network using Ensemble K-means++ and Random Forest," 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2020, pp. 164-169, doi: 10.1109/ISRITI51436.2020.9315521.
12. J. Xie, Y. Liang, J. Fang and X. Kang, "Two-stage uplink training for pilot spoofing attack detection and secure transmission," 2017 IEEE International Conference on Communications (ICC), 2017, pp. 1-6, doi: 10.1109/ICC.2017.7996989.
13. S. Moothedath et al., "A Game-Theoretic Approach for Dynamic Information Flow Tracking to Detect Multistage Advanced Persistent Threats," in IEEE Transactions on Automatic Control, vol. 65, no. 12, pp. 5248-5263, Dec. 2020, doi: 10.1109/TAC.2020.2976040.
14. Belej O., Halkiv L. (2020) *Development of a network attack detection system based on hybrid neuro-fuzzy algorithms*. CEUR Workshop Proceedings, Vol. 2608, 926-938.
15. O. Belej, "Development of a Technique for Detecting "Distributed Denial-of-Service Attacks" in Security Systems of Wireless Sensor Network," 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), 2020, pp. 316-319, doi: 10.1109/CSIT49958.2020.9321942.
16. O. Belej, M. Karpinski, A. Shaikhanova, O. Veselska and A. Azatov, "Development of

Intrusion Monitoring System in Wireless Sensor Networks Based on Neural Networks," 2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), Dortmund, Germany, 2020, pp. 1-6, doi: https://doi.org/10.1109/IDAACS-SWS50031.2020.9297080.

17. S. OUIAZZANE, M. ADDOU and F. BARRAMOU, "A Multi-Agent Model for Network Intrusion Detection," 2019 1st International Conference on Smart Systems and Data Science (ICSSD), 2019, pp. 1-5, doi: 10.1109/ICSSD47982.2019.9003119.
18. Z. Feng and G. Hu, "Distributed secure leader-following consensus of multi-agent systems under DoS attacks and directed topology," 2017 IEEE International Conference on Information and Automation (ICIA), 2017, pp. 73-79, doi: 10.1109/ICInfA.2017.8078885.
19. C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer and G. Narayansamy, "Intrusion Detection System for Internet of Things based on a Machine Learning approach," 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 2019, pp. 1-6, doi: 10.1109/ViTECoN.2019.8899448.
20. N. Nestor, O. Belej and V. Tomyuk, "Application of Hybridization Methods to Detect Network Attacks in Wireless Sensor Networks," 2021 IEEE 16th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), 2021, pp. 44-48, doi: 10.1109/CADSM52681.2021.9385215.
21. S. Berrani, A. Yachir, B. Djemaa and M. Aissani, "Extended multi-agent system based service composition in the Internet of things," 2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS), 2018, pp. 1-8, doi: 10.1109/PAIS.2018.8598503.
22. M. Zouai, O. Kazar, B. Haba and H. Saouli, "Smart house simulation based multi-agent system and internet of things," 2017 International Conference on Mathematics and Information Technology (ICMIT), 2017, pp. 201-203, doi: 10.1109/MATHIT.2017.8259717.
23. S. M. A. Mavee and E. M. Ehlers, "A Multi-agent Immunologically-inspired Model for Critical Information Infrastructure Protection -- An Immunologically-inspired Conceptual Model for Security on the Power Grid," 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 2012, pp. 1089-1096, doi: 10.1109/TrustCom.2012.40.
24. B. U. I. Khan, F. Anwar, R. F. Olanrewaju, B. R. Pampori and R. N. Mir, "A Novel Multi-Agent and Multilayered Game Formulation for Intrusion Detection in Internet of Things (IoT)," in IEEE Access, vol. 8, pp. 98481-98490, 2020, doi: 10.1109/ACCESS.2020.2997711.

Received 03.05.21
Accepted 16.06.21